МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное учреждение науки
ИНСТИТУТ ПРОБЛЕМ РЕГИОНАЛЬНОЙ ЭКОНОМИКИ РОССИЙСКОЙ АКАДЕМИИ НАУК

Федеральное государственное бюджетное учреждение науки
ЦЕНТРАЛЬНЫЙ ЭКОНОМИКО-МАТЕМАТИЧЕСКИЙ ИНСТИТУТ РОССИЙСКОЙ АКАДЕМИИ НАУК

К 300-летию
Российской академии наук

В. Л. Макаров, В. В. Окрепилов, А. Р. Бахтизин

НАУЧНЫЕ РЕШЕНИЯ СЛОЖНЫХ ЭКОНОМИЧЕСКИХ И СОЦИАЛЬНЫХ ЗАДАЧ С ПОМОЩЬЮ СУПЕРКОМПЬЮТЕРОВ

URSS
МОСКВА

© URSS

ББК 65.050

Рецензенты:

академик РАН, научный руководитель ФГУ ФНЦ НИИСИ РАН, д-р физ.-мат. наук, проф. *В. Б. Бетелин*;
член-корреспондент РАН, д-р экон. наук, директор ИЭ РАН *М. Ю. Головнин*

Утверждено к печати в качестве монографии
Ученым советом ФГБУН ИПРЭ РАН (Протокол № 9 от 17 октября 2022 г.);
Ученым советом ФГБУН ЦЭМИ РАН (Протокол № 11 от 17 октября 2022 г.)

Макаров Валерий Леонидович,
Окрепилов Владимир Валентинович,
Бахтизин Альберт Рауфович

Научные решения сложных экономических и социальных задач с помощью суперкомпьютеров. — М.: ЛЕНАНД, 2023. — 416 с.; цв. ил.

В монографии, посвященной 300-летию Российской академии наук, представлены результаты совместных исследований группы ведущих российских ученых, разрабатывающих новые теоретико-методологические подходы, основанные на применении инструментов экономики качества и больших агент-ориентированных моделей для решения сложных экономических и социальных задач. Используя новые возможности цифровой экономики, авторы предлагают уникальные разработки, воссоздающие в искусственной среде многомерные модели функционирования реальных социально-экономических систем. Применение таких моделей возможно на любом уровне управления — от предприятия до страны, способствует обеспечению роста эффективности управленческих решений, важнейшим из которых является повышение качества жизни населения.

Книга адресована студентам, аспирантам, преподавателям, специалистам и руководителям предприятий, организаций и органов власти, занимающимся проблемами совершенствования научных основ управления и путями их практического решения в период цифровой трансформации экономики и социальной сферы.

Авторы выражают большую благодарность за участие в подготовке монографии сотрудникам Института проблем региональной экономики РАН д-ру физ.-мат. наук В. Т. Перекресту, канд. физ.-мат. наук Н. Л. Гагулиной, А. Г. Гридасову, сотрудникам Центрального экономико-математического института РАН канд. экон. наук Е. Д. Сушко, канд. экон. наук Б. Р. Хабриеву, ООО «Яндекс», PhD Г. Б. Сушко, начальнику департамента Государственного академического университета гуманитарных наук М. Ю. Сидоренко

ООО «ЛЕНАНД». 117312, Москва, пр-т Шестидесятилетия Октября, д. 11А, стр. 11.
Формат 60×90/16. Печ. л. 26. Зак. №

Отпечатано в АО «Т 8 Издательские Технологии».
109316, Москва, Волгоградский проспект, д. 42, корп. 5.

ISBN 978–5–9710–6284–4

978–5–9519–3858–9
(твердый переплет)

34465 ID 298911

© В. Л. Макаров, В. В. Окрепилов, А. Р. Бахтизин, 2023
© ЛЕНАНД, 2023

НАУЧНАЯ И УЧЕБНАЯ ЛИТЕРАТУРА
E-mail: URSS@URSS.ru
Каталог изданий в Интернете:
http://URSS.ru
Тел./факс (многоканальный):
+ 7 (499) 724 25 45
URSS

Все права защищены. Никакая часть настоящей книги не может быть воспроизведена или передана в какой бы то ни было форме и какими бы то ни было средствами, будь то электронные или механические, включая фотокопирование и запись на магнитный носитель, а также размещение в Интернете, если на то нет письменного разрешения владельцев.

Оглавление

Введение **7**

Литература к введению 17

Глава 1. Цифровизация экономики и ее влияние на изменение условий хозяйствования **18**

1.1. Цифровизация как объективный процесс технологического развития 18

1.2. Основа и тенденции цифровой трансформации 23

1.3. Развитие цифровизации — необходимый ответ на современные вызовы экономике России 28

1.4. Реализация национальных программ и проектов РФ в интересах цифровой трансформации экономики страны 39

1.5. Социально-экономические системы как объект управления и моделирования их развития 52

Литература к главе 1 65

Глава 2. Методология разработки агент-ориентированных моделей для применения суперкомпьютеров при исследовании экономических и социальных проектов **67**

2.1. Программные платформы для реализации агентных моделей на суперкомпьютерах 77

© URSS

2.1.1. Пандора — высокопроизводительное программное обеспечение для построения агент-ориентированных моделей ... 77
2.1.2. RepastHPC — среда проектирования агентных моделей для суперкомпьютеров ... 82
2.1.3. EpiFast — эффективный инструмент для агентного моделирования эпидемий на системах с распределенной памятью ... 88
2.1.4. SWAGES — расширяемая распределенная среда для крупномасштабного агент-ориентированного моделирования ... 99
2.1.5. Программное обеспечение для высокопроизводительных вычислений агентных моделей пешеходной динамики с использованием графических процессоров ... 111
2.1.6. CyberGIS — высокопроизводительный инструмент для пространственного анализа и агентного моделирования ... 117
2.1.7. HPABM — иерархическая платформа для параллельной реализации пространственно-распределенных агент-ориентированных моделей ... 122
2.1.8. Анатомические исследования с использованием агент-ориентированного подхода и параллельных вычислений ... 134
2.1.9. Агент-ориентированные модели для систем с эксафлопсной производительностью ... 143
2.1.10. Microsoft AXUM — агент-ориентированное программирование ... 148
2.1.11. Платформа для моделирования городской мобильности на основе агент-ориентированного подхода, геоинформационных и суперкомпьютерных технологий ... 154
2.1.12. POLARIS — программная платформа для построения транспортных симуляторов на базе агент-ориентированного подхода ... 184

© URSS

2.1.13. Симулятор поведения толпы с использованием агентного подхода, клеточных автоматов и высокопроизводительных параллельных вычислений.... 201

2.1.14. D-MASON — масштабируемая распределенная среда построения агент-ориентированных моделей 217

2.1.15. Care HPS — высокопроизводительный инструмент для параллельного и распределенного агентного моделирования 237

2.2. Построение цифровых агент-ориентированных моделей и их применение в социально-экономических исследованиях 249

2.2.1. EURACE — крупномасштабная модель европейской экономики 250

2.2.2. Система мониторинга состояния планеты, реализуемая на суперкомпьютерах 253

2.2.3. Компьютерная модель мира от Microsoft 258

2.2.4. Параллельный крупномасштабный симулятор транспортных потоков с использованием агент-ориентированного подхода и языка X10 259

2.2.5. Другие проекты в сфере экономики 265

2.3. Разработка на основе агент-ориентированных моделей и суперкомпьютерных технологий возможных сценариев развития социально-экономической системы России..... 274

2.3.1. Концепция разработанных демографических моделей 275

2.3.2. Конструкция первой разработанной модели 279

2.3.3. Результаты апробации первой модели 283

2.3.4. Отличия второй модели 286

2.3.5. Адаптация моделей для запуска на суперкомпьютере..... 289

2.3.6. Альтернативный способ реализации агентных моделей на суперкомпьютерах..... 296

2.3.7. Развитие конструкции агент-ориентированной демографической модели 314

© URSS

2.4. Применение цифровых моделей для оценки стратегий социального развития 327

Литература к главе 2 331

Глава 3. Методическое обеспечение исследования качества жизни в цифровой экономике 336

3.1. Качество жизни как сфера применения стандартизации, метрологии и управления качеством в условиях цифровизации 336

3.2. Разработка информационно-методического обеспечения оценки качества жизни 364

3.3. Исследование качества жизни в российских регионах 379

Литература к главе 3 408

Заключение 413

© URSS

Введение

Так уж устроен человек, что ему всегда хочется, оценивая настоящее, обязательно заглянуть в будущее, представить себе жизнь, которая будет через десятки лет или даже более продолжительный период. Для большинства людей такое стремление чаще всего сводится к размышлениям об обычных понятиях, связанных с условиями работы, быта и отдыха в грядущие времена. И это нормальное желание представить горизонты бытия, в которых предстоит жить не только нынешнему, но и последующим поколениям, нашим детям и внукам.

На этом желании увидеть будущее в течение нескольких последних веков выросла целая плеяда известных писателей-фантастов, таких как Герберт Уэллс, Жюль Верн, Айзек Азимов, Артур Кларк, Рэй Бредбери, Роберт Хайнлайн, а также наши соотечественники: Александр Беляев, Иван Ефремов, братья Стругацкие и многие другие, убедительно рисовавшие в своих книгах картины будущего мира и даже предсказывавшие выдающиеся научные открытия. В связи с этим обычно вспоминают, например, книгу Алексея Толстого «Гиперболоид инженера Гарина», впервые изданную в 1927 г. С середины 1950-х гг. советскими учеными Александром Прохоровым и Николаем Басовым разрабатывался квантовый генератор, который во многом реализовал на практике фантастическую идею этого романа. Аналогичный проект параллельно выполнил в США профессор Чарльз Таунс. В 1964 г. всем троим ученым была присуждена Нобелевская премия за это уникальное открытие, ставшее основой для применения лазеров в технике, в медицине и во многих отраслях экономики, а также в научных исследованиях.

Перспективы технического прогресса всегда привлекали создателей научной фантастики. Многие произведения этого направле-

ния литературы были связаны с попытками предугадать, как будет развиваться вычислительная техника, способствующая проведению сложнейших аналитических операций. Правда, в художественной форме авторы подобных произведений стремились представить результаты подобных исследований в виде появления уникальных бытовых изделий, роботов, способных действовать на основе искусственного интеллекта.

Однако в реальности первоначальное развитие вычислительной техники было связано с решением конкретных математических задач. Вовсе не случайно пальму первенства в создании прообраза современного компьютера чаще всего отдают английскому математику Чарльзу Бэббиджу. Именно он в 1822 г. (то есть 200 лет назад) создал машину для табулирования, которая включала арифметическое устройство, регистры памяти и перфокарты для ввода данных и вывода результатов вычислений. При этом машина успешно производила все основные арифметические действия.

Более века спустя, в 1941 г., американский ученый Говард Эйкин на основе чертежей Бэббиджа разработал машину, отдаленно напоминающую современные компьютеры. Ее основное предназначение соответствовало времени — на ней производились расчеты при создании военного оборудования. Впрочем, первенство в создании компьютера вправе оспаривать еще несколько других ученых, решавших с помощью своих аппаратов в период Второй мировой войны задачи расчетов аэродинамики самолетов (Конрад Цузе — Германия), баллистики (Джон Мокли — США), дешифровки секретных сообщений (Алан Тьюринг — Англия).

Дальнейшее развитие компьютерных технологий связано с появлением в 1959 г. и постоянным совершенствованием основы их деятельности — микросхем. Один из основателей компании Intel Гордон Мур в 1965 г. вывел на основе расчетов так называемый «Закон Мура», согласно которому количество транзисторов в микросхемах удваивается каждые 2 года. Такие темпы позволяют постоянно наращивать мощности вычислительных устройств и решать с их помощью все более сложные задачи.

Термин «супервычисления» впервые появился в обиходе в конце 1920-х гг., когда газета «New York World» рассказала о создании по заказу Колумбийского университета компанией IBM табулятора, способного с большой скоростью производить автоматическую обработку числовой и буквенной информации, записанной на перфокартах [1]. Фактически это был существенно улучшенный вариант машины Чарльза Бэббиджа.

Однако временем реального появления первых суперкомпьютеров принято считать 1965 г., когда по заказу американского космического агентства NASA была создана ЭВМ, впервые работающая на принципе параллельной вычислительной системы. Эта машина — ILLIAC IV — могла выполнять 150 млн операций с плавающей точкой в секунду (150 мегафлопсов). Это был качественный скачок в производительности ЭВМ, поскольку прежние компьютеры обладали производительностью менее 1 мегафлопса.

Следующей вехой стало изобретение американцем Сеймуром Креем в 1976 г. первой малогабаритной супер-ЭВМ CRAY-1, имевшей производительность 180 мегафлопсов и запущенной в серию для использования в проектах государственного управления, решения задач промышленности и в научных исследованиях США. Именно в этот период за машинами с максимальной производительностью окончательно закрепился термин «суперкомпьютер».

В последующие годы между крупнейшими производителями вычислительной техники происходила гонка по созданию самого быстродействующего суперкомпьютера. Последовательно в 1983 г. был преодолен рубеж в производительности в 1 миллиард флопс, в 1996-м — 1 триллион флопс, в 2008-м — 1 квадриллион флопс (1 петафлопс), наконец, в 2022 г. — 1 квинтиллион флопс (1 эксафлопс).

В рейтинге 500 самых мощных в мире суперкомпьютеров [2], который обновляется дважды в год (в июне и декабре), первенство в 2022 г. перешло к суперкомпьютеру «Frontier», установленному в Ок-Риджской национальной лаборатории США. Его производительность составляет 1,102 эксафлопса, или 10^{18} вычислительных операций с плавающей запятой в секунду. Он в два с половиной

раза перекрыл показатели прежнего лидера рейтинга — суперкомпьютера «Фугаку», применяемого в Центре вычислительных наук Института физико-химических исследований в японском городе Кобе.

Если на начальной стадии создания суперкомпьютеров они применялись исключительно для решения задач военного предназначения, то сегодня сложно найти отрасль, в которой не использовались бы возможности самых быстродействующих электронных систем. В первую очередь они стали применяться для проведения сложных расчетов в математике, физике, химии, биологии, при моделировании процессов, происходящих в земной среде обитания человека.

Динамику применения суперкомпьютеров можно оценить на примере их использования в такой сложнейшей для точности расчетов области, как прогнозирование погоды. Гидрометцентры крупнейших стран уже достаточно давно применяют в своей деятельности суперкомпьютеры. Сегодня это позволяет предсказывать погоду с высокой точностью, на срок от одного до трех часов, обновляя прогнозы с такой периодичностью. В конце 2022 г. в Великобритании в службе «Met Office» планировали приступить к использованию нового суперкомпьютера с производительностью свыше 40 петафлопс, что в 6 раз мощнее ныне действующего [3]. Британские метеорологи будут обладать самым мощным суперкомпьютером среди всех метеослужб мира. Затраты на его приобретение и обслуживание в течение 10 лет составят более 1,5 млрд долларов. Но его использование позволит не только перейти к решению задачи составления «прогноза для каждой улицы», но и попытаться осуществить проект так называемой «бесшовной модели атмосферы», который бы объединил раздельно осуществляемые сегодня наблюдения глобального характера (охватывающие крупнейшие атмосферные процессы по всему земному шару), регионального (слежение за изменениями погодных явлений с точностью до километров) и микроклиматические (вплоть до оценки силы порыва ветра на конкретной улице и у углов зданий, когда

состояние погоды оценивается и прогнозируется с точностью до метров).

Решение такой сложной задачи могло бы стать важнейшим переходным этапом к осуществлению еще более амбициозного проекта. По оценке специалистов в этой сфере, создание к 2030 г. зеттафлопных суперкомпьютеров может обеспечить разработку точной модели погодной системы всей планеты на срок не менее двух недель, учитывающей как все глобальные, так и местные вариации. Трудно переоценить практическое значение реализации такого проекта для раннего предупреждения о таких аномальных природных явлениях, как ураганы, торнадо, наводнения, цунами и другие катаклизмы, что позволит значительно сократить ущерб от их воздействий. Точный прогноз погоды на такой срок поможет оптимизировать режимы работы в сельском хозяйстве, воздушном и водном транспорте, в других отраслях, зависящих от воздействия земных, водных и воздушных аномалий. Это знание поможет и на бытовом уровне — позволит людям планировать свой график хозяйственных работ на улице, поездки на природу, выбирать варианты отдыха, ориентируясь заранее на точный прогноз погоды на достаточно продолжительный период в конкретной местности.

К задачам еще более сложного порядка относятся проблемы, к решению которых исследователи только начинают приступать с помощью суперкомпьютеров и необходимого программного обеспечения. Речь идет о применении суперкомпьютерных технологий для оценки текущего состояния и моделирования дальнейшего развития социально-экономических систем. Такая задача включает в себя анализ и прогноз огромного комплекса данных экономического, социального, политического, экологического характера, в котором, — для понимания сложности такой задачи, — точный прогноз погоды может являться одним из небольших штрихов при выработке общих решений.

Лучшие умы человечества в своих философских и социально-экономических теориях в течение многих веков стремились установить долговременные закономерности общественного развития.

В трудах Адама Смита, Дэвида Рикардо, Георга Вильгельма Фридриха Гегеля, Карла Маркса, Джона Мейнарда Кейнса и других ученых эти закономерности выявлялись на основе разных подходов, обогащая тем самым сферу научного познания мира и прогнозирования его будущего элементами разносторонних оценок и доказательств.

Выдающиеся российские ученые считали своим главным делом помощь в решении экономических и социальных задач нашей страны. Один из первых академиков России, стоявший у истоков отечественной науки и высшего образования, Михаил Васильевич Ломоносов в одном из наиболее значимых своих научных трудов «О сохранении и размножении российского народа» [4] предложил детальную программу успешного развития производительных сил России. В середине XVIII в. путь к «внутреннему изобилию» он видел через улучшение земледелия, совершенствование здравоохранения, широкое просвещение народа.

Всемирно известный ученый, создатель периодической системы химических элементов Дмитрий Иванович Менделеев проявил себя истинным патриотом России в своих разнообразных и многочисленных трудах, посвященных укреплению промышленности, сельского хозяйства, транспорта, финансов, таможенного дела и других отраслей экономики. Сам Менделеев однажды заметил: «Мне говорят, ведь вы химик, а не экономист, зачем же входите не в свое дело?» [5]. На это ученый ответил, что быть химиком вовсе не означает не интересоваться экономическими проблемами государства, верное решение которых зачастую зависит от применения в том числе и методов естествознания. Своим энциклопедическим умом Д. И. Менделеев стремился охватить многие сферы знания, пытаясь установить общие закономерности экономического и социального развития и определить наилучший путь к общественному благополучию.

Значительный вклад в этот процесс внес в 1920-е гг. выдающийся российский экономист Николай Дмитриевич Кондратьев. Он предложил системную теорию волнового характера социально-эконо-

мического развития, получившую впоследствии название «циклы Кондратьева».

К своим выводам Николай Кондратьев пришел на основе тщательного анализа колоссального объема статистических данных, характеризующих развитие большой группы стран на протяжении 100–150 лет, фактически с момента начала промышленной революции в мире. Эти данные включали такие показатели, как добыча угля, золота, производство чугуна, свинца, тканей и другие результаты работы промышленности. Анализу были подвергнуты также показатели внешнеторгового оборота стран, динамика стоимости государственных деловых бумаг, индекса цен, номинальной заработной платы, изменения в численности населения, данные, характеризующие влияние Первой мировой войны на различные страны, и целый ряд других показателей.

Обладая большим опытом научной и практической работы в сфере статистических исследований и высокой культурой математических расчетов, Н. Д. Кондратьев в результате своего анализа установил, что в общем социально-экономическом развитии стран с определенной закономерностью наблюдаются волны подъемов и спадов, повторяющиеся с периодичностью в 50 лет, плюс-минус 10 лет, которые он предложил рассматривать как большие циклы, в ходе которых фазы относительно высоких темпов экономического роста сменяются на фазы с более низкими показателями.

В своем докладе «Большие циклы экономической конъюнктуры» [6] ученый выделил четыре основных вывода своих исследований, назвав их «эмпирическими правильностями». Первая из них состоит в том, что перед началом или в самом начале «повышательной» волны большого цикла наблюдаются значительные изменения в основных условиях хозяйственной жизни общества. Эти изменения обычно выражаются (в той или иной комбинации) в глубоких трансформациях техники производства и обмена (которым, в свою очередь, предшествуют значительные технические изобретения и открытия), в трансформации условий денежного обращения, в усилении роли новых стран в мировой хозяйственной жизни.

Вторая эмпирическая правильность состоит в том, что периоды «повышательных» волн больших циклов, как правило, значительно богаче на крупные социальные потрясения и перевороты в жизни общества (революции, войны), чем периоды «понижательных» волн.

Третья эмпирическая правильность, согласно выводу Кондратьева, заключается в том, что «понижательные» волны этих циклов сопровождаются длительной депрессией сельского хозяйства.

Четвертая правильность объясняет характер взаимодействия больших и средних (7–12 лет) циклов экономического развития. Большие циклы экономической конъюнктуры выявляются в том же едином процессе динамики экономического развития, в котором выявляются и средние циклы с их фазами подъема, кризиса и депрессии. Средние циклы поэтому как бы нанизываются на волны больших циклов. При этом «повышательный» тренд большого цикла усиливает аналогичную тенденцию в среднем цикле, а в фазе понижения волны большого цикла усиливается такая же тенденция среднего цикла.

Сегодня теория больших циклов Н. Д. Кондратьева широко признана учеными всего мира и постоянно применяется многими специалистами для оценки и прогнозирования социально-экономических явлений.

Важно отметить, что Николай Кондратьев, опираясь на свой опыт исследователя экономических процессов, в 1920-е гг. занимался практическим планированием развития народного хозяйства СССР. В этот период, благодаря организационным усилиям Кондратьева и его научному подходу, была создана обширная информационно-статистическая база для изучения динамических закономерностей в развитии народного хозяйства СССР. Под его руководством был разработан перспективный план развития сельского и лесного хозяйства РСФСР на 1923–1928 гг. (сельскохозяйственная пятилетка Кондратьева), сочетавший плановые и рыночные принципы.

Это позволило ученому сформулировать принципиальные подходы к составлению планов, изложенные в его работе «План и

предвидение» (1927). Н. Д. Кондратьев писал: «Совершенно бесспорно… что сама плановая работа требует планового упорядочения. Она требует упорядочения организационного. Не в меньшей, если не в большей степени она требует усовершенствования со стороны самих методов построения планов» [7].

Важнейшим среди предложений Кондратьева в сфере планирования было соблюдение правильного соотношения показателей, которые подлежат точному расчету и последующему учету, с теми, предвидеть точное количественное выражение которых не представляется возможным. По мнению Кондратьева, в перспективных планах должно быть углубленное экономическое освещение вероятных и желательных тенденций развития народного хозяйства и его отраслей, а также направления мероприятий. При этом нужно учитывать особенности и практическую необходимость количественных расчетов, возможностей в различных областях хозяйства.

И еще один принципиальный вывод можно сделать, применяя научное наследие Николая Кондратьева к практике современных социально-экономических исследований. Анализируя огромное количество факторов при анализе их взаимозависимости в ходе разработки теории больших циклов, ученый уделял особое внимание тем тенденциям, которые отражали динамику качества рабочей силы, благосостояния людей, их доходов (в том числе по странам на душу населения), показатели рождаемости и смертности, другие данные, характеризующие современное понятие «качество жизни».

Таким образом, можно с полным правом утверждать, что сформулированные Н. Д. Кондратьевым методологические подходы к оценке различных экономических и социальных процессов остаются и сегодня актуальными и могут использоваться в современных исследованиях.

Выдающиеся ученые прошлого на основе научного предвидения стремились не только установить закономерности общественного развития, но и найти алгоритмы их дальнейшей реализации. Однако они не могли полностью учитывать огромную множествен-

ность вариантов, поскольку не обладали для этого необходимыми инструментами вычислений.

Применение суперкомпьютерных систем и других цифровых технологий создает принципиально новые возможности для реализации таких проектов. В данной монографии представлены результаты научно-исследовательских работ, проведенных авторскими коллективами Центрального экономико-математического института (ЦЭМИ) РАН и Института проблем региональной экономики (ИПРЭ) РАН в рамках выполнения программы Президиума РАН «Фундаментальные проблемы математического моделирования», а также программы Президиума РАН «Механизмы обеспечения отказоустойчивости современных высокопроизводительных и высоконадежных вычислений» и подпрограммы «Фундаментальные проблемы решения сложных практических задач с помощью суперкомпьютеров».

В первой главе монографии рассмотрены процессы, связанные с цифровизацией мировой экономики, влиянием цифровой трансформации на современное социально-экономическое развитие России и с особенностями разработки и выполнения Национальной программы «Цифровая экономика Российской Федерации».

Во второй главе раскрывается методология разработки агент-ориентированных моделей для применения суперкомпьютеров при исследовании экономических и социальных проектов.

В третьей главе представлен новый теоретико-методологический подход, основанный на применении инструментов экономики качества — стандартизации, метрологии и управления качеством, который в условиях цифровой экономики создает дополнительные возможности для исследования и повышения качества жизни.

Авторы монографии в своих исследованиях рассматривают повышение качества жизни как самую важную и сложнейшую социальную, экономическую и управленческую задачу, поиску научного решения которой в значительной степени может способствовать совместное применение экономики качества и агент-ориентированных моделей с использованием суперкомпьютерных технологий.

Литература к введению

1. *Гузик В. Ф.* Проектирование высокопроизводительных проблемно-ориентированных вычислительных систем: монография / В. Ф. Гузик, Е. В. Ляпунцова, Д. А. Беспалов, М. Ю. Поленов. Изд. 2-е, испр. и доп. Ростов-на-Дону; Таганрог: Изд-во Южного федерального университета, 2017. С. 21.
2. https://www.top500.org/lists/top500/2022/06/
3. Новый британский погодный суперкомпьютер: что он сможет и как это изменит нашу жизнь. URL: https://itnan.ru/post.php?c=1&p=489506
4. *Ломоносов М. В.* О сохранении и размножении российского народа // Журнал древней и новой словесности. 1819. Ч. 5. Март. № 6. С. 52–78.
5. *Окрепилов В. В.* Дмитрий Иванович Менделеев — ученый, педагог, гражданин // Труды XIV Всероссийской научно-практической конференции с международным участием «Здоровье — основа человеческого потенциала: проблемы и пути их решения». 2019. № 1. Т. 14. С. 36.
6. Тезисы доклада Н. Д. Кондратьева «Большие циклы экономической конъюнктуры» // Кондратьев Н. Д., Яковец Ю. В., Абалкин Л. И. Большие циклы конъюнктуры и теория предвидения. Избранные труды. М.: Экономика, 2002. С. 397–400.
7. *Кондратьев Н. Д.* План и предвидение // Кондратьев Н. Д., Яковец Ю. В., Абалкин Л. И. Большие циклы конъюнктуры и теория предвидения. Избранные труды. М.: Экономика, 2002. С. 568.

© URSS

Глава 1
Цифровизация экономики и ее влияние на изменение условий хозяйствования

1.1. Цифровизация как объективный процесс технологического развития

Во второй половине XX в. традиционный характер развития экономики в прежние эпохи с опорой на расширение материального производства коренным образом изменился. Превращение науки в ведущий фактор развития в результате научно-технической революции обусловило формирование новой ключевой технологии — интеллектуальной, что обозначило переход к новому этапу научно-технического прогресса и интенсификации общественного производства.

Сменились и приоритеты общественного развития. Так, 200 лет назад главными ценностями были размер, территория государства, 100 лет назад на первый план вышли развитая промышленность и природные ресурсы. Сегодня же основной ценностью являются знания и информация, отражающиеся в объеме инноваций, повышающих устойчивость экономического развития и качество жизни населения.

На современном этапе развития доминирующими стали виды деятельности, для которых на первом плане создание новых нематериальных ценностей. Стремительное совершенствование информационных технологий и повсеместное внедрение автоматизации очень заметно влияют на уровень знаний и компетенций людей, увеличивая их возможности и интеллектуальный вклад в развитие экономики. Возрастает доля наукоемких отраслей, соз-

дающих продукцию на основе высоких технологий. Общей закономерностью изменений в отраслевой структуре мирового хозяйства является рост удельного веса сферы услуг, что способствует формированию обширной инфраструктуры экономики, усилению ее диверсификации.

В числе основных причин быстрой трансформации можно выделить как экономические, так и технологические изменения:

- экспоненциальный рост всемирной информационной «включенности» вследствие уменьшения затрат на стоимость подключения к информационным сетям;
- рост вычислительных мощностей на фоне снижения затрат;
- появление дешевых средств и способов хранения больших объемов данных;
- применимость современных технологий в контексте больших массивов данных;
- возможность обрабатывать исходные данные на уровне рядового потребителя.

Исходя из опыта научно-технического развития, можно говорить о том, что современная цифровизация экономики является предшественницей новой научно-технической революции, которая приведет к созданию новой индустрии, основанной на комбинации интеллектуальных производственных и информационных технологий, то есть симбиозе человеческого и машинного интеллекта.

Содержание понятия «цифровая экономика» постоянно менялось с того момента, как вошло в обиход в 1995 г. Впервые термин «цифровая экономика» был использован американским ученым из Массачусетского университета Николасом Негропонте. Сегодня понятие «цифровая экономика» в широком смысле означает цифровую трансформацию всей существующей экономики при лидирующей роли цифровых технологий. Цифровая экономика в узком смысле подразумевает цифровой сектор национальной экономики.

Цифровая трансформация экономики в целом — закономерный процесс, так как является проявлением системных изменений в эко-

номике, что выражается в переносе центров создания добавленной стоимости в сферу выстраивания цифровых ресурсов и сквозных цифровых процессов. Цифровая трансформация порождает новый технологический уклад, формирование которого сопровождается появлением новых отраслей экономики.

От цифровой революции общество, бизнес, население ожидают новых, более эффективных, моделей управления. Огромная значимость появления таких моделей обусловлена тем, что на мировых рынках в эпоху цифровой экономики конкуренция товаров и услуг уступает место конкуренции моделей управления.

Обращаясь к анализу цифровой экономики с позиций системного подхода, можно утверждать, что она представляет собой модифицированную социо-эколого-экономическую *цифровую* систему (далее — «цифровая экосистема») со своими законами и правилами. Такая система относится к классу кибернетических систем со всеми присущими ей свойствами: эмерджентностью, изменчивостью, включенностью, массовым характером экономических процессов и наличием субъективного фактора и фактора случайности (подробнее эти свойства рассматриваются в разделе 1.5). Когда объектом управления является экономическая система, главными критериями ее оптимизации выступают доход, прибыль, минимизация затрат. Переход к цифровой экосистеме существенно меняет задачи управления. Главными критериями оптимизации становятся критерии соразмерности, а условием поддержания устойчивости — достаточность информационного обеспечения ее потенциалов.

Отталкиваясь от стандартного представления модели социо-эколого-экономической системы в виде трех крупных взаимодействующих подсистем — социальной, экологической и экономической, можно представить критерии оптимизации в модели цифровой экосистемы как совокупность критериев оптимизации подсистем (рис. 1.1.1).

В экологической подсистеме основной критерий оптимизации: стабильная продуктивность экосистем, которая обеспечена их мак-

Рис. 1.1.1. Критерии оптимизации цифровой экосистемы в условиях цифровой экономики

симальной устойчивостью к внешним воздействиям и, соответственно, минимальными изменениями. Критерием оптимизации в социальной подсистеме служит достижение наивысшего общественного благосостояния в стабильных институциональных условиях, когда сильные институты длительное время «работают» на повышение качества жизни, что сопровождается установлением высоких значений показателей, формирующих индекс человеческого развития: продолжительности жизни, валового объема производства в расчете на душу населения и уровня образования населения. При выполнении критериев оптимизации в отдельных составляющих модели, представленной на рис. 1.1.1, уровень жизни в цифровой экономике

будет расти на фоне роста экономической эффективности и минимизации природоемкости производства.

Приведенная модель хорошо укладывается в определение цифровой экономики, сформулированное в **Программе развития цифровой экономики в Российской Федерации до 2035 г., разработанной в 2017 г.** Центром изучения цифровой (электронной) экономики. В этой программе цифровая (электронная) экономика определена как **«совокупность общественных отношений, складывающихся при использовании электронных технологий, электронной инфраструктуры и услуг, технологий анализа больших объемов данных и прогнозирования в целях оптимизации производства, распределения, обмена, потребления и повышения уровня социально-экономического развития государств»** [1].

Для достижения оптимального развития территории при удовлетворении настоящих и будущих потребностей населения необходимо соблюдать соответствие производственных мощностей территории ее природно-ресурсному потенциалу. Все критерии, приведенные на рис. 1.1.1, должны быть задействованы в практике управления цифровой экономики. Критерию соизмерения отводится отдельная роль — роль регулятора, направленного на долговременное сохранение устойчивости системы.

Цифровая экономика — молодая система, в которой огромный потенциал развития сопровождается значительными рисками. Джон Чемберс, председатель совета директоров одной из крупнейших компаний, занимающих лидирующие позиции в области сетевых технологий, связи и инфокоммуникаций, — «Cisco Systems», высказал свою точку зрения на развитие бизнеса в условиях цифровой экономики. По его словам «как минимум 40 % бизнесов умрут в течение ближайших 10 лет, если они не решат, как изменить работу их компаний с использованием новых технологий» [2]. Возможно, реальная угроза потери бизнеса не столь велика, но она существует и подстегивает внедрение инноваций, по крайней мере в успешных производствах и предприятиях.

1.2. Основа и тенденции цифровой трансформации

Основу цифровой трансформации составляют:

- создание ключевых институтов цифровой экономики;
- создание основных инфраструктурных элементов цифровой экономики.

В рамках создания ключевых институтов цифровой экономики необходимо осуществить ряд мер по построению нормативно-правовой базы и правил, обеспечивающих эффективное взаимодействие внутри цифровой экосистемы. В числе правил в первую очередь следует обозначить направление «Нормативное регулирование», нацеленное на формирование новой регуляторной среды, обеспечивающей благоприятный правовой режим для возникновения и развития современных технологий, а также для осуществления экономической деятельности, связанной с их использованием. В числе «аналоговых» основ развития цифровой трансформации наряду с нормативно-правовым регулированием выделяются информационная инфраструктура, информационная безопасность и другие.

Создание основных инфраструктурных элементов цифровой экономики предполагает прежде всего развитие сетей связи, развитие системы центров обработки данных, внедрение цифровых платформ работы с данными для обеспечения потребностей граждан, власти и бизнеса. Технологическая совместимость цифровых решений, общих цифровых платформ и цифровой инфраструктуры в целом будет способствовать ускорению обмена товарами и услугами, развитию экономики и повышению ее эффективности. Для поддержания жизнеспособности и в целях дальнейшего развития «аналоговых» основ цифровой экосистемы необходимо усиление роли науки и кадрового обеспечения перечисленных приоритетных направлений.

В числе основных тенденций цифровой трансформации выделяются:

- интернет вещей (Internet of Things, IoT);
- дополненная (Augmented Reality, AR) и виртуальная реальность (Virtual Reality, VR);

- интерфейсы прикладных программ (Application Programming Interfaces, API);
- умные машины и искусственный интеллект (Artificial Intelligence, AI);
- уничтожение функциональных колодцев;
- гиперсвязь 5G;
- развитие технологии блокчейн.

Интернет вещей

Прогнозируемый среднегодовой темп глобального рынка интернета вещей составляет 28,5 %. Сегменты, на которые опирается рост рынка IoT: умные города (26 %), промышленный интернет вещей (24 %), здоровье (20 %), умные дома (14 %), автомобили (7 %), умные утилиты — вспомогательные компьютерные программы (4 %) и носимые устройства (3 %). Корпоративный и промышленный сегмент рынка интернета вещей составляют самые конкурентоспособные решения. При этом наибольшая доля инвестиций (порядка 40 млрд долл. на направление) поступит в дискретное производство, транспорт и логистику, утилиты.

Интернет вещей позволит пересмотреть системы управления качеством в связи с новым подходом к решению проблем точности, скорости и уровня контроля.

Дополненная и виртуальная реальность

Программы виртуальной и дополненной реальности находят применение в разных сферах деятельности. Это видеоигры, здравоохранение, проектирование, реализация медиапроектов в прямом эфире, образование и другие сферы. Развитие технологии дополненной реальности значительно отстает от развития виртуальной, которая активнее всего разрабатывается в области игровой индустрии.

Интерфейсы прикладных программ

Серьезные изменения в стандартизации информационного взаимодействия связаны с созданием и расширением интерфейса

прикладного программирования. API представляет собой определенный набор правил и спецификаций, которые могут выполнять программные программы для связи друг с другом. Подобно тому, как пользовательский интерфейс облегчает взаимодействие между людьми и компьютерами, API служит интерфейсом между различными программными программами и облегчает их взаимодействие, работая на повышение эффективности решаемых задач.

Умные машины и искусственный интеллект

Машинный интеллект, основанный на возросших компьютерных мощностях и лучшем понимании математических моделей, находит наиболее широкое распространение в финансовых и транспортных сегментах современной экономики, но его использование не сводится только к данным сегментам. В настоящее время машинный интеллект, использующий технологию нейронных сетей, ускоряет процесс принятия решений за счет расширения возможностей подбора своевременной релевантной информации. Однако уже сейчас есть примеры выбора искусственным интеллектом самостоятельных решений, например игровых, — компьютерная программа «AlfaGo», которая в матче по сложнейшей настольной игре го, состоявшемся в марте 2016 г., обыграла многократного чемпиона по этой игре Ли Седоля [3].

Уничтожение функциональных колодцев

Функциональные колодцы опираются на распределение полномочий и ответственности между менеджерами функциональных подразделений. Цифровая трансформация способствует созданию такой системы непрерывных ценностных улучшений в организации, когда продукт можно короткими итерациями изменять в соответствии с требованиями потребителей сколь угодно продолжительное время.

Наиболее подходящим инструментом для уничтожения функциональных колодцев представляются системы менеджмента качества.

Гиперсвязь 5G

Революционные изменения в связи с цифровой трансформацией в недалеком будущем станут опираться на возможности, которые открываются в связи с развитием технологии 5G: пятого поколения беспроводных мобильных сетей. Стандарты для 5G (IMT-2020) окончательно были разработаны в 2020 г., к их внедрению, например, уже приступили в 13 странах Евросоюза, преимущественно в крупных городских районах. Сети 5G уже работают в ряде городов США, Китая, Южной Кореи, Великобритании.

Развитие технологии Блокчейн

Технология блокчейн, или «интернет ценностей» предполагает полную свободу и независимость цепи без единого администратора. Живой интерес к технологии привел к тому, что уже сейчас появились централизованные технологии блокчейна, позволяющие установить централизованную систему контроля. Одно из преимуществ данной технологии состоит в том, что с помощью сети блокчейн можно хранить любые данные, которые нуждаются в отдельной записи и возможности проверки в будущем.

Анализ основных тенденций цифровой трансформации показывает, что отрасли экономики, имеющие доступ к большим массивам данных и использующие их наиболее эффективно, получат возможность радикально повысить качество принимаемых решений на их основе.

Для успешной реализации стратегии цифровой трансформации можно назвать следующие условия:

- создание нормативно-правовой базы и институтов управления и регулирования цифровой экономики;
- повышение степени вовлеченности всех субъектов современной экономики: органов государственной власти, предпринимательского сектора, общественных организаций, средств массовой информации, исследовательских и образовательных учреждений, населения;

- усиление заинтересованности участников цифровой трансформации в улучшениях.

Выявленные тенденции цифровой трансформации способствуют формированию ярко выраженных трендов, среди которых можно выделить следующие:

- массовое внедрение интеллектуальных (квантовых) датчиков в оборудование и производственные линии (технологии индустриального интернета вещей);
- переход на массовое внедрение роботизированных технологий;
- перевод хранения информации и проведения вычислений на перераспределенные ресурсы;
- создание единой интегрированной автоматизированной информационной системы производственных и управленческих процессов;
- внедрение «безбумажных» технологий посредством обязательного использования только оцифрованной технической документации;
- цифровое проектирование и моделирование технологических процессов на всем жизненном цикле;
- применение технологий наращивания материалов взамен технологий среза;
- применение мобильных технологий для мониторинга, контроля и управления процессами;
- развитие технологий промышленной аналитики.

Цифровые технологии повышают эффективность выполнения торговых транзакций, улучшают прозрачность и отчетность, устраняют задержки и снижают коррупционные риски. Снижение стоимости транзакций, связанных с перемещением нецифровых товаров и услуг (например, путем региональных таможенных реформ), также может дать значительный экономический эффект, в том числе за счет стимулирования роста цифровой торговли.

© URSS

1.3. Развитие цифровизации — необходимый ответ на современные вызовы экономике России

Следует признать, что начало периода развития цифровой экономики пришлось в России на неблагоприятное время. В середине 1990-х гг. страна переживала переход от прежней социально-экономической формации к новым условиям хозяйствования и управления, в рамках которого происходил разрыв хозяйственных связей и упадок производства во многих отраслях, что прямо затронуло и сферу информационных технологий.

Первопроходцами в создании компьютерной техники в Советском Союзе принято считать лабораторию моделирования и вычислительной техники, которая начала свою деятельность в 1947 г. в составе только что учрежденного Института электротехники Академии наук Украинской ССР. Инициатором появления такой лаборатории был назначенный директором института С. А. Лебедев. Под его руководством в течение двух лет была разработана и в 1951 г. авторитетной комиссией АН СССР под председательством академика М. В. Келдыша принята в эксплуатацию Малая электронная счетная машина (МЭСМ), положившая начало успешной работе этой лаборатории. Всего с участием С. А. Лебедева было разработано 18 вариантов ЭВМ, из которых 15 было запущено затем в серийное производство. Коллеги называли Сергея Алексеевича Лебедева основоположником советского компьютеростроения. В 1953 г. он был избран академиком АН СССР, годом ранее стал в Москве директором Института точной механики и вычислительной техники АН СССР, которым руководил более 20 лет.

В Киеве на базе созданной С. А. Лебедевым лаборатории в 1957 г. был создан Вычислительный центр, через 5 лет преобразованный в Институт кибернетики АН УССР. Сегодня он носит имя его первого руководителя, которым стал академик В. М. Глушков.

В Институте кибернетики в 1960-е гг. начались исследования по теории искусственного интеллекта и теории программирования. Институт являлся лидером в разработке методологических подходов

к применению кибернетики для моделирования и решения сложных задач в таких отраслях науки, как экономика, биология, медицина. Академик Глушков в 1963 г. предложил создать на основе объединения всех вычислительных центров страны единую общегосударственную автоматизированную систему учета и обработки информации (ОГАС). В этот и последующие периоды работы по созданию компьютерной техники и соответствующие научные исследования активно развивались в Ереване, Минске, Пензе, в подмосковном Зеленограде и в ряде других городов страны.

В 1970-е гг. благодаря таланту выдающегося российского инженера и конструктора, академика РАН В. С. Бурцева были развернуты работы по созданию отечественной вычислительной техники с максимальной на тот период производительностью. Первоначально использование таких машин предназначалось для систем наведения истребителей на воздушные цели, высокоточной работы радиолокационных станций в составе комплексов противоракетной обороны. Работы проводились в закрытом режиме и полностью основывались на отечественных разработках. В дальнейшем они были использованы при создании серии советских суперкомпьютеров «Эльбрус», достигавших производительности более 100 млн операций в секунду. Реализованные в этих машинах технические идеи, такие как суперскалярность процессорной обработки, симметричная многопроцессорная архитектура с общей памятью, реализация защищенного программирования с аппаратными типами данных, были абсолютно революционными в создании вычислительной техники и заметно опережали разработки западных специалистов.

Значительную роль в развитии не только отечественных, но и мировых достижений в сфере информационных технологий сыграли открытия академика РАН Ж. И. Алферова, путь к которым был начат им в Физико-техническом институте имени А. Ф. Иоффе еще в 1960–1970-х гг. Вручение ему в 2000 г. Нобелевской премии за развитие полупроводниковых гетероструктур для высокоскоростной оптоэлектроники подтвердило его выдающийся вклад в создание основ современной ИТ-индустрии.

Академик РАН Г. Я. Красников, представляющий следующее поколение российских ученых, работающих в сфере физики полупроводников, добился уникальных результатов в разработке технологии создания сверхбольших интегральных схем и обеспечении высокого качества их промышленного производства в российских условиях, что крайне важно сегодня для самостоятельного развития отечественного электронного приборостроения.

Отметим, что избрание в сентябре 2022 г. Г. Я. Красникова президентом Российской академии наук является свидетельством того, что в отечественной науке наступает новый этап, когда на первый план выходят не только исследования в сфере цифровых технологий, но и особый акцент делается на практическое применение научных достижений на этом важнейшем в данный период направлении развития страны.

Такая закономерность в деятельности Академии наук прослеживается постоянно на протяжении многих десятилетий, когда решалась судьба крупнейших научных проектов, обеспечивавших надежную обороноспособность государства и приоритет в самых передовых отраслях науки и техники.

Первым в этом ряду следует назвать выдающегося российского ученого академика АН СССР Игоря Васильевича Курчатова, «отца» советской атомной бомбы. Он был научным руководителем атомного проекта, реализация которого позволила в кратчайшие сроки обеспечить создание ядерного щита страны, который и поныне является главным звеном ее надежной обороны. Курчатов внес также огромный научный вклад в мирное использование атомной энергии. Созданный им в 1943 г. Институт атомной энергии, которым он руководил до своей кончины в 1960 г., ныне — Национальный исследовательский центр «Курчатовский институт», является сегодня одним из крупнейших научных учреждений России, развивая исследования в сфере применения атомной энергии и в других областях физики. Заложенные И. В. Курчатовым традиции высокоэффективной работы успешно продолжаются коллективом Института, который с 2005 г. возглавляет член-корреспондент РАН, член

президиума Совета при Президенте РФ по науке и образованию М. В. Ковальчук.

Анатолий Петрович Александров, президент АН СССР в 1975–1986 гг., был одним из тех, кто также олицетворял осуществление атомного проекта — от создания в 1940-х гг. ядерного оружия до развития атомной энергетики и использования мирного атома в различных отраслях в последующие десятилетия. Выдающийся ученый в области прикладной математики и механики Мстислав Всеволодович Келдыш был одним из идеологов практического использования космоса и создателей ракетно-космических систем. Он возглавлял Академию наук СССР в 1961–1975 гг., в наиболее успешный период реализации советской космической программы.

Академик Евгений Павлович Велихов, избиравшийся вице-президентом АН СССР (1978–1991) и РАН (1991–1996) объединяет два важнейших научных направления. Он был директором Института атомной энергии, возглавлял советскую программу разработки управляемых термоядерных реакторов. С 1983 г. Е. П. Велихов в течение восьми лет был академиком-секретарем Отделения информатики, вычислительной техники и автоматизации АН СССР, а затем вновь возглавлял его уже в РАН в 2007–2017 гг. с новым названием — «Отделение нанотехнологий и информационных технологий». Именно при его активном участии сформировался нынешний состав отделения, руководитель которого (2019–2022 гг.) академик Г. Я. Красников был избран президентом РАН. На посту академика-секретаря отделения его сменил академик РАН В. Я. Панченко, научный руководитель Института проблем лазерных и информационных технологий РАН, избранный вице-президентом РАН.

С именем входящего в состав Отделения нанотехнологий и информационных технологий РАН академика РАН В. Б. Бетелина связано развитие инструментальных систем программного обеспечения и компьютерного моделирования сложных аппаратных комплексов. Работая с 1990 г. директором, а с 2016 г. — научным руководителем Научно-исследовательского института системных исследований РАН, В. Б. Бетелин обеспечил создание и внедрение в

серийное производство высокопроизводительных ЭВМ специального назначения. В самом начале своей научной карьеры в 1970-е гг. с группой молодых ученых он приступил к разработке методологии программного и объектно-ориентированного программирования, которая в дальнейшем стала применяться не только к производственным процессам, но и к широкому кругу явлений экономики и социальной жизни, что в том числе составляет основу данной монографии.

Можно с полным правом утверждать, что в нашей стране еще полвека назад имелись научные основы и благоприятные возможности перспективного развития информационных технологий с опорой на отечественный кадровый потенциал.

Однако именно в 1970-е гг. на правительственном уровне было принято решение сделать ставку на копирование западной компьютерной техники и программного обеспечения, что, по оценке многих отечественных и западных экспертов, крайне негативно сказалось на деятельности по дальнейшему развитию самостоятельных исследований. Это стало одной из основных причин существенного отставания СССР от Запада в сфере компьютерных технологий на рубеже 1990-х гг.

С распадом СССР внимание к этим проблемам на уровне государственной политики свелось к минимуму, заводы, производившие компьютерную технику, перепрофилировали производство на другие заказы или вовсе прекратили свою деятельность, а исследовательские группы предпочитали заниматься выполнением работ, ориентированных на игровые запросы потребителей компьютерных технологий.

Для фундаментальных научных исследований в сфере компьютерных технологий это был сложнейший период, связанный с сокращением или почти полным отсутствием государственного финансирования. Начался значительный отток специалистов в зарубежные страны, остановка многих перспективных исследований.

Запущенные в тот период процессы негативно сказывались на развитии ИТ-технологий, а также научных исследований в этой

сфере и в последующие десятилетия. При этом влияние некоторых из них с годами только усиливались.

Прежде всего это выражалось в постоянном ужесточении политики санкций. Компьютерная техника, являясь продукцией двойного назначения, всегда была сферой особого контроля над ее экспортом в Советский Союз и Россию со стороны США и других западных стран.

Последние две волны санкций в этой сфере связаны с событиями на Украине. Ограничения, введенные в 2014 г., имея существенные последствия для всей экономики России, заметно затронули и ИТ-отрасль. Если в 2013 г. общая годовая выручка российских ИТ-компаний составляла 33 млрд долл., то в 2018 г. она достигла только 22,6 млрд долл. При этом падение доходов в 2014–2016 гг. было еще сильнее, и только затем в течение двух лет удалось вернуться к положительным темпам роста выручки (2017 — +17 %, 2018 — +4 %) [4].

Увереннее чувствовали себя на рынке российские компании, производящие программное обеспечение. Это связано с достаточно высоким уровнем ряда таких компаний, создающих востребованные на мировом рынке продукты. Безусловным лидером в этой сфере является «Лаборатория Касперского», производящая антивирусное программное обеспечение. Даже после введения санкций и сокращения своих доходов от продаж в западных странах, выручка от продаж на рынках Северной Америки и Европы в 2018 г. составляла 50 % от общих доходов этой компании. В целом же выручка российских компаний, создающих компьютерные программы, несмотря на санкции, выросла с 11 млрд долл. в 2013 г. до 16,9 в 2018-м. И это свидетельствует о жизнеспособности этого сегмента российского рынка ИТ-индустрии.

Вместе с тем следует отдавать отчет в том, что нынешние позиции России на рынке ИТ не соответствуют ее экономическому и интеллектуальному потенциалу. Допущенные в предыдущие десятилетия провалы и отставания преодолеваются крайне медленно. Если в 2004 г. на долю России приходилось 0,7 % мирового рынка, то в 2016-м экспорт компьютерных, телекоммуникационных и ин-

формационных услуг составлял всего 0,8 %, а в 2018 г. он достиг лишь 1 % мирового рынка.

Для решения сложнейших задач с помощью суперкомпьютерных технологий требуется наличие в достаточном количестве самых высокопроизводительных вычислительных машин. Однако на этом направлении Россия заметно отстает от мировых лидеров. В TOP500 — рейтинге самых мощных вычислительных систем мира, опубликованном 30 мая 2022 г., Россия представлена только семью суперкомпьютерами, причем самый мощный из них занимает 22-ю позицию (по данным https://ru.wikipedia.org/wiki/Top500).

Таблица 1.3.1

Позиции суперкомпьютеров России в рейтинге TOP500 (июнь 2022 г.)

Позиция	R_{max} / R_{peak}, петафлопс	Принадлежность	Название	Год создания
22	21,530 / 29,415	Яндекс	Червоненкис*	2021
40	16,020 / 20,636	Яндекс	Галушкин	2021
43	12,810 / 20,029	Яндекс	Ляпунов	2020
46	11,950 / 14,909	Сбербанк	Кристофари Нео	2021
80	6,669 / 8,790	Сбербанк	Кристофари	2019
262	2,478 / 4,947	МГУ	Ломоносов-2	2018
318	2,258 / 3,012	МТС	МТС GROM	2021

* Червоненкис, Галушкин, Ляпунов — фамилии выдающихся советских и российских ученых. Кристофари — владелец первой сберегательной книжки в истории России.

В этом рейтинге с большим отрывом лидируют Китай (173 суперкомпьютера входят в TOP500) и США (127). За ними следуют Япония (34), Германия (31), Франция (22), Канада (14), Великобритания (12). После России располагаются Италия, Нидерланды, Саудовская Аравия, Бразилия и Южная Корея, которые представлены в TOP500 шестью суперкомпьютерами.

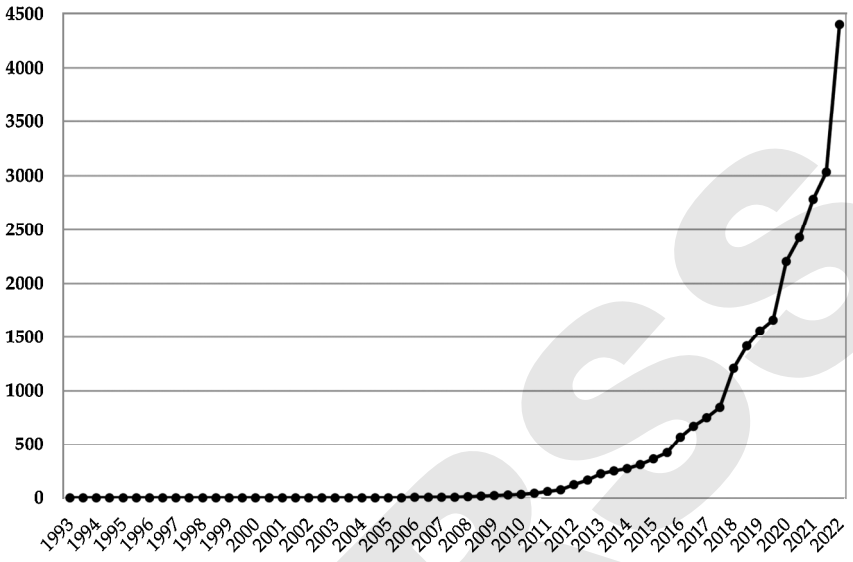


Рис. 1.3.1. Суммарная производительность входящих в TOP500 суперкомпьютеров мира, петафлопс (*по данным портала https://www.top500.org*)

Отметим, что если до начала 2000-х гг. превосходство США в гонке производительности суперкомпьютеров было практически абсолютным, то за последние 10 лет сильно прибавил Китай (рис. 1.3.2), несмотря на некоторые усилия США приостановить этот рост. Так, в 2015 г. компания Intel подала заявку на экспорт процессоров для обновления китайского суперкомпьютера «Млечный путь — 2», который на тот момент возглавлял список TOP500, однако правительство США заблокировало данную инициативу из-за опасения развития ядерной программы КНР. Тем не менее уже в 2016 г. в Китае был введен в эксплуатацию суперкомпьютер «Sunway TaihuLight», построенный на 260-ядерных процессорах собственного производства SW26010, причем он лидировал в списке с июня 2016 по июнь 2018 гг., демонстрируя производительность на тесте Linpack в 93,01 петафлопс.

Что касается России, то на рис. 1.3.3 показана доля России в суммарной производительности суперкомпьютеров из TOP500. Пики в 2011 и 2014 гг. обусловлены вводом в эксплуатацию суперкомпьютеров

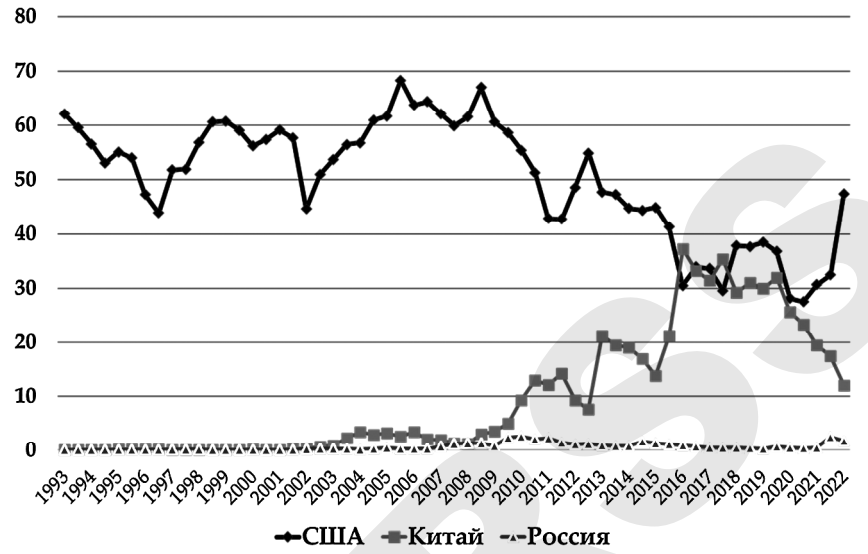


Рис. 1.3.2. Доля США, Китая и России в суммарной производительности суперкомпьютеров из TOP500, % (*по данным портала https://www.top500.org*)

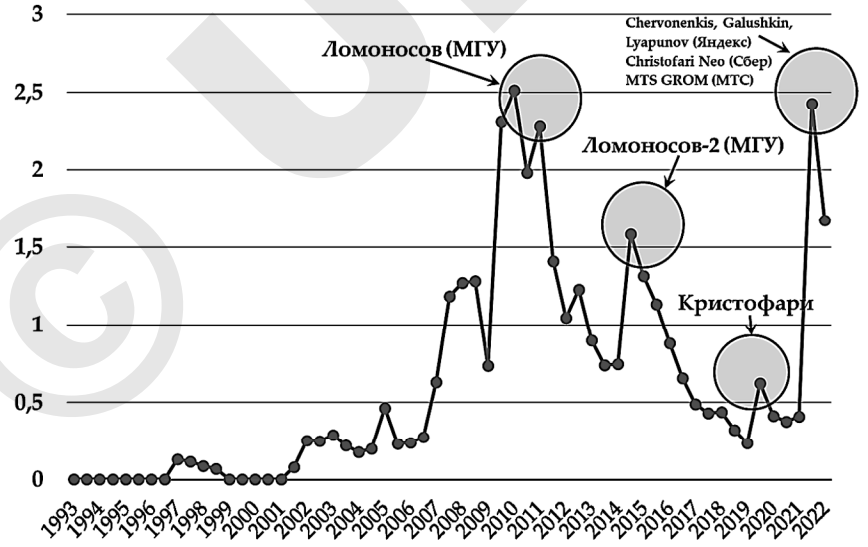


Рис. 1.3.3. Изменение по годам доли России в суммарной производительности суперкомпьютеров из TOP500, % (*по данным портала https://www.top500.org*)

«Ломоносов» и «Ломоносов-2». Затем некоторое время доля снижалась, но запуск суперкомпьютеров от Сбера и Яндекса в 2021 г. вернул позиции России практически до 2,5 %. В 2022 г. доля России в суммарной производительности суперкомпьютеров из TOP500 составила 1,67 %.

Развитие суперкомпьютерных технологий является приоритетным направлением модернизации экономики России, и в дальнейшем планируется значительное увеличение мощности суперкомпьютеров Московского государственного университета имени М. В. Ломоносова, что позволит дополнительно повысить рейтинг страны в списке TOP500.

При этом необходимо осознавать, что возникшее отставание от стран-лидеров в сфере суперкомпьютерного обеспечения, как и в целом в отрасли информационных технологий, невозможно решить наскоком или в одиночку, не имея доступа к ключевым составляющим развития этого рынка, включая все его технические, технологические и программные особенности. Зависимость России от внешних поставок в сфере ИТ очень высокая, в 2017 г. 46 % импорта приходилось на оборудование и аппаратуру, 35 % — на интеллектуальные решения. По оценке генерального директора ГК «Цифра» Игоря Богачева, доля импортного программного обеспечения, предназначенного для автоматизации производственных процессов, управления предприятием и решения других задач, в ряде отраслей российской промышленности достигает 90 % [5].

В условиях усиления политики санкций продвижение вперед на этом направлении существенно затруднено.

Ситуацию осложняет и человеческий фактор. В обеспечении рынка ИТ квалифицированным персоналом накладываются друг на друга несколько негативных тенденций. Общая из них связана с демографической проблемой 1990-х гг., когда в период 1993–2002 гг. коэффициент рождаемости снизился до самых низких значений в истории России, составляя по годам менее 10 на 1000 человек. В реальности это означает, что в период, начиная с 2010 г., рынок труда получает минимальное пополнение новыми работниками, причем

молодыми и наиболее открытыми к восприятию инноваций и современных информационных технологий.

При этом сфера ИТ, являясь одним из наиболее быстро растущих рынков в экономике России, постоянно испытывала дефицит квалифицированных кадров, получивших необходимое образование и имеющих опыт работы в отрасли.

Весьма существенной тенденцией на рынке труда ИТ-специалистов является его сверхконцентрация в Москве и Санкт-Петербурге, на долю которых, как следует из приведенных на рис. 1.3.4 данных, приходится 60 % всех резюме. Такое неравномерное распространение специалистов отрасли во многом отражает ее состояние в России, где все крупнейшие работодатели в этой сфере продолжают сосредотачиваться в столичных регионах, ограничивая возможности развития отрасли на периферии. Эту тенденцию подтверждает и тот факт, что в 2020 г. ИТ-компании Москвы сделали 3,2 млн приглашений на собеседования для иногородних соискателей вакансий.

Рис. 1.3.4. Распределение спроса на вакансии рынка труда в ИТ-сфере (*по данным https://habr.com/ru/company/hh/blog/656981/*)

Такое неравномерное распределение специалистов рынка ИТ-индустрии по территории страны создает диспропорции в осуществлении программ цифровизации российской экономики уже на начальной стадии их осуществления.

1.4. Реализация национальных программ и проектов РФ в интересах цифровой трансформации экономики страны

Цифровая трансформация экономики, как и любой процесс революционных преобразований, несет в себе высокий уровень неопределенности. Средством повышения устойчивости являются институты управления как фундаментальные факторы функционирования цифровой экосистемы. В роли институтов управления выступают организационные структуры, а также законодательные акты и программы. Они определяют правила и регулируют взаимоотношения экономических субъектов.

Практическим шагом в создании в нашей стране нового института управления стала принятая в 2017 г. Правительством Российской Федерации программа по созданию условий для перехода страны к цифровой экономике «Цифровая экономика Российской Федерации», ее выполнение было рассчитано до 2024 г.

На презентации этой программы на ежегодной конференции «Цифровая индустрия промышленной России» (ЦИПР) в мае 2017 г. возглавлявший тогда министерство связи и массовых коммуникаций Николай Никифоров привел в своем выступлении исходные данные, характеризующие положение России в сфере цифровых технологий. Наша страна занимала 38-е место в мире по использованию цифровых технологий для получения экономических и инновационных результатов. Лидерами рейтинга являлись Финляндия, Швейцария, Швеция, Израиль, Сингапур. Россия находилась на 41-м месте по индексу NRI, характеризующему готовность страны к цифровой экономике. В 2016–2017 гг. России отводилось 43-е место в рейтинге глобальной конкурентоспособности (рис. 1.4.1).

38-е место с точки зрения экономических и инновационных результатов использования цифровых технологий, отставая от стран-лидеров (Финляндия, Швейцария, Швеция, Израиль, Сингапур)	**41-е место** по готовности к цифровой экономике, согласно индексу сетевой готовности (NRI), со значительным разрывом от десятки лидирующих стран (Сингапур, Финляндия, Швеция, Норвегия, США)	**43-е место** согласно рейтингу глобальной конкурентоспособности 2016–2017 гг.
Согласно индексу I-DESI, опубликованному Европейской Комиссией в 2016 г., Россия отстала в развитии цифровой экономики от ЕС, Австралии и Канады, но опережала Китай, Турцию, Бразилию. По доступности фиксированной широкополосной связи Россия опережала ЕС и остальные страны. В отношении человеческого капитала Россия имела лучшие позиции, чем в среднем по ЕС, Турция, Мексика, Бразилия, но значительно отстала от Японии, Кореи, Швеции, Финляндии, Великобритании и лидирующих стран ЕС. В отношении частоты использования сети Интернет (в среднем ежедневно и регулярно) Россия продемонстрировала не очень высокие позиции в сравнении с ЕС, США, Новой Зеландией и Австралией, но опережала Китай, Бразилию, Мексику. В области внедрения цифровых технологий предприятиями Россия значительно отстала от ЕС и остальных стран, лишь немного опередив Турцию, Китай и Мексику		

Рис. 1.4.1. Положение России на глобальном цифровом рынке в 2017 г. (*по данным Минкомсвязи России, https://disk.yandex.ru/a/rmrsU69p3YWNFK/5b3439fcd98703df6312b1d9*)

В программе «Цифровая экономика Российской Федерации», утвержденной распоряжением Правительства РФ от 28 июля 2017 г. № 1632-р, были четко обозначены три ее основные цели:

- создание экосистемы цифровой экономики Российской Федерации, в которой данные в цифровой форме являются ключевым фактором производства во всех сферах социально-экономической деятельности и в которой обеспечено эффективное взаимодействие, включая трансграничное, бизнеса, научно-образовательного сообщества, государства и граждан;

- создание необходимых и достаточных условий институционального и инфраструктурного характера, устранение имеющихся препятствий и ограничений для создания и/или развития высокотехнологических бизнесов и недопущение появления новых препятствий и ограничений как в традиционных отраслях экономики, так и в новых отраслях и высокотехнологичных рынках;
- повышение конкурентоспособности на глобальном рынке как отдельных отраслей экономики Российской Федерации, так и экономики в целом [6].

Важным указанием в представлении программы было выделение трех уровней цифровой экономики, которые, тесно взаимодействуя, влияют на жизнь граждан и общества в целом:

- рынки и отрасли экономики (сферы деятельности), где осуществляется взаимодействие конкретных субъектов (поставщиков и потребителей товаров, работ и услуг);
- платформы и технологии, где формируются компетенции для развития рынков и отраслей экономики (сфер деятельности);
- среда, создающая условия для развития платформ и технологий и эффективного взаимодействия субъектов рынков и отраслей экономики (сфер деятельности) и охватывающая нормативное регулирование, информационную инфраструктуру, кадры и информационную безопасность.

Тем самым в программе на базовом нормативном фундаменте были определены подходы к формированию новой регуляторной среды, обеспечивающей благоприятный правовой режим для возникновения и развития современных технологий, а также для осуществления экономической деятельности, связанной с их использованием. В числе «аналоговых» основ развития наряду с нормативно-правовым регулированием в программе были выделены следующие направления: кадры, формирование исследовательских компетенций и технологических заделов, информационная инфра-

структура, информационная безопасность. По каждому из направлений были разработаны дорожные карты, содержащие конкретные задачи, ключевые события (вехи) и сроки исполнения.

После того, как в соответствии с майскими (2018 г.) указами Президента РФ В. В. Путина было положено начало формированию национальных проектов, в основу разработки нового проекта «Цифровая экономика Российской Федерации» легла уже действовавшая на тот момент программа, отвечавшая по подходам и срокам выполнения общим нормативным правилам составления национальных проектов.

Национальный проект в сфере цифровизации был утвержден как Национальная программа «Цифровая экономика Российской Федерации» протоколом заседания президиума Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам от 4 июня 2019 г. № 7 [6].

В ходе доработки программы ее состав видоизменялся и расширялся, в настоящий момент в нее включены 9 федеральных проектов:

- «Нормативное регулирование цифровой среды»;
- «Кадры для цифровой экономики»;
- «Информационная инфраструктура»;
- «Информационная безопасность»;
- «Цифровые технологии»;
- «Цифровое государственное управление»;
- «Искусственный интеллект»;
- «Обеспечение доступа в Интернет за счет развития спутниковой связи»;
- «Развитие кадрового потенциала ИТ-отрасли».

Первоначальный вариант структуры и финансирования программы включал 6 проектов, и на их реализацию предполагалось направить из бюджета в течение шести лет (2019–2024) 1080 млрд руб. Распределение планировавшегося финансирования представлено на рис. 1.4.2.

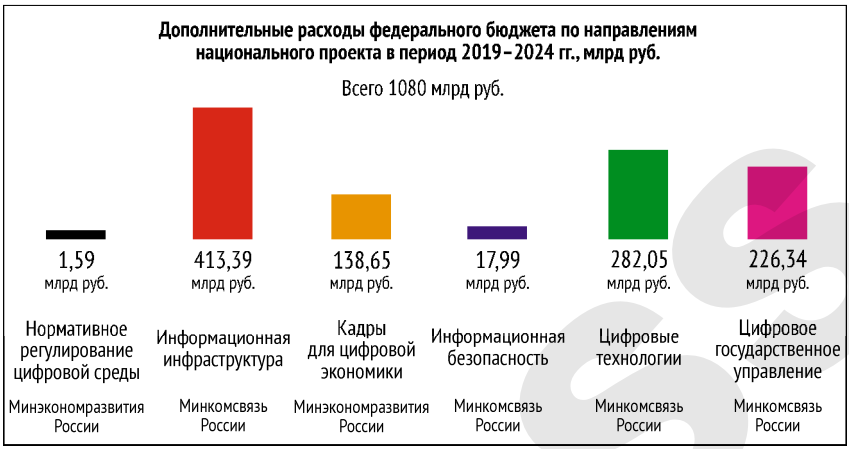


Рис. 1.4.2. Структура и финансирование нацпроекта «Цифровая экономика РФ» по планам 2018 г.

Дополнительно на финансирование национального проекта планировалось привлечь из внебюджетных источников свыше 535 млрд руб., поэтому общий бюджет проекта оценивался на сумму более 1,6 трлн руб.

Однако в течение нескольких лет программа «Цифровая экономика РФ» подвергалась существенным изменениям как по ее структуре, так и по объемам финансирования. Во многом по этой причине на первых порах она постоянно фигурировала в числе самых отстающих по освоению бюджетных ассигнований среди всех 13 национальных проектов. Так, за первое полугодие 2019 г. исполнение бюджетных расходов по программе составило только 8,3 %, по итогам 9 месяцев этого года — 12, 3 %. В целом же по итогам 2019 г. исполнение бюджетных расходов составило 73,3 %, тем самым более четверти бюджетных ассигнований не было освоено в установленный срок.

Руководители программы объясняли такое положение задержками в принятии нормативных актов, — что отражалось на поздних сроках подписания контрактов с исполнителями работ, — длительностью и сложностью процедур по размещению госзакупок. В после-

дующие годы исполнение требований по бюджетным расходам улучшилось — по итогам 2021 г. было освоено 95,8 % бюджетных ассигнований.

В результате проведенных корректировок на осуществление программы «Цифровая экономика Российской Федерации» из бюджета страны было уже выделено и предполагается в дальнейшем направить в общей сложности 952 млрд руб. (табл. 1.4.1), в том числе в 2022–2024 гг. около 580 млрд руб. По оценке заместителя министра цифрового развития, связи и массовых коммуникаций Н. С. Яцеленко, являющегося администратором проекта, это позволит достичь национального показателя «цифровая зрелость» и соответствовать выполнению Указа Президента РФ о национальных целях до 2030 г.

Таблица 1.4.1

Выделенные и планируемые затраты бюджета РФ на программу «Цифровая экономика Российской Федерации в 2019–2024 гг.

Год	Сумма, млрд руб.
2019	108,0
2020	108,6
2021	156,2
2022	199,4
2023	191,1
2024	188,8

Определенное сокращение бюджетного финансирования программы «Цифровая экономика» (более чем на 100 млрд руб. по сравнению с первоначальными планами) во многом связано с изменившимися макроэкономическими условиями, в которых оказалась Россия в последние годы. Сказалось глобальное влияние пандемии, заставившей в 2021–2022 гг. перераспределять бюджетные ресурсы внутри основных программ развития страны. При этом такое перераспределение происходило как между национальными

проектами, так и внутри самих проектов, когда по мере их реализации выяснялась необходимость сконцентрировать затраты на наиболее приоритетных направлениях.

Так, например, получил дополнительную поддержку федеральный проект «Цифровые технологии», из которого было выделено в отдельный федеральный проект направление «Искусственный интеллект».

Аналогичное перераспределение произошло с проектом «Кадры для цифровой экономики», который был дополнен новым федеральным проектом «Развитие кадрового потенциала ИТ-отрасли». Такое изменение стало признанием базового положения ИТ-отрасли в процессе цифровизации всей экономики страны, поэтому совершенствование подготовки кадров для успешного развития этой отрасли определяет темпы цифрового прогресса и во всех других отраслях.

Согласно данным Росстата, общее число специалистов в сфере информационно-коммуникативных технологий составляло в 2020 г. 1764,6 тыс. человек. По сравнению с 2019 г. (1665,5 тыс.), их число увеличилось почти на 6 %, что отражает динамику постоянно растущей общей численности специалистов этой сферы в России. Однако доля таких специалистов к общему числу занятых в России еще заметно уступает ведущим странам в этой сфере (рис. 1.4.3). Вместе с тем анализ возрастного состава специалистов ИКТ показывает, что в России половина из них моложе 35 лет, что отражает как растущее состояние этого рынка труда в стране, так и перспективы его дальнейшего развития за счет повышения квалификации работающих (рис. 1.4.4) [7].

Как следует из общих данных, представленных на рис. 1.4.5, в 2020 г. специальное образование среднего и высшего звена по научной области «Информационно-коммуникационные технологии» получили в России около 79 тыс. специалистов, при этом их доля в процентном соотношении по каждой из категорий выпуска (среднее, высшее, аспирантура) к общему числу выпускников немного уступает Эстонии, Финляндии, а по среднему образованию — Швеции и Италии, имеющим более долгую традицию развития ИКТ-отрасли в своих странах.

Рис. 1.4.3. Доля специалистов по ИКТ в общей численности занятых в ряде ведущих стран в сфере развития ИКТ (*по данным: Цифровая экономика: 2022: краткий статистический сборник*)

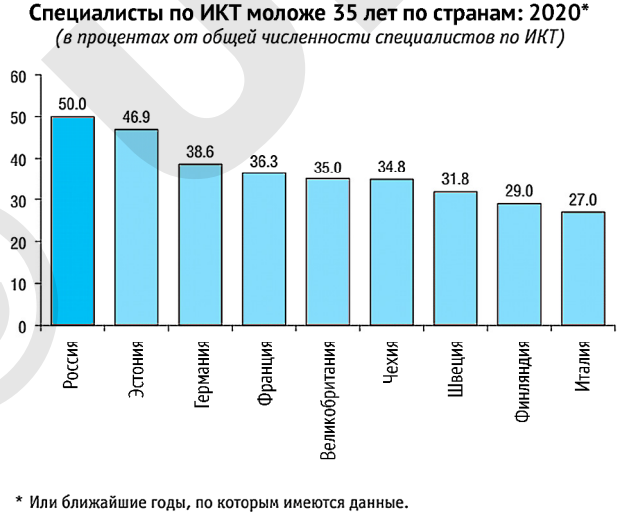


Рис. 1.4.4. Доля специалистов по ИКТ моложе 35 лет в общей численности специалистов этой отрасли (*по данным: Цифровая экономика: 2022: краткий статистический сборник*)

©URSS

Выпуск со средним профессиональным по программам подготовки специалистов среднего звена и высшим образованием по научной области «Информационно-коммуникационные технологии» по странам: 2020*

	Среднее профессиональное образование по программам подготовки специалистов среднего звена (МСКО 5)		Высшее образование по программам бакалавриата, специалитета, магистратуры (МСКО 6, 7)		Высшее образование по программам подготовки научно-педагогических кадров в аспирантуре (МСКО 8)	
	Тысячи человек	В процентах от общего выпуска	Тысячи человек	В процентах от общего выпуска	Тысячи человек	В процентах от общего выпуска
Россия	**35.8**	**6.5**	**42.2**	**5.0**	**0.9**	**6.6**
Великобритания	6.3	5.4	25.3	3.8	1.2	4.1
Германия	–	–	26.6	5.0	0.9	3.1
Италия	0.4	10.2	4.5	1.2	0.2	2.4
Канада	8.5	4.3	6.7	2.5	0.3	3.4
Республика Корея	8.3	5.0	19.3	4.5	0.5	3.4
США	36.3	3.6	132.9	4.5	2.0	2.7
Финляндия	–	–	3.9	7.0	0.1	7.8
Франция	6.3	2.9	20.8	3.7	0.7	5.2
Чехия	–	–	3.7	5.0	0.1	3.1
Швеция	1.0	11.5	2.0	3.2	0.2	5.3
Эстония	–	–	0.6	6.5	0.0	11.9

* По зарубежным странам – данные 2018 г.

Рис. 1.4.5. Данные по подготовке специалистов ИКТ в ряде ведущих стран отрасли (*по данным: Цифровая экономика: 2022: краткий статистический сборник*)

Базовым документом для управления Национальным проектом Национальная программа «Цифровая экономика Российской Федерации» является его Паспорт [8], включающий цели, задачи, планы мероприятий, ответственных должностных лиц за их осуществление и сроки выполнения работ. Паспорт включает в себя также отчетность о проделанной работе по каждому из федеральных проектов, находящихся в составе этой программы.

Как уже отмечалось, из-за постоянных изменений, вносимых в содержательную часть национального проекта, а также в выделяемые финансы из бюджета на его осуществление, достаточно трудно на данном этапе проследить всю систему выполнения намеченных

планов. Отчетность по наиболее сложным рубежам, которые были установлены при утверждении программы «Цифровая экономика РФ», сегодня для оценки общественности не представлена.

Например, планировалось, что внутренние затраты на развитие цифровой экономики за счет всех источников по доле в валовом внутреннем продукте страны увеличатся с базового показателя на конец 2017 г. в 1,7 % до 3,0 % по итогам 2021 г. Доля Российской Федерации в мировом объеме оказания услуг по хранению и обработке данных, которая на 31.07.2018 оценивалась в 0,9 %, должна была к концу 2021 г. достичь 2,0 %, а к концу 2024 г. составить уже 5,0 %. Ни по одному из этих показателей положительная динамика в отчетах по программе не была представлена.

В связи с этим необходимо отметить наличие серьезной критики к составителям самой программы в среде специалистов ИТ-отрасли [9, 10, 11]. Если суммировать эти критические оценки, то можно выделить следующие основные замечания.

1. Программа при ее составлении с самого начала не имела целостной концепции, в которой бы учитывалось влияние политических, экономических, технологических и других факторов, способных серьезно повлиять на ее реализацию, не было рассмотрено и предложено альтернативных вариантов по результатам ее выполнения.
2. В программе не отражена взаимосвязь с цифровыми составляющими других национальных проектов, нет четко прописанных каналов взаимодействия с другими программами, отсутствует возможность взаимного использования и софинансирования отдельных мероприятий в интересах улучшения реализации нескольких программ.
3. Были подвергнуты критике декларируемые в программе ее основные цели. В частности, цель увеличения внутренних затрат на развитие цифровой экономики, по мнению специалистов, не стимулирует повышение экономической эффективности от внедрения цифровых технологий, а отражает прежний экстен-

сивный подход к экономическому развитию. По оценке ряда экспертов, в принятой в 2017 г. программе «Цифровая экономика Российской Федерации» основные цели были сформулированы гораздо точнее и содержательнее.

4. Программа по многим планируемым направлениям и показателям не содержит объективных критериев точной оценки и измерений достигнутых результатов, что размывает ее эффективность и ответственность исполнителей.
5. Замена содержания паспортов национальной программы с пятиуровневого (проект — цель — задача — веха (контрольное событие) — мероприятие — результат), которое применялось на начальной стадии ее составления, на двухуровневое (задачи — результаты) лишает программу конкретики в управлении достижением намеченных результатов. Фактически оно нередко сводится к отчетности о проведенных мероприятиях без оценки их эффективности.
6. В программе не представлен механизм общественного контроля над ее выполнением, она в значительной степени носит закрытый характер, что затрудняет оперативный мониторинг оценки выполнения программы, имеющей огромное значение не только для развития экономики, но и всего общества.

К этому следует добавить, что в программе не нашло достаточного отражения решение проблемы цифрового неравенства регионов, на которое указывают специалисты, занимающиеся анализом рынка ИКТ в стране [12]. Согласно данным Международной лаборатории исследований проблем устойчивого развития Центра экономической географии и регионалистики Института прикладных экономических исследований РАНХиГС, в 2020 г. разрыв между регионами по доле домохозяйств, имеющих доступ в Интернет, был более чем двукратным: в Санкт-Петербурге — 85 %, в Ингушетии — 41,3 %; по доле населения, использующего Интернет для заказа товаров и услуг, — четырехкратным: в Ямало-Ненецком автономном округе — 71,9 %, в Ульяновской области — 15,8 %; по доле населения, использующего механизм получения государственных и муници-

пальных услуг в электронном виде, — двукратным: в Тыве — 94,8 %, в Магаданской области — 49,1 %.

Лидерами в освоении цифровых технологий и Интернета являются в России крупнейшие агломерации: Москва и Московская область, Санкт-Петербург, а также наиболее обеспеченные финансами Ханты-Мансийский и Ямало-Ненецкий автономные округа, где расположены основные районы газо- и нефтедобычи. В числе отстающих — регионы с преимущественно сельским населением и низкими доходами, такие как Тамбовская область, республики Мордовия, Ингушетия, Дагестан.

Между тем в национальной программе «Цифровая экономика Российской Федерации» это цифровое неравенство не учитывается, предложен единый подход ко всем субъектам РФ в создании цифровой инфраструктуры и обучении населения и представителей бизнеса навыкам работы с применением цифровых технологий. Исключение сделано только для Чукотского автономного округа, для которого был разработан план-график создания телекоммуникационной инфраструктуры магистральных каналов связи с выходом в единую сеть электросвязи Российской Федерации. В то же время запланировано приоритетное развитие условий для создания сетей связи 5G, обеспечивающих многократное увеличение скорости передачи данных и емкости сети по сравнению с 4G, не менее чем в 10 городах с населением более 1 млн человек, что в очередной раз покажет с лучшей стороны цифровые возможности этих городов по сравнению с остальными населенными пунктами страны.

В соответствии с федеральным проектом «Информационная инфраструктура», составной частью национальной программы «Цифровая экономика РФ», в первую очередь планируется обеспечить широкополосным высокоскоростным Интернетом социально значимые объекты на всей территории страны — больницы, школы, учреждения культуры. Это, в свою очередь, позволит решить проблемы доступа к Интернету всего населения. Но техническое решение проблемы доступа не снимает более сложной задачи примене-

ния Интернета для полноценного развития населения отдаленных территорий, использования людьми новых возможностей в интересах создания и ведения бизнеса, самообразования и освоения достижений культуры. И эту задачу призваны решать отнюдь не технические специалисты, а организации соответствующего направления и с высокой квалификацией кадров.

Примером комплексного решения подобных вопросов могут служить стратегии регионального развития. Впервые такой подход был применен при разработке Стратегии социально-экономического развития Санкт-Петербурга до 2030 г., впоследствии получившей продолжение на период до 2035 г. В действующем варианте «Стратегии 2035», утвержденном в декабре 2018 г. [13], цифровая трансформация экономики, повышение уровня информатизации общества и его готовности к инновационным изменениям, наряду с увеличением инвестиций в человеческий капитал и повышением эффективности фундаментальных и прикладных научных исследований, названы основными факторами, обеспечивающими переход Санкт-Петербурга к инновационному развитию.

Эти приоритеты остаются важнейшими во всей системе целей и задач социально-экономической политики Санкт-Петербурга на ближайшие годы и дальнейшую перспективу, определенную «Стратегией 2035». При этом задачи цифровой трансформации и использования информационных технологий занимают ведущее положение в решении наиболее актуальных проблем по всем основным направлениям развития города. В «Стратегии 2035» к ним относятся следующие задачи.

- Повышение уровня информатизации системы здравоохранения, развитие электронного здравоохранения и телемедицины.
- Развитие современной и безопасной цифровой образовательной среды.
- Повышение правовой, финансовой и цифровой грамотности населения.
- Активное внедрение в деятельность организаций культуры современных информационных технологий.

- Стимулирование развития интеллектуальных систем комплексного учета коммунальных энергоресурсов, обеспечение доступа потребителей к информации государственной информационной системы жилищно-коммунального хозяйства.
- Развитие информационных технологий и цифровой экономики.
- Активное использование инструментов сети Интернет для продвижения туристского продукта Санкт-Петербурга на целевых рынках и в целевых сегментах.
- Формирование единого информационного пространства по профессиональной ориентации граждан в сети Интернет.
- Внедрение современных информационных технологий в работу органов государственной власти.
- Совершенствование системы обеспечения информационной безопасности.

Такой комплексный подход к применению цифровых технологий во всех основных сферах жизнедеятельности позволяет рассматривать Стратегию социально-экономического развития города в качестве главного документа цифровой трансформации управления Санкт-Петербургом как единой социально-экономической системой, имеющей все предпосылки для реализации концепции «умного» города.

Именно для решения задачи этого и подобного уровней необходимо использование суперкомпьютерной техники и соответствующего программного обеспечения.

1.5. Социально-экономические системы как объект управления и моделирования их развития

Основным понятием, применяемым в социально-экономическом моделировании, является социально-экономическая система — сложная вероятностная динамическая система, охватывающая процессы производства, обмена, распределения и потребления материальных и других благ. Она относится к классу кибернетических систем, то есть систем управляемых. В качестве таких систем для моделирования могут быть выбраны отдельные страны и объедине-

ния стран, регионы, края и области, города, муниципальные образования и другие административно-территориальные единицы [14].

Вот уникальные свойства, которыми обладают такие системы.

- *Эмерджентность*, то есть наличие качеств, не присущих ни одному из составляющих систему элементу, взятому в отдельности, вне системы. Это свойство возникает как следствие установления связей между элементами системы. Поэтому такие системы необходимо исследовать и моделировать в целом.
- *Массовый характер экономических процессов*. Вследствие этого выводы о тенденциях в экономике могут основываться только на массовых исследованиях.
- *Изменчивость*. Параметры и структура системы могут активно изменяться под воздействием внешних факторов.
- *Включенность*. Мы не можем наблюдать и исследовать экономические процессы в чистом виде, так как такие системы являются открытыми.
- *Наличие субъективного фактора и фактора случайности*. Вследствие этого в таких системах могут происходить не всегда предсказуемые действия, поэтому экономические прогнозы и модели носят в основном вероятностный характер.

Кроме того, в данной системе действует и такой фактор, как сознательное начало, которое способно сочетать объективные и субъективные аспекты общественного развития и оказывать целесообразное воздействие на изменения. С этой точки зрения функционирование социально-экономической системы представляется в форме взаимодействия двух крупных подсистем — организационно-управленческой и ценностно-нормативной, конфликт между которыми основан на расхождении между культурно предписанными стремлениями и социально структурированными средствами их реализации. Возникающее при этом напряжение потенциально способно дезорганизовать устойчивое развитие системы.

Как показывает опыт, причиной кризисного состояния общества как социально-экономической системы является деформация

социальной структуры в целом, то есть изменения в различных сферах общества (экономической, политической, культурной), в системе ценностных ориентаций, а также в социальных институтах и социальной стратификации. Несмотря на всю сложность социально-экономической системы, ее центральным участником является человек, как ее субъект. Именно его потребности, требующие удовлетворения, «запускают мотор» экономики, а расширение и рост потребностей человечества — одна из движущих сил развития экономики, а следовательно и социально-экономической системы. Социально-экономические системы носят антропоцентрический характер, а основным их процессом является процесс производства и распределения материальных и духовных благ.

Субъекты и объекты социально-экономической системы взаимодействуют между собой путем выбора и обмена (рис. 1.5.1). Очевидно, что цель взаимодействия субъектов социально-экономической системы — улучшение качества жизни.

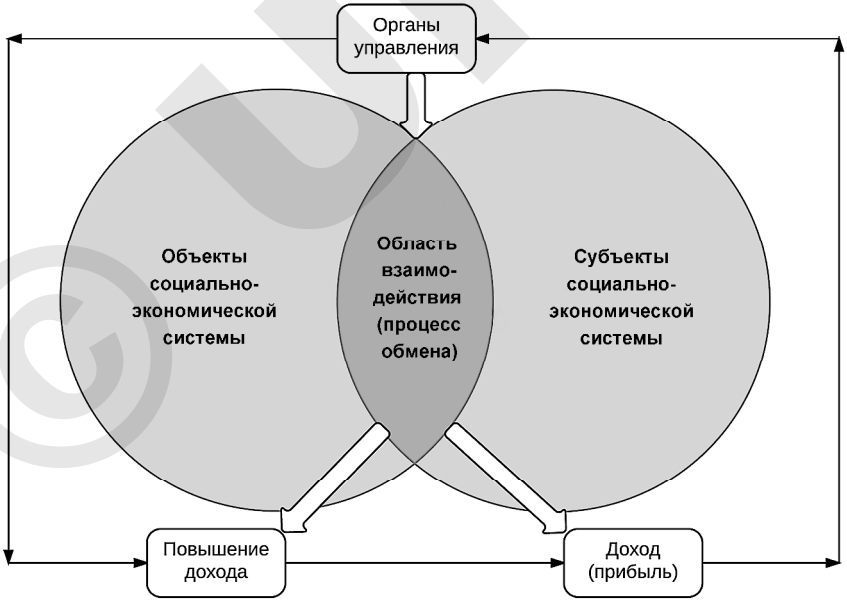


Рис. 1.5.1. Взаимодействие субъектов и объектов социально-экономической системы

Задачей органов управления является влияние на условия процесса взаимодействия с целью улучшения его качества. Например, предоставление больших возможностей для выбора, сокращение времени выбора. При этом, как показывает практика, методы воздействия органов управления на процесс взаимодействия могут быть правовыми, административными, прямыми и косвенными.

Правовые методы воздействия заключаются в установлении «правил экономической игры», то есть законодательной базы, которая определяет формы и права собственности, условия заключения контрактов и функционирования трудовых отношений, профсоюзов и работодателей и т. д.

Административные методы представляют собой комплекс обязательных требований и распоряжений. Например, меры по регулированию, лицензированию, квотированию и т. п.

Прямые методы воздействия базируются на властно-распределительных отношениях и могут быть сведены к административному воздействию на функционирование и результативность объектов. В частности, это целевое финансирование в виде субвенций и субсидий, включающих дотации, пособия, доплаты из специальных бюджетных и внебюджетных фондов общенационального и регионального уровня, а также льготные кредиты.

К числу косвенных методов воздействия можно отнести широкое информирование потребителей, материальные и моральные меры по стимулированию качества продукции и услуг, развитие обществ потребителей, создание свободных экономических зон и т. д.

Наиболее существенным объективным фактором, влияющим на условия процесса взаимодействия в социально-экономической системе, является характер экономического роста, ведь именно от него зависит увеличение количества объектов социально-экономического пространства, объема благ, а значит, увеличение возможностей для повышения качества жизни и дохода.

Поскольку социально-экономические системы являются антропоцентрическими, то они всегда несут в себе зачатки дезорганизации, которая сопровождается такими явлениями, как диспропорции,

нестабильность, неравномерность и прерывность развития. Поэтому задачей управления развитием данных систем является баланс организующего и дезорганизующего начала. Иными словами, каждое из возникающих изменений тут же уравновешивается другим, ему противоположным. Эта концепция динамического равновесия как оптимального режима функционирования системы оформилась в результате попыток разрешения важной дилеммы — соотнесения двух противоположных потребностей социально-экономической системы: развития и стремления сохранить стабильное состояние.

Две тенденции развития социально-экономической системы обусловливают возможность кризисных форм ее состояния, которые могут иметь два исхода: регенерацию или разрушение системы. Если дезорганизующий момент «перевешивает» силу организующих связей, это ведет к общему крушению существующей организационной формы, вследствие чего структура или преобразуется, или распадается.

Изучение проблемы соотношения организующего и дезорганизующего начал развития социально-экономических систем, раскрытие законов поддержания динамического равновесия и закономерностей перехода системы от одного состояния к другому позволяют усовершенствовать концепцию управления, необходимыми компонентами которой должны стать распознавание и поддержание самоорганизующихся сил и свойств систем. Познание закономерностей эволюционного развития систем и возможных путей их движения способствует разработке методологии социального прогнозирования — необходимого этапа в стратегии управления.

Анализ свойств сложных социально-экономических систем дает основание указать, что их развитие при подчинении природным и экономическим закономерностям приводит к стабильности. Однако в ходе своего развития системы подвергаются воздействию разнородных случайных факторов, а значит, они подвержены риску потери равновесного, устойчивого состояния.

Рассматривая развитие систем в терминах устойчивости, мы опираемся на определение, предложенное международным сооб-

ществом (ООН) и окончательно утвержденное в 1992 г. на конференции в Рио-де-Жанейро: **под *устойчивым развитием* понимается такое развитие, которое позволяет удовлетворить потребности нынешних поколений, не нанося при этом ущерба возможностям, оставляемым в наследство будущим поколениям для удовлетворения их собственных потребностей**.

В Российской Федерации с 1 декабря 2017 г. был введен в действие национальный стандарт ГОСТ Р 56548–2015/ISO/DIS/37101 «Устойчивое развитие административно-территориальных образований. Системы менеджмента. Общие принципы и требования», подготовленный Техническим комитетом по стандартизации ТК 115 [15]. В данном стандарте в разделе «Термины и определения» **устойчивое развитие** получило следующее определение: **«Развитие, отвечающее текущим потребностям и не ущемляющее возможностей будущих поколений с точки зрения удовлетворения собственных потребностей».** Это определение в полной мере соответствует подходу к устойчивому развитию, предложенному ООН.

Национальный стандарт ГОСТ Р 56548–2015/ISO/DIS/37101 предназначен для применения в административно-территориальных образованиях любых размеров, структуры и типа, на местном, региональном или национальном уровне. Он устанавливает требования к разработке, внедрению, обеспечению работоспособности и совершенствованию систем менеджмента, позволяющих сообществам использовать системный подход к устойчивому развитию.

Условие управляемости и контролируемости исключительно важно, поскольку устойчивое развитие отнюдь не является стабильно неизменным, наоборот, его темпы могут замедляться или ускоряться.

Можно выделить следующие основные характеристики данной концепции.

1. *Перспективность.* Развитие социально-экономической системы ориентировано на удовлетворение сегодняшних потребностей, однако при этом учитываются потребности и будущих поколений.

2. *Относительность пределов роста*. По мере развития технологий и социальной организации социально-экономическая система может преодолеть существующие ограничения в области эксплуатации природных ресурсов и улучшить состояние биосферы.
3. *Обязательность минимальных социальных стандартов*. Устойчивое и долговременное развитие невозможно без удовлетворения базовых потребностей всех людей.
4. *Сбалансированность потребления*. Для обеспечения устойчивого развития очень важно согласование потребностей всех субъектов социально-экономических систем.

С учетом международного опыта основные цели устойчивого развития социально-экономических систем следующие:

- улучшение качества жизни;
- гарантированное здоровье людей;
- удовлетворение основных жизненных потребностей как текущего населения, так и будущих поколений;
- борьба с бедностью;
- рациональная структура производства и потребления;
- рациональное природопользование;
- сохранение экосистем, защита климата и озонового слоя;
- обеспечение экологической безопасности;
- межсекторальное сотрудничество;
- экологическое сознание, экологическая этика;
- становление гражданского общества;
- устранение всех форм насилия над человеком и природой (предупреждение войн, терроризма и экоцида);
- глобальное партнерство.

В таком комплексе принципов социально-экономическая система выступает в качестве природосберегающей системы, объединяя в одно целое экологические, экономические и социальные аспекты. Таким образом, основными результатами устойчивого развития социально-экономических систем являются:

©URSS

- поддержание устойчивости экологических систем, достижение баланса между природной и искусственной средой;
- стабильные темпы экономического развития (то есть экономический рост);
- социальное развитие, то есть создание благоприятных условий жизнедеятельности.

В условиях перманентной неопределенности среды, в которой формируется устойчивость, можно выделить ряд рисков развития.

Социально-экономические риски

Господство «философии потребления». На протяжении многих веков человечество придерживалось «ресурсного» пути развития, таких его принципов, как «потребление ради процветания», «биосфера для человека», «человек — царь природы». Результатом этого стало истощение ресурсного потенциала, деградация окружающей природной среды и нарастание глобальных экологических проблем.

Высокая конкурентоспособность низкоэффективных технологий определяется приоритетом экономической выгоды и иллюзией неисчерпаемости ресурсного потенциала.

Неадекватность механизма ценообразования на природные ресурсы их истинной стоимости, а также динамике цен на ресурсы по мере использования возобновляемых и исчерпаемости невозобновляемых ресурсов. К недостаткам системы ценообразования относится и отсутствие системы выплат за сохранение не нарушенных в процессе хозяйственной деятельности территорий и обесценение «нересурсных» природных ценностей.

Проблема «Север — Юг». Суть ее заключается в специфике взаимоотношений между развитыми и развивающимися странами. Относительно низкий уровень цен на сырье и стоимость рабочей силы в развивающихся странах, с одной стороны, и высокий технологический и промышленный потенциал развитых государств — с другой, способствовали возникновению проблемы «Север — Юг». Результатом этого является диспропорция в уровнях экономического развития.

Экологические риски

Загрязнение окружающей среды, формирование новых техногенных геохимических зон, нарушение биогеохимических циклов как на глобальном, так и на локальном уровнях.

Техногенное нарушение целостности поверхности ландшафтов: обезлесивание, опустынивание, заболачивание и др.

Уменьшение видового разнообразия мира живых организмов.

Проблема качества продуктов питания и питьевой воды: накопление загрязняющих веществ в продуктах питания, поверхностных и подземных водах.

Возникновение экологической патологии: заболевания, генетические отклонения, сокращение продолжительности жизни и др.

Вероятностный характер экономических прогнозов и моделей, описывающих социально-экономические системы, приводит к появлению отклонения фактических значений от прогнозных, что само по себе делает проблематичным сохранение устойчивости и в модели, и в практике ее применения.

По оценкам аналитической компании «International Data Corporation» (IDC), общий объем данных в мире к 2025 г. достигнет 175 зеттабайт, что означает почти 150-кратный рост за 15 лет (1200 экзабайт в 2010 г.). Как полагают аналитики, наибольший вклад в этот огромный прирост внесут следующие сферы: Интернет (блоги, социальные сети и т. д.); финансы (биржевые индексы, аналитические материалы и т. д.); здравоохранение (данные о пациентах, лекарствах, способах лечения и т. д.); астрономия (детализированные изображения галактик); биоинформатика (данные о 3,3 млрд оснований нуклеиновых кислот в геноме человека, протеиновые последовательности и их анализ); библиотеки (текстовые данные, фото, карты и т. д.).

Необходимость обработки такого объема данных обусловливает появление аналитических систем нового поколения, включающих усовершенствованные методы вычислений, распознавания образов, организации хранилищ, сбора статистики с целью извле-

чения смысла из данных и получения информационного контекста. Ставка делается в том числе и на агент-ориентированные модели, относящиеся к классу моделей, основанных на индивидуальном поведении атомарных сущностей и создаваемых для компьютерных симуляций. Основная идея, реализуемая в агентной модели, заключается в построении вычислительного инструмента, представляющего собой совокупность агентов с определенным набором свойств и позволяющего проводить симуляции реальных явлений [16].

С помощью агент-ориентированных моделей можно смоделировать систему, максимально приближенную к реальности. Их появление следует трактовать как результат эволюции методологии моделирования: переход от мономоделей (одна модель — один алгоритм) к мультимоделям (одна модель — множество независимых алгоритмов). При этом агент в модели является автономной сущностью, имеющей, как правило, графическое представление, с определенной целью функционирования и возможностью обучения в процессе существования до уровня, определяемого разработчиками соответствующей модели.

В качестве примеров агентов могут рассматриваться: 1) люди (или иные живые организмы), роботы, автомобили и другие подвижные объекты; 2) недвижимые объекты, а также 3) совокупности однотипных объектов. В целом агентами в моделях могут быть любые наблюдаемые в реальной жизни объекты, однако основной задачей учета этих объектов в рамках модели является их корректная спецификация. Отметим, что общая особенность и одновременно главное отличие всех агентных моделей от моделей других классов — наличие в них большого числа взаимодействующих друг с другом индивидов.

В силу фактически экспоненциального роста данных обозначается дальнейший тренд развития агентных моделей — их построение с использованием суперкомпьютерных технологий (в том числе на базе геоинформационных систем). Это направление активно развивается, а на мировых конгрессах, посвященных агент-ориентированному моделированию, оно уже давно обсуждается не толь-

ко на специализированных сессиях, но и в рамках пленарных выступлений.

Актуальность использования суперкомпьютерных технологий для разработки агентных моделей обусловлена тем, что оперативная память обычного персонального компьютера не способна вместить то количество объектов программной среды, которое соответствует, например, населению земного шара или даже отдельных густонаселенных стран. Запуск оригинальной модели в специализированных средах для разработки агентных моделей с количеством агентов, превышающим несколько миллионов, уже приводит к исчерпанию оперативной памяти персонального компьютера.

Подобным же образом дела обстоят и с производительностью. Для пересчета состояния масштабной системы с нетривиальной логикой поведения и взаимодействия агентов требуются значительные вычислительные ресурсы, сопоставимые с потребностями вычислительных методов математической физики с аналогичным количеством расчетных ячеек. Но, в отличие от использования точных расчетов в математике и физике, поведение агентов включает элементы случайности, поэтому требуется провести целую серию расчетов и найти вероятностное распределение ключевых характеристик итогового состояния моделируемой среды.

Названные факторы обусловливают необходимость масштабных экспериментов с использованием суперкомпьютерных версий моделей, в которых популяция агентов распределяется по множеству узлов суперкомпьютера и расчеты выполняются параллельно. При этом возникает задача адаптации разрабатываемых в традиционных программных средах моделей для суперкомпьютеров.

Как и при создании суперкомпьютерных программ для решения многих физических задач, потенциал для распараллеливания многоагентных систем кроется в использовании локальности межагентного взаимодействия. В модели, как и в реальной жизни, большая часть взаимодействий происходит между субъектами, находящимися неподалеку. Это позволяет использовать подход распараллеливания «по пространству», то есть разместить популяцию

агентов на узлах суперкомпьютера наиболее равномерно и с учетом близости их географического положения. Таким образом, разбиение территории, на которой проживают агенты, на так называемые кварталы обеспечивает базовую возможность для распараллеливания задачи. Это наиболее часто встречающийся на практике подход для случаев, когда взаимодействие элементов моделируемой системы, будь то агенты в многоагентных системах или отдельные расчетные ячейки с усредненными параметрами моделируемой физической среды, удовлетворяет принципу пространственной локализации: связи и обмен данными имеют место преимущественно для элементов с близкими координатами и отрабатываются практически мгновенно в пределах каждого вычислительного узла.

Обратимся к зарубежным разработкам в области построения агент-ориентированных моделей с использованием суперкомпьютерных технологий, прежде всего к тем, которые направлены на моделирование социальных систем и процессов, связанных с их функционированием.

На основе анализа большого количества зарубежных научных публикаций были конкретизированы основные направления использования суперкомпьютерных технологий применительно к общественным наукам. Ниже перечислены наиболее цитируемые из них с указанием конкретных проектов.

- **Прогнозирование развития социально-экономических систем — стран, регионов, городов.**

  Примеры: проекты EURACE и FuturICT.

- **Воспроизведение исторических событий.**

  Примеры: 1) исследование гоминидов (Университет Цюриха); 2) исследование средневековых военных походов, например похода византийской армии на Манцикерт 1071 г. н. э. (Школа информатики Бирмингемского университета).

- **Моделирование миграционных процессов.**

  Примеры: проекты Pandora, EURACE и FuturICT.

- **Моделирование распространения эпидемий.**

  Примеры: 1) Глобальная агентная модель пандемии (Global-Scale Agent Model, GSAM) (Центр социальной и экономической динамики Брукингского института); 2) проект EpiFast (Институт биоинформатики Вирджинии).

- **Моделирование транспортных систем.**

  Примеры: 1) транспортный симулятор X10 (Токийская лаборатория компании IBM); 2) платформа POLARIS (Аргоннская национальная лаборатория).

- **Имитация и оптимизация пешеходного движения.**

  Пример: проект Университета Шеффилда.

- **Предсказание политических событий на основе сканирования новостей.**

  Пример: анализ контекстного содержания статей в Центре вычислительных наук Университета Теннесси.

- **Прогнозирование экологического состояния окружающей среды.**

  Примеры: EURACE; FuturICT; оценка Сценария национального планирования № 1 (National Planning Scenario 1, NPS1) — плана федерального правительства Соединенных Штатов на случай ядерной атаки.

  И т. д.

Отставание России на рынках программного обеспечения, где лидерами являются Microsoft, Google и другие компании, создает риски для реализации национального проекта «Цифровая экономика» на многих направлениях.

Применительно к разработке имитационных моделей социально-экономических систем можно отметить, что на данный момент существует несколько десятков специализированных пакетов (той или иной степени эффективности), но за редким исключением это все зарубежная продукция. Но даже в этих редких случаях

© URSS

средства имитационного моделирования были построены в программных средах зарубежного производителя. Такая же ситуация в смежных видах анализа данных для социально-экономических систем (эконометрические, статистические и нейросетевые пакеты).

Здесь хотелось бы сослаться на опыт наших зарубежных коллег из Китая, которые для аналогичных задач используют свое собственное программное обеспечение. С его помощью, к примеру, был разработан симулятор для прогнозирования социально-экономической динамики (Social Economic Dynamics, SED). Платформа SED охватывает более 100 стран, каждая их которых представлена совокупностью различных агентов (домашние хозяйства, фирмы, отрасли, банки, правительства и т. д.). С использованием собственного программного обеспечения проводится глубокий анализ динамики развития новых отраслей экономики в рамках четырнадцатой пятилетки КНР.

Авторы монографии посчитали возможным поделиться собственным немалым опытом разработки агент-ориентированных моделей с использованием суперкомпьютерных технологий для научного решения сложных экономических и социальных задач.

Литература к главе 1

1. Программа развития цифровой экономики в Российской Федерации до 2035 г. URL: https://aetp.ru/news/item/410256
2. Эволюция или революция: как технологии изменят мировой финансовый сектор? URL: https://tass.ru/pmef-2017/4274429
3. Искусственный интеллект разгромил чемпиона мира по игре в го. URL: https://lenta.ru/news/2016/03/15/gofinal/
4. Эксперты назвали самые изменившиеся из-за санкций отрасли экономики. Ограничения повлияли в первую очередь на сельское хозяйство и ИТ. URL: https://www.rbc.ru/economics/28/10/2019/5db1a76a9a794744a5d6e13a
5. Глава ГК «Цифра»: иллюзий быть не должно — иностранные вендоры не будут работать с компаниями под санкциями. URL: https://www.interfax.ru/interview/833644
6. Программа «Цифровая экономика Российской Федерации». С. 2. URL: http://static.government.ru/media/files/9gFM4FHj4PsB79I5v7yLVuPgu4bvR7M0.pdf

7. Цифровая экономика: 2022: краткий статистический сборник / Г. И. Абдрахманова, С. А. Васильковский, К. О. Вишневский и др.; Нац. исслед. ун-т «Высшая школа экономики». М.: НИУ ВШЭ, 2022. 124 с.
8. Паспорт национального проекта «Национальная программа „Цифровая экономика Российской Федерации“» (утв. президиумом Совета при Президенте РФ по стратегическому развитию и национальным проектам, протокол от 04.06.2019 № 7) // www.consultant.ru. URL: https://turov.pro/wp-content/uploads/2022/02/pasport_naczionalnogo_proekta_naczionalnaya_programma_czifro.pdf (дата обращения: 03.02.2022).
9. «Цифровая экономика». Как реорганизовать нацпрограмму, чтобы она заработала в полную силу. URL: https://www.cnews.ru/articles/2019–10–22_tsifrovaya_ekonomikakak_reorganizovat
10. Простые данные о больших числах, или как реализуется нацпрограмма «Цифровая экономика РФ»: успехи, провалы и бюджеты. URL: https://www.bigdataschool.ru/blog/digitalization-problems-and-challenges-2020.html
11. Национальные проекты: ожидания, результаты, перспективы. URL: https://raexpert.ru/researches/national_project_2020/
12. Цифровое неравенство в регионах России и возможные пути его преодоления. URL: https://www.researchgate.net/publication/355198503_Cifrovoe_neravenstvo_v_regionah_Rossii_i_vozmoznye_puti_ego_preodolenia
13. Закон Санкт-Петербурга «О Стратегии социально-экономического развития Санкт-Петербурга на период до 2035 г.» (с изменениями на 26 ноября 2020 г.). URL: https://docs.cntd.ru/document/551979680
14. *Окрепилов В. В.* Моделирование социально-экономических систем с использованием инструментов экономики качества // Экономика и управление. № 11, 2017. С. 8–20.
15. Национальный стандарт ГОСТ Р 56548–2015/ISO/DIS/37101 «Устойчивое развитие административно-территориальных образований. Системы менеджмента. Общие принципы и требования». https://docs.cntd.ru/document/1200124250 (с. 42)
16. *Макаров В. Л., Бахтизин А. Р., Сушко Е. Д., Васенин В. А., Борисов В. А., Роганов В. А.* Агент-ориентированные модели: мировой опыт и технические возможности реализации на суперкомпьютерах // Вестник Российской академии наук. 2016. Т. 86. № 3. С. 252–262.

Глава 2
Методология разработки агент-ориентированных моделей для применения суперкомпьютеров при исследовании экономических и социальных проектов

Современный мир в последние десятилетия не только показывает впечатляющую динамику по основным показателям, характеризующим уровень его развития, но и демонстрирует возрастающую сложность взаимодействия субъектов социально-экономических систем большинства стран. Численность населения Земли с 1960 г. (3,031 млрд чел.[1]) выросла к июлю 2022 г. в 2,6 раза (7,975 млрд чел.[2]), а к 2060 г., по прогнозам, превысит 10 млрд чел.[3] За счет развития транспорта, информационных и прочих технологий объем мировой торговли в процентах от мирового ВВП с 1960-х гг. увеличился с 25 % до 60 % к началу XXI в.[4] В 2020–2021 гг. он несколько снизился, но тенденция последнего полувека свидетельствует о возрастающей взаимозависимости экономик большинства стран мира.

При этом многими специалистами отмечается, что частота мировых кризисов, связанных с дисбалансами в экономике, в последние десятилетия также возросла. (К примеру, аналитиками Банка Германии (Deutsche Bank) было показано, что периодичность финансовых кризисов после ратификации Бреттон-Вудского соглаше-

[1] Данные Всемирного банка. URL: https://data.worldbank.org/indicator/SP.POP.TOTL

[2] Данные Отдела народонаселения Департамента по экономическим и социальным вопросам ООН. URL: https://population.un.org//

[3] Там же.

[4] Данные Всемирного банка. URL: https://data.worldbank.org/indicator/NE.TRD.GNFS.ZS

ния оказалась заметно выше, чем до нее[1].) К этому можно добавить участившиеся военные конфликты, гибридное противостояние ключевых мировых игроков и прочие события, последствия которых являются малопредсказуемыми, а мультипликативные эффекты весьма значимыми практически для всех стран.

В этой связи остро встает проблема поиска аналитических инструментов нового поколения, адекватных задачам современного сложного мира. До недавнего времени для определения взаимосвязей между социально-экономическими показателями в основном применялись методы многомерного статистического анализа, например регрессионные уравнения, но их спецификация осуществляется с использованием данных прошедших периодов, поэтому использование этих инструментов для прогноза хотя и возможно, но с долей условности, предполагающей определенный специалистами набор сценариев. Поэтому в условиях сильной флуктуации проходящих в социально-экономической сфере процессов данный инструмент малопригоден к использованию.

С 1990-х гг. благодаря широкому распространению персональных компьютеров стали применяться более продвинутые методы компьютерного моделирования социально-экономических процессов (оптимизационные модели, вычислимые модели общего равновесия, дискретно-событийные, системно-динамические, агент-ориентированные модели, нейронные сети, экспертные системы, генетические алгоритмы и т. д.). Некоторые из них имели достаточно хорошо проработанную теоретическую базу и даже примеры простой практической реализации, но реальное их использование стало возможным лишь благодаря возросшим вычислительным возможностям.

Таким образом, возрастающая сложность мира и последние глобальные потрясения (пандемия, разрыв производственных цепочек, торговые войны, политические катаклизмы и т. д.), с одной стороны, и колоссальная производительность современных систем —

[1] Портал Deutsche Bank Research. URL: https://www.dbresearch.com

с другой, несомненно, приведут к появлению искомых инструментов, способных оценивать мультипликативные эффекты с прямыми и обратными связями на социально-экономические системы большинства государств.

Отметим, что множество научных центров различных стран мира уже вовлечено в этот процесс, хотя такая возможность доступна преимущественно для крупных научных центров и международных организаций в силу ресурсоемкости (как научной, так и финансовой) этого вида деятельности. Вообще говоря, подобного рода модели, основанные прежде всего на равновесном подходе, уже имеются, и можно перечислить наиболее известные среди них: GTAP, MIRAGRODEP, MIRAGE, GLOBE, MULTIMOD, GEM, Global Macrofinancial Model, The Long Term Growth Model, Moody's Research Labs Inc. Model, WorldScan, LINK, WEFM, KPMG-MACRO, NiGEM. А более подробный обзор приведен в статье журнала «Вестник РАН» [1, 2]. Тем не менее, не умаляя их достоинств, отметим, что предсказывать будущие события они практически не способны — «черные лебеди» 2020 г. это хорошо показали.

Имея большой практический опыт в разработке множества агент-ориентированных моделей как инструментов для «выращивания» и изучения искусственных обществ, авторы монографии берут на себя смелость рассматривать модели этого класса в качестве потенциальных инструментов решения не только перечисленных выше, но и прочих проблем. При этом мы не утверждаем, что агент-ориентированное моделирование является долгожданной панацеей, поскольку существует множество других, возможно более эффективных для ряда задач аналитических инструментов, имеющих программную реализацию (например, нейронные сети и прочие методы машинного обучения), но, дополняя друг друга, эти методы могут существенно обогатить арсенал исследователей. К настоящему времени разработаны сотни специализированных программных средств для построения агентных моделей (NetLogo, RePast, MASON, AnyLogic и многие другие), но помимо этого есть готовые решения от ведущих ИТ-компаний (Microsoft, IBM, ESRI,

Wolfram Research и др.) по использованию средств разработки широкого профиля, реализующих модели этого класса.

Одна из задач данной монографии — пригласить читателя к дискуссии о развитии новых инструментов исследования окружающего нас мира и подвести промежуточные итоги в этом направлении с формулировкой следующих направлений развития агентного подхода. Авторы уже не раз публиковали научные труды, в которых давали характеристики и описывали успешные примеры реализации агент-ориентированных моделей [3, 4], поэтому здесь мы сосредоточимся только на отдельных моментах, показывающих преимущества подхода. Кроме того, это настолько бурно развивающееся направление (анализ крупнейших библиографических баз Web of Science и Scopus показал их прирост за период 2000–2021 гг. более чем в 120 раз), что любой обзор будет неполным.

Исторический экскурс

Трудно однозначно выявить момент зарождения агент-ориентированного подхода к моделированию наблюдаемых в реальной жизни процессов. Некоторые исследователи увязывают появление агентных моделей с клеточными автоматами 1940-х гг. Один из величайших математиков своего времени Джон фон Нейман, заложивший фундамент в архитектуру современных компьютеров (архитектура фон Неймана) и участвовавший в Манхэттенском проекте, а также занимавшийся вопросами самовоспроизводящихся систем, предложил концепцию клеточных автоматов, далее развитую его коллегой — Станиславом Уламом. За последующие десятилетия было разработано большое количество клеточных автоматов с самыми разнообразными правилами перехода между состояниями, написаны тысячи статей и книг по этой тематике[1]. Одной из наиболее известных среди этих моделей является игра «Жизнь», разработанная английским математиком Джоном Хортоном Конвеем в

[1] Данные библиографической и реферативной базы научных публикаций «Scopus». URL: https://www.scopus.com

1970 г., при этом не только ставшая классической, но и породившая огромное количество всевозможных вариаций [5].

Американский экономист и лауреат Нобелевской премии 2005 г., один из крупнейших специалистов в области теории игр Томас Шеллинг в 1971 г. в статье «Динамические модели сегрегации» предложил клеточный автомат, описывающий процесс сегрегации и показавший, что «черные» районы образуются вовсе не из-за расовой неприязни, а вследствие выбора людей, предпочитающих проживать по соседству с похожими на себя [6].

Клеточные автоматы оказали очень серьезное влияние на многие области науки (естественные, технические и гуманитарные) — астрономию, физику, медицину, геологию, математику, биологию, информатику, робототехнику, социологию, экономику и др., в связи с чем для обзора даже самых цитируемых моделей этого класса понадобится отдельная монография. Поэтому сошлемся на основной вывод научно-популярного бестселлера «Новый вид науки» («A New Kind of Science»), написанного более чем за 10 лет одним из известных современных математиков, разработчиком системы компьютерной алгебры «Mathematica» Стивеном Вольфрамом (Stephen Wolfram), и во многом посвященного клеточным автоматам. Автор приглашает к дискуссии о том, что накопленный к настоящему времени математический аппарат не позволяет понять сложность окружающего нас мира, а для его понимания, помимо совершенно новых подходов, необходимо проведение исследований, предполагающих осуществление многочисленных экспериментов, которые сами по себе имеют большой практический смысл и позволяют накапливать знания. В свою очередь, методология клеточных автоматов, по мнению ученого, позволяет рассмотреть системы любой степени сложности путем их дезагрегирования на простые элементы.

Отметим, что в этом убеждении Вольфрам не одинок. Так, создатель первого в мире языка программирования высокого уровня «Планкалкюль» (Plankalkül) и программируемого компьютера, немецкий ученый Конрад Цузе еще в 1969 г. опубликовал книгу

«Вычислительное пространство» («Rechnender Raum»), в которой бросил вызов сложившейся доктрине о непрерывности физических явлений, предположив дискретность всех наблюдаемых процессов. В этой связи он выдвинул тезис о том, что Вселенная представляет собой невообразимо сложный клеточный автомат с пока неизвестными правилами перехода между состояниями.

Отметим, что разработка и техническая реализация клеточных автоматов — далеко не разработанная сфера приложения исследователей, а, скорее, наоборот, с использованием высокопроизводительных вычислений это направление получит дополнительный импульс к дальнейшему развитию.

* * *

Один из основоположников агентного подхода в мире — директор Лаборатории агентного моделирования Нью-Йоркского университета, аффилированной с Курантовским институтом математических наук, Джошуа М. Эпштейн написал фундаментальную трилогию об агент-ориентированном моделировании. В первом томе «Growing artificial societies: Social science from the bottom up» [7] рассматривается ставшая классической модель Sugarscape, являющаяся симбиозом клеточного автомата и агент-ориентированной модели, и, несмотря на относительную простоту, она внесла серьезный вклад в развитие эпистемологии.

Второй том «Generative Social Science: Studies in Agent-Based Computational Modeling» [8] раскрывает потенциал агентного подхода применительно к самым различным научным сферам (социологии, биологии, экономике, эпидемиологии, археологии, конфликтологии и др.). «Растущие» искусственные общества первого тома эволюционировали во втором в более реалистичные практические приложения.

Третий том «Agent_Zero: Toward Neurocognitive Foundations for Generative Social Science» [9] стал квинтэссенцией многочисленных работ в области агент-ориентированного моделирования и вплотную подвел к пониманию того, что для разработки по-настоящему

реалистичных моделей и получения новых знаний агенты должны вырасти до сложных и противоречивых сущностей, обладающих целеполаганием и разумом, но также и подверженных страстям. Соответственно, перечень учитываемых в моделях параметров значительно расширяется от традиционных (уровень здоровья, дохода и др.) до более сложных, определяющих социальное поведение агентов. Один из посылов книги — «вырасти Раскольникова» — агента с девиантным поведением, значительно повышает уровень сложности формализации социальных и экономических процессов. Для этого была введена методология *Agent_Zero* (*агент нулевого уровня*), предусматривающая создание программных сущностей, характеризующихся не только разноплановыми параметрами, но и поведение которых моделируется с использованием современных представлений когнитивной нейробиологии.

* * *

Взрывное развитие информационных технологий способствовало прогрессу агент-ориентированного подхода со стороны самых разных сфер жизни, к примеру, кинопромышленности. Так, в начале 2000-х гг. был разработан программный продукт MASSIVE (Multiple Agent Simulation System in Virtual Environment — «Система многоагентного моделирования в виртуальной среде»), который может использоваться для одновременной обработки десятков, сотен тысяч и вплоть до миллионов объектов[1].

MASSIVE был первоначально разработан в процессе съемок фильма «Властелин колец». Режиссер сделал запрос на программное обеспечение, позволяющее реалистично отображать взаимодействие множества программных сущностей — виртуальных людей, растений и т. д. Основатель компании, предложившей продукт MASSIVE, Стивен Регелус при его разработке реализовал придуманную им в 1990-е гг. концепцию искусственных обществ, основанную на агент-ориентированном подходе.

[1] Сайт продукта MASSIVE. URL: http://www.massivesoftware.com

В то время это решение стало настоящим прорывом в киноиндустрии, поскольку используемые ранее технологии покадровой анимации уже не удовлетворяли запросам кинопродюсеров и зрителей. Для «Властелина колец» было сгенерировано около 200 000 агентов, каждый из которых был очень примитивным в плане поведения — по сути, задачи агентов заключались в следующем: оставлять ощущение непохожести на ближайших соседей по толпе, уклоняться от препятствий, следовать направлению общего движения и т. д. Как видно, никаких сложных действий в плане самостоятельного принятия решений и прочего не предусматривалось.

Стивен Регелус получил множество наград от организаций киноиндустрии за достижения в развитии технологий, легших в основу MASSIVE — одной из первых в отрасли систем, использующих автономных агентов для создания анимации. Главную награду — премию «Оскар» — Регелус получил за научные и инженерные достижения (Scientific and Engineering Award) и за автономную систему анимации агентов, используемую для боевых сцен в трилогии «Властелин колец»[1].

В фильме «Аватар» продукт был использован уже не только для отображения фауны, но и флоры. Список кинокартин, для производства которых был использован MASSIVE, включает большинство самых кассовых фильмов, среди которых: «Грань будущего», «Трон: наследие», «2012», «Гарри Поттер», «Пираты Карибского моря», «Кинг Конг», «Бен-гур», «Аквамен», «Обитель зла», «300 спартанцев», «Я, робот», «Годзилла», «Помпеи», «Игра престолов» и многие другие.

Конечно, как правило, над фильмом работает не одна компания, специализирующаяся на компьютерной графике, но MASSIVE — пример очень успешного применения агент-ориентированного подхода в киноиндустрии, и на данный момент более эффективного метода анимации географически распределенных батальных сцен еще не придумали.

[1] Раздел официального сайта американской Академии кинематографических искусств и наук, посвященный церемонии награждения в 2004 г. URL: https://www.oscars.org/sci-tech/ceremonies/2004

* * *

С середины 1990-х гг. агент-ориентированное моделирование проникло в самые различные научно-практические сферы и успешно применяется при решении множества задач. На основе анализа многих публикаций, посвященных агент-ориентированным моделям, можно обозначить несколько устоявшихся направлений их использования.

1. *Эпидемиология* (в первую очередь это оценка скорости распространения эпидемий с учетом различных ограничительных мер). Самая большая агент-ориентированная модель в мире (6,5 млрд агентов) была разработана в 2009 г. в США, первоначально для сценарных расчетов различных вариантов профилактических мероприятий в отношении вируса H1N1 (авторы — Джон Паркер и Джошуа Эпштейн).
2. *Пешеходное движение* (включая имитацию процессов эвакуации).
3. *Киноиндустрия* (так, в начале 2000-х гг. был разработан программный продукт MASSIVE, подробно описанный выше[1]).
4. *Демографические процессы* (использование агентного подхода позволяет не только более точно прогнозировать численность населения, но и оценивать эффективность различных мер государственной политики, направленной на решение демографических проблем; также модели этого класса хорошо себя зарекомендовали в вопросах прогнозирования миграционных процессов).
5. *Транспортные системы* (к настоящему моменту разработаны агентные модели транспортных систем для более 100 мегаполисов мира).
6. *Окружающая среда* (учитывая заметные климатические изменения и сформировавшийся тренд на декарбонизацию производств, модели этого класса активно применяются для мони-

[1] Раздел официального сайта американской Академии кинематографических искусств и наук, посвященный церемонии награждения в 2004 г. URL: https://www.oscars.org/sci-tech/ceremonies/2004

торинга и прогнозирования состояния окружающей человека среды. Более того, третий год подряд специалистами в области высокопроизводительных вычислений стран БРИКС проект «Цифровая Земля», учитывающий большое количество факторов (атмосферные, океанические изменения, биогеохимию океана, динамику ионосферы, эволюцию ледяного покрова) и реализуемый в том числе с использованием агентного подхода, обозначается как приоритетный).

7. *Землепользование* (это направление активно развивается, и ряд построенных моделей, в частности, позволяют учитывать влияние изменений режима землепользования на качество жизни населения).
8. *Городская динамика* (моделирование городских агломераций — более сложный процесс, нежели симуляции окружающей среды, поскольку такие модели должны включать в себя социальные, транспортные, экологические, экономические и другие системы; в силу трудоемкости построения таких инструментов разрабатываются они редко, но примеры есть — Гренобль, Лос-Анджелес, Вашингтон и др.).
9. *Реконструкция исторических событий* (при описании имевших место исторических событий часто не хватает подтверждающих документов, и возникающие пробелы интерпретируются историками во многих случаях с некоторой долей субъективизма, без убедительных доказательств. В качестве альтернативы представители Школы информатики Бирмингемского университета и исторического факультета Принстонского университета предлагают агент-ориентированный подход как инструмент детального воссоздания изучаемых событий, например похода византийской армии на Манцикерт (1071 г. н. э.)).
10. *Военные конфликты* (существует множество симуляторов, позволяющих моделировать военные операции и проигрывать различные варианты боевых действий с целью нахождения наиболее эффективной стратегии для заданной местности с учетом имеющегося арсенала и количества боевых единиц,

к примеру, продукт от компании «AgentFly Technologies» или проект «The System Effectiveness Analysis Simulation», финансируемый командованием военно-воздушных сил США).

11. *Социальные сети* (использование агентного подхода для моделирования и исследования социальных сетей открывает широкий спектр новых возможностей для изучения эволюции человеческого общества; возможно изучать скорость распространения информации, выявлять закономерности формирования социальных групп, продолжительность различных видов отношений между участниками и т. д.).
12. *Экономика* (в последние годы этот инструментарий прочно вошел в арсенал средств, используемых научными организациями, аналитическими центрами, корпорациями, органами государственного управления и другими учреждениями применительно к вопросам моделирования и прогнозирования экономических процессов; наиболее известными примерами являются модель рынка труда США с количеством агентов, равным 120 млн (автор — Роберт Экстелл), фондового рынка США (автор — Лебарон Блейк) и модель экономики стран ЕС — EURACE).

Ниже мы рассмотрим наиболее известные программные платформы для реализации агентных моделей на суперкомпьютерах.

2.1. Программные платформы для реализации агентных моделей на суперкомпьютерах

2.1.1. Пандора — высокопроизводительное программное обеспечение для построения агент-ориентированных моделей

В суперкомпьютерном центре Барселоны (Barcelona Supercomputing Centre) разработано высокопроизводительное программное обеспечение для построения крупномасштабных агентных моделей — Пандора (Pandora). Также Пандора обеспечивает полную поддержку геоинформационных систем в том случае, если для функционирования моделей необходима географическая привязка агентов.

Результат каждой симуляции хранится в иерархическом формате данных (Hierarchical Data Format, HDF), предназначенном для хранения большого количества цифровой информации. Этот формат поддерживается большинством ГИС.

Пандора дополняется программой Кассандра (Cassandra), позволяющей визуализировать процесс выполнения симуляции с использованием 2D и 3D графики.

Ниже перечислены ключевые особенности Пандоры и Кассандры:

- все возможности C++ для разработки, выполнения и анализа агентных моделей и клеточных автоматов;
- параллельное выполнение программного кода с использованием интерфейсов MPI/OpenMP (Message Passing Interface / Open Multi-Processing);
- высокая масштабируемость;
- поддержка ГИС;
- возможность разработки распределенных агентов, построенных с использованием технологий искусственного интеллекта.

Разработчики предлагают использовать Пандору для следующих исследовательских направлений:

- социальные явления, анализ социальных сетей и межличностные взаимодействия;
- сложные социально-экономические системы;
- миграционные процессы и др.

Механизм работы Пандоры отображен на рис. 2.1.1.1. После концептуального определения каркаса создаваемой модели ее первая реализация может быть разработана с помощью скриптов языка Python, а полученные результаты затем обрабатываются с использованием нескольких инструментов анализа. Если для реализации модели потребуются дополнительные аппаратные ресурсы, Пандора может конвертировать программный код в C++ версию, при этом автоматически его распараллелив. Таким образом, по заверению разработчиков, от пользователя не требуется дополнительных усилий по адаптации программного кода для узлов суперкомпьютера.

©URSS

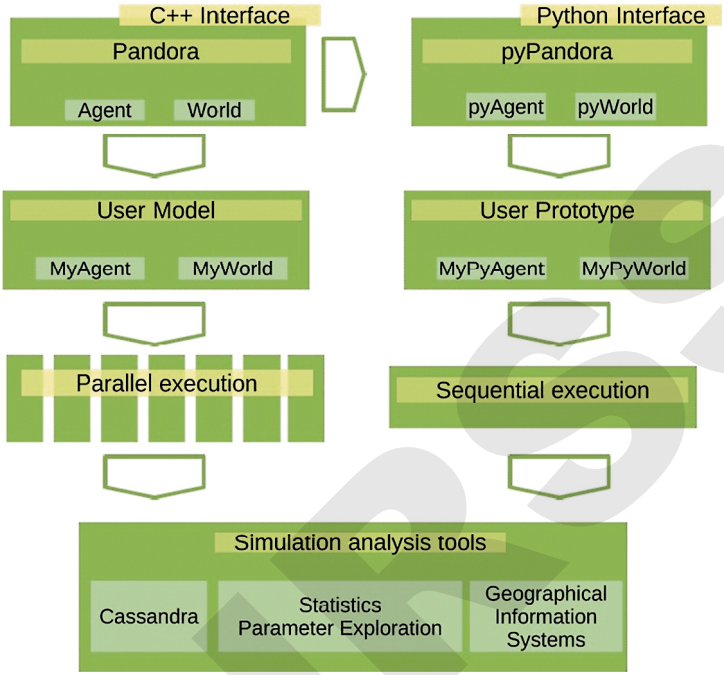


Рис. 2.1.1.1. Механизм реализации агентных моделей с использованием Пандоры. *Суперкомпьютерный центр Барселоны, https://www.bsc.es*

Среда функционирования агентов в Пандоре определяется как набор слоев, содержащих растровые карты, соответствующие стандартам ГИС. В процессе симуляции моделируемая среда, а также населяющие ее агенты равномерно распределяются по узлам суперкомпьютера.

Такой подход наилучшим образом решает проблему распределения вычислительной нагрузки при моделировании древних сообществ, в которых не было средств телекоммуникации.

Разработанные модели являются хорошо масштабируемыми, хотя есть проблема с синхронизацией действий агентов. В Пандоре она решается путем разделения каждого узла на четыре равные части (0, 1, 2 и 3). Программный код всех агентов, обрабатываемых в части 0, выполняется одновременно и без конфликтов, поскольку

эта часть узла не является смежной по отношения к аналогичной части другого узла (рис. 2.1.1.2). Полученные в результате выполнения кода результаты далее передаются для последовательного выполнения соседним частям (1, 2 и 3). После этого данные всех узлов синхронизуются, и выполнение модели переходит на следующий шаг.

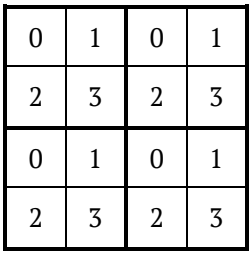


Рис. 2.1.1.2. Схема типового узла, поделенного на четыре раздела, последовательно выполняющих вычисления. *Суперкомпьютерный центр Барселоны, https://www.bsc.es*

Синхронизация в Пандоре осуществляется с помощью интерфейса MPI (Message Passing Interface) — программного интерфейса для обмена сообщениями между процессами, выполняющими задачу в параллельном режиме. Кроме того, директивы OpenMP используются для распределения вычислительной нагрузки по различным процессорам внутри узла. При анализе разных пакетов для разработки агентных моделей было выявлено, что наиболее ресурсоемкими участками кода являются моменты сбора информации и решения о порядке выполнения программных инструкций на следующем шаге. В Пандоре для одного процессора выполняемый шаг модели содержит в себе три составляющих метода. В рамках первого из них (updateKnowledge) агенты собирают информацию, но не могут изменять параметры среды и состояния других агентов. Второй метод (SelectAction) позволяет агентам принимать решения и генерировать последовательность дальнейших действий, но попрежнему не разрешает проводить какие-либо изменения. Наконец, во время третьего метода (updateState) агенты меняют свое состояние. В случае использования множества процессоров, Пандора равномерно распараллеливает между ними объем вычислений первых двух методов (наиболее ресурсоемких). Что касается про-

цедур третьего метода, то они выполняются последовательно с тем, чтобы избежать возможных конфликтов.

На первый взгляд разделение шага на три исполняемых метода кажется усложнением. Но в то же время реализация шага с помощью одного метода требует усилий по корректному определению последовательности выполнения различных процедур в рамках моделируемых сценариев (рис. 2.1.1.3). Таким образом, разделение шага на три метода позволяет снять эти проблемы.

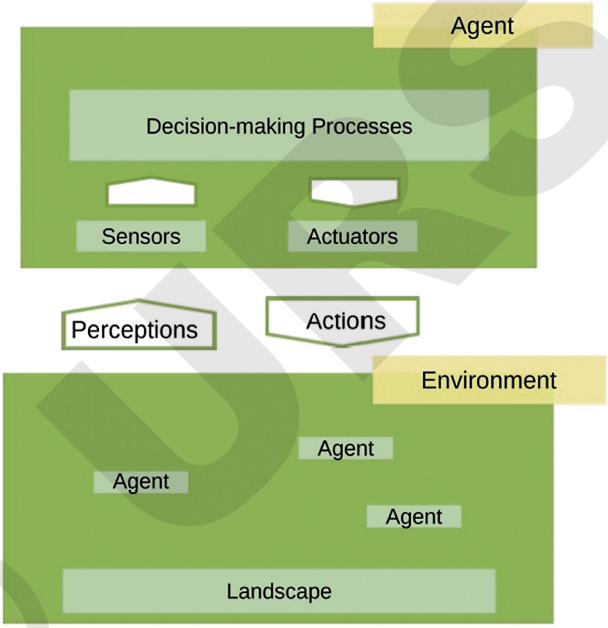


Рис. 2.1.1.3. Программная реализация агента в рамках цикла. *Суперкомпьютерный центр Барселоны, https://www.bsc.es*

Более подробно про проект можно прочитать здесь:

https://www.bsc.es/research-and-development/software-and-apps/software-list/pandora-hpc-agent-based-modelling-framework

Дистрибутив Пандоры можно скачать отсюда:

https://github.com/xrubio/pandora

2.1.2. RepastHPC — среда проектирования агентных моделей для суперкомпьютеров

В Аргоннской национальной лаборатории (рис. 2.1.2.1) разработано программное обеспечение (ПО) для проектирования агентных моделей с целью последующего запуска на суперкомпьютерах — Repast for High Performance Computing (RepastHPC). Данный пакет реализован с использованием языка C++ и MPI, а также библиотеки Boost, расширяющей C++.

В рамках RepastHPC реализован динамический дискретно-событийный планировщик выполнения программных инструкций с консервативными алгоритмами синхронизации, предусматривающими задержку процессов для соблюдения определенной очередности их выполнения.

В RepastHPC агенты распределяются между процессами, и каждый процесс связан с агентом, являющимся *локальным* по отношению к данному процессу. В свою очередь, агент локален к процессу,

Рис. 2.1.2.1. Суперкомпьютер IBM Blue Gene/P в Аргоннской национальной лаборатории, на котором многократно тестировался RepastHPC. *Аргоннская национальная лаборатория, CC BY-SA 2.0, https://www.anl.gov*

выполняющему программный код, описывающий поведение данного агента. При этом копии остальных — *нелокальных* — агентов могут присутствовать в любом процессе, что позволяет агентам всей модели взаимодействовать с этими копиями. К примеру, пусть пользователь в своей модели, предполагающей параллельные вычисления, использует два процесса — P1 и P2, каждый из которых создает определенное количество агентов и имеет собственный планировщик выполнения программных инструкций. Агенты, поведение которых рассчитывается на процессе P1, являются локальными по отношению к данному процессу, и только в рамках данного процесса программный код может изменить их состояние (аналогично и для процесса P2). Предположим, процесс P1 запрашивает копию агента A2 из процесса P2. Агент A2 не является локальным по отношению к процессу P1, и соответственно программный код, выполняемый в рамках процесса P1, не может изменить состояние агента A2. При этом агенты, реализуемые в рамках процесса P1, при необходимости могут запросить состояние агента A2, но копия A2 останется неизменной. Изменение оригинального A2 возможно только в рамках процесса P2, но в этом случае RepastHPC синхронизирует изменения состояния агента между всеми процессами.

Пример агентной модели в RepastHPC

Как правило, для реализации агентной модели в рамках RepastHPC требуются: 1) классы объектов типа «Агент» (листинг 2.1.2.1); 2) класс верхнего уровня — Model; 3) функция main.

Листинг 2.1.2.1. Пример интерфейса класса «Агент»

```
...
class Agent {
public :
    virtual ~Agent () {}
    virtual AgentId & getId () = 0;
    virtual const AgentId & getId () const = 0;
};
...
```

В листинге 2.1.2.2 создается конструктор класса с единственной переменной **_state** типа **int**, которая определяет состояние агента. Естественно, в моделях агенты обычно имеют более сложную структуру.

Листинг 2.1.2.2. Конструктор класса, определяющего состояние агента

```
...
class ModelAgent : public repast :: Agent {
private :
    repast :: AgentId _id ;
    int _state ;
public :
    ModelAgent ( repast :: AgentId id , int state );
    virtual ~ ModelAgent ();
    int state () const {
        return _state ;
    }
    void state (int val ) {
        _state = val ;
    }
    repast :: AgentId & getId () {
        return _id ;
    }
    const repast :: AgentId & getId () const {
        return _id ;
    }
    void flipState ();
};
...
```

Далее создается структура, позволяющая скопировать агента от одного процесса к другому (листинг 2.1.2.3).

© URSS

Листинг 2.1.2.3. Пример структуры, осуществляющей копирование агента между процессами

```
...
struct ModelAgentPackage {
    friend class boost :: serialization :: access ;
    template < class Archive >
    void serialize ( Archive & ar , const unsigned int version ) {
        ar & id;
        ar & state ;
    }
    repast :: AgentId id;
    int state ;
    repast :: AgentId getId () const {
        return id;
    }
};
...
```

Класс верхнего уровня Model нужен для начальной инициализации модели и для старта симуляции. Как правило, в рамках данного класса осуществляется распределение агентов по процессам, а также создается среда для функционирования агентов (решетка или непрерывное пространство). Кроме того, происходит инициализация данных для модели и планирование расписания для синхронизации. В листинге 2.1.2.4 приведен пример определения класса Model.

Листинг 2.1.2.4. Инициализация модели (пример класса Model)

```
...
class Model {
private :
    int rank ;
public :
    repast :: SharedContext < ModelAgent > agents ;
```

© URSS

```
    repast :: SharedNetwork < ModelAgent , ModelEdge >* net ;
    repast :: SharedGrids < ModelAgent >:: SharedWrappedGrid * grid ;
    repast :: DataSet * dataSet ;
    Model ();
    virtual ~ Model ();
    void initSchedule ();
    void step ();
}
...
```

В листинге 2.1.2.5 создается конструктор класса Model. Сначала определяется ранг процессов, в рамках которых выполняются симуляции, а затем в качестве примера создаются 4 агента с уникальным номером и определением обслуживающего процесса. Также определяется среда для взаимодействия агентов — решетка размерностью 40 × 60 — и назначается источник данных.

Листинг 2.1.2.5. Пример конструктора класса Model

```
...
const MODEL_AGENT_TYPE = 0;
Model :: Model()
{
    rank = RepastProcess :: instance() ->rank();
    for (int i = 0; i < 4; i++) {
        AgentId id(i, rank , MODEL_AGENT_TYPE);
        agents.addAgent (new ModelAgent (id, rank));
    }
    net = new SharedNetwork < ModelAgent , ModelEdge > ("network", true);
    agents.addProjection(net);
    grid = new SharedGrids < ModelAgent >:: SharedWrappedGrid ("grid",
        GridDimensions (Point <int> (40, 60)),
        std :: vector <int>(2, 2), 2);
    agents.addProjection(grid);
    NCDataSetBuilder builder ("./output/data.ncf", RepastProcess
        :: instance () -> getScheduleRunner(). schedule());
    StateSum * sum = new StateSum (this);
```

© URSS

```
    builder . addDataSource ( repast :: createNCDataSource ("state_sum",
        sum, std :: plus <int>()));
    dataSet = builder . createDataSet ();
    Provider provider (this);
    AgentsCreator creator (this);
    grid -> synchBuffer < ModelPackage >(agents, provider, creator);
}
...
```

В листинге 2.1.2.6 приведен пример инициализации планировщика. Как видно, остановка предусмотрена на 2000-м шаге, при этом на каждом шаге вызывается метод **step**, а также осуществляется изменение состояния агентов и запись данных.

Листинг 2.1.2.6. Инициализация планировщика модели

```
...
void Model :: initSchedule () {
    ScheduleRunner & runner = RepastProcess :: instance () ->
        getScheduleRunner ();
    runner.scheduleStop (2000) ;
    runner.scheduleEvent (1, 1, Schedule :: FunctorPtr (new
        MethodFunctor <Model > (this , & Model :: step)));
    runner.scheduleEvent (1.1 , 1, Schedule :: FunctorPtr (new
        MethodFunctor < DataSet > (dataSet , & DataSet :: record)));
    Schedule :: FunctorPtr dsWrite = Schedule :: FunctorPtr (new
        MethodFunctor < DataSet > (dataSet , & DataSet :: write));
    runner.scheduleEvent (25.2, 25, dsWrite);
    runner.scheduleEndEvent (dsWrite);
}
...
```

И, наконец, функция main обычно осуществляет инициализацию MPI в рамках RepastHPC, создает класс Model и запускает планировщик модели (листинг 2.1.2.7).

© URSS

Листинг 2.1.2.7. Пример функции main

```
...
int main (int argc, char ** argv ) {
    mpi :: environment env(argc, argv );
    std :: string config = argv [1];
    std :: string propsfile = argv [2];
    try {
        RepastProcess :: init (config);
        Properties props (propsfile);
        repast :: initializeRandom (props);
        Model model ();
        model.initSchedule ();
        ScheduleRunner & runner = RepastProcess :: instance () ->
            getScheduleRunner ();
        runner.run ();
    } catch (std :: exception & ex) {
        std :: cerr << "Error while running the rumor model : " <<
            ex. what () << std :: endl ;
        throw ex;
    }
    RepastProcess :: instance () ->done ();
    return 0;
}
...
```

Более подробно с данным ПО и руководством пользователя можно ознакомиться по адресу: https://repast.github.io/repast_hpc.html, где также можно скачать версию пакета 2.3.1 от 21 октября 2021 г.

2.1.3. EpiFast — эффективный инструмент для агентного моделирования эпидемий на системах с распределенной памятью

Специалисты исследовательской группы из Института биоинформатики Вирджинии разработали инструмент для изучения особенностей распространения инфекционных заболеваний — EpiFast, к достоинствам которого можно отнести масштабируемость и высокую

скорость исполнения. В рамках реализованного алгоритма болезнь распространяется стохастически, а масштабируемость осуществляется за счет использования программной парадигмы «ведущий — ведомый», которую мы рассмотрим ниже.

EpiFast — инструмент для высокопроизводительных вычислений на суперкомпьютерах, который может быть использован для оценки последствий различного рода решений в области борьбы с инфекционными заболеваниями, с учетом различных топологий сетей контактов между агентами моделей [10].

По заверениям разработчиков EpiFast за счет применяемых эффективных алгоритмов распараллеливания работает на порядок быстрее аналогичных программ. К примеру, имитация социальной активности (в течение 180 дней) жителей Большого Лос-Анджелеса (агломерации с населением свыше 16 млн человек) с 900 млн связей между людьми на кластере с 224 двухъядерными процессорами POWER5 заняла менее 15 минут. Такая высокая производительность связана с механизмом распараллеливания, предложенным авторами.

A. Концептуальное описание модели

В рамках модели рассматриваются:

1. агенты — люди, каждый из которых представлен подробным расписанием действий на день с учетом передвижений;
2. болезнь с характеристиками по отношению к агентам модели (восприимчивость, способность к выздоровлению и к заражению других людей);
3. меры по борьбе с болезнью.

A.1. Контакты

Контакты между людьми образуют сеть, которая называется «социальная сеть контактов». Она определяется как $G(V, E, w)$ — направленный граф с взвешенными ребрами. Узлы графа — это люди, ребра — осуществляемые в течение дня контакты, а веса — их продолжительность. Ребро (u, v) с весом $w(u, v)$ означает, что узел (человек) u контактирует с узлом v с продолжительностью $w(u, v)$, и во

время взаимодействия возможно заражение с вероятностью $p(w(u, v))$. В большинстве случаев ребра являются двунаправленными с одинаковыми весами для каждого направления. По предположению разработчиков, сама сеть контактов G_t во все периоды работы модели t не меняется, но может изменяться при проигрывании различных сценариев карантина для оценки последствий эпидемии.

A.2. Агенты

Каждый агент может находиться в одном из четырех состояний SEIR (S — восприимчивый к болезни (susceptible), E — инкубационный период (exposed), I — инфекционная болезнь (infectious), R — вылечившийся (recovered)). Обозначим $\theta(v, t)$ состояние здоровья агента v в момент времени t. Инкубационный период для агента v — $\Delta t_E(v)$, а период болезни — $\Delta t_I(v)$.

Состояние $\theta(v)$ агента v в процессе всей эпидемии может быть эквивалентно представлено посредством следующего набора переходов $\tau(v) = \left(t_{S\to E}(v), t_{E\to I}(v), t_{I\to R}(v)\right)$, где $t_{S\to E}(v)$ — день, когда для агента наступил инкубационный период, $t_{E\to I}(v) = t_{S\to E}(v) + \Delta t_E(v)$ — день, когда агент заболел, а $t_{I\to R}(v) = t_{E\to I}(v) + \Delta t_I(v)$ — день, когда агент выздоровел. При этом периоды $\Delta t_E(v)$ и $\Delta t_I(v)$ для каждого агента предполагаются известными до начала симуляции. Состояния SEIR приведены на рис. 2.1.3.1. Как видно, переход между состояниями является односторонним и только в виде единственной возможной последовательности.

Внутри социума инфекция распространяется от уже инфицированного к восприимчивому к болезни агенту модели. В любой из дней,

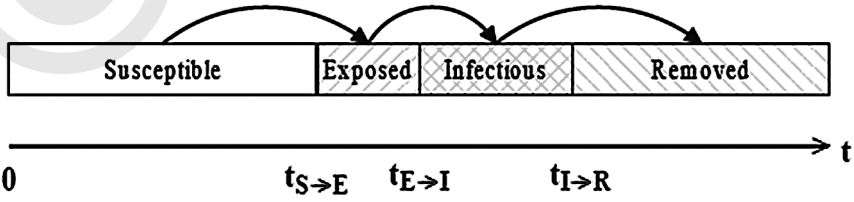


Рис. 2.1.3.1. SEIR состояния агентов. *EpiFast: A Fast Algorithm for Large Scale Realistic Epidemic Simulations on Distributed Memory Systems* [10]

если узел u инфицирован, а узел v восприимчив, то болезнь может быть передана с вероятностью $p\big(w(u,v)\big) = 1 - (1-r)^{w(u,v)}$, где r — вероятность передачи инфекции в единицу времени во время контакта.

Рис. 2.1.3.2 показывает, как болезнь распространяется через социальную сеть из четырех узлов и пяти ребер и как меняются их состояния.

Предположим, что для каждого узла $\Delta t_E = 0$ и $\Delta t_I = 2$, а вероятность передачи инфекции между инфицированным узлом (красного цвета) и восприимчивым узлом (зеленого цвета) равняется 0,5. Узел A становится инфицированным при инициализации модели и сразу заражает узел B, но в то же время узел D не успевает от него заразиться. На следующий день оба узла A и B заражены, причем узел B еще и заражает C, а D заражается от узла A (хотя потенциально мог заразиться от узла B). Других восприимчивых к инфекции агентов больше нет, и рассматриваемый социум окончательно выздоравливает к окончанию четвертого дня.

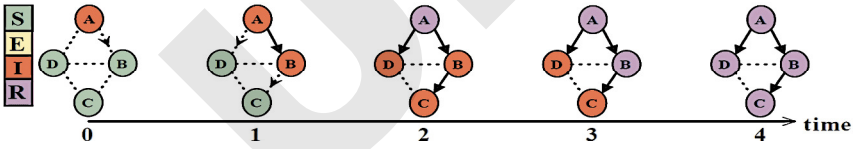


Рис. 2.1.3.2. Пример распространения инфекции через простую социальную сеть. *EpiFast: A Fast Algorithm for Large Scale Realistic Epidemic Simulations on Distributed Memory Systems* [10]

B. Алгоритм

В листинге 2.1.3.1 приведена концептуальная версия алгоритма EpiFast, предполагающая последовательное исполнение.

Листинг 2.1.3.1. Алгоритм EpiFast: последовательная версия

Вход: $(G, \Delta\tau, A, \Lambda)$, где G — социальная сеть, $\Delta\tau$ — продолжительность инкубационного и инфекционного периодов, A — совокупность мероприятий по борьбе с эпидемией, а Λ — набор начальных условий.

Выход: (τ, П), где τ — количество переходов между состояниями здоровья агентов, а П — количество агентов, заразивших других людей.

```
1 инициализация модели
2 for t = 0 to T do
3   foreach v ∈ V do
4   обновляется состояние здоровья агента в соответствии с τ(v)
5   применяются меры по борьбе с эпидемией (A_t), в том числе обновление G
6   foreach инфицированного узла u ∈ V do
7       foreach контакта v узла u do
8           if v восприимчив к инфекции then
9               возможна передача инфекции от u к v
10              с вероятностью p(w(u, v))
11              if передача инфекции от u к v произошла then
12                  обновление τ(v) и π(v)
13 вывод (τ, П)
```

Для распараллеливания данного алгоритма необходимо разделить данные и объем вычислений. Основным объемом данных является социальная сеть, и в этой связи главная проблема при распараллеливании — разделить ее на части. Как правило, для реализации социума на группе процессоров используют несколько естественных способов. К примеру, один из них базируется на представлении социума в виде набора неструктурированных связей между индивидами. При распараллеливании эти связи равномерно делятся на группы. Минус такого подхода заключается в сложности отслеживания статуса отдельного человека, то есть если, к примеру, индивид v инфицирован, то информация об этом должна быть синхронизирована во всех группах, так как мы не знаем, в каких из них содержатся связи данного человека. В свою очередь, это влечет за собой большую вычислительную нагрузку. Другой подход основан на разделении людей — агентов модели в соответствии с их географическим месторасположением. Однако в этом случае, поскольку население обычно распределено неравномерно, вычислительная нагрузка также будет распределена неравномерно, а ее выравнивание требует применения достаточно сложных алгоритмов, что создает дополнительную вычислительную нагрузку.

Предлагаемый авторами метод заключается в равномерном распределении агентов с соответствующими им исходящими связями по группам, количество которых равным числу процессоров.

На рис. 2.1.3.3 приведен пример разделения социальной сети из пяти узлов по двум процессорам. Предположим, что агент v отнесен к процессору γ. Тогда разработчики определяют процессор γ в качестве «владельца» агента v, который, в свою очередь, является «локальным» по отношению к γ. Вся информация об агенте v хранится на локальном для него процессоре. На рис. 2.1.3.3 видно, что узлы A и B отнесены к процессору 1, а узлы C, D и E — к процессору 2. При этом число обслуживаемых процессорами связей одинаково и равно 7.

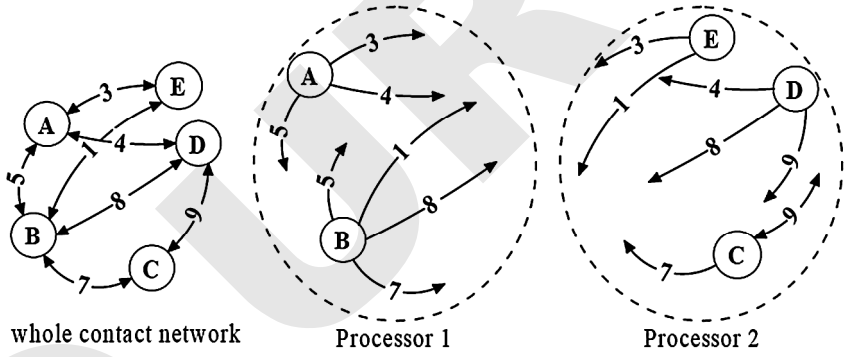


Рис. 2.1.3.3. Пример деления социальной сети по двум процессорам. *EpiFast: A Fast Algorithm for Large Scale Realistic Epidemic Simulations on Distributed Memory Systems* [10]

На рис. 2.1.3.4 приведена схема неоптимизированного алгоритма распараллеливания рассматриваемой задачи. Последовательность операций при этом следующая: a) вычисление вероятностей передачи инфекции от локальных узлов; b) обновление $\tau(v)$ для локальных узлов; c) передача информации нелокальным процессорам; f) обновление $\tau(v)$ для получивших информацию процессоров.

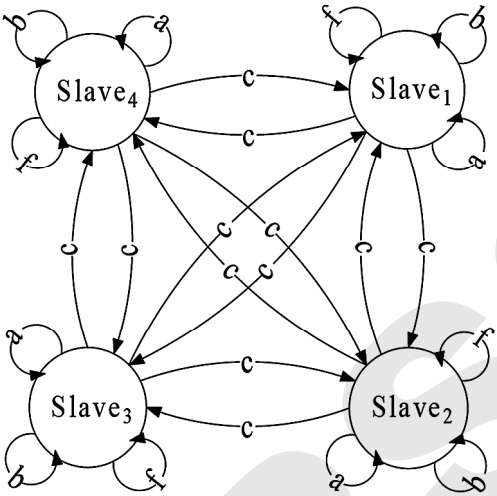


Рис. 2.1.3.4. Схема неоптимизированного алгоритма EpiFast. *EpiFast: A Fast Algorithm for Large Scale Realistic Epidemic Simulations on Distributed Memory Systems* [10]

Данный алгоритм продемонстрировал значительное ухудшение производительности по сравнению с последовательной версией.

В этой связи разработчики предлагают модифицированную версию вычислительного алгоритма, основанную на взаимодействии ведущих (master) и ведомых (slave) процессоров, организованном следующим образом: ведущий процессор «знает», какой из процессоров обслуживает конкретного агента, а каждый из ведомых процессоров «знает» только обслуживаемых (локальных) агентов. В процессе вычислений ведомые процессоры посылают запросы ведущим процессорам относительно связей с нелокальными агентами. В свою очередь, ведущие процессоры не осуществляют никаких расчетов, кроме поиска нелокального агента для каждой внешней связи и пересылают запросы соответствующим процессорам.

На рис. 2.1.3.5 приведена схема оптимизированного алгоритма распараллеливания рассматриваемой задачи. Последователь-

ность операций при этом следующая: a) на ведомых процессорах вычисляются вероятности передачи инфекции от локальных узлов; b) на ведомых процессорах происходит обновление $\tau(v)$ для локальных узлов; c) ведомые процессоры передают информацию нелокальным процессорам; d) ведущие процессоры проверяют владельцев полученных сообщений; e) ведущие процессоры пересылают информацию (u, v) владельцу v; f) ведомые процессоры обновляют $\tau(v)$.

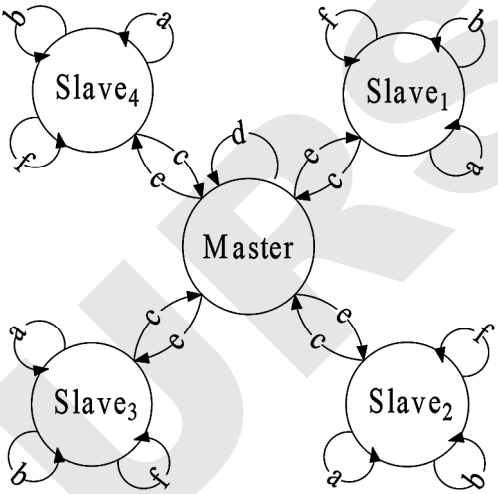


Рис. 2.1.3.5. Схема оптимизированного алгоритма EpiFast. *EpiFast: A Fast Algorithm for Large Scale Realistic Epidemic Simulations on Distributed Memory Systems* [10]

Используя схему оптимизированного алгоритма EpiFast, конвертируем алгоритм из листинга 2.1.3.1 в параллельную версию (листинг 2.1.3.2).

Листинг 2.1.3.2. Алгоритм EpiFast: параллельная версия

Вход/Выход: аналогичные алгоритму № 1.

1 разделение социальной сети

2 инициализация модели

```
3 for t = 0 to T do
4   if процессор ведомый then
5   foreach v do
6       обновляется состояние здоровья агента в соответствии с τ(v)
7       применяются меры по борьбе с эпидемией (A_t) и обновляется
8       локальный раздел G для каждого узла
9       foreach локального инфицированного узла u do
10          foreach контакта v узла u do
11              (a): передача инфекции от u к v
12              с вероятностью p(w(u, v))
13              if передача инфекции от u к v произошла then
14                  if v является локальным к процессору then
15                          (b): обновление τ(v) и π(v)
16                  else
17                          (c): пересылка информации ведущему
18                          процессору о передачи инфекции от u к v
19 if процессор ведущий then
20      получение информации от ведомых процессоров
21      (d): проверка отправителей информации
22      (e): пересылка информации владельцам v
23 if процессор ведомый then
24      (f): обновление τ(v) и π(v)
25 вывод (τ, Π)
```

Алгоритм EpiFast реализован на C++ с использованием MPI.

C. Расчеты

Первая серия расчетов заключалась в оценке производительности алгоритма в зависимости от применения различных мер по борьбе с инфекцией. Для сценария без применения мер были использованы следующие настройки: 1) значения инкубационного и инфекционного периодов для каждого человека подбирались с тем, чтобы их средние значения составляли 1,9 и 4,1 суток соответственно; 2) для инициации эпидемии были выбраны случайные 20 человек; 3) эпидемия имеет показатель пораженности (attack rate) 42 %, рассчитываемый как отношение людей, которые

будут инфицированы к общему числу людей, находящихся в зоне риска.

Для базового сценария разработчики дополнительно рассмотрели два сценария применения мер.

1. *С использованием антивирусных препаратов.* Сценарий следующий: с 20-го по 39-й дни с момента начала распространения вируса 10 % людей применяли препараты на протяжении 20 дней. Эти препараты способны на 87 % снизить риск заражения, а если человек уже заражен, то сделать его на 80 % безопаснее для окружающих.
2. *С использованием ограничений на передвижение.* Сценарий запускается в случае, если 0,1 % людей уже заразились, и предусматривает ограничение на лишние передвижения для 10 % населения (то есть разрешается передвижение только на работу и домой).

На рис. 2.1.3.6 приведено время расчетов для базового сценария и двух сценариев с применением описанных мер. Расчеты

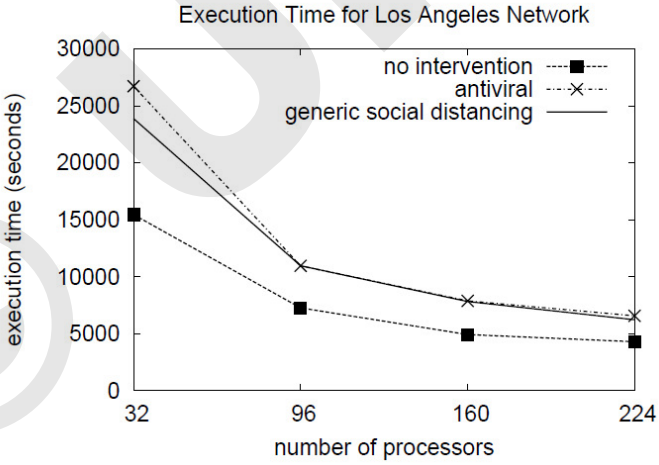


Рис. 2.1.3.6. Среднее время компьютерных симуляций для трех модельных сценариев в зависимости от числа использованных процессоров (вычислено на основе 25 запусков для каждого случая). *EpiFast: A Fast Algorithm for Large Scale Realistic Epidemic Simulations on Distributed Memory Systems* [10]

© URSS

проводились для социальной сети Большого Лос-Анджелеса (более 16 млн человек, около 900 млн социальных связей) на 180 дней. Как видно, производительность алгоритма зависит от того, применяются ли в модели меры по борьбе с инфекцией. Тем не менее EpiFast продемонстрировал очень высокую скорость — так, средняя скорость одной симуляции — менее 15 минут.

Далее были проведены расчеты для других социальных сетей (Майами — 2,09 млн чел., Бостон — 4,15 млн чел. и Чикаго — 9,05 млн чел.). На рис. 2.1.3.7 и 2.1.3.8 приведены результаты, полученные с теми же модельными настройками. Как видно, увеличение числа процессоров в 7 раз позволяет увеличить скорость вычислений в среднем до 4 раз, что говорит о хорошей масштабируемости.

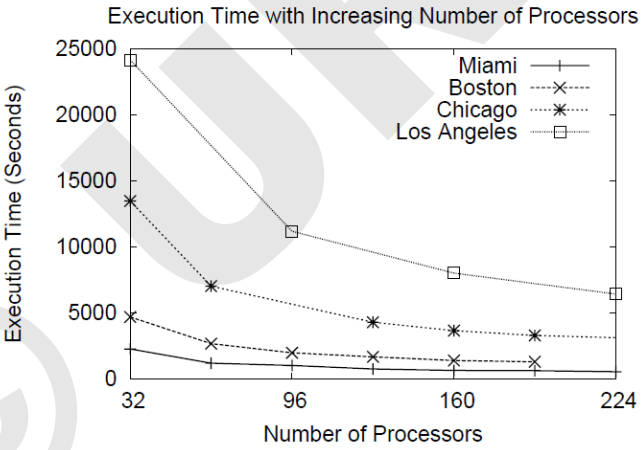


Рис. 2.1.3.7. Среднее время компьютерных симуляций для четырех социальных сетей в зависимости от числа использованных процессоров (вычислено на основе 25 запусков для каждого случая). *Оригинальное изображение опубликовано в работе «EpiFast: A Fast Algorithm for Large Scale Realistic Epidemic Simulations on Distributed Memory Systems»* [10]

© URSS

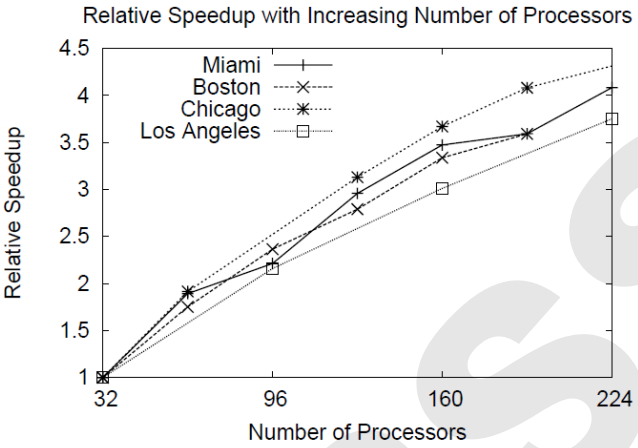


Рис. 2.1.3.8. Зависимость скорости выполнения симуляций от числа использованных процессоров (вычислено на основе 25 запусков для каждого случая). *EpiFast: A Fast Algorithm for Large Scale Realistic Epidemic Simulations on Distributed Memory Systems* [10]

Таким образом, разработанный алгоритм позволяет проводить эффективные и высокопроизводительные симуляции систем с очень большим числом агентов.

2.1.4. SWAGES — расширяемая распределенная среда для крупномасштабного агент-ориентированного моделирования

Ученые из Университета Тафтса (Медфорд, Массачусетс, США) представили пакет SWAGES (Software Agent Grid Experimentation System) — среду для разработки распределенных агентных моделей с возможностью автоматического динамического распараллеливания программного кода. SWAGES предоставляет пользователям поддержку нескольких языков программирования, а также возможность подключения плагинов для визуализации, статистического анализа и автоматической обработки ошибок. Результаты

вычислений могут быть сконвертированы в файлы для последующего использования в общедоступных средствах обработки данных (типа R или Scilab).

SWAGES содержит несколько компонент клиент-серверной архитектуры, серверные компоненты которой предоставляют возможности для распределенных вычислений (планировка, распределение, запуск, контроль выполнения, восстановление после сбоев и др.). На сервере определяются наборы экспериментов, отличающиеся начальными состояниями, приоритетом выполнения, а также формируются коллекции выходных данных, результаты дополнительного анализа данных и т. д.

Планировщик отвечает за приоритет выполнения экспериментов из имеющихся наборов, а также динамически подгружает освободившиеся ресурсы для экспериментов из очереди.

В процессе работы приложения поддерживается открытый канал связи между клиентом и сервером с целью отслеживания текущего состояния агентной модели. Этот канал также крайне важен для обнаружения ошибок, возникающих в ходе выполнения программного кода, и возможного восстановления последствий в зависимости от причины нарушения (сбой в работе операционной системы, проблема с сетью или непредвиденная ошибка в самом приложении). В случае невозможности продолжения симуляции серверная часть может возобновить работу приложения с последнего сохраненного состояния.

Помимо этого, в рамках системы функционирует сторожевой таймер (watchdog), который регулярно проверяет состояние отдельных компонент приложения на предмет стабильности их работы и перезапускает их в случае необходимости.

В свою очередь, компоненты клиентской части приложения обеспечивают возможность управления запуском определенной симуляции и получают информацию о ходе выполнения приложения (через соответствующий программный модуль SimWorld).

SimWorld построен на базе среды SimAgent — специализированного программного обеспечения для разработки агентных моделей, которое создано с использованием кроссплатформенного компилятора Poplog со встроенным интерпретатором OPS5 (Official Production System, Version 5). SimWorld запускается как с использованием графического интерфейса, так и в режиме пакетной обработки, и может быть использован для разработки агентных моделей любой степени сложности.

Автоматическое распараллеливание в SimWorld может запускаться в режиме жесткой конфигурации (когда все параллельные процессы обновляются одновременно, в рамках одного цикла) или в асинхронном режиме, когда отдельные процессы обновляются независимо, по мере необходимости. Последний режим используется при реализации агентных моделей, в которых важно пространственное распределение агентов, и где агенты могут оказывать влияние на окружающую среду в рамках определенного диапазона. Так, симуляция S с n агентами может быть распределена по k параллельным процессам S_1, S_2, ..., S_k с n_1, n_2, ..., n_k агентами, распределенными по k узлам, которые будут обновляться асинхронно. В рамках текущей реализации SimWorld каждый процесс S_i посылает информацию о состоянии обслуживаемых им агентов серверному приложению, которое хранит эти данные в рамках структуры, доступной всем остальным процессам данной симуляции.

Экономия времени расчетов в зависимости от использования различного количества узлов для модели, имитирующей перемещение роя, видна на рис. 2.1.4.1.

Результаты были получены на основе прогона 20 симуляций модели со 100 циклами в каждом случае на кластере под управлением Linux с двухъядерными процессорами Xeon (2,4 GHz, 1 GB RAM).

Как уже говорилось, в SimWorld реализована поддержка внешних плагинов. К примеру, в качестве демонстрации возможностей

авторы подключили к SimWorld открытый физический движок ODE (Open Dynamics Engine), который хорошо подходит для симуляции движущихся объектов в рамках неизменного пространства, и объектно-ориентированный графический движок OGRE (Object-Oriented Graphics Rendering Engine) — кроссплатформенный движок с удобным дизайном и архитектурой.

В рамках каждого цикла физический движок рассчитывает текущую позицию и направление движения агента и может передать эту информацию для графического движка, причем не требующие визуализации агенты на выбор разработчика могут быть исключены из показа. На рис. 2.1.4.2 показано рабочее окно агентной модели с одновременным отображением SimWorld и OGRE.

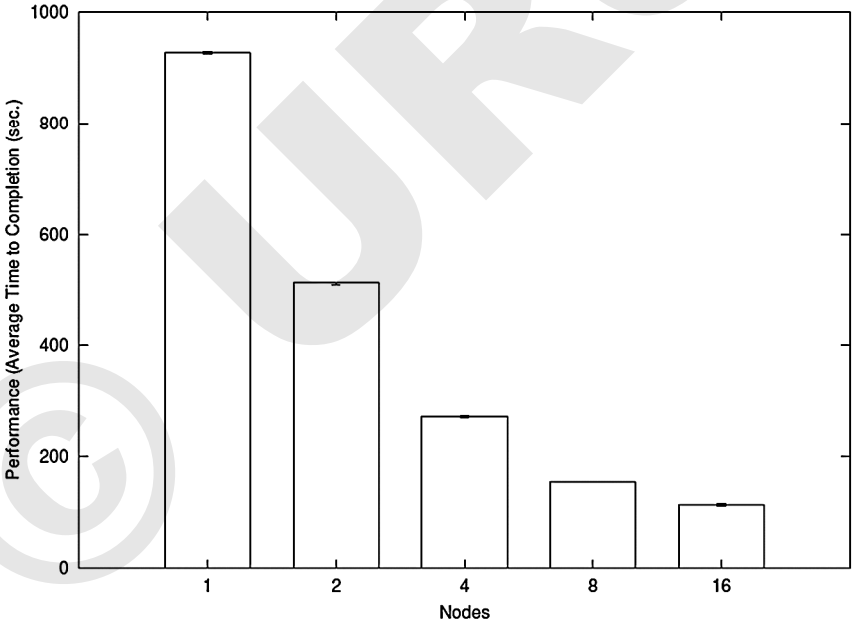


Рис. 2.1.4.1. Среднее время реализации задачи по максимально быстрому поиску и перемещению агентов к ближайшей точке сбора в рамках агентной модели, имитирующей перемещение роя для 1, 2, 4, 8, 16 узлов. *Университет Тафтса, https://hrilab.tufts.edu*

© URSS

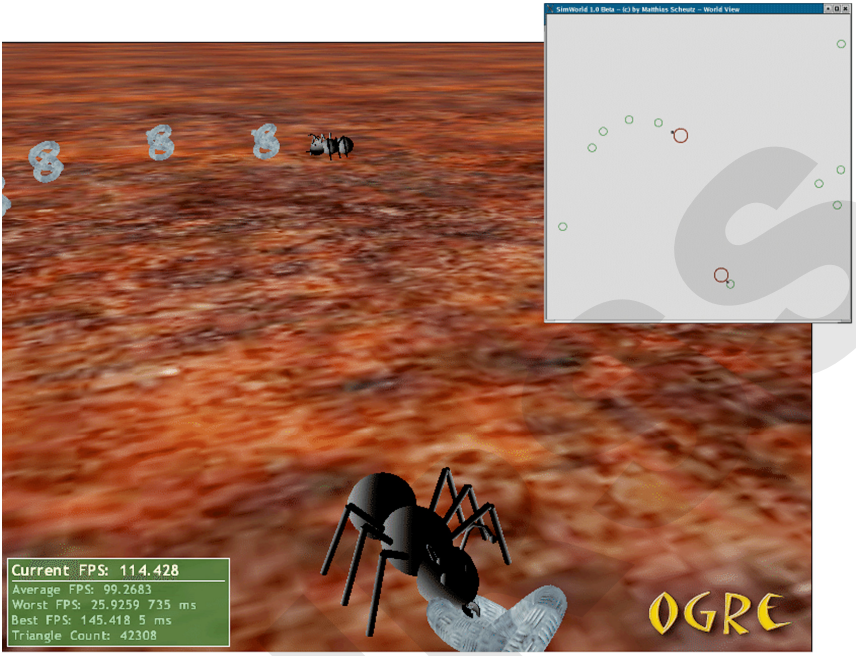


Рис. 2.1.4.2. Пример отображения простых агентов-муравьев с помощью OGRE в простом двумерном графическом интерфейсе SimWorld.
Scheutz M., Schermerhorn P., Connaughton R., Dingler A. SWAGES — An Extendable Parallel Grid Experimentation System for Large-Scale Agent-Based Alife Simulations // Artificial Life X: Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems. 2006. P. 412–418

Моделирование в SWAGES

Шаг. 1. Разработка агентной модели

Агенты могут быть описаны с использованием любого программного языка, поддерживаемого Poplog (к примеру, Pop11, Prolog, ML, Scheme, C-Lisp). Кроме того, для разработчика предусмотрена возможность вызова внешних функций, написанных на других языках программирования, посредством специального интерфейса.

В листинге 2.1.4.1 показан пример кода простого агента, который получает энергию, потребляя продукты питания. SimWorld позволяет модифицировать программный код во время работы приложения. К примеру, в процессе запуска симуляции, в коде, приведенном в листинге 2.1.4.1, можно заменить метод eating, реализующий определенное поведение пустым методом (для возможного изучения последствий отказа от еды), то есть на такую строчку: 'define :method eating(a:simple agent); enddefine'.

Листинг 2.1.4.1. Структура простого агента в SimWorld, реализованная с помощью языка Pop11, включающая базовый класс, производный класс и правила потребления

```
define :class thing; is sim_object; /* базовый класс */
    slot geometry == undef;
    slot position = undef;
    slot heading = undef;
    slot mass = undef;
enddefine;

define :class simple_agent; is thing; /* производный класс */
    slot speed = 3;
    slot energy = 1000;
    slot maxenergy = 2000;
    slot food = undef;
    slot intake = 50;
enddefine;

define :ruleset eating_ruleset; /* правила для потребления */
    RULE start_eating
        [ingest food] [NOT halting_for eating]
                ==>
        [do halting_for eating]
        [do eating]
        [STOP]
```

©URSS

```
    RULE end_eating
        [NOT ingest food] [halting_for eating]
                ==>
        [NOT halting_for eating]
        [STOP]
    RULE continue_eating
        [ingest food] [halting_for eating]
                ==>
        [do eating]
enddefine;

define :method eating(agent:simple_agent);
    if energy(agent) >= maxenergy(agent) then
        remove([halting_for eating]); /* агент насытился и перестал есть */
    else /* добавление энергии агенту */
        energy(agent) + intake(agent) -> energy(agent);
        energy(food(agent)) — intake(agent)
                -> energy(food(agent));
    endif;
enddefine;
```

Таким же образом возможно определять новые типы агентов, добавлять или удалять агентов в процессе запуска симуляции, а также отслеживать и записывать значения свойств агентов в процессе их функционирования.

Шаг. 2. Настройка и проведение экспериментов

После построения агентной модели можно настроить порядок проведения экспериментов. В листинге 2.1.4.2 приведен пример настройки параметров для проведения множественных экспериментов с целью выяснения живучести агентов. Как видно, определяется число агентов модели (количество объектов типа «AGENT»), объем еды (количество объектов типа «FOOD») и вероятность («PROB») появления объектов типа «FOOD» в случайном месте пространства размерности 1000 × 1000. Таким образом, в рамках трех-

© URSS

мерного параметрического пространства настройки экспериментов AGENT × FOOD × PROB мы получим 10 × 20 × 10 × 40 = 80 000 уникальных экспериментов.

Листинг 2.1.4.2. Настройка параметров для проведения экспериментов

```
[[ /*** настройка среды моделирования ***/
    [initfile 'reactive.p'] /* определения пользователя */
    [quitif 10000] /* остановка после 10 000 циклов */
    [world WIDTH 1000 HEIGHT 1000]/* пространственные ограничения */
[ /*** настройка агентов ***/
    [simple_agent /* использование данного объекта */
        [startup [variate AGENT from 1 to 10 step 1]]
        [record [at death [sim_x][sim_y]]
            [at end [energy]]]
    [food /* использование данного объекта */
        [startup [variate FOOD from 10 to 200 step 10]]]]
[ /*** расписание событий ***/
    [random [[variate PROB from 0.1 to 0.5 step 0.05] [food]]]]
[ /*** настройка SWAGES ***/
    [name 'simple' 'resultsdir'] /* выбор места для хранения результатов */
    [user 'airolab'] /* запуск это имени данного пользователя */
    [priority 5] /* запуск с определенным приоритетом */
    [parallelize] /* разрешить распараллеливание */
    [ranseed 29187] /* начальные условия */
    [replicates 40] /* для всех повторов */
    [watch 120 30] /* контроль исполнения */
    [email 'mscheutz@cse.nd.edu'] /* уведомить пользователя о выполнении */
    [copyimages '/tmp/'] /* место для хранения информации о симуляции */
    [copystats 'statsdir']]] /* место для хранения статистики */
```

Настройки экспериментов могут быть произведены через web-интерфейс пакета SWAGES (рис. 2.1.4.3).

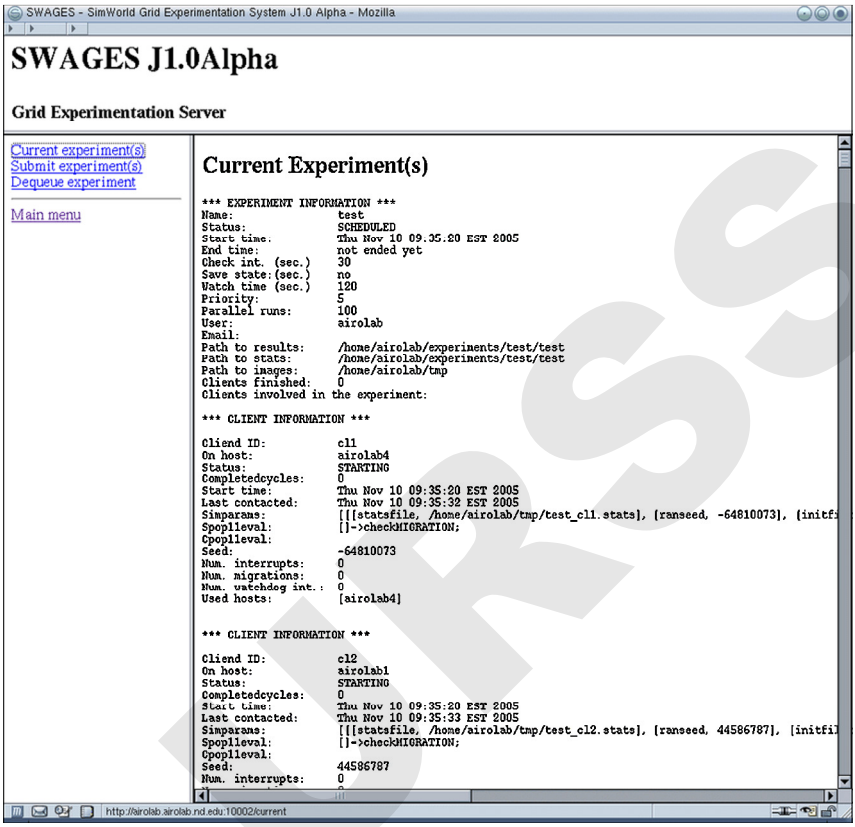
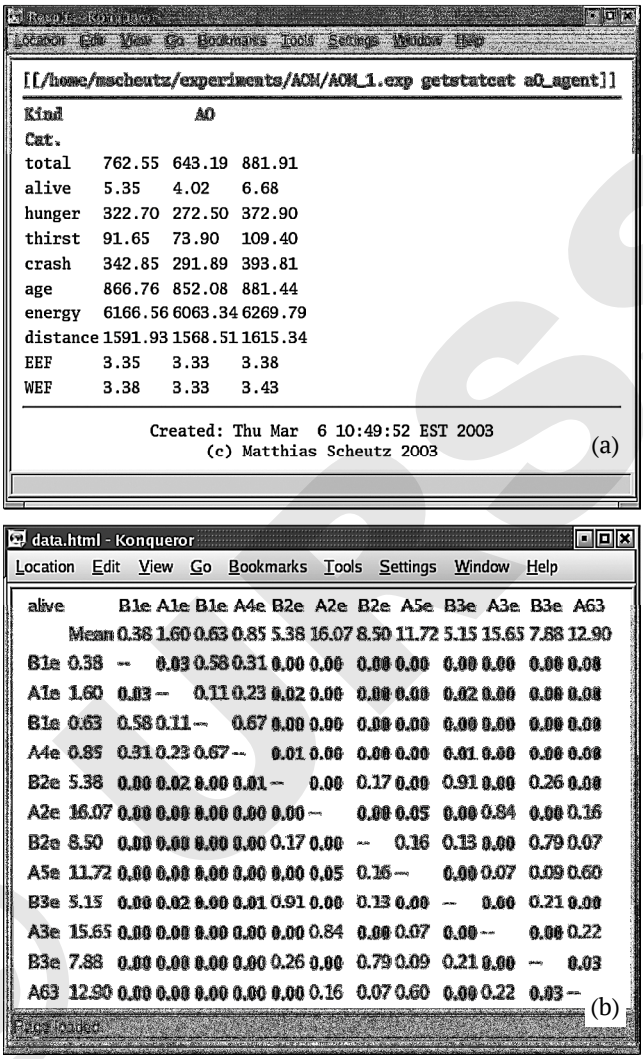


Рис. 2.1.4.3. Настройка расписания запуска набора экспериментов на нескольких узлах. *Scheutz M., Schermerhorn P., Connaughton R., Dingler A. SWAGES — An Extendable Parallel Grid Experimentation System for Large-Scale Agent-Based Alife Simulations // Artificial Life X: Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems. 2006. P. 412–418*

Шаг 3. Анализ полученных результатов

SWAGES содержит встроенные инструменты для статистического анализа, а также библиотеки для добычи данных (в различных форматах, включая HTML, TeX, простой текст). На рис. 2.1.4.4 отображена типовая статистическая операция, проведенная через web-интерфейс SWAGE, а на рис. 2.1.4.5 выходной файл R и результаты динамических скриптов, выполненных в gnuplot и Scilab).

© ДМК



Result - Konqueror

Location Edit View Go Bookmarks Tools Settings Window Help

[[/home/mscheutz/experiments/AON/AON_1.exp getstatcat a0_agent]]

Kind	A0		
Cat.			
total	762.55	643.19	881.91
alive	5.35	4.02	6.68
hunger	322.70	272.50	372.90
thirst	91.65	73.90	109.40
crash	342.85	291.89	393.81
age	866.76	852.08	881.44
energy	6166.56	6063.34	6269.79
distance	1591.93	1568.51	1615.34
EEF	3.35	3.33	3.38
WEF	3.38	3.33	3.43

Created: Thu Mar 6 10:49:52 EST 2003
(c) Matthias Scheutz 2003

(a)

data.html - Konqueror

Location Edit View Go Bookmarks Tools Settings Window Help

alive		B1e	A1e	B1e	A4e	B2e	A2e	B2e	A5e	B3e	A3e	B3e	A63
	Mean	0.38	1.60	0.63	0.85	5.38	16.07	8.50	11.72	5.15	15.65	7.88	12.90
B1e	0.38	--	0.03	0.58	0.31	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
A1e	1.60	0.03	--	0.11	0.23	0.02	0.00	0.00	0.00	0.02	0.00	0.00	0.00
B1e	0.63	0.58	0.11	--	0.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
A4e	0.85	0.31	0.23	0.67	--	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00
B2e	5.38	0.00	0.02	0.00	0.01	--	0.00	0.17	0.00	0.91	0.00	0.26	0.00
A2e	16.07	0.00	0.00	0.00	0.00	0.00	--	0.00	0.05	0.00	0.84	0.00	0.16
B2e	8.50	0.00	0.00	0.00	0.00	0.17	0.00	--	0.16	0.13	0.00	0.79	0.07
A5e	11.72	0.00	0.00	0.00	0.00	0.00	0.05	0.16	--	0.00	0.07	0.09	0.60
B3e	5.15	0.00	0.02	0.00	0.01	0.91	0.00	0.13	0.00	--	0.00	0.21	0.00
A3e	15.65	0.00	0.00	0.00	0.00	0.00	0.84	0.00	0.07	0.00	--	0.00	0.22
B3e	7.88	0.00	0.00	0.00	0.00	0.26	0.00	0.79	0.09	0.21	0.00	--	0.03
A63	12.90	0.00	0.00	0.00	0.00	0.00	0.16	0.07	0.60	0.00	0.22	0.03	--

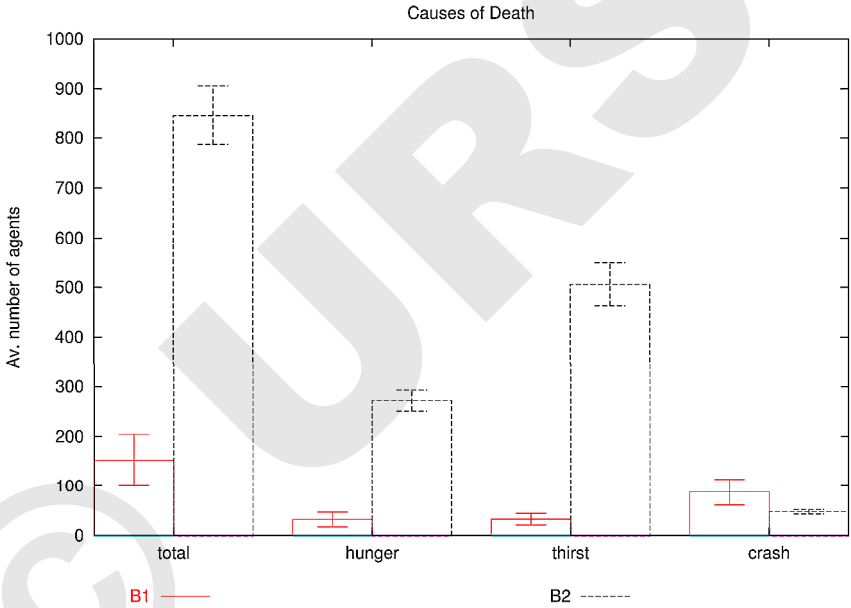
Page loaded.

(b)

Рис. 2.1.4.4. Выходные результирующие данные после эксперимента для одного агента A0 (a) и сравнение сводных результатов для среднего числа выживших (b). *Scheutz M., Schermerhorn P., Connaughton R., Dingler A. SWAGES — An Extendable Parallel Grid Experimentation System for Large-Scale Agent-Based Alife Simulations // Artificial Life X: Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems. 2006. P. 412–418*

```
"N"  "Repl"  "Spread"  "Females"  "Time"  "Males"  "Distance"
1    1       0.8       1          147.83  0.00000  274.512
1    2       0.8       1          175.15  0.05000  276.763
1    3       0.8       1          201.63  0.13333  281.357
1    1       0.8       2          226.10  0.21250  290.448
1    2       0.8       2          245.87  0.25200  300.530
...
```

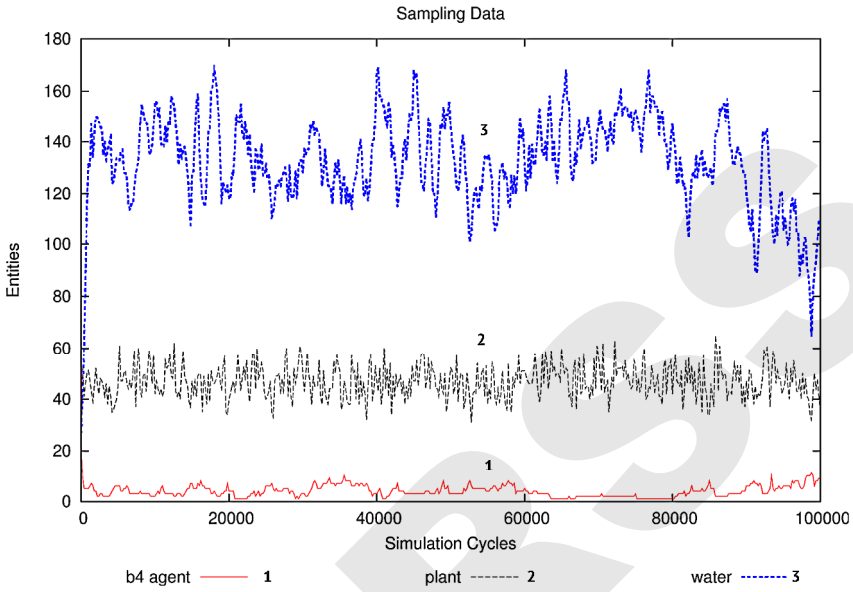
(a)



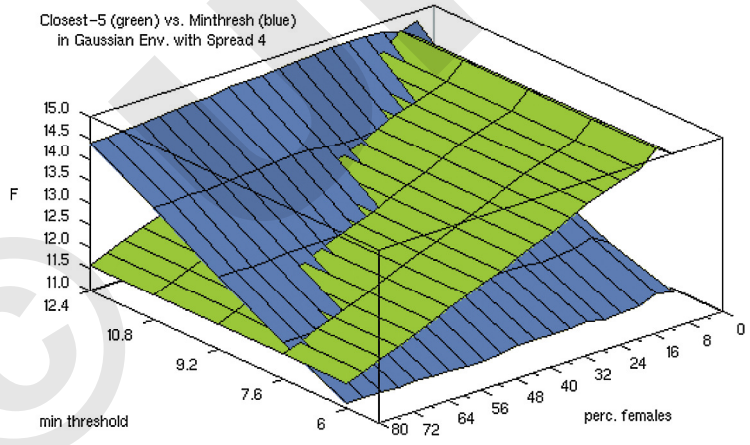
(b)

Рис. 2.1.4.5 (a, b). Выходной файл R (a). Сравнение двух видов агентов по показателям причин смертности (b)— график получен посредством gnuplot..
Scheutz M., Schermerhorn P., Connaughton R., Dingler A. SWAGES — An Extendable Parallel Grid Experimentation System for Large-Scale Agent-Based Alife Simulations // Artificial Life X: Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems. 2006. P. 412–418

© URSS



(c)



(d)

Рис. 2.1.4.5 (c, d). Динамика популяции агентов и ресурсов (c) — график получен посредством gnuplot. Сравнение нескольких показателей двух видов агентов (зеленых и синих), реализованное с помощью Scilab (d). *Scheutz M., Schermerhorn P., Connaughton R., Dingler A. SWAGES — An Extendable Parallel Grid Experimentation System for Large-Scale Agent-Based Alife Simulations // Artificial Life X: Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems. 2006. P. 412–418*

Разработчики SWAGES позиционируют свой продукт как проработанную среду для построения крупномасштабных агентных моделей, успешно использованную как минимум для нескольких десятков научно-исследовательских проектов. Более подробно про SWAGES можно прочитать на сайте лаборатории взаимодействия между человеком и роботом (факультет информатики Университета Тафтса): https://hrilab.tufts.edu/software.

2.1.5. Программное обеспечение для высокопроизводительных вычислений агентных моделей пешеходной динамики с использованием графических процессоров

Работа исследовательской группы из Университета Шеффилда (The University of Sheffield, UK) посвящена разработке программного обеспечения для построения крупномасштабных пешеходных агентных моделей с использованием графических процессоров (Graphics Processing Unit, GPU), которые потенциально обеспечивают высокую производительность приложений. Это особенно важно для визуализации городской среды, представляющей собой сложную пространственную структуру с большим количеством движущихся объектов.

Разработанное программное обеспечение позволяет описывать поведение агентов, а также изменение их данных на высоком уровне (в виде написания соответствующих скриптов) — без непосредственной работы с памятью GPU. Такая возможность обеспечивается специальной функцией (‘*F*’), отображающей информацию об агентах в набор двумерных упорядоченных 32-битных текстур с данными 4 каналов (3 цветных — красный (Red), зеленый (Green), синий (Blue) и альфа-канал (Alpha), или RGBA), см. рис. 2.1.5.1.

Расширение OpenGL — Frame buffer object (FBO) предоставляет возможность внеэкранного рендеринга сцен в несколько текстур (за счет применение метода одновременного рендеринга MRT — Multiple Render Targets).

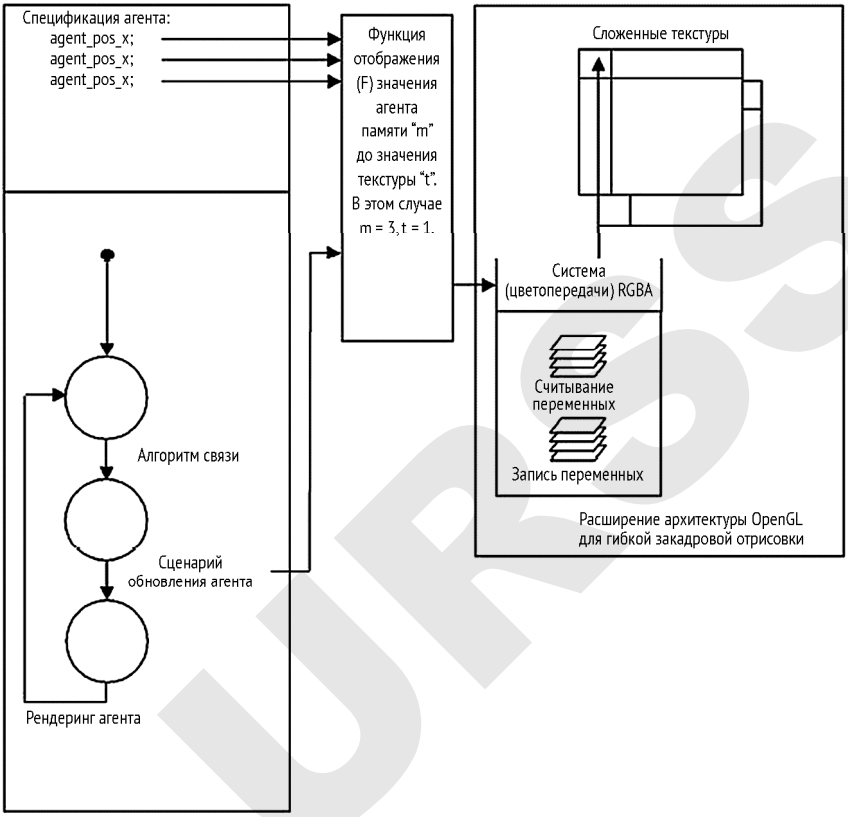


Рис. 2.1.5.1. Пример отображения трех переменных агента в двумерную текстуру

Как правило, части пространства используются либо для непосредственного хранения данных агентов, либо они содержат указатели на данные в пространстве самого агента. В соответствии с этой методологией моделируемое пространство делится на подпространства, размер которых пропорционален радиусу видимости агентов, содержащихся в подпространстве. Затем вычисляются все связи каждого агента внутри собственного и внешних подпространств.

© URSS

Определение поведения агентов

Поведение агентов задается с использованием соответствующего API в виде скриптов на языке Си, а также путем переопределения функций базового класса.

Приведенное ниже уравнение рассчитывает силу, являющуюся результирующей нескольких других сил, действующую на каждого агента и определяющую параметры его движения (скорость и направление):

$$F_i = R_i + Cr_i + G_i + M_i. \qquad (2.1.5.1)$$

В этом соотношении R_i — переменная, характеризующая воздействие на агента со стороны социума, Cr_i — сила, действующая на близком расстоянии (корректирующая движение для избегания столкновений), G_i — сила, зависящая от цели движения агента и приближающая его к ней, M_i — сила, воздействующая на агента со стороны окружающей среды.

У агентов в тестируемой модели ограниченный горизонт видимости, который упрощает межагентное взаимодействие, так как исключает из расчетов невидимых для агента членов моделируемого социума. Уравнение (2.1.5.2) описывает силу, воздействующую на агента со стороны видимых для него агентов:

$$R_i = S \sum_{j}^{0} \lambda_{ij} \left[P_i = P_j\right]_i \left(\frac{I}{\left(\left|P_i = P_j\right|\right)^2} \right), \qquad (2.1.5.2)$$

где S — регулируемый параметр, определяющий силу воздействия, P_i и P_j — пространственные координаты агентов, λ — значение угла между линией движения агента i и линией, связывающей координаты агентов i и j, а переменная I регулирует уровень влияния расстояния между агентами на силу воздействия.

Сила Cr_i, в отличие от R_i, не зависит от направления движения агентов, но проявляет себя в случае повышенной концентрации агентов и позволяет предотвращать столкновения между ними.

Визуализация

Рендеринг агентной модели достигается двумя различными методами. Первый — использование треугольных примитивов, каждый из которых представлен текстурными координатами, обрабатываемыми вершинным шейдером, вычисляющим окончательные параметры объектов. Поскольку пешеходы — это не статичные объекты, важно анимировать их передвижения реалистично, с учетом телодвижений. Для этого была использована технология создания анимации по ключевым кадрам, с одной стороны, упрощающая визуализацию, а с другой — снижающая вычислительную нагрузку.

Посредством первого метода каждый пешеход анимируется с одинаковым уровнем детализации. С целью экономии вычислительных ресурсов в рендеринге агентной модели применяется второй метод — LOD (Levels Of Detail — «уровни детализации»), заключающийся в создании нескольких вариантов одного и того же объекта с разными уровнями детализации. Эти уровни меняются в зависимости от степени удаленности виртуальной камеры (находящиеся на далеком расстоянии объекты подробно не детализируются).

У каждого агента есть переменная, используемая для хранения соответствующего ему номера уровня LOD, значение которой обновляется на каждом шаге.

Результат применения второго метода показан на рис. 2.1.5.2, где пешеходы окрашены согласно их принадлежности уровням LOD.

Производительность и результаты

Для всех расчетов поведение агентов определялось уравнением 2.1.5.1.

На рис. 2.1.5.2 показана зависимость между горизонтом видимости у агентов, измеряемым в метрах, и количеством пешеходов, входящих в систему поиска (lookups) для возможной связи

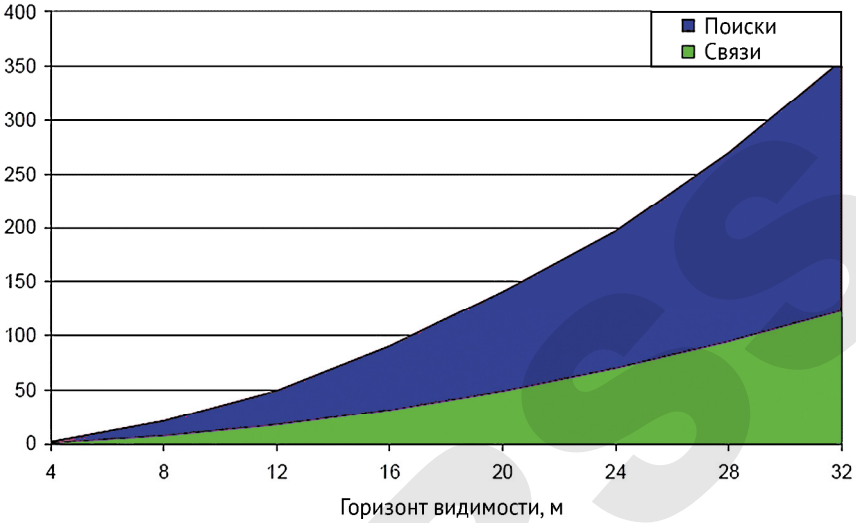


Рис. 2.1.5.2. Зависимость между горизонтом видимости агентов, их потенциальными контактами и связями

(communications). Соотношение между состоявшимися связями у агентов и количеством видимых агентов, потенциально готовых к взаимодействию, для всех случаев составляет примерно 35 %.

График производительности приложения, показывающий зависимость между количеством агентов и кадровой частотой (Frames Per Second) с учетом различных горизонтов видимости, приведен на рис. 2.1.5.3. Отметим, что в случае более дальнего горизонта (32 метра) производительность тестируется с учетом большего объема данных для планирования движения агентов.

Вообще говоря, применение двух упомянутых выше методов рендеринга агентной модели показало вполне удовлетворительные результаты. Так, при видимости агентов равной 4 метрам, производительность модели для 262 144 агентов составляет 13 кадров в секунду, а для 65 536 агентов — 42 кадра в секунду.

Далее оценка рендеринга осуществлялась посредством использования различных уровней детализации (нулевой уровень — самая

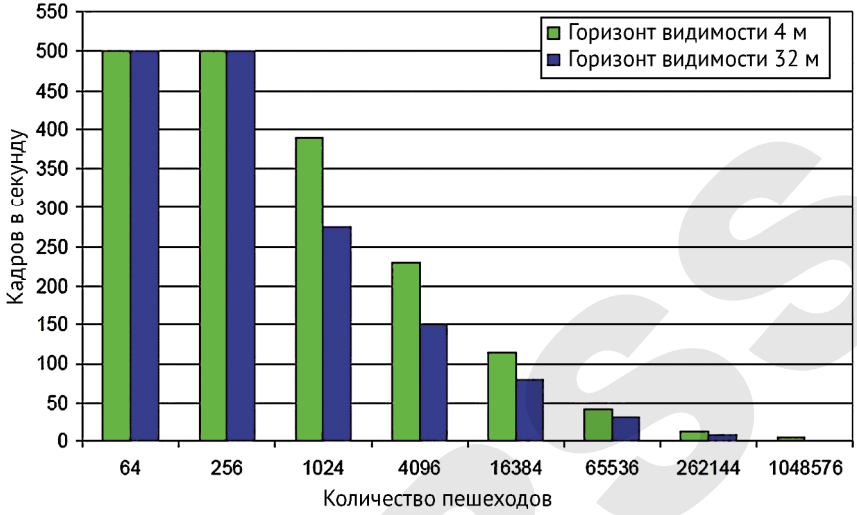


Рис. 2.1.5.3. Производительность рендеринга для примитивных агентов с горизонтом видимости в 4 и 32 метра

низкая степень детализации, а второй уровень — самый высокий). Рис. 2.1.5.4 показывает, что для 16 384 агентов при высокой степени детализации второго уровня (1000 полигонов) производительность составляет менее 20 кадров в секунду, а при детализации первого уровня (400 полигонов) составляет уже 40 кадров в секунду. В свою очередь, применение алгоритма интеллектуального рендеринга с использованием LOD позволяет увеличить производительность до 50 кадров в секунду. С другой стороны, применение такого алгоритма не оправдано при незначительном числе агентов (ниже 1024).

Дальнейшее развитие модели авторы видят в увеличении численности пешеходов до размеров среднего города с одновременным усложнением ландшафта. Помимо этого предполагается настройка агентов путем разработки индивидуальной походки для отдельных пешеходов, а также увеличение разнообразия их внешнего вида.

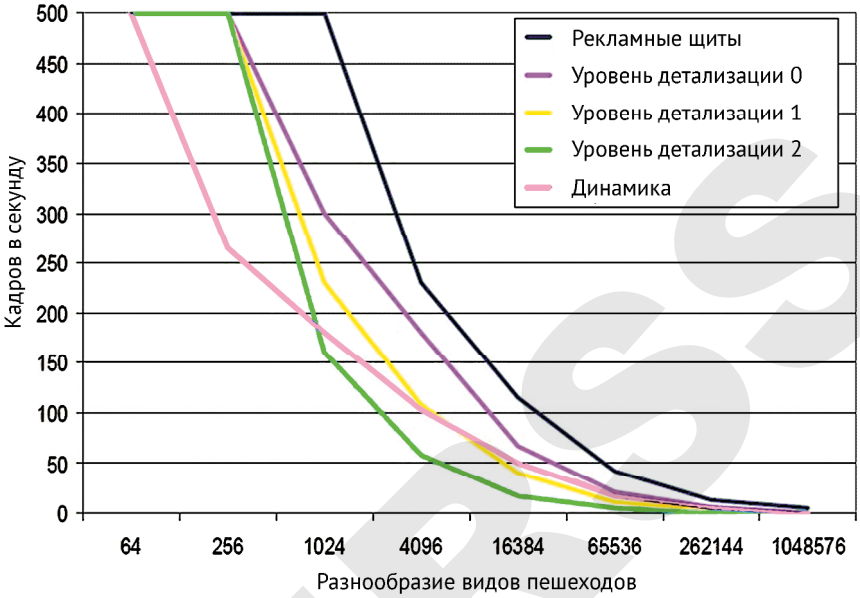


Рис. 2.1.5.4. Производительность рендеринга агентной модели для различных уровней детализации

Более подробно про модель можно прочитать на сайте разработчика, а также на сайте компании Nvidia (в разделе с примерами моделей, разработанных с помощью программно-аппаратной архитектуры CUDA):

1) https://paulrichmond.shef.ac.uk/
2) https://www.nvidia.com/content/cuda/spotlights/paul-richmond-sheffield.html

2.1.6. CyberGIS — высокопроизводительный инструмент для пространственного анализа и агентного моделирования

Интернациональный междисциплинарный проект CyberGIS, вовлекший как научно-исследовательские центры нескольких стран (среди которых крупнейшие суперкомпьютерные центры),

так и лидеров ИТ-рынка (в их числе безусловный лидер ГИС-индустрии — компания ESRI), стартовал в августе 2012 года. К настоящему моменту в рамках проекта реализован набор инструментальных средств CyberGIS, представляющих собой несколько геопространственных программных компонент с открытым кодом, открывающих широкие возможности для пространственного анализа. Важной особенностью CyberGIS является поддержка параллельных вычислений, а также реализация агент-ориентированного подхода при моделировании.

Процесс разработки приложений в CyberGIS включает в себя три основных этапа:

1. Сборка и тестирование программного обеспечения, разработанного исследователями посредством сервисов типа Travis CI.
2. Тестирование возможности интегрирования и масштабируемости с использованием соответствующего программного пакета (поддерживаемого организацией «National Science Foundation Middleware Initiative» и финансируемого Национальным научным фондом США), который содержит большое количество утилит для проведения научных и прикладных исследований.
3. Тестирование производительности программного обеспечения посредством среды разработки XSEDE (Extreme Science and Engineering Discovery Environment), включающей в себя коллекцию передовых ресурсов для научных исследований. Посредством XSEDE выявляются узкие места для последующей эффективной реализации приложений на суперкомпьютерах.

CyberGIS содержит следующие программные компоненты:

1. TauDEM (Terrain Analysis Using Digital Elevation Models) — набор программных компонент для гидрологического анализа цифровых моделей рельефа с возможностью запуска на суперкомпьютерах.
2. Параллельная версия PySAL — библиотека функций для пространственного анализа данных, реализованных с использованием языка Python.

3. pRasterBlaster — среда для высокопроизводительных вычислений картографических проекций.
4. PGAP — масштабируемое средство реализации параллельного генетического алгоритма (Parallel Genetical Algorithm (PGA)) для решения обобщенной задачи о назначениях (Generalized Assignment Problem (GAP)). Предлагаемое разработчиками решение масштабируется для аппаратной платформы с 262 тыс. ядер.
5. PABM (Parallel Agent-Based Modeling) — программная платформа для построения параллельных агентных моделей.

Особенности перечисленных компонент приведены в табл. 2.1.6.1.

Таблица 2.1.6.1

Особенности программных компонент

Компоненты	Программная модель и методология	Разработчики
PABM	1) MPI; 2) параллельный ввод-вывод	Иллинойский университет в Урбане-Шампейне (University of Illinois at Urbana-Champaign, UIUC)
Параллельная версия PySAL	Параллельная реализация Python	Университет штата Аризона (Arizona State University, ASU)
pRasterBlaster	1) MPI; 2) параллельный ввод-вывод	1) Центр лучших практик по геопространственной информации (Center of Excellence for Geospatial Information Science, CEGIS) 2) Геологическая служба США (United States Geological Survey, USGS)
SpatialText	Текстовое геокодирование большого объема новостей	1) Исследовательская группа Калева Леетару (Kalev Leetaru) 2) Иллинойский университет в Урбане-Шампейне

© URSS

Используемые программные компоненты и их связь с программными интерфейсами приведена на рис. 2.1.6.1.

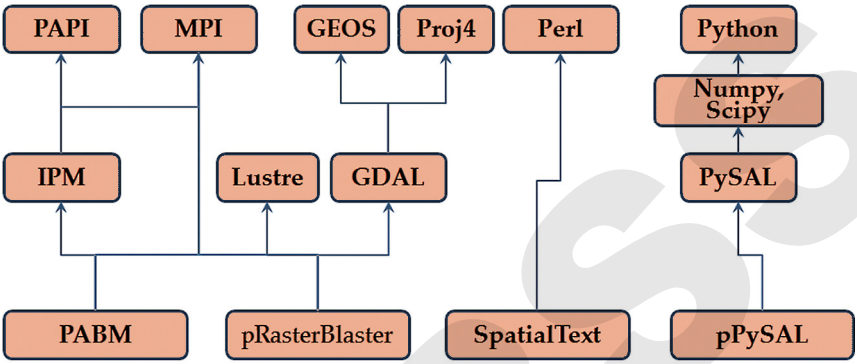


Рис. 2.1.6.1. Граф зависимостей в CyberGIS. *https://cybergis.illinois.edu*

Приоритеты развития CyberGIS:

1. Участие междисциплинарных групп исследователей с целью сбалансированного развития CyberGIS для возможности решения широкого спектра задач.
2. Интеграция и поддержка совместимых, хорошо управляемых программных компонент с открытым кодом (последнее свойство является обязательным).
3. Повышение производительности и масштабируемости приложений CyberGIS на основе эффективного распараллеливания процедур обработки пространственных данных.
4. Широкое внедрение CyberGIS в образовательные программы.
5. Вовлечение большого числа пользователей из разных стран с целью расширения CyberGIS путем решения множества задач для лучшего понимания человеческой природы.
6. Совершенствование интерфейса CyberGIS для легкой настройки и использования со стороны широкого круга пользователей.

© URSS

Примеры рабочего окна CyberGIS приведены на рис. 2.1.6.2 и 2.1.6.3.

Более подробно про CyberGIS можно прочитать на сайте проекта: https://cybergis.illinois.edu

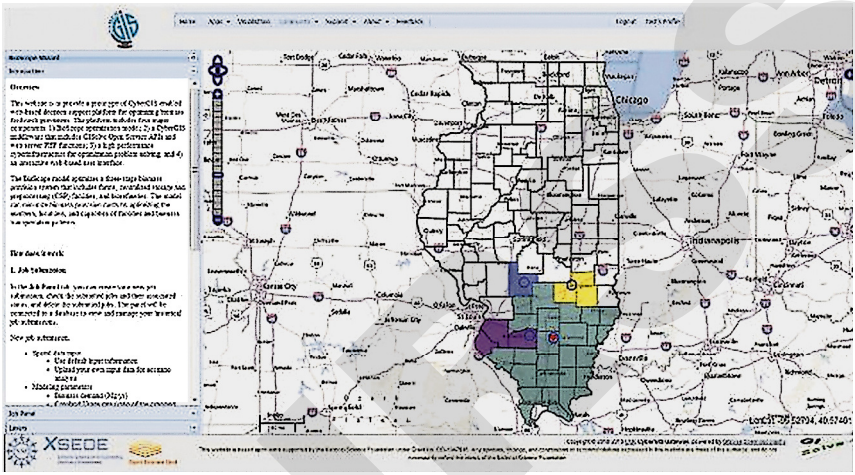


Рис. 2.1.6.2. Рабочее окно CyberGIS. *https://cybergis.illinois.edu*

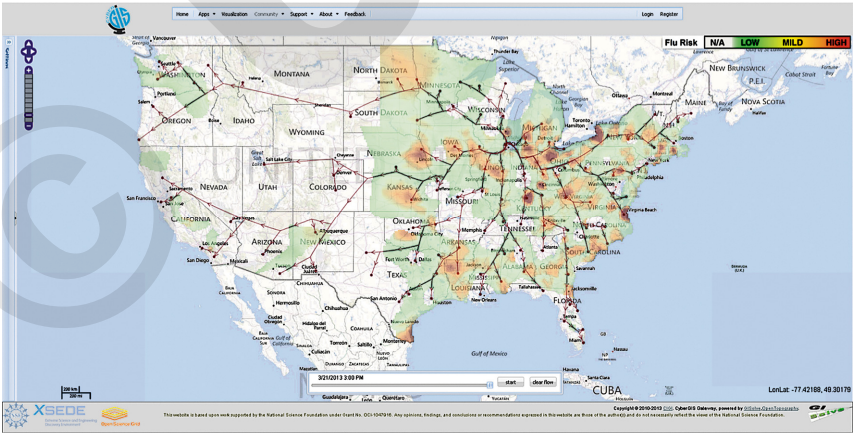


Рис. 2.1.6.3. Рабочее окно CyberGIS. *https://cybergis.illinois.edu*

2.1.7. HPABM — иерархическая платформа для параллельной реализации пространственно-распределенных агент-ориентированных моделей

Разработанная в Национальном центре суперкомпьютерных приложений (National Center for Supercomputing Applications, NCSA) Иллинойского университета иерархическая платформа для параллельной реализации пространственно-распределенных моделей (A Hierarchical Parallel Simulation Framework for Spatially-explicit Agent-based Models, HPABM) позволяет реализовывать модели, построенные на базе геоинформационных систем с использованием высокопроизводительных вычислительных ресурсов. При расчетах исходная агентная модель декомпозируется на несколько субмоделей, каждая их которых представляет собой отдельный блок для параллельных вычислений, содержащий набор агентов, имеющих географическую привязку.

Соотношение (2.1.7.1) определяет формальную структуру HPABM, а на рис. 2.1.7.1 в концептуальном виде представлена конструкция HPABM. Каждая модель, разработанная с использованием HPABM, содержит в себе наборы субмоделей, агрегированных в группы супермоделей. Субмодель является простой вычислимой единицей, содержащей набор агентов и соответствующую им среду, в то время как супермодели агрегируют информацию от своих субмоделей и организуют ее обмен с другими супермоделями.

$$HPABM = \langle G, G' \rangle \tag{2.1.7.1}$$

при условиях:

$G = \langle V, E \rangle;\ G' = \langle V', E' \rangle;$

$V = \{v_i | v_i = \langle m_i^{sp} | G_i' \rangle, m_i^{sp} \in SuperM, \forall i \in [1, n^{sp}]\};$

$E = \{A_{ij}^{sp} | A_{ij}^{sp} = 1, \text{если } m_i^{sp} u\ m_j^{sp} \text{ соединены; иначе } 0; \forall i, j \in [1, n^{sp}]\};$

$V' = \{v_k' | v_k' = \langle m_k^{sb} \rangle, m_k^{sb} \in SubM, \forall k \in [1, n^{sb}]\};$

$E' = \{A_{jl}^{sb} | A_{jl}^{sb} = 1, \text{если } m_j^{sb} u\ m_l^{sb} \text{ соединены; иначе } 0; \forall j, l \in [1, n^{sb}]\},$

где

G, G' — графы, содержащие супер- и субмодели;

SuperM, *SubM* — наборы супер- и субмоделей (будут определены позже);

V, E — вершины и ребра графа G, содержащего супермодели;

V', E' — вершины и ребра графа G', содержащего субмодели;

G_i' — субграф, содержащий субмодели i-й супермодели $G_i' = \langle V_i', E_i' \rangle$;

V_i', E_i' — вершины и ребра субграфа G_i';

$V_i' = \{v_j' | v_j' = \langle m_j' \rangle, m_j' \in SubM, m_j' \in m_i^{sp}, \forall j \in [1, n_i^{sb}], \forall i \in [1, n^{sp}]\}$;

$E_i' = \left\{ A_{kl}^{sb} \middle| \begin{matrix} A_{kl}^{sb} = 1, \text{если } m_k' \text{и } m_l' \text{ соединены и взаимодействуют} \\ (m_k' \in m_i^{sp} \text{ или } m_l' \in m_i^{sp}); \text{иначе } 0; \forall k, l \in [1, n^{sb}] \end{matrix} \right\}$;

$A_{ij}^{sp}, A_{jl}^{sb}, A_{kl}^{sb}$ — матрицы смежности для соответствующих графов;

v_i, v_j', v_k' — вершины соответствующих графов;

m_i^{sp}, m_j^{sp} — супермодели i и j в наборе супермоделей;

n^{sp}, n^{sb} — общее количество супер- и субмоделей;

m_j', m_k', m_l' — субмодели в i-й супермодели m_i^{sp};

n_i^{sb} — количество субмоделей в i-й супермодели;

$m_k^{sb}, m_j^{sb}, m_l^{sb}$ — субмодели, входящие в набор субмоделей.

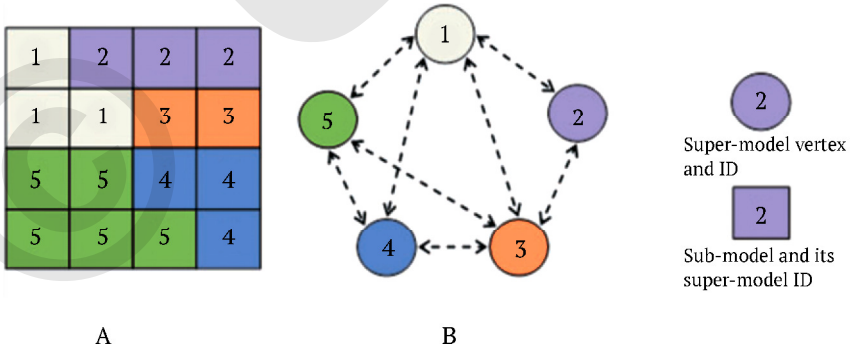


Рис. 2.1.7.1. Концептуальное представление модели, разработанной в HPABM и реализуемой в параллельном режиме (A — распределение супер- и субмоделей; B — граф, показывающий топологию супермоделей). *https://pages.charlotte.edu/wenwu-tang*

Упрощенно каркас HPABM может быть также представлен четырьмя уровнями (с соответствующими графами): модель, субмодель, супермодель, процессор (рис. 2.1.7.2). Агентам, связям между ними и среде их функционирования соответствует граф, представляющий динамику всей системы. Процессоры могут быть организованы в граф, отображающий аппаратную архитектуру, а супер- и субмодели, в свою очередь, представляются графами, связывающими агентную модель с вычислительными ресурсами. При использовании решеток для отображения набора субмоделей (см. рис. 2.1.7.1) набор вершин графа субмоделей может быть определен следующим образом:

$$V' = \left\{ v'_{ij} \middle| v'_{ij} = \langle m_{ij}^{sb} \rangle, m_{ij}^{sb} \in SubM, \forall i \in [1, nr], \forall j \in [1, nc] \right\}, \qquad (2.1.7.2)$$

где v'_{ij} — вершина в наборе V'; m_{ij}^{sb} — субмодель, находящаяся в i-м ряду и j-м столбце решетки набора субмоделей, nr (nc) — количество рядов (столбцов) решетки.

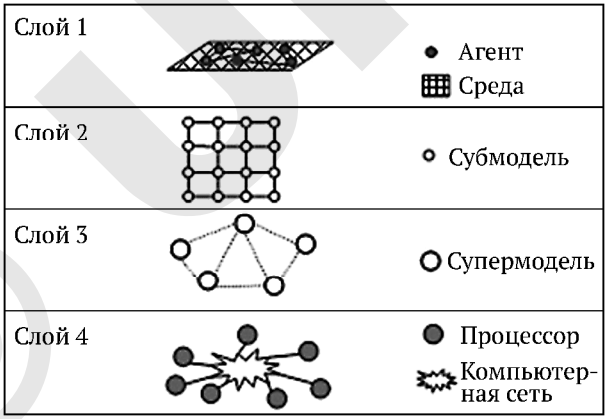


Рис. 2.1.7.2. Архитектура HPABM.
https://pages.charlotte.edu/wenwu-tang

Супермодели

Каждая супермодель организует порядок вычислений набора субмоделей, а также межпроцессорную связь и обмен информаци-

ей с другими супермоделями. Набор супермоделей *SuperM* (см. соотношение 2.1.7.1) может быть определен таким соотношением:

$$SuperM = \left\{ m_i^{sp} \middle| \begin{matrix} m_i^{sp} = \langle spid, Model^{sb}, NB^{sp}, ps, Processors, Buffer^{sp} \rangle, \\ \forall i \in [1, n^{sp}] \end{matrix} \right\}, \quad (2.1.7.3)$$

где m_i^{sp} — i-я супермодель, представляющая собой шестикомпонентную структуру; n^{sp} — общее количество супермоделей; *spid* — идентификационный номер супермодели, определяемый набор субмоделей $Model^{sb}$. Далее NB^{sp} обозначает супермодели, являющиеся соседями m_i^{sp}, а *ps* хранит информацию о процессоре, с которым супермодель связана. *Processors* — набор процессоров, используемых для вычисления агентной модели, и он может динамически обновляться в процессе расчетов. *Buffer*^{sp} является буфером обмена информацией относительно состояния агентов и их среды между супермоделями.

Таким образом, супермодели обеспечивают механизм связывания между процессорами суперкомпьютера и агентными моделями. Супермодели могут быть представлены в виде направленного графа (рис. 2.1.7.3), в котором каждая супермодель является вершиной, а отношения между ними — ребрами.

Субмодели

Выше было сказано, что субмодели являются основными вычислимыми единицами НРАВМ, и при этом соотношение (2.1.7.4) определяет набор субмоделей *SubM* (упомянутый в соотношении 2.1.7.1). Каждая субмодель связана с соответствующей супермоделью посредством информации *model*^{sp}. Используя информацию о своих соседях, хранящуюся в *NB*^{sb}, субмодель может сообщить, информация от какой супермодели ей нужна для дальнейшей оптимизации межпроцессорного сообщения.

$$SubM = \left\{ m_k^{sb} \middle| m_k^{sb} = \langle sbid, AG, EN, model^{sp}, NB^{sb} \rangle, \forall k \in [1, n^{sb}] \right\}, \quad (2.1.7.4)$$

где

m_k^{sb} — k-я субмодель из набора субмоделей;

sbid — идентификационный номер субмодели;

AG, *EN* — наборы агентов и их среда, рассматриваемые в субмодели;
$model^{sp}$ — супермодель к которой соотносится m_k^{sb};
NB^{sb} — набор субмоделей, являющихся соседями m_k^{sb};
n^{sb} — общее количество субмоделей.

Набор субмоделей может быть представлен в виде направленного графа (рис. 2.1.7.3), описывающего топологическое соотношение между ними. В свою очередь, топологическое соотношение необходимо для передачи данных, вызванной межагентным взаимодействием.

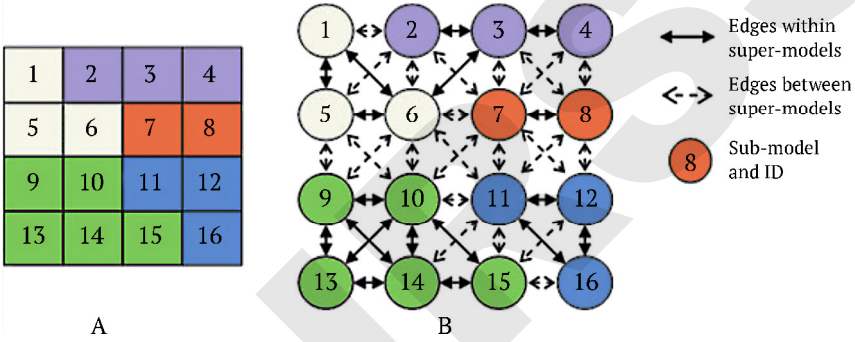


Рис. 2.1.7.3. Набор субмоделей, разработанных в HPABM (A — распределение субмоделей; B — граф субмоделей, где супермодели дифференцированы по цвету). *https://pages.charlotte.edu/wenwu-tang*

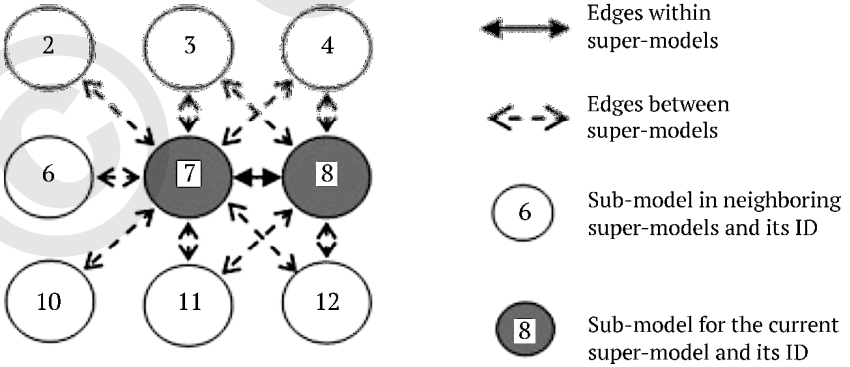


Рис. 2.1.7.4. Субграф, связывающий суб- и супермодели. *https://pages.charlotte.edu/wenwu-tang*

Субмодели в рамках супермодели также могут быть представлены в виде субграфа (см. G_i' в соотношении 2.1.7.1 и рис. 2.1.7.4) более общего графа всех субмоделей. Таким образом может быть получена иерархическая структура всех элементов, входящих в агентную модель.

Техническая реализация иерархической платформы для параллельных агентных моделей

Реализация параллельных агентных моделей в HPABM осуществляется следующим типовым алгоритмом (листинг 2.1.7.1). В силу того, что агенты и их среда динамически изменяются, синхронизация и обмен данными между процессорами в значительной степени определяют производительность агентной модели. HPABM представляет возможность разработчикам распараллеливать модели на уровне супер- и субмоделей, не касаясь программирования на более низком уровне вычислительных ресурсов.

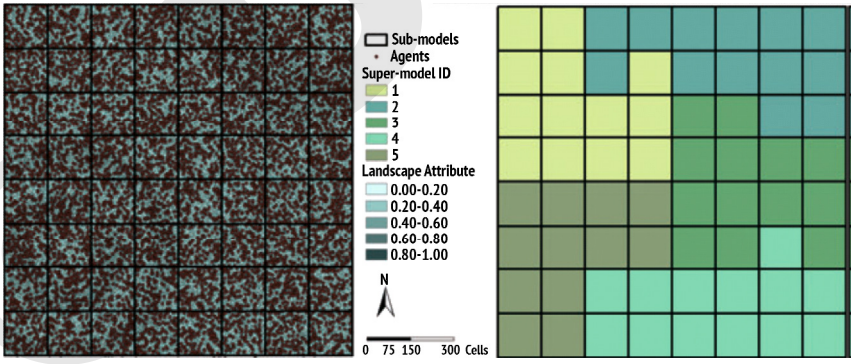
Листинг 2.1.7.1. Алгоритм реализации агентной модели, разработанной в HPABM

```
Начало алгоритма
1. Инициализация модели
   1.1. Определение степени необходимой детализации модели
   1.2. Дезагрегирование модели на субмодели
   1.3. Построение топологии субмоделей
   1.4. Присваивание субмоделей к супермоделям
   1.5. Построение топологии супермоделей
2. Выполнение модели
   For на каждой итерации выполняется проверка
      If Is_ReBuild_SuperModel_Topology() == True Then
         Перепостроение топологии супермоделей
         Перенос субмоделей между супермоделями
      EndIf
      For супермодели sp, рассчитываемой на процессоре p
         For каждой субмодели sb, входящей в супермодель sp
                                   Обновление субмодели sb
```

```
            End For
        End For
        Синхронизация модели
    End For
3. Получение результатов от супермоделей
4. Окончание работы модели
Окончание алгоритма
```

Пример использования

В качестве примера использования HPABM разработчиками была реализована простая пространственная агентная модель «Хищник — жертва». Пространство задано двумерной решеткой, ячейки которой содержат различное количество ресурсов, изменяемое в единицу времени. Агенты модели отличаются несколькими характеристиками (размер, возраст и др.), а также правилами поведения, что поразному влияет на отношения «агент — среда». Представленная на решетке модель разделена на области (содержащие агентов и часть среды), каждая из которых обрабатывается субмоделью. В свою очередь, субмодели объединены в супермодели (рис. 2.1.7.5).



(a) (b)

Рис. 2.1.7.5. Реализация простой агентной модели в HPABM ((a)— агенты и среда, (b) — супер- и субмодели; размер ландшафта — 1200 × 1200, количество агентов — 10 000; пространство субмоделей — 8 × 8, ландшафт, обрабатываемый одной субмоделью — 150 × 150, количество супермоделей — 5).
https://pages.charlotte.edu/wenwu-tang

Поскольку агенты в модели подвижны, на каждом шаге супермодели обрабатывают перемещающихся агентов через буфер (см. соотношение 2.1.7.3 и листинг 2.1.7.1).

Эксперименты

Для расчетов авторы использовали суперкомпьютер Mercury, расположенный в Национальном центре суперкомпьютерных приложений Иллинойского университета. Платформа HPABM реализована на Си с использованием программного интерфейса MPI для распараллеливания. При проведении экспериментов с упомянутой выше моделью отслеживалось влияние детализации пространства на производительность параллельной версии с использованием двух вычисляемых характеристик — ускорение ($S_{пр}$, уравнение 2.1.7.5) и эффективность ($E_{пр}$, уравнение 2.1.7.6).

$$S_{пр} = T_1 / T_n, \tag{2.1.7.5}$$

$$E_{пр} = S_{пр} / n, \tag{2.1.7.6}$$

где T_1 — время выполнения последовательной версии алгоритма, T_n — время выполнения параллельной версии на n процессорах.

Для оценки влияния детализации пространства были проведены расчеты по четырем сценариям: G_1, G_2, G_3, G_4, которые отличаются размером решетки (см. табл. 2.1.7.1). Размер ландшафта в тестируемой модели 12 800 × 12 800, количество агентов — 409 600, а количество итераций — 200. В рамках каждого сценария модель запускалась с различным числом процессоров от 2 до 16, изменяемым с шагом 2, со средним числом прогонов равным 10 (для получения усредненных оценок).

В табл. 2.1.7.2 приведено время выполнения агентной модели для четырех сценариев. Как видно, время расчетов уменьшается в зависимости от использования дополнительных процессоров. Отметим, что для последовательной версии модели среднее время вычислений составило 1126,3 секунд.

Таблица 2.1.7.1

Параметры сценариев для получения оценок производительности

Сценарии	**Размер решетки субмоделей**	**Размер ландшафта для субмоделей**	**Количество агентов на одну субмодель**
G_1	8 × 8	1600 × 1600	6400
G_2	16 × 16	800 × 800	1600
G_3	32 × 32	400 × 400	400
G_4	64 × 64	200 × 200	100

Таблица 2.1.7.2

Время выполнения агентной модели для четырех сценариев в секундах

Число процессоров	G_1	G_2	G_3	G_4
2	1009,6	962,1	1026,5	2162,7
4	569,6	530,7	519,5	735,7
6	458,7	437,0	415,6	492,8
8	447,3	402,5	389,7	422,3
10	411,6	383,2	365,2	384,9
12	407,5	376,5	366,1	358,3
14	390,3	367,4	350,2	333,9
16	385,4	359,3	349,2	318,5

На рис. 2.1.7.6 показаны результаты оценок производительности модели, рассчитанные по формулам (2.1.7.5) и (2.1.7.6). Как видно, ускорение увеличивается, а эффективность расчетов снижается в зависимости от прироста числа процессоров.

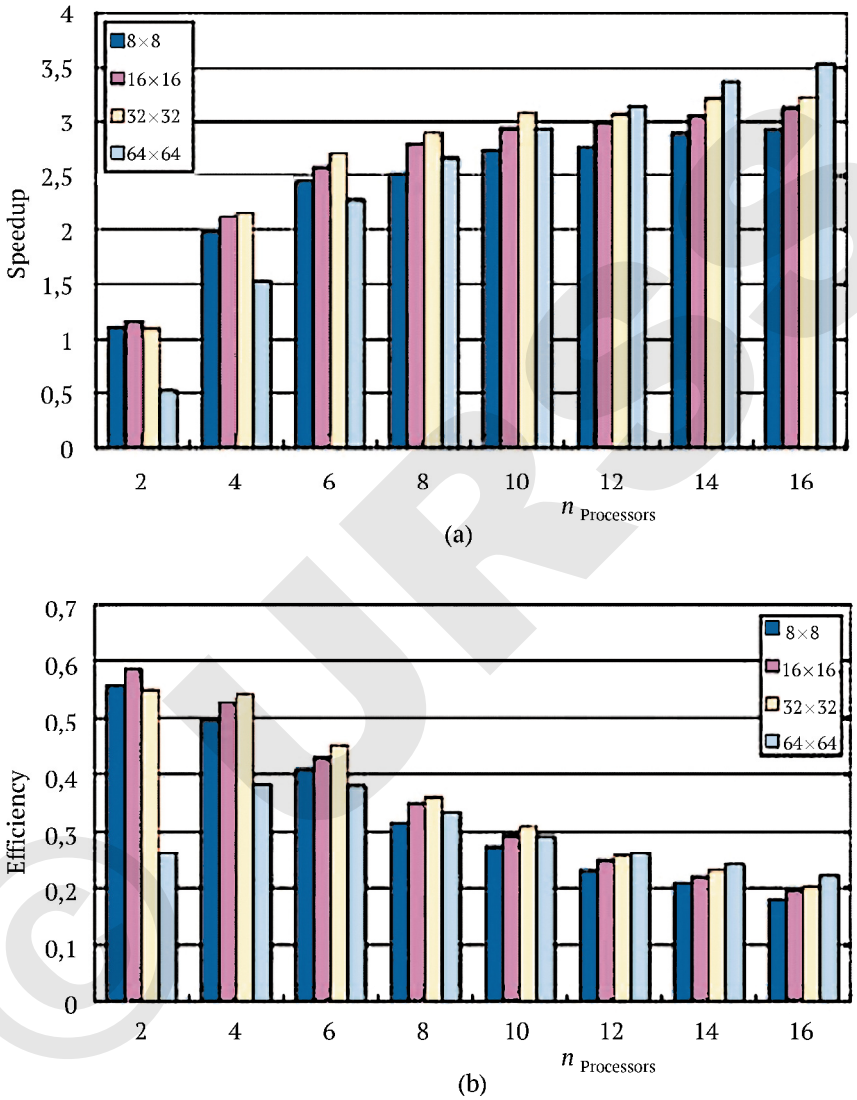


Рис. 2.1.7.6. Результаты оценки производительности для параллельной версии модели (a: ось абсцисс — число процессоров, ось ординат — ускорение; b: ось абсцисс — число процессоров, ось ординат — эффективность). *https://pages.charlotte.edu/wenwu-tang*

Далее были проведены два эксперимента (Exp_1 и Exp_2) для оценки масштабируемости модели в зависимости от размера пространства и количества агентов. В табл. 2.1.7.3 приведены параметры для экспериментов — для каждого сценария размер решетки для субмоделей составляет 32 × 32, а количество супермоделей для всей агентной модели — 16. Число прогонов для каждого сценария — 10, а количество итераций в рамках каждого прогона — 200.

Таблица 2.1.7.3

Параметры для проведения экспериментов

Эксперименты	Параметры для субмоделей		Параметры для всей модели	
	Размер ландшафта	**Количество агентов**	**Размер ландшафта**	**Количество агентов**
Exp_1	200 × 200	400	6400 × 6400	409 600
	400 × 400	400	12 800 × 12 800	409 600
	600 × 600	400	19 200 × 19 200	409 600
	800 × 800	400	25 600 × 25 600	409 600
	1000 × 1000	400	32 000 × 32 000	409 600
Exp_2	400 × 400	200	12 800 × 12 800	204 800
	400 × 400	400	12 800 × 12 800	409 600
	400 × 400	600	12 800 × 12 800	614 400
	400 × 400	800	12 800 × 12 800	819 200
	400 × 400	1000	12 800 × 12 800	1 024 000
	400 × 400	1200	12 800 × 12 800	1 228 800

На рис. 2.1.7.7 и 2.1.7.8 показаны результаты расчетов для параллельной версии агентной модели. Как видно, ускорение и эффективность прирастают в зависимости от увеличения ландшафта и снижаются при увеличении числа агентов.

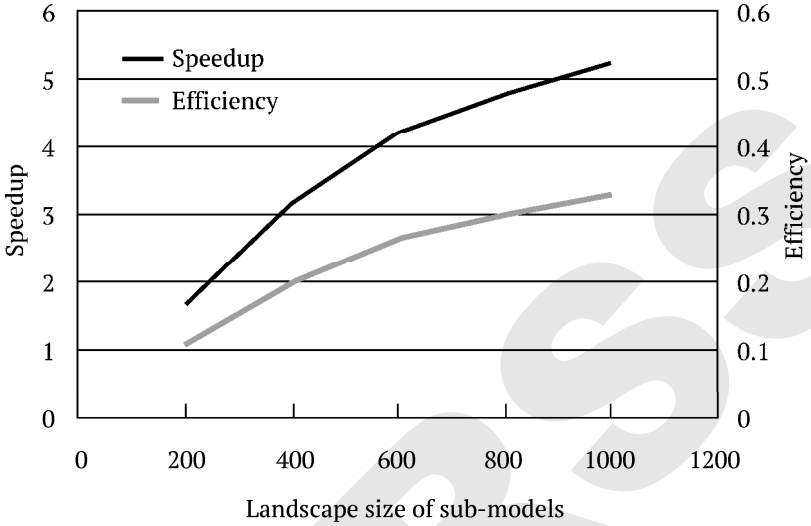


Рис. 2.1.7.7. Оценки производительности агентной модели в зависимости от размера ландшафта. *https://pages.charlotte.edu/wenwu-tang*

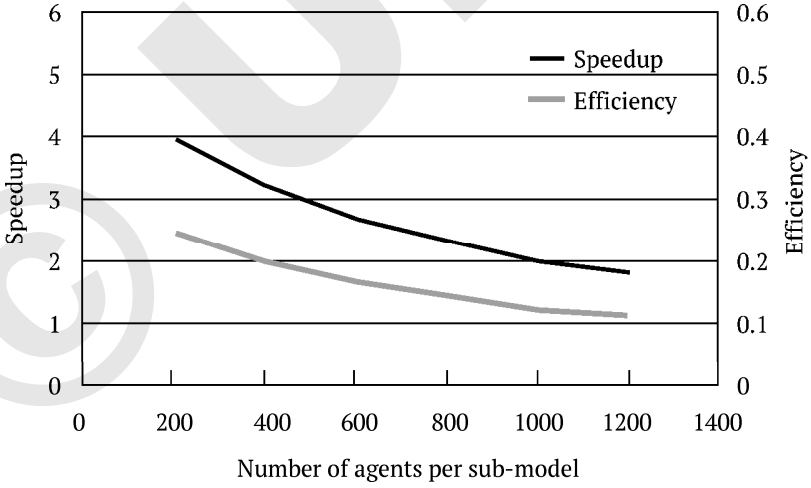


Рис. 2.1.7.8. Оценки производительности агентной модели в зависимости от числа агентов. *https://pages.charlotte.edu/wenwu-tang*

В табл. 2.1.7.4 приведены результаты оценки производительности модели с размером ландшафта 12 800 × 12 800, числом агентов 409 600 и размером решетки для субмоделей 16 × 16 в зависимости от изменения числа процессоров. При этом отслеживаемые величины ведут себя вполне ожидаемо.

Таблица 2.1.7.4

Производительность параллельной версии модели

Число процессоров	Время выполнения	Ускорение	Эффективность
1	5291	1,00	1,00
10	990	5,34	0,53
20	495	10,69	0,53
30	346	15,29	0,51
40	269	19,67	0,49
50	232	22,81	0,46

HPABM также использовалась для агентных моделей большой размерности, техническая реализация которых была аналогична описанной выше, и во всех случаях показала свою эффективность. Более подробно про HPABM можно прочитать в публикациях автора — Wenwu Tang: https://pages.charlotte.edu/wenwu-tang

2.1.8. Анатомические исследования с использованием агент-ориентированного подхода и параллельных вычислений

Одна из проблем, стоящих перед исследователями, специализирующимися в области биомедицины, заключается в возможности интеграции детализированных молекулярных механизмов различных болезней с конструированием эффективных средств соответствующей терапии. В идеале было бы хорошо иметь в арсенале симуляторы, позволяющие проводить оценку развития болезней на различных уровнях (клеточном, отдельных органов и организма в

целом). В Чикагском университете была разработана модель эпителия кишечника с использованием агент-ориентированного подхода и параллельных вычислений, позволяющая имитировать процесс воспаления подвздошной кишки в случае язвенного колита [11]. Хотя это исследование напрямую не относится к общественным наукам, однако разработанное высокопроизводительное программное обеспечение может быть использовано для реализации агентных моделей в других областях (в частности, при моделировании социальных систем).

По мнению ученых, использование агентного подхода как нельзя лучше подходит для отображения деятельности миллиардов клеток, содержащих ДНК и РНК, но, с другой стороны, обработка такого массива информации требует использования суперкомпьютеров, и в этой связи исследователями была разработана пространственная модель общего назначения для исследования кишечной ткани (Spatially Explicit General-purpose Model of Enteric Tissue, SEGMEnT).

SEGMEnT разработана на языке C++ с использованием программного интерфейса MPI для выполнения межпроцессорных коммуникаций. Можно выделить следующие ключевые особенности приложения.

1. Использование набора двумерных сеток, которые затем преобразуются в трехмерную конструкцию, имитирующую поверхность исследуемой ткани (рис. 2.1.8.1). В ходе симуляции используемые сетки могут расти или сжиматься для большего соответствия реальным физиологическим процессам.
2. SEGMEnT использует ряд неблокирующих коммуникационных операций, позволяющих эффективно масштабировать систему.
3. Несмотря на изначальную направленность SEGMEnT в качестве инструмента для исследования кишечной ткани, это программное средство может использоваться для разработки агентных моделей и для других областей.

При распараллеливании модели в SEGMEnT, вся площадь исследуемой ткани делится равномерно по числу используемых процессоров (рис. 2.1.8.1). На панели *A* изображен илеоанальный

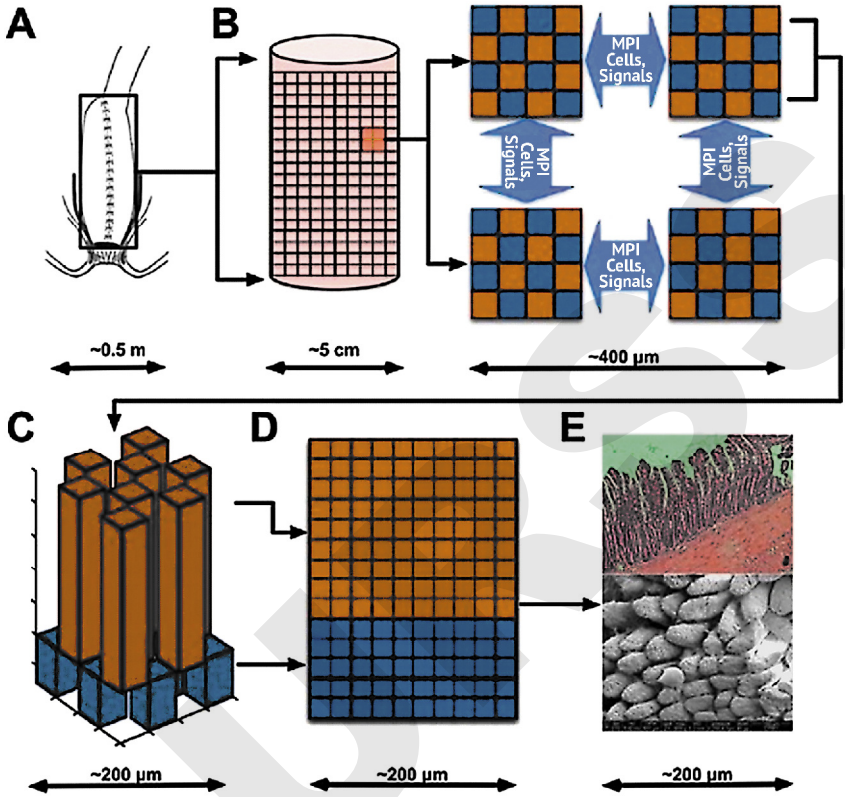


Рис. 2.1.8.1. Симуляция работы илеоанального резервуара типа J (J-pouch). *Towards Anatomic Scale Agent-Based Modeling with a Massively Parallel Spatially Explicit General-Purpose Model of Enteric Tissue* (*SEGMEnT_HPC*) [11]

резервуар типа *J* (*J-pouch*). Панель *B* представляет собой виртуальную проекцию *J-pouch* в виде цилиндра, распределяемую между процессорами суперкомпьютера. Панель *C* показывает, как крипты (crypt) и ворсинки (villus) представлены на одном процессоре, а на панели *D* показывается как они преобразуются в двумерную сетку, где коричневым цветом обозначены ворсинки, а синим — крипты. И, наконец, на панели *E* изображена часть биологической системы, обрабатываемой с помощью SEGMEnT, причем в верхней части рисунка — результат гистологического анализа, а в нижней — изо-

бражение, полученное с помощью сканирующего электронного микроскопа.

Сигнальные молекулы и пример передающей их сети приведены на рис. 2.1.8.2. Морфогены окрашены синим цветом, а воспалительные компоненты — оранжевым. В свою очередь, стимулирующие контакты изображены зелеными стрелками, а подавляющие — красными. Сети включают в себя сигнальный путь Wnt (от англ. «wingless» — «бескрылый» и «int» — «ген»), костные морфогенетические белки (BMP, bone morphogenetic proteins), фосфатазу и гомолог тензина / фосфоинозитид-3-киназы (phosphotase and tensin homolog / phosphoinositide 3-kinase (PTEN/PI3K)) и др.

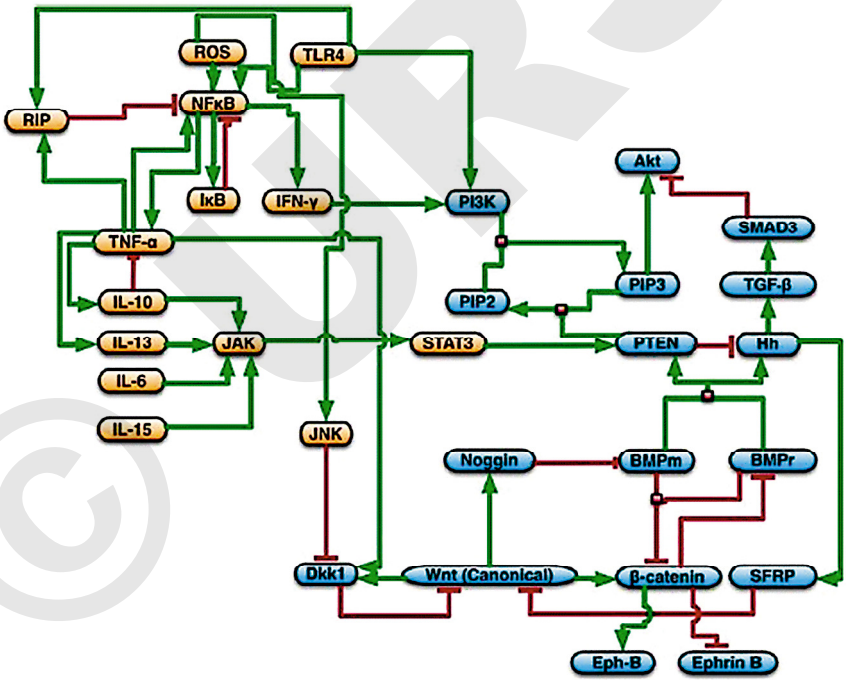


Рис. 2.1.8.2. Передающая сеть в SEGMEnT. *Towards Anatomic Scale Agent-Based Modeling with a Massively Parallel Spatially Explicit General-Purpose Model of Enteric Tissue* (*SEGMEnT_HPC*) [11]

Виртуальные проекции криптов и ворсинок, расположенные на границе обслуживаемого процессорами пространства содержат дополнительные буфера для сохранения связи с соседними процессорами.

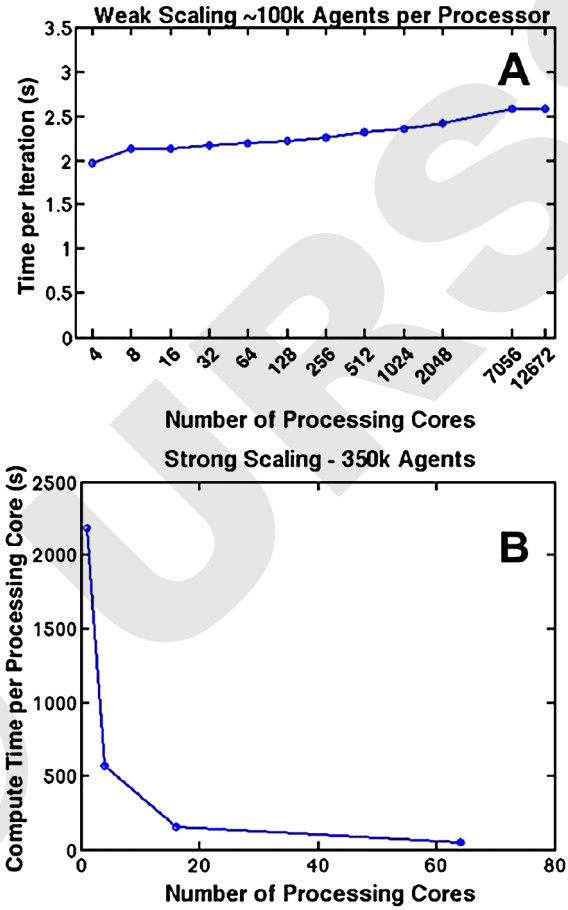


Рис. 2.1.8.3. Результаты масштабирования. *Towards Anatomic Scale Agent-Based Modeling with a Massively Parallel Spatially Explicit General-Purpose Model of Enteric Tissue* (*SEGMEnT_HPC*) [11]

© URSS

Расчеты проводились на суперкомпьютерах Beagle и Midway, расположенных в Чикагском университете, а на рис. 2.1.8.3 показаны результаты масштабирования.

На Панели *A* показано время выполнения итерации в зависимости от количества используемых процессорных ядер, при этом каждый процессор рассчитывал результат для 100 000 агентов. По мере увеличения используемых процессорных ядер с 4 до 12 672 среднее время выполнения итераций увеличилось примерно на 25 %, и, как видно, по мере приближения к 10 000 ядрам, кривая начинает постепенно выравниваться. В свою очередь, на панели *B* показана производительность приложения для 350 000 агентов, в зависимости от количества используемых процессорных ядер (1, 4, 16 и 64), при этом каждое ядро рассчитывало 256, 64, 16 и 4 конструкций из криптов и ворсинок.

На рис. 2.1.8.4 приведены другие результаты масштабируемости модели — время расчетов в зависимости от числа используемых процессоров для конструкций из криптов и ворсинок размерности 4, 16 и 64. Как видно, по мере увеличения размера ткани эффективность масштабируемости приближается к идеальной (панель *C*). Максимальное число используемых ядер составляло 12 672, но, по мнению разработчиков, у SEGMEnT есть отличный потенциал для масштабирования на суперкомпьютерах с числом ядер более 100 000.

На рис. 2.1.8.5 показана зависимость общего количества моделируемых клеток (агентов) для образца ткани размером 1129 мм^2 от расчетного времени с использованием 7056 процессоров. Соотношение ворсинок к криптам составляет примерно 4 к 1, что соответствует результатам гистологии. Максимальное количество моделируемых клеток в рамках проведенной симуляции составило 1 038 136 974, причем первоначальный рост клеточной популяции связан с процедурой инициализации модели.

На рис. 2.1.8.6 показан результат визуализации фрагмента выходных данных SEGMEnT. Панель *A* демонстрирует здоровую ткань, *B* — небольшое воспаление, *C* — восстановленную ткань после тяжелого воспаления, а *D* — ткань с уменьшенной высотой ворсинок.

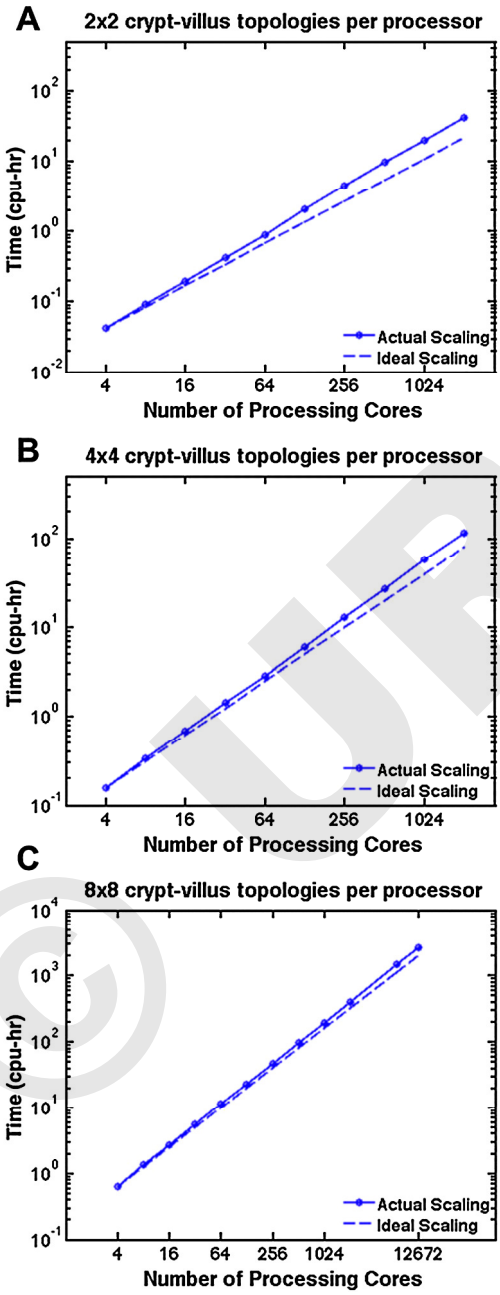


Рис. 2.1.8.4. Масштабируемость в зависимости от размеров кишечной ткани. *Towards Anatomic Scale Agent-Based Modeling with a Massively Parallel Spatially Explicit General-Purpose Model of Enteric Tissue* (*SEGMEnT_HPC*) [11]

© URSS

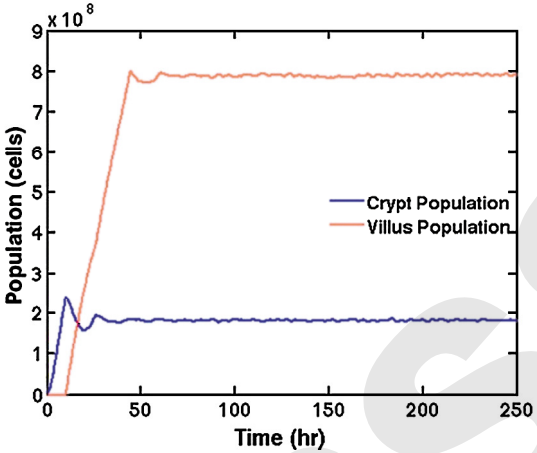


Рис. 2.1.8.5. Общее количество клеток моделируемой ткани. *Towards Anatomic Scale Agent-Based Modeling with a Massively Parallel Spatially Explicit General-Purpose Model of Enteric Tissue* (*SEGMEnT_HPC*) [11]

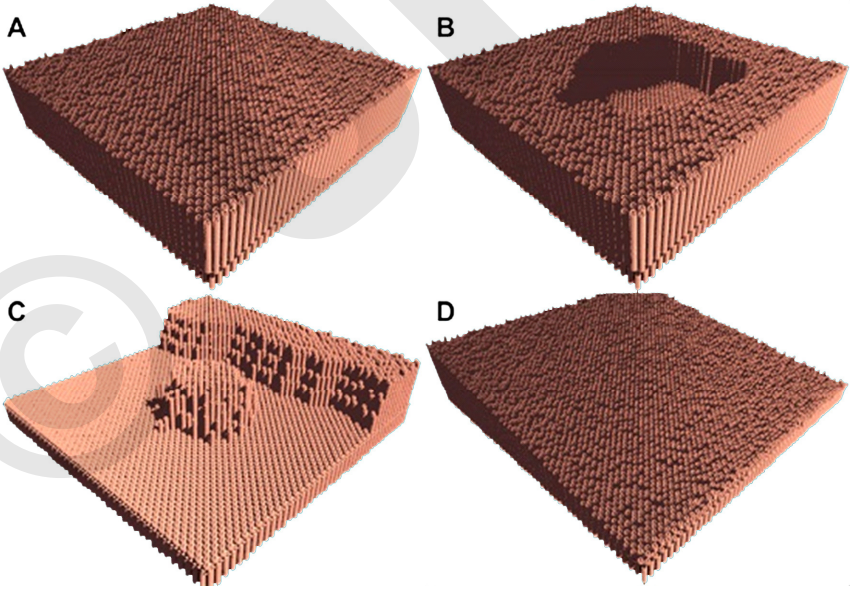


Рис. 2.1.8.6. Рендеринг ткани. *Towards Anatomic Scale Agent-Based Modeling with a Massively Parallel Spatially Explicit General-Purpose Model of Enteric Tissue* (*SEGMEnT_HPC*) [11]

Успешная реализация и использование модели SEGMEnT позволяет рассчитывать на дальнейшее развитие системы для исследования других органов человеческого организма. Разработчики проанализировали аналогичные инструментальные средства, позволяющие разрабатывать агентные модели для суперкомпьютеров. Среди протестированных программ:

1. FLAME, позволяющая распараллеливать агентные модели для систем с несколькими сотнями процессоров. В настоящее время это ПО используется не только для анализа биологических систем (для чего была разработана изначально), но и для построения моделей больших социально-экономических систем.
2. RepastHPC — разработка Аргоннской национальной лаборатории, изначально используемая для симуляций систем в социальных науках, но может быть адаптирована и для анализа биологических систем. Тестирование этого ПО со стороны ученых Чикагского университета показало возможность масштабирования до реализации на 32 678 процессорных ядрах.
3. Biocellion — средство для реализации агентных моделей биологических систем на системах с 4000 ядрами (также по оценкам чикагских ученых).

В свою очередь, исследователи отмечают, что все эти средства не лишены недостатков, главный из которых — недостаточная масштабируемость. Если речь идет о моделировании более сложных процессов, к примеру, имитации работы всех клеток толстой кишки в течение одного года, то по оценкам разработчиков SEGMEnT потребуется задействование 450 000 процессоров в течение 30 часов. Для имитации работы всех клеток тонкого и толстого кишечников в течение года по мнению разработчиков потребуется работа уже 2,4 млн процессорных ядер в течение 40 часов. А если моделировать работу клеток всего организма с учетом развития различных заболеваний, то исследовательская группа увязывает

эту возможность с появлением суперкомпьютеров эксафлопсной производительностью, причем теоретически SEGMEnT к этому уже готова.

2.1.9. Агент-ориентированные модели для систем с экзафлопсной производительностью

Исследователи из Аргоннской национальной лаборатории (Argonne National Laboratory) работают над созданием агентных моделей нового поколения, которые, по мнению разработчиков, должны позволить достичь существенного прогресса в науке и технике (и, в том числе, в общественных дисциплинах). Эти модели будут вычисляться в рамках систем с эксафлопсной производительностью (10^{18} флоп/с), которые могут появиться уже в ближайшие годы (2016–2018 гг.) на смену существующим суперкомпьютерам петафлопсного уровня (то есть порядка 10^{15} флоп/с).

На данный момент ученые разработали и используют крупномасштабную модель — комплексную адаптивную систему рынка электроэнергии (Electricity Markets Complex Adaptive Systems, EMCAS), используемую для оценки последствий дерегулирования рынка электроэнергии в штате Иллинойс (рис. 2.1.9.1).

В этой модели рассматривается подробная электрическая сеть с действующими на дерегулируемом рынке агентами — производителями и потребителями электроэнергии. Агенты — генерирующие компании — адаптируют свое поведение в зависимости от изменений рыночных условий, максимально приближенных к реальности (рис. 2.1.9.2).

Помимо этого Аргоннская лаборатория известна своими суперкомпьютерными моделями в области биологии, в военном деле, в социальных науках и др.

Вернемся к агентным моделям нового поколения, которые предполагается реализовывать в средах с эксафлопсной производительностью.

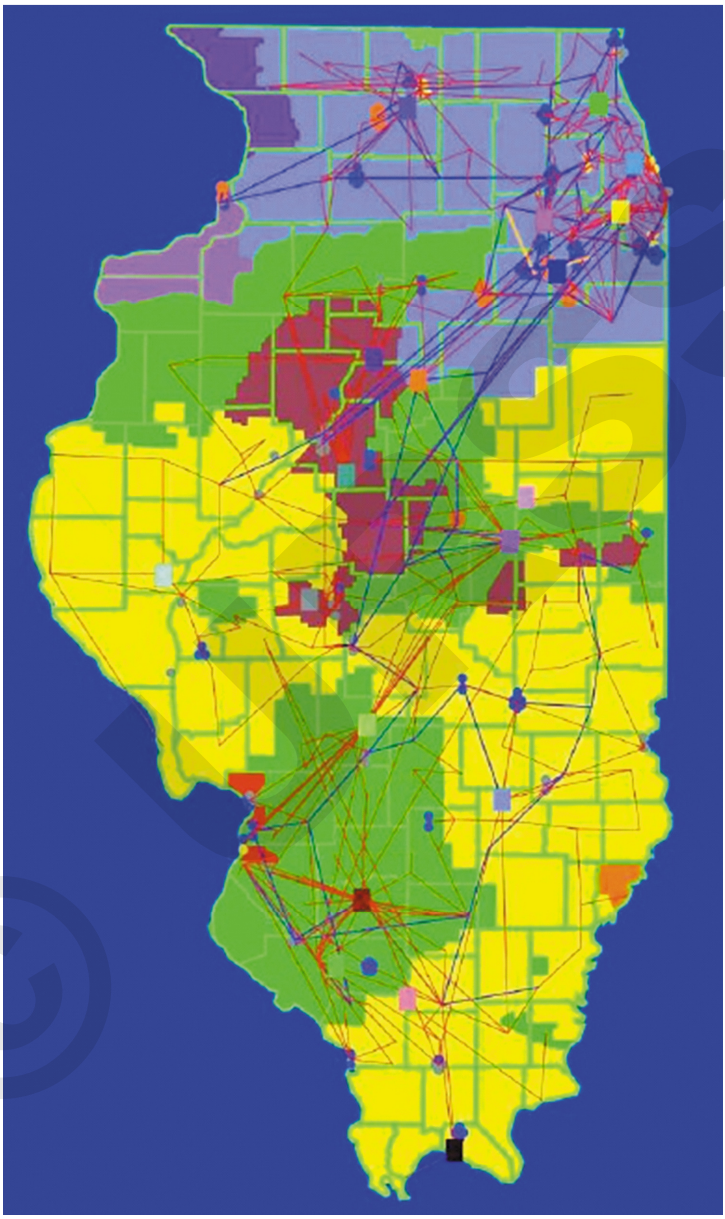


Рис. 2.1.9.1. Электрическая сеть штата Иллинойс и пункты обслуживания. *Аргоннская национальная лаборатория, http://www.alcf.anl.gov*

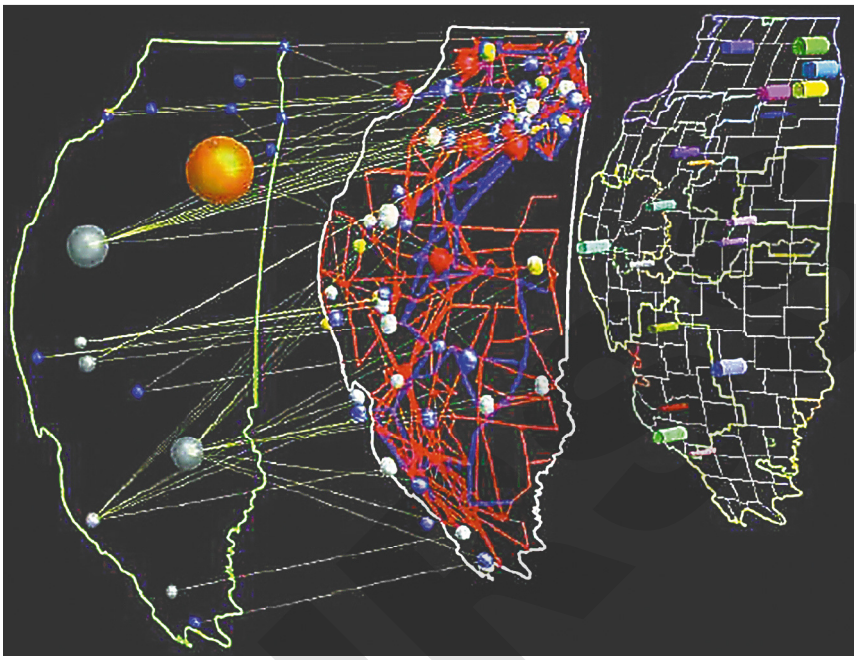


Рис. 2.1.9.2. Элементы рынка электроэнергии штата Иллинойс: генерирующие компании (слева), электрическая сеть (в центре) и пункты обслуживания (справа). *Аргоннская национальная лаборатория, http://www.alcf.anl.gov*

По словам заместителя директора Аргоннского центра комплексного адаптивного агентного моделирования (Argonne's Center for Complex Adaptive Agent Systems Simulation) Майкла Норта, (Michael North) разработка агентных моделей для эксафлопсных систем — далеко не тривиальная задача: «...Переход от моделирования нескольких взаимодействий до миллионов не просто увеличение в разрядности, а принципиальное изменение технологии моделирования. Меняется пространство имен, используются другие методы работы с памятью и т. д. Также следует отметить, что переключение работы моделей на архитектуру с кратным увеличением узлов не является линейно масштабируемым процессом, а требует дополнительных и весьма больших усилий по их переносу...»

Пилотные проекты Аргоннской национальной лаборатории в области агентных моделей с эксафлопсной производительностью

На данный момент таких проектов три: 1) биоразнообразие микроорганизмов; 2) кибербезопасность; 3) социальные аспекты изменения климата.

В качестве возможного инструмента для построения соответствующих агентных моделей исследователи предполагают использовать продукт Аргоннской национальной лаборатории — RepastHPC.

Этот инструмент был неоднократно использован для широкого диапазона приложений. К примеру, для Национального управления по воздухоплаванию и исследованию космического пространства (NASA) была построена агентная модель, имитирующая передвижение по марсианской поверхности (рис. 2.1.9.3).

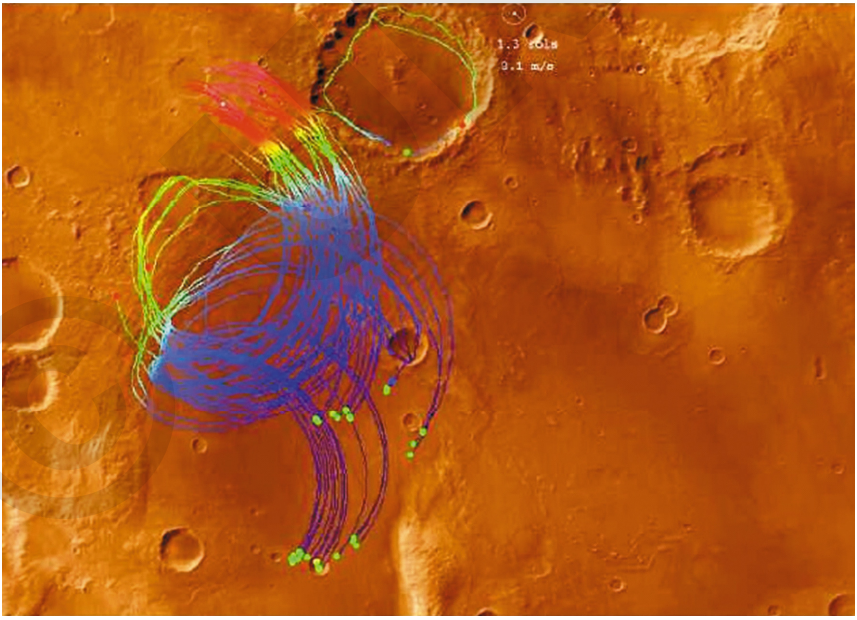


Рис. 2.1.9.3. Автономные вездеходы, координирующие друг с другом свои действия и осуществляющие движение по поверхности виртуального Марса в рамках агентной модели. *Аргоннская национальная лаборатория, http://www.alcf.anl.gov*

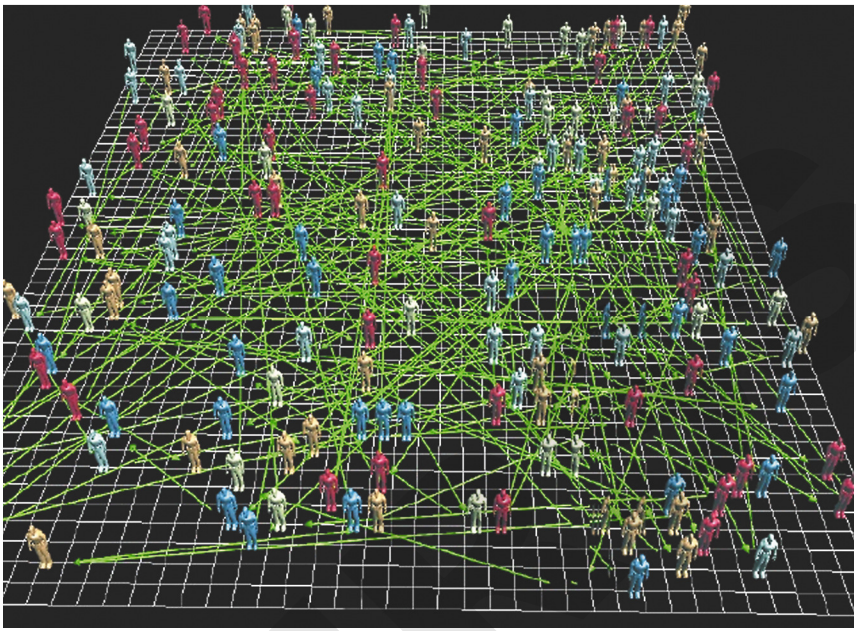


Рис. 2.1.9.4. Динамическое моделирование социальной сети в рамках агентной модели. *Аргоннская национальная лаборатория, http://www.alcf.anl.gov*

Другое использование упомянутого инструмента заключалось в моделировании динамической социальной сети (рис. 2.1.9.4).

В рамках упомянутого выше третьего пилотного проекта Аргоннской национальной лаборатории в области агентных моделей с эксафлопсной производительностью, связанного с социальными аспектами изменения климата, предполагается углубленно рассмотреть экономические и социальные эффекты от изменения окружающей среды. По мнению разработчиков, они будут важны для долгосрочного прогнозирования климата, поскольку изменения в этих областях могут оказать существенное влияние на физические факторы климата. К примеру, выбор топлива, а также переход на новые технологии в энергетике, скорее всего, повлияют на уровень концентрации двуокиси углерода в атмосфере и наоборот.

Если в области моделирования физических аспектов климата (атмосферные потоки, океаническая циркуляция и др.) исследователи добились ощутимых успехов, то в вопросах моделирования населения планеты на долгосрочный период (когда численность может достичь 10^{10} чел.) предстоит большая работа. Предполагается, что в модели у каждого агента будет 10^3 (!) внутренних состояний, и, кроме того, планируется рассмотреть широкий спектр поведенческих типов. Работа началась недавно, и на данный момент информации о моделировании населения исследователи в открытый доступ не выкладывают.

Более подробно про описанные выше разработки можно прочитать на следующем сайте: http://www.alcf.anl.gov.

2.1.10. Microsoft AXUM — агент-ориентированное программирование

В 1990 г. профессор Стэнфордского университета Йоав Шохам предложил новую парадигму программирования — *агент-ориентированный подход*, концепция которого заключается в представлении агента в качестве программного модуля, выполняющего поставленные задачи. Поведение агента зависит от окружающей среды, которую он воспринимает с помощью датчиков и он, в свою очередь, может воздействовать на нее посредством исполнительных механизмов.

По мнению Шохама, агент-ориентированный подход является частным случаем более общей и широко известной парадигмы программирования — объектно-ориентированного программирования. В табл. 2.1.10.1 приведены сравнительные характеристики этих двух подходов [12].

Если объект в рамках объектно-ориентированной парадигмы программирования определяется набором методов и атрибутов, то агент задается описанием своего поведения в более широком смысле.

Идея агент-ориентированного программирования предполагает, что агент является программной сущностью с определенной степенью автономности и целью выполнения задач для пользователя.

Таблица 2.1.10.1

Сравнение двух парадигм программирования

	Объектно-ориентированное программирование	**Агент-ориентированное программирование**
Базовая единица	Объект	Агент
Параметры, которые определяют состояние базовой единицы	Без ограничений	Убеждения, обязательства, возможности, выбор и др.
Процесс вычислений	Получение сообщений и методы ответа	Получение сообщений и методы ответа
Виды сообщений	Без ограничений	Информирование, запрос, предложение, обещание, отказ и др.
Ограничения используемых методов	Нет	Честность, последовательность и др.

При этом считается, что агенты автономнее, чем объекты, и их поведение более гибкое, а также более социальное — в том смысле, что соответствующий программный код выполняется не автоматически, а по решению самого агента в зависимости от ситуации, задач, текущих приоритетов и параметров среды.

В 2009 г. одна из крупнейших транснациональных компаний по производству программного обеспечения — Microsoft (а точнее DevLabs — подразделение Microsoft, занимающееся экспериментальными и инновационными программными разработками) анонсировала новый язык программирования — Axum, который был изначально предназначен для написания многопоточных параллельных приложений, оптимизированных для работы на многоядерных процессорах. Другой, не менее важной особенностью нового языка является реализация в нем агент-ориентированной парадигмы программирования. Таким образом, экспериментальный язык вобрал в себя многие прогрессивные концепции программирования, а основной целью его создания стала проверка

совместимости перечисленных выше парадигм в рамках одного языка.

По синтаксису и ряду конструкций Axum наиболее близок к языку C#, однако они концептуально разные. Ниже приведены их общие элементы и основные различия.

Общее

- все C# 3.0 выражения, в том числе LINQ (Language Integrated Query) запросы;
- все C# 3.0 операторы;
- объявление полей и методов;
- делегаты и перечисляемые типы.

Не поддерживаемые в Axum элементы языка C#

- классы, интерфейсы и структуры;
- объявление операторов;
- свойства;
- константные поля и константные переменные;
- статические поля и методы.

Уникальные особенности Axum

- агенты и домены;
- каналы;
- схемы;
- сети;
- асинхронные потоки.

Таким образом, главные концепции Axum — каналы, схемы, слои, домены и агенты. Первое, что бросается в глаза, — отсутствие в Axum классов. Вместо них в новом языке есть новый абстрактный тип — *агенты*, которые хоть и похожи на классы, но на них нельзя ссылаться. Однако взаимодействие между агентами происходит посредством передачи сообщений в рамках каналов. Агенты близки по смыслу потокам, но при этом они гораздо «легче» потоков, и, вообще говоря, количество агентов может составлять сотни, тысячи или даже миллионы в рамках одной программы.

© URSS

Каналы представляют собой набор портов, каждый из которых поддерживает асинхронную буферизацию со строгой семантикой. Каналы могут использоваться для обеспечения связи между распределенными частями приложения, использующего одинаковую агент-ориентированную программную модель. *Порты* — это основные элементы для определения канала, и при этом каждый порт определяется посредством ориентации, названия и типа.

Проиллюстрируем идею каналов на примере [13, 14]. Рассмотрим канал *Adder*, который принимает два числа (Num1 и Num2) и выдает сумму этих чисел (Sum). Пользователь канала посылает эти числа во входные порты *Num*1 и *Num*2 и получает результат в выходной порт *Sum*. Канал используется агентом *MainAgent* и осуществляется агентом *AdderAgent* (рис. 2.1.10.1).

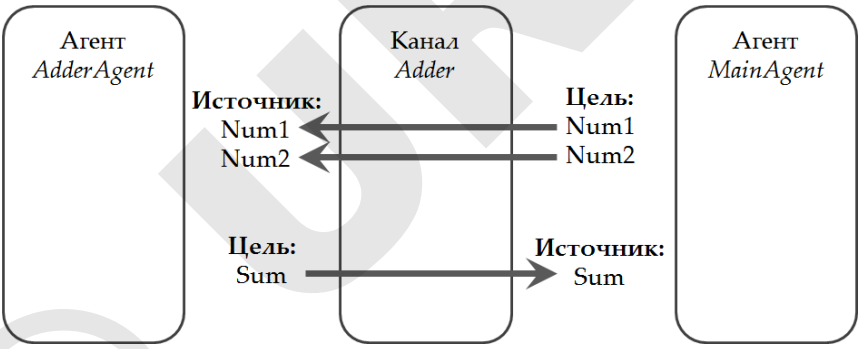


Рис. 2.1.10.1. Двойственность портов при реализации канала Adder [13]

Канал определяется с помощью ключевого слова *channel*, а его порты определяются с помощью ключевых слов *input* и *output*. В конструкторе агента *MainAgent* с помощью статического метода *CreateInNewDomain* агента *AdderAgent* создается экземпляр канала *Adder*. Таким образом, создаются два порта канала *Adder* (являющиеся экземплярами различных типов), а также создается экземпляр агента *AdderAgent*, который реализует канал, а затем возвращает значение, передаваемое в выходной порт канала. Далее мы

посылаем сообщения в порты *Num*1 и *Num*2 и ожидаем результата от порта *Sum* (листинг 2.1.10.1).

Листинг 2.1.10.1. Программный код на языке Axum, реализующий канал *Adder*

```
using System;
using System.Concurrency;
using Microsoft.Axum;
channel Adder
{
    input int Num1;
    input int Num2;
    output int Sum;
}
agent AdderAgent : channel Adder
{
    public AdderAgent()
    {
        int result = receive(PrimaryChannel::Num1) +
        receive(PrimaryChannel::Num2);
        PrimaryChannel::Sum <-- result;
    }
}
agent MainAgent : channel Microsoft.Axum.Application
{
    public MainAgent()
    {
    var adder = AdderAgent.CreateInNewDomain();
    adder::Num1 <-- 10;
    adder::Num2 <-- 20;
    var sum = receive(adder::Sum);
    Console.WriteLine(sum);
    PrimaryChannel::ExitCode <-- 0;
    }
}
```

© URSS

Каждая программа, написанная на языке Axum, содержит по крайней мере один *домен*, обеспечивающий точку входа в приложение. Однако главная цель доменов заключается в обеспечении изоляции памяти между параллельными прикладными компонентами.

При асинхронной обработке данных, как правило, необходимо обеспечить буферизацию данных, которая может иметь различные формы (к примеру, в виде неограниченной очереди изменяемых переменных). В Axum реализован механизм, основанный на концепции абстрактных *точек взаимодействий*, которые могут быть составлены в виде *сетей* потоков данных или же могут использоваться непосредственно в асинхронных операциях. Точки взаимодействий представляют собой основу всех асинхронных операций в Axum и позволяют программисту осуществлять маршрутизацию асинхронных данных в рамках одной модели. В свою очередь, *сети* позволяют комбинировать точки взаимодействия в сложные структуры, используя семь бинарных операторов и один унарный.

Схемы позволяют осуществлять операции сериализации данных, а также корректное выполнение асинхронных операций (к примеру, чтобы реализуемые параллельно агенты до определенного момента не имели доступа к общим изменяемым состояниям). Концептуально схемы Axum похожи на схемы XML и в меньшей степени — на структуры в C#.

Работа над инновационным программным языком Axum на данный момент закончена, и в виде самостоятельного продукта он на рынок не выйдет. По мнению разработчиков, этот проект был весьма успешным и апробированные в нем концепции найдут применение в будущих релизах .NET: C# и VB будут поддерживать асинхронные режимы выполнения программного кода, а также реализованные в Axum парадигмы агент-ориентированного подхода в программировании. Наверное, с коммерческой точки зрения действительно эффективнее расширять функционал имеющихся и уже достаточно распространенных языков программиро-

вания, чем выводить на рынок абсолютно новые. Для нас тем не менее важно, что крупнейший вендор программного обеспечения в числе прочего взял курс на поддержку агентного подхода в своих продуктах.

2.1.11. Платформа для моделирования городской мобильности на основе агент-ориентированного подхода, геоинформационных и суперкомпьютерных технологий

В совместной работе исследователей университета им. Иоганна Кеплера в Линце (Австрия), Лондонского университета (Великобритания), высшей технической школы Ингольштадта (Германия) и университета Бахри (Исламабад, Пакистан) рассматривается программная платформа для моделирования городской мобильности на основе агент-ориентированного подхода, геоинформационных и суперкомпьютерных технологий. Разработчики использовали реальную растровую карту небольшого города в Центральной Европе с очень высоким разрешением, преобразованную в мелкозернистую (1,25 м^2) двумерную решетку. Соответствующий клеточный автомат связывает перемещающихся агентов с их жизненным пространством. Вместе с тем, учитывая масштаб моделируемого объекта, техническая реализация модели практически невозможна без использования параллельных вычислений.

Параллельная симуляция на основе агент-ориентированного подхода: компоненты платформы

Разработанная программная платформа для агент-ориентированного моделирования включает в себя три основных компонента, показанных на рис. 2.1.11.1 в виде трех блоков: 1) блок для обработки изображений; 2) блок для параллельных симуляций с использованием агент-ориентированного подхода; 3) блок для обработки и визуализации выходных данных.

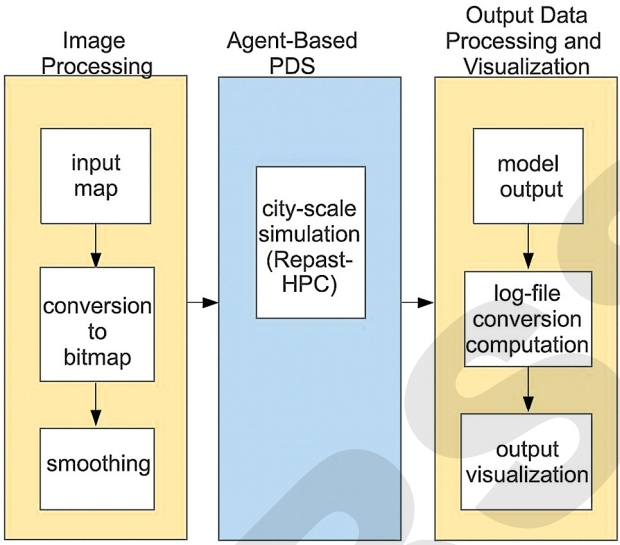


Рис. 2.1.11.1. Схема программной платформы для параллельных симуляций с использованием агент-ориентированного подхода: основные компоненты. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Обработка изображений

Подробная карта города, представляющая собой детализированное растровое изображение, передается в модель с обозначением соответствующих областей (рис. 2.1.11.2). Затем отмеченная часть разбивается на сегменты в зависимости от вычислительных ресурсов и затрат на межпроцессорную синхронизацию — в рассматриваемом примере на 200 квадратов равных размеров (то есть карта представляется в виде матрицы из 20 строк и 10 столбцов (рис. 2.1.11.3)). Каждый сегмент обрабатывается одним процессором. Для моделирования эвакуации в масштабах города, сектора 22, 64, 96, 124 и 196 определяются как *выходы* (exits), а в процессе симуляции сегмент 22 затопляется. На рис. 2.1.11.4 отображен детализированный сектор 33. Для обработки растрового изображения используется алгоритм низкочастотной фильтрации, а полученные при этом 200 файлов сохраняются в формате ASCII для последующего использования в ГИС.

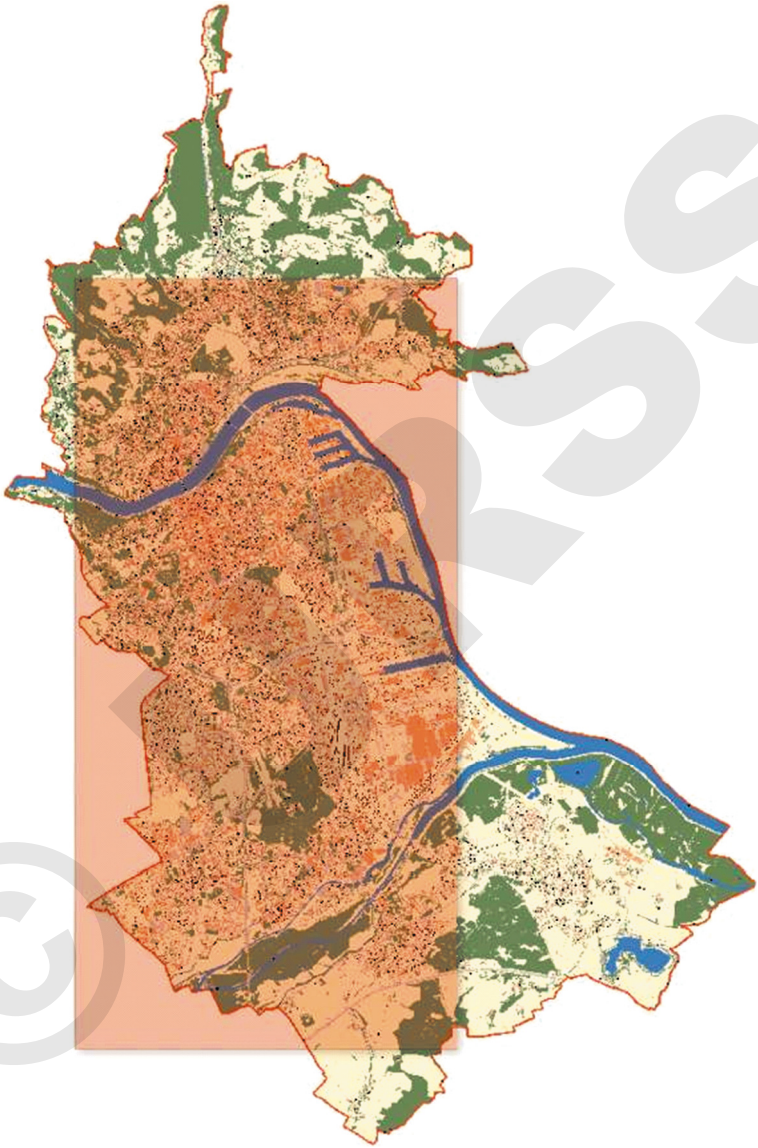


Рис. 2.1.11.2. Растровая карта города. Разработанная модель относится к выделенной части пространства. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

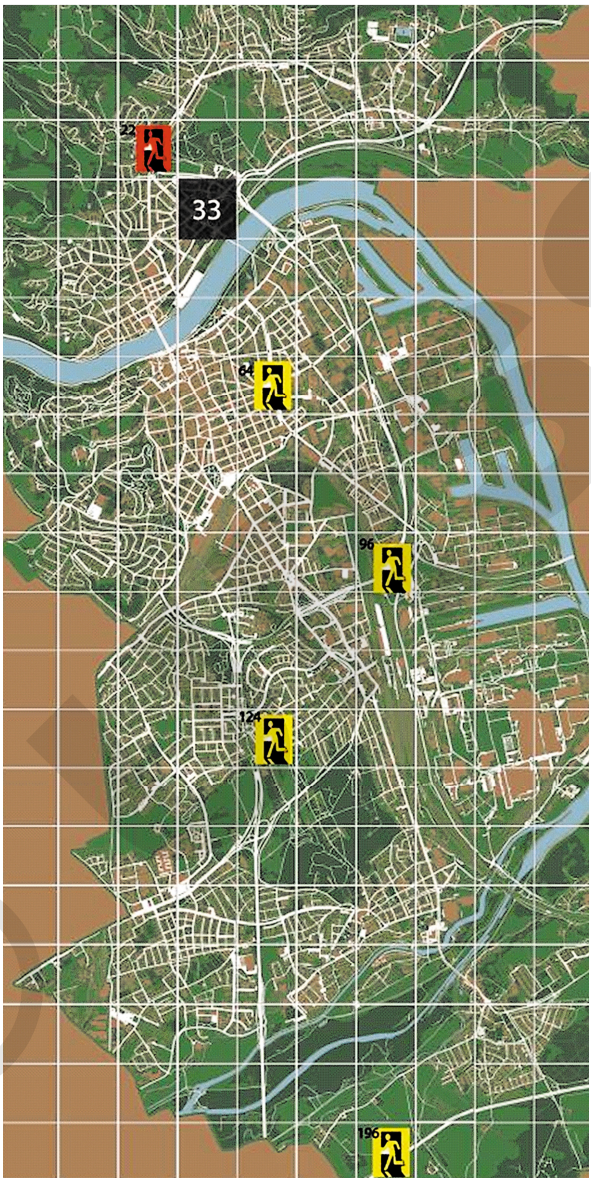
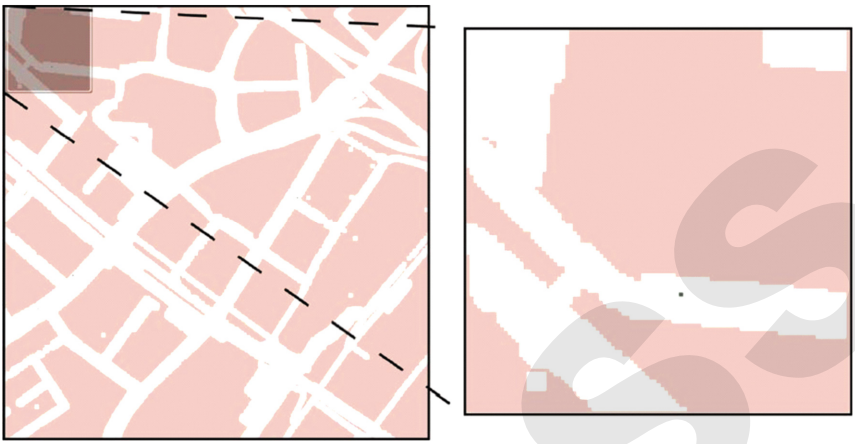


Рис. 2.1.11.3. Используемая для симуляций карта города, поделенная на 200 сегментов — квадратов одинакового размера. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]



(a) (b)

Рис. 2.1.11.4. (a) — часть обрабатываемого 25 (5 × 5) процессорами пространства с выделенным сектором (33 на рис. 2.1.11.3), составляющим примерно 1/500-я часть от всей карты города (рис. 2.1.11.2) и 1/200-я от используемого для симуляций пространства. (b) — увеличенный (один из 25) фрагмент карты (a), обрабатываемый одним процессором. Области пространства белого цвета соответствуют улицам и доступны для перемещающихся агентов. Соответственно, все остальные области для движения недоступны. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Агент-ориентированный параллельный симулятор

Моделируемое пространство разделено между двумя видами памяти: локальной (для каждого процессора) и разделяемой, используемой процессами для обмена информацией посредством общего сегмента памяти.

Настройка работы каждого процесса может осуществляться во время симуляции и с помощью файлов конфигурации модели, в которых определяются различные параметры. Наиболее важными из них являются: значения границ пространства (*x* и *y* координаты), количество процессоров и их вертикальное (и горизонтальное) распределение относительно моделируемого ландшафта, параметры синхронизации и время симуляции в рамках итерационного алгоритма. Каждому процессу автоматически присваивается идентификационный номер, значение которого зависит от вертикаль-

ного (и горизонтального) местоположения процесса в рамках разделенного на сегменты пространства. Нумерация начинается с 0 и заканчивается $N-1$, где N — общее количество сегментов. На рис. 2.1.11.5 видно, что для расчетов используется 200 процессов со значениями идентификаторов от 0 до 199, а каждый отдельный процесс включает в себя три составляющих модуля: 1) настройка; 2) модель; 3) синхронизация.

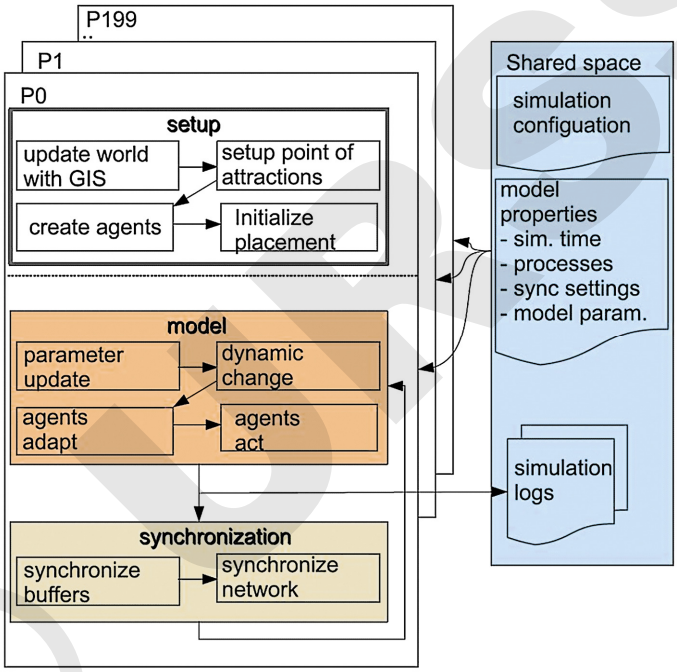


Рис. 2.1.11.5. Схема работы агент-ориентированного параллельного симулятора. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Настройка

Инициализация модуля предполагает выполнение следующих шагов:

- обновление карты (с первоначальными настройками ячеек) с использованием информации из файлов ASCII;

- настройка «центров притяжения» (point of attractions) для агентов с использованием информации из файлов конфигурации модели;
- создание агентов с использованием информации (относительно типов агентов) из файлов конфигурации модели;
- распределение агентов по пространству с использованием информации (координаты, плотность расселения и др.) из файлов конфигурации модели.

Модуль настройки выполняется только один раз перед началом симуляции.

Модель

В процессе выполнения модели происходит выполнение следующих процедур:

- обновление параметров — к примеру, параметров совместно используемых процессами глобальных переменных;
- обновление пространственных переменных;
- изменение параметров агентов с учетом изменения пространства и поведения агентов, находящихся в непосредственной близости;
- изменение поведения агентов, то есть выполнение запланированных ранее действий.

Синхронизация

Основной целью этой процедуры является синхронизация буферного и сетевого пространства.

Параметры эффективности

Для масштабируемости системы (то есть эффективного использования дополнительных вычислительных ресурсов для ускорения счета) разработчики определили ряд факторов, в наибольшей степени влияющих на производительность.

1. Интенсивность взаимодействия агентов и горизонт их видимости.

© URSS

2. Сложность в поведении агентов. В своем предыдущем исследовании авторы показали, что во многих случаях увеличение сложности агентов и изменение их поведения в целом снижает эффективность распараллеливания соответствующей модели.
3. Распределение пространства между процессорами. Здесь имеет значение равномерность распределения агентов по сегментам пространства.
4. Алгоритм синхронизации.

Модель

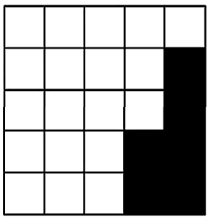
Карта

Изначальная карта (г. Линц, Австрия) представляла собой растровый рисунок размерностью 10 000 × 15 000 ячеек (соответствующих квадратам с размером 1.25 м^2). Далее карта была обрезана до центрального района города (рис. 2.1.11.2) размерностью 5000 × 10 000 ячеек и поделена на 200 сегментов, каждый из которых (с площадью 250 000 ячеек) при распараллеливании обслуживался отдельным процессом p_i, где i — его идентификационный номер (от 0 до 199).

Также разработчиками введена пространственная единица — патч (patch). В терминах разработанной платформы патч — это структурная единица с информацией о типе структуры и ее координатами (к примеру, улица, здание, зеленая зона и т. д.).

1	1	1	1	1
1	1	1	1	0
1	1	1	1	0
1	1	1	0	0
1	1	1	0	0

(a)



(b)

Рис. 2.1.11.6. Пример сетки патчей (справа) созданной на основе данных из файла (слева). Согласно правилам, белые патчи доступны для агентов, а черные — нет. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Для упрощения вычислений у пространственной единицы могут быть только два получаемых агентом состояния: (1) доступно для прохода; (0) недоступно для прохода. На рис. 2.1.11.6 показан пример пространства, созданного на основе данных из файла ASCII.

Полевая модель (Floor Field Model)

В теории клеточных автоматов есть тип моделей, в основе которых лежит использование *полей*, в которых хранится информация, оказывающая влияние на навигацию агентов. Поля могут быть динамическими и статическими, в зависимости от характера используемой информации. К примеру, динамическое поле, имеющее собственную динамику, может рассеивать и забывать виртуальные следы агентов. В свою очередь статические поля не меняются, и информация обо всех недвижимых препятствиях сохраняется. В рамках текущей модели, знания и память агентов также могут влиять на информацию поля, а каждый патч сетки имеет три дополнительные переменные:

1. направление (Direction) из текущего патча до выхода — DOM;
2. расстояние (Distance) от текущего патча до выхода — HOPC;
3. маршрут — ROUTE.

Типовая структура переменных для стандартного патча показана на рис. 2.1.11.7.

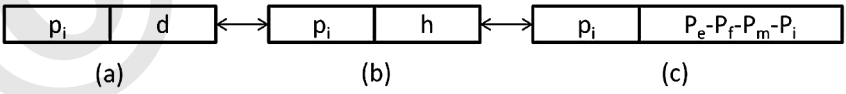


Рис. 2.1.11.7. Структура поля стандартного патча: (a) направление *d* варьируется в определенном диапазоне и рассчитывается как относительный угол между отправляющим и принимающим патчами; (b) расстояние *h* определяется перемещениями до целевого патча; (c) маршрут представляет собой последовательность передающих информацию процессов. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Изначально этим переменным не присвоено значений, и соответствующие патчи являются недоступными для агентов препятствиями (рис. 2.1.11.8 (d)).

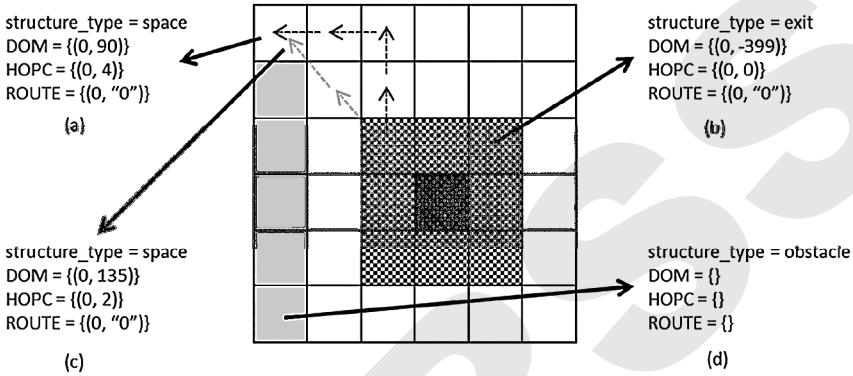


Рис. 2.1.11.8. Примеры патча-выхода (заштрихованная область) и значений дополнительных переменных. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Каждый доступный для перемещения патч ***p*** (не являющийся выходом) во время итеративного пересчета обновляет имеющуюся у него информацию (в окрестности Мура) о патчах-выходах. Если соседний с ним патч имеет информацию о новом выходе в своей коллекции DOM или же имеет информацию о расстоянии до него, то он сообщает эти сведения патчу ***p*** со ссылкой на источник (рис. 2.1.11.8 (a) и рис. 2.1.11.8 (c)). Это позволяет вычислить правильное направление и необходимое количество переходов (в используемой нотации они записываются последовательно через символ «.»). Для более частых случаев, когда выход обслуживается другим процессом, маршрут от патча должен содержать примерно следующую информацию {(15, 2.12.13.14.15) (0, 2.1.0)}, где патч относится к процессу 2 и для него есть два возможных выхода на процессах 15 и 0. Пример изменения полей модели показан на рис. 2.1.11.9, а соответствующий псевдокод приведен в листинге 2.1.11.1.

Листинг 2.1.11.1. Псевдокод, генерирующий поля для патчей

```
1   if patch p structure_Type == “walkable” and is not a “exit”
2       for each neighboring patch n
3       // рассматриваются патчи в окрестности Мура
4           for each pair p <key, value> of DOMn
5           // сравниваются пары key и value среди доступных выходов в DOMp
6                   if key does not exists in DOMp
7                   // если подходящих нет, то происходит обновление
8                       DOMp [key] = relative-angle to n
9                       HOPCp [key] = value (HOPCn [key]) + 1
10                      ROUTEp [key] = value (ROUTEn [key]).ProcessID
```

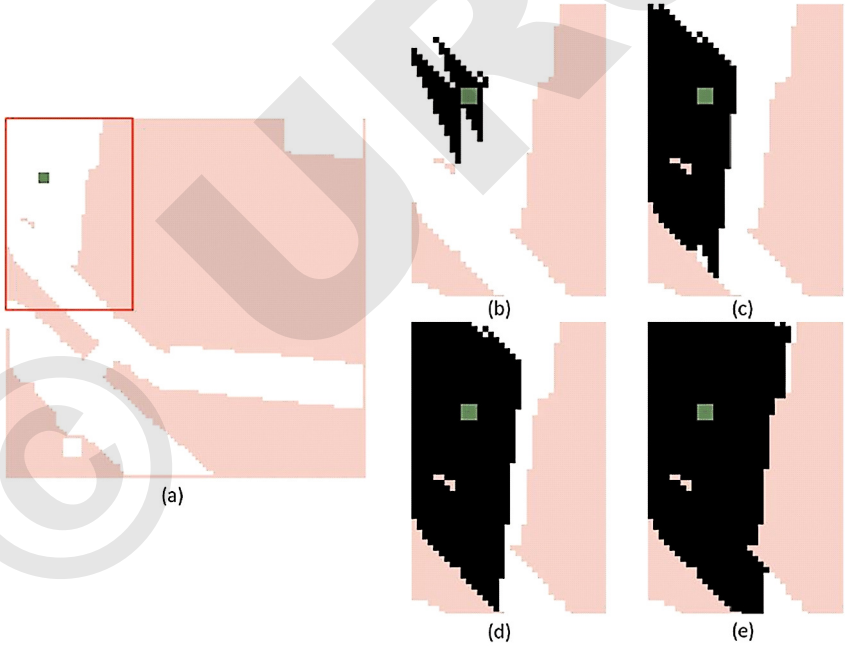


Рис. 2.1.11.9. Распространение информации о патче-выходе, находящемся внутри помеченной области (a); через 1 итерацию (b); 3 (c); 6 (d) и 10 (e). *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Механизмы взаимодействий и синхронизация

Разработанная платформа поддерживает два режима синхронизации — (1) синхронизация буфера и (2) сетевая синхронизация. Оба режима позволяют работать как с патчами, так и с мобильными агентами.

Синхронизация буфера (*Buffer synchronization*). В моделируемом пространстве используемая процессами информация может быть синхронизирована с использованием буферов обмена. На рис. 2.1.11.10 (a) показана синхронизация данных между процессом 64 и процессами 54, 63, 74 и 65 с использованием буферов с 1 размером. Диагональные процессы также синхронизируются, но на схеме это не показано.

Сетевая синхронизация (*Network synchronization*). В этом случае информация синхронизируется между связанными патчами (или агентами) из соседних процессов. Пример синхронизации двух связанных патчей, находящихся в процессах 64 и 74, показан на рис. 2.1.11.10 (a).

Патчи и мобильные агенты используют три переменные (“agent_id”, “process_id”, “agent_type”), идентифицирующие их в процессе синхронизации, а другие переменные необходимы для спецификации некоторых функций. Часть из них приведена в табл. 2.1.11.1.

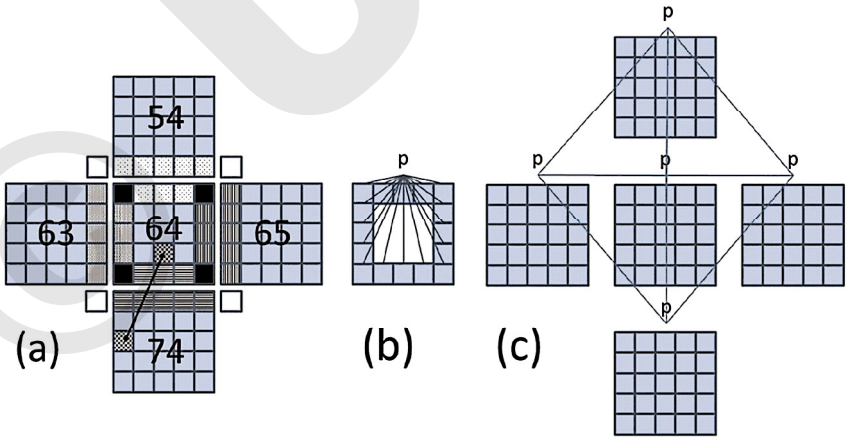


Рис. 2.1.11.10. Обмен данных между процессами: (a) синхронизация буфера и сетевая синхронизация; (b) параметры процесса; (c) совместное использование параметров процесса. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Таблица 2.1.11.1

Переменные, используемые при синхронизации и спецификации

Переменная	Тип агента	Краткое описание
agent_id	Любой	Уникальный идентификатор агента
process_id	Любой	Идентификатор процесса, который в настоящий момент обслуживает агента
agent_type	Любой	Тип агента
structure_type	Патч	Тип участка ГИС пространства
DOM	Патч	Переменная поля
HOPC	Патч	Переменная поля
ROUTE	Патч	Переменная поля
orientation	Мобильный	Относительный угол направляющей между текущим патчем и патчем-выходом
curr_exit	Мобильный	Текущий патч-выход (пункт назначения) для агента
wait	Мобильный	Длительность ожидания перехода с позиции агента (в терминах итерационного пересчета)
speed	Мобильный	Скорость агента при выполнении текущего перехода
following?	Мобильный	Проверка — является ли агент ведомым по отношению к агенту, обладающему большей информацией, чем он сам?

Параметры процесса

Во время симуляций возможны ситуации, при которых информация на уровне процессов должна быть разделена. Например, «процессная единица плотности»:

$$\textit{Процессная единица плотности} = pop / wal, \qquad (2.1.11.1)$$

где *pop* — количество перемещающихся агентов, обслуживаемых текущим процессом и *wal* — количество доступных для перемещения патчей, также обслуживаемых текущим процессом. Этот параметр рассчитывается и хранится на уровне одного процесса, но при

этом доступен всем агентам для расчета скорости передвижения (рис. 2.1.11.10 (b)).

Глобальные параметры

Значения упомянутого ранее параметра крайне важны для планирования маршрутов, но в силу больших вычислительных затрат на каждой итерации они не рассчитываются. В этой связи в разработанной платформе используются глобальные параметры, используемые несколькими процессами (рис. 2.1.11.10 (c)), сохраняющие необходимые данные.

Поведение и адаптация агентов

Все агенты в модели являются пешеходами двух типов:

- интеллектуальные агенты, имеющие доступ ко всей информации в рамках симуляции;
- неинтеллектуальные агенты, имеющие доступ только к локальной информации (на уровне обслуживающего их процесса).

Создание агентов и их случайное размещение

Количество создаваемых агентов рассчитывается по приведенной ниже формуле:

$$count_{Type} = (Perc_{Type} / 100) \times \\ \times ((ProccessW / TotalW) \times TotalP), \qquad (2.1.11.2)$$

где $Perc_{Type}$ — доля агентов определенного типа, $ProccessW$ — количество доступных для перемещения патчей в рамках текущего процесса, $TotalW$ — количество всех доступных патчей в моделируемом пространстве, $TotalP$ — общее количество мобильных агентов.

При инициализации, каждому агенту присваиваются следующие значения основных переменных (независимо от типа агента): orientation (– 999), curr_exit (– 1), speed(0), wait(– 1), confidence(– 1) и following(false). Агент генерируется на еще незанятом патче и ему присваивается тип “walkable”.

Скорость агентов

Скорость конкретного агента зависит от количества агентов в локальном пространстве и рассчитывается следующим образом:

$$speed_on_density = v_0 \times (1 - count\ (agents) / count\ (walkables) / max_density_per_patch) \quad (2.1.11.3)$$

$$speed = \max \{v_{min}, speed_on_density\} \quad (2.1.11.4)$$

Процедура принятия решения агентом по перемещению приведена в листинге 2.1.11.2. Она выполняется для каждого агента в соответствии с правилами, показанными на рис. 2.1.11.11, в результате применения которых обновляются переменные агента.

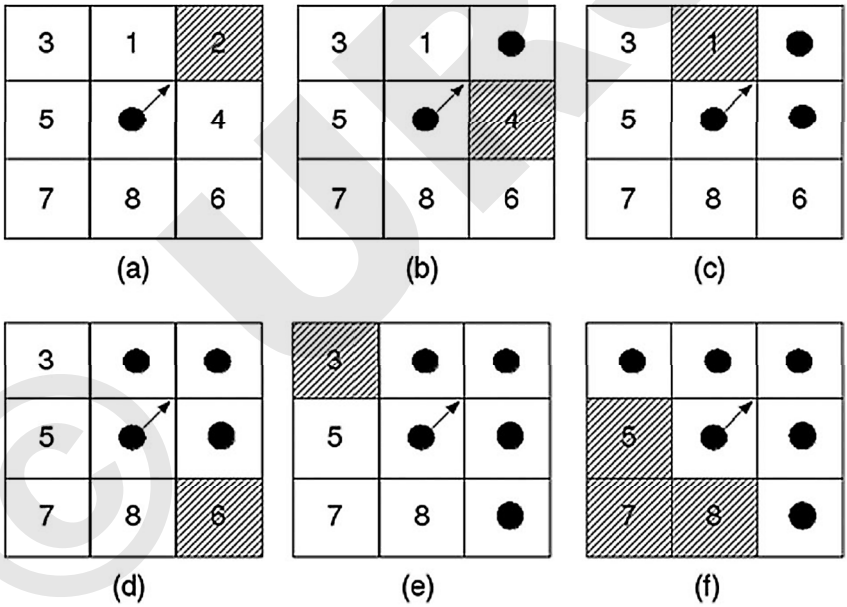


Рис. 2.1.11.11. Возможные предпочтения объекта, движущегося под углом 45°. Черные кружки обозначают занятые объектом ячейки, а заштрихованные — целевые. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Листинг 2.1.11.2. Псевдокод, описывающий процедуру принятия агентом решения по перемещению

```
1   heading (orientation)
2   p = patchAtHeadingAndDistance (heading, speed)
3       if (p → turtlesHere.size > 0 || p → turtlesHere.size > 0
4           if (scanNeighborinOrderRight (angle))
5               heading (angle)
6               move (speed)
7               reset_wait ()
8               confidence (x)
9           else
10              inc_wait ()
11      else
12          move (speed)
13          reset_wait ()
14          confidence (100)
```

Моделирование передвижения агентов

При моделировании передвижения учитывается множество факторов, среди которых важнейшими являются следующие.

1) Тип агентов. Для проведения экспериментов разработчики предусмотрели три вариации:

- все агенты являются неинтеллектуальными: стратегия 1 и стратегия 4 (см. ниже);
- все агенты являются интеллектуальными: стратегия 2;
- часть агентов является интеллектуальными, а часть — нет: стратегия 3 и стратегия 5.

2) Уровень информационной осведомленности, влияющий на поиск оптимального для агентов патча-выхода.

3) Аппроксимация результатов. Метод, позволяющий сократить время расчетов, если не так важна точность полученных значений (реализуется путем использования стратегии 4 вместо стратегии 1 и стратегии 5 вместо стратегии 3).

© URSS

Далее переходим к рассмотрению различных стратегий передвижения, реализованных в рамках экспериментальных расчетов.

Стратегия 1

Все агенты модели являются неинтеллектуальными и каждый из них использует только стратегию с локальными маршрутами к ближайшему патчу-выходу. Пример реализации стратегии 1 приведен в листинге 2.1.11.3.

Листинг 2.1.11.3. Псевдокод, реализующий стратегию 1

```
1  p = patchHere
2  selected_exit = min {key (HOPC_p)}
3  if (selected_exit == null)
4      inc_wait()
5  else
6      find pair <key, value> in DOM_p
7      orientation (value (DOM_p [key]))
8      current_exit (selected exit)
9      Proceed_to_Next_Cell()
```

Стратегия 2

Все агенты модели являются интеллектуальными, и все они реализуют стратегию с учетом всех возможных вариантов (листинг 2.1.11.4).

Листинг 2.1.11.4. Псевдокод, реализующий стратегию 2

```
1  if there exits an intelligent agent a in interation_range
3      if (a → current_exit != -1)
4      following (true)
5      current_exit (a → current_exit)
6  if (following())
7      p = patchHere
8      find pair <key, value> in DOM_p where key = current_exit
9      orientation (value (DOM_p [key]))
```

© URSS

```
10      Proceed_to_Next_Cell()
11  else
12      Move_to_Nearest()
```

Стратегия 3

В случае гибридной популяции неинтеллектуальные агенты (95 % от общего числа) ищут в доступной окрестности интеллектуальных (5 %) и следуют за ними. В листинге 2.1.11.5 приведен псевдокод, реализующий стратегию неинтеллектуальных агентов, а агенты другого типа, в свою очередь, реализуют стратегию 2.

Листинг 2.1.11.5. Псевдокод, реализующий стратегию 3

```
1   if (current_exit != -1)
2       p = patchHere
3       if (size (p → DOMp) >= 2)
4           set_OptimalExit()
5           find pair <key, value> in DOMp where key = current_exit
6           orientation (value (DOMp [key]))
7           Proceed_to_Next_Cell()
8   else
9       if (size (p → DOMp) >= 2)
10          set_OptimalExit()
11          inc_wait()
12      else
13          if (size (p → DOMp) == 1)
14              set_AvailableExit()
15              inc_wait()
16          else
17              inc_wait()
18  set_OptimalExit()
19      current_weight = 999999;
20      aggregate_route_density = 0;
21      selectedExit = -1;
22      for each pair <key, value> in ROUTEp
23          current_process = key
24          current_route_sequence = value
25          for each element e in current_route_sequence
26              find pair <key, value> in DENSITIESprocess where key = e
```

©URSS

```
27              current_density = value (DENSITIES_process [key])
28              aggregate_route_density = aggregate_route_density + current_density
29          aggregate_route_density — aggregate_route_density /
                                size (current_route_sequence)
30          find pair <key, value> in DOM_p where key = current_process
31          temp_value = aggregate_route_density × value (DOM_p [key])
32          if (temp_value <= current_weight)
33              current_weight = temp_value
34              selectedExit = current_process
35          current_exit (selectedExit)
36  set_AvailableExit()
37  current_exit (p → DOM_p .first[key])
```

Стратегия 4

Данная стратегия аналогична первой, но, с целью сокращения времени расчетов, первый же найденный патч-выход считается оптимальным.

Стратегия 5

Стратегия аналогична третьей, но в реализующем ее коде первая строка выполняется не каждый раз, а только в случае потери маршрута.

Для соответствия базовым принципам агентного подхода, разработанная платформа должна обладать функциональностью, позволяющей агентам реагировать на изменение окружающей среды, к примеру, на изменения состояний патчей-выходов.

Сценарий 1. Один из выходов *x* становится недоступным, но с течением времени может снова стать доступным. При этом информация о недоступности выхода распространяется мгновенно, а о повторной доступности появляется постепенно (как в листинге 2.1.11.1).

Сценарий 2. Один из выходов *x* становится недоступным и остается таким на протяжении всей симуляции, а информация о его недоступности также как в сценарии 1 распространяется мгновенно.

URSS

Сценарий 3. Один из выходов *x* становится недоступным и остается таким на протяжении всей симуляции, но информация о его недоступности распространяется постепенно.

Описанные сценарии реализованы в следующих стратегиях, основанных на базе стратегии 5.

Стратегия 6

Во время выполнения *i*-й итерации информация о выходе *x* стирается из коллекций DOM, HOPC и ROUTE для всех патчей. На итерации *i* + 1 запускается алгоритм, генерирующий поля для патчей (листинг 2.1.11.1).

Стратегия 7

Во время выполнения *i*-й итерации информация о выходе *x* стирается из коллекций DOM, HOPC и ROUTE для всех патчей.

Стратегия 8

Во время выполнения *i* + 1-й итерации запускается измененный алгоритм из листинга 2.1.11.1. Модификация касается распространения информации о недоступности выхода — теперь она распространяется постепенно и пока патчи ее не получат, они будут дезинформированы и считать выход валидным.

Симуляция процесса эвакуации и некоторые результаты

Для тестового запуска пространство было ограничено размерностью 500 × 500, что составляет примерно 1/500-ю часть от всей карты города (рис. 2.1.11.2) и обрабатывается 25 (5 × 5) процессами, на каждый из которых приходится сегмент размерностью 100 × 100 (рис. 2.1.11.12). В процессе симуляции патч-выход 12 через определенное время (на 150-й итерации) меняет свое состояние от «доступного» на «недоступный». Всего в тестовом режиме было задействовано 12 500 агентов, отработано все 8 стратегий, а общее количество итераций — 500.

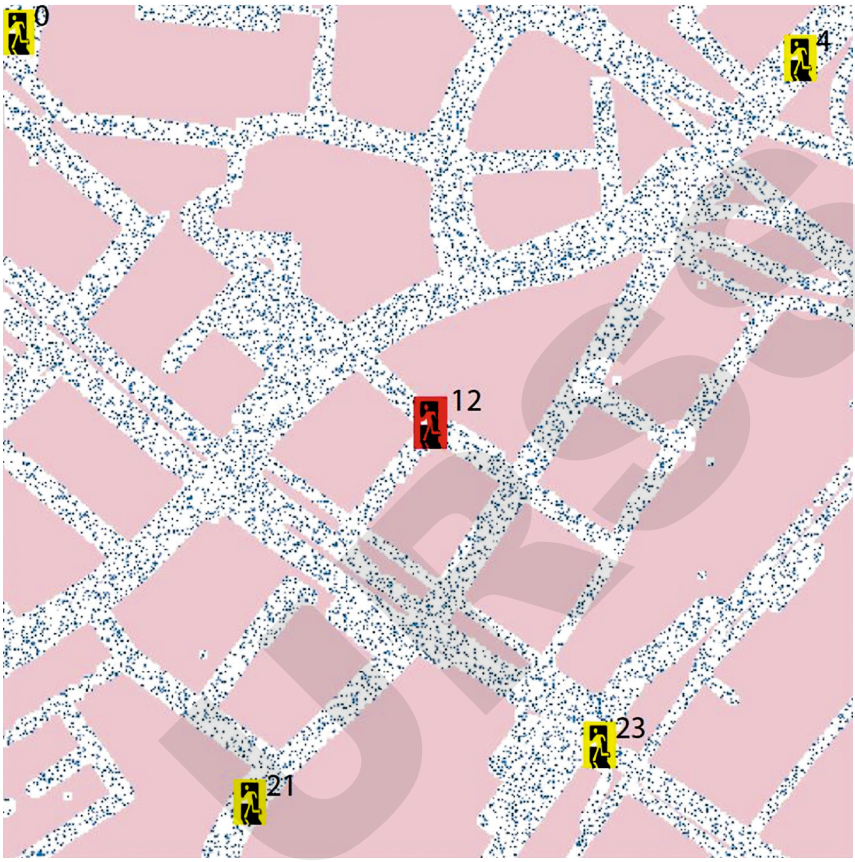


Рис. 2.1.11.12. Пространство для тестового запуска. Сектора 0, 4, 12, 21 и 23 являются патчами-выходами, причем выход 12 в процессе симуляции затапливается. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Поскольку цель разработки модели — оценка эффективности той или иной стратегии эвакуации, то особое внимание в результатах уделяется затраченному времени. В табл. 2.1.11.2 приведены соответствующие оценки и, как видно, разница во времени между 1-й и 4-й стратегиями весьма существенна, в то время как между 2-й и 5-й незначительна.

Таблица 2.1.11.2

Сравнительная оценка производительности параллельной версии симулятора для различных стратегий передвижений в рамках тестового пространства

Стратегии передвижений	**Процессорное время,** ***чч:мм:сс***	**Память,** ***Кб***	**Виртуальная память,** ***Кб***	**Затраченное время,** ***чч:мм:сс***
Стратегия 1 (100 % неинтеллектуальных агентов)	11:32:22	1 068 744	20 562 304 660	00:27:58
Стратегия 2 (100 % интеллектуальных агентов)	01:56:21	1 067 968	20 562 303 596	00:04:53
Стратегия 3 (5 % интеллектуальных и 95 % неинтеллектуальных агентов)	06:56:40	1 067 512	20 562 302 904	00:16:57
Стратегия 4 (100 % неинтеллектуальных агентов)	01:57:28	1 065 856	20 562 301 428	00:04:54
Стратегия 5 (5 % интеллектуальных и 95 % неинтеллектуальных агентов)	02:24:3	1 068 116	20 562 303 472	00:06:02
Стратегия 6 (5 % интеллектуальных и 95 % неинтеллектуальных агентов)	02:27:33	1 067 432	20 562 302 588	00:06:37
Стратегия 7 (5 % интеллектуальных и 95 % неинтеллектуальных агентов)	01:52:50	1 052 036	20 562 287 592	00:04:42
Стратегия 8 (5 % интеллектуальных и 95 % неинтеллектуальных агентов)	02:13:37	1 061 136	20 562 296 924	00:05:33

© URSS

Если сравнивать стратегии 1 и 4 на основе рис. 2.1.11.13 и табл. 2.1.11.3, то видно, что процент агентов, эвакуировавшихся через выходы 0 и 21, в случае применения стратегии 4 снизился в силу того, что эти выходы имеют меньшую пропускную способность (по сравнению с 4 и 23), а агенты в модели не тратят времени на поиск другого, возможно более оптимального выхода. На рис. 2.1.11.14 видно, что общее число эвакуированных агентов в случае применения стратегий 1 и 4 практически не отличается, но судя по данным из табл. 2.11.2, время реализации стратегии 1 почти в семь раз превышает время реализации стратегии 4. В этой связи для симуляции всего пространства была выбрана стратегия 4.

Таблица 2.1.11.3

Результаты применения восьми стратегий передвижений для параллельной версии симулятора в рамках тестового пространства: количество агентов, эвакуированных через разные патчи-выходы (0, 4, 12, 21 и 23), в процентах

Стратегии	Выход 0	Выход 4	Выход 12	Выход 21	Выход 23
1	8 %	26 %	32 %	14 %	20 %
4	4 %	33 %	27 %	12 %	22 %
3	10 %	27 %	26 %	14 %	23 %
5	9 %	27 %	25 %	14 %	24 %
7	9 %	33 %	10 %	19 %	31 %
8	8 %	35 %	8 %	16 %	33 %

При реализации стратегий 3 и 5 эвакуация агентов через выходы происходит более равномерно, то есть даже небольшой процент интеллектуальных агентов может оказать влияние на весь процесс. Сравнивая результативность обеих стратегий (рис. 2.1.11.13 и табл. 2.1.11.3) можно отметить, что большой разницы в скорости эвакуации и использовании выходов нет. Однако время реализации стратегии 3 почти в 3 раза превышает время реализации стратегии 5 (табл. 2.1.11.2), поэтому для симуляции всего пространства была выбрана последняя.

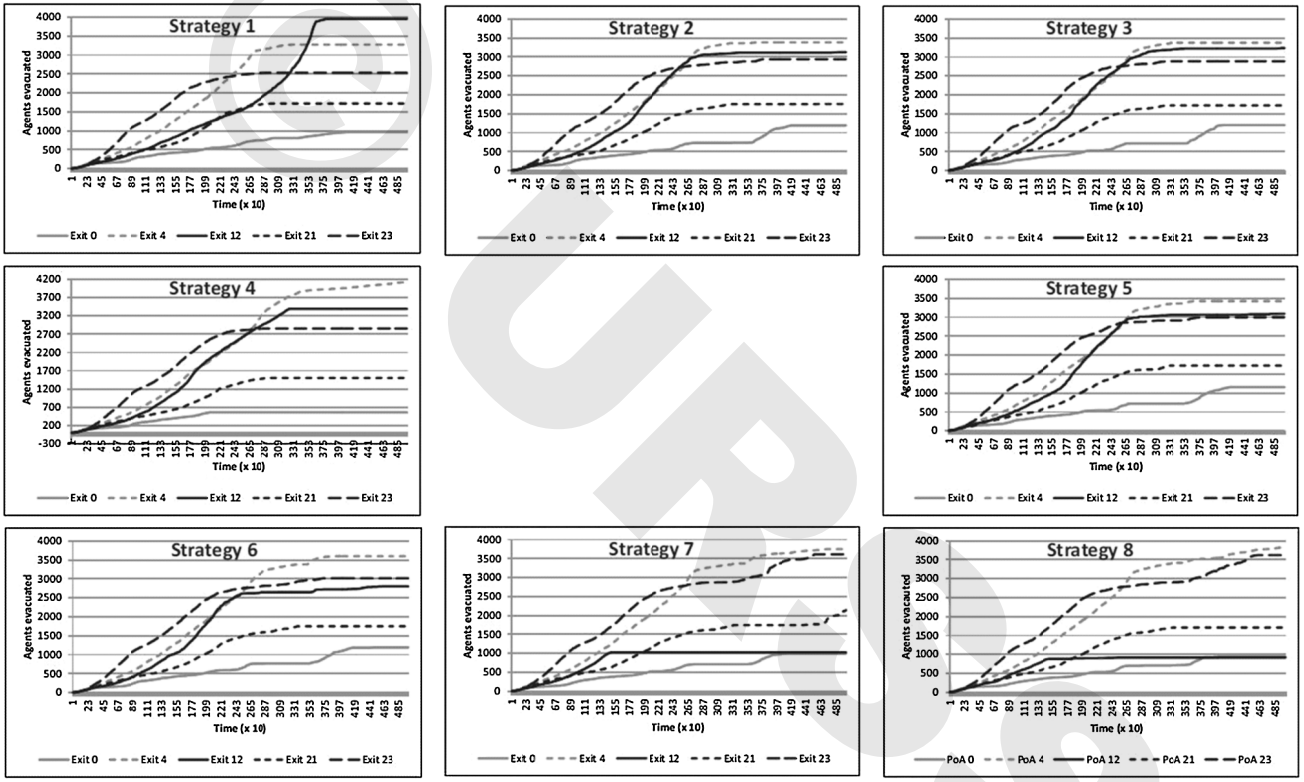


Рис. 2.1.11.13. Результаты применения 8 стратегий передвижений для параллельной версии симулятора в рамках тестового пространства: количество эвакуированных агентов через разные патчи-выходы (0, 4, 12, 21 и 23) в течение 500 итераций. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

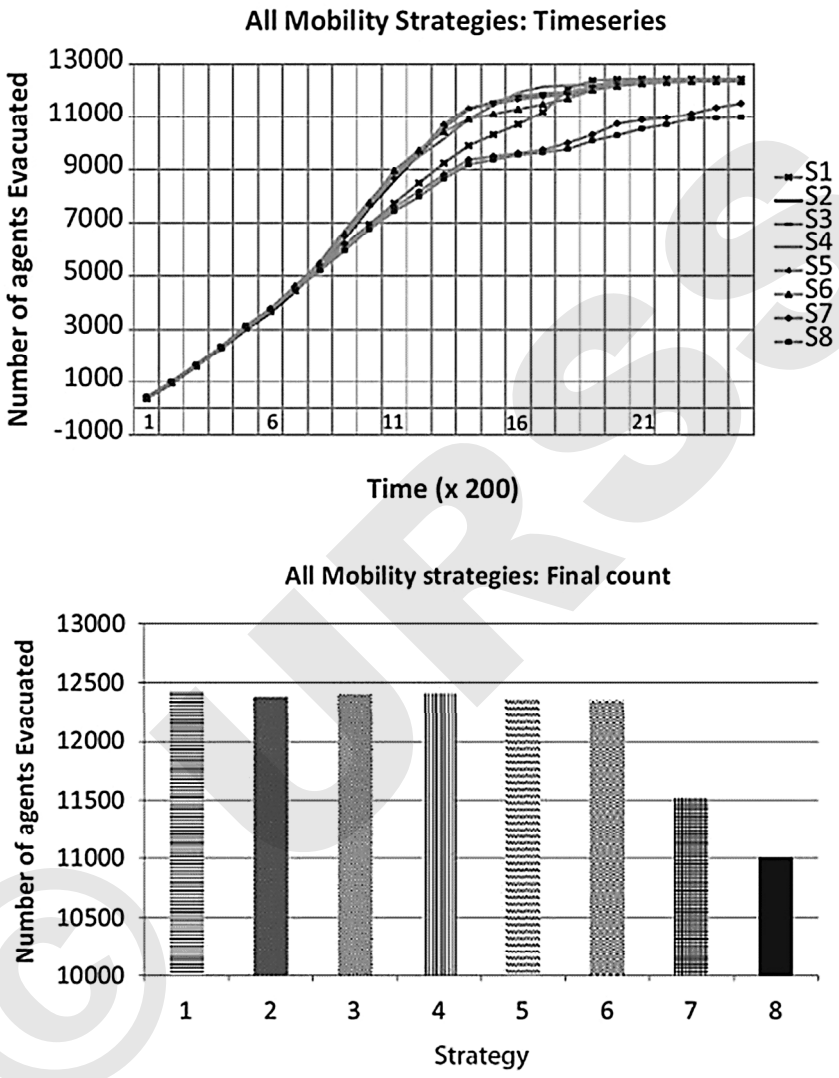


Рис. 2.1.11.14. Результаты применения 8 стратегий передвижений для параллельной версии симулятора в рамках тестового пространства: количество эвакуированных агентов безотносительно выходов (*Sn* — стратегии, *n* — номер конкретной стратегии). *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

©URSS

Предпосылки стратегий 2 и 6 не являются реалистичными, поэтому они были обсчитаны только для теоретического сравнения и для симуляции всего пространства использованы не будут.

Стратегии 7 и 8 позволяют перенаправить агентов к новому выходу, если в процессе симуляции выход 12 станет недоступным. По данным рис. 2.1.11.13 и табл. 2.1.11.3 видно, что агенты, изначально направляющиеся к выходу 12, далее были перенаправлены к выходам 4, 21 и 23. Стратегия 7 производительнее, чем стратегия 8 (табл. 2.1.11.2), в связи с чем она и была выбрана для симуляции всего пространства.

Моделирование эвакуации в городском масштабе

Для симуляций в масштабах города карта была поделена на 200 сегментов — квадратов одинакового размера (рис. 2.1.11.3), распределенных по прямоугольнику размерностью 10 × 20. Каждый из сегментов, обслуживаемый отдельный процессором, содержит подпространство размерностью 500 × 500. Количество агентов, соответствующее населению города, составляет 200 000. Патчи-выходы представляют собой городские достопримечательности: центр города — 64-й патч, промышленная зона — 96-й, стадион — 124-й, пригородный центр — 22-й, выезд на автомагистраль — 196-й. В рамках одной симуляции осуществлялось 5000 итераций, а выход 22 меняет свое состояние с «доступного» на «недоступный» на 1000-й итерации.

Результаты

В ходе экспериментов, предусматривающих применение 4-й, 5-й и 7-й стратегий, вычислялось общее количество эвакуировавшихся в процессе симуляции агентов. На рис. 2.1.11.15 приведены соответствующие результаты, показывающие эффективность стратегии 5. Так, при ее применении 80 % агентов эвакуировались на итерации 3690, а при реализации стратегии 4 тоже количество агентов эвакуируется только на 4540 итерации. Также очевидно, что 7-я стратегия по сравнению с 5-й является проигрышной, поскольку переориентация агентов с 22-го патча на другие выходы потребовала дополнительного времени, и за 5000 итераций удалось эвакуировать только 60 % агентов.

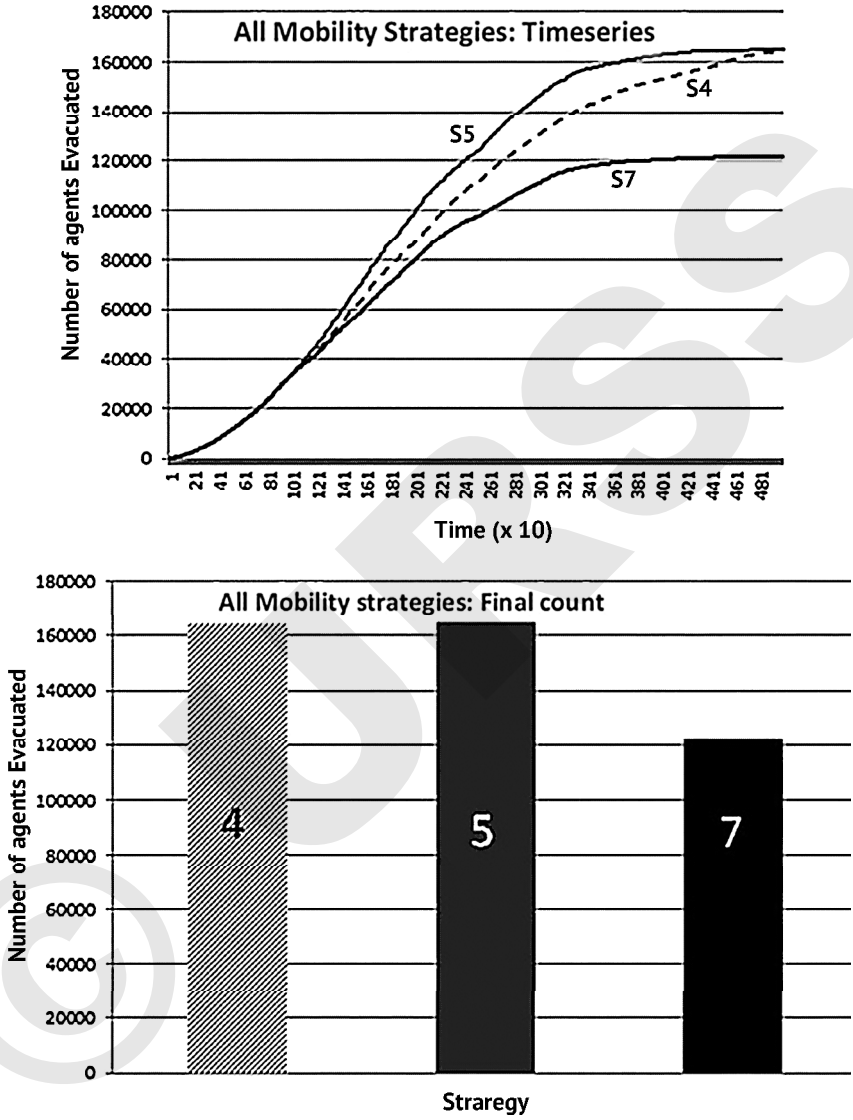


Рис. 2.1.11.15. Результаты применения отобранных стратегий (4, 5 и 7) передвижений для параллельной версии симулятора в рамках масштаба всего города: количество эвакуированных агентов безотносительно выходов (*Sn* — стратегии, *n* — номер конкретной стратегии). *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

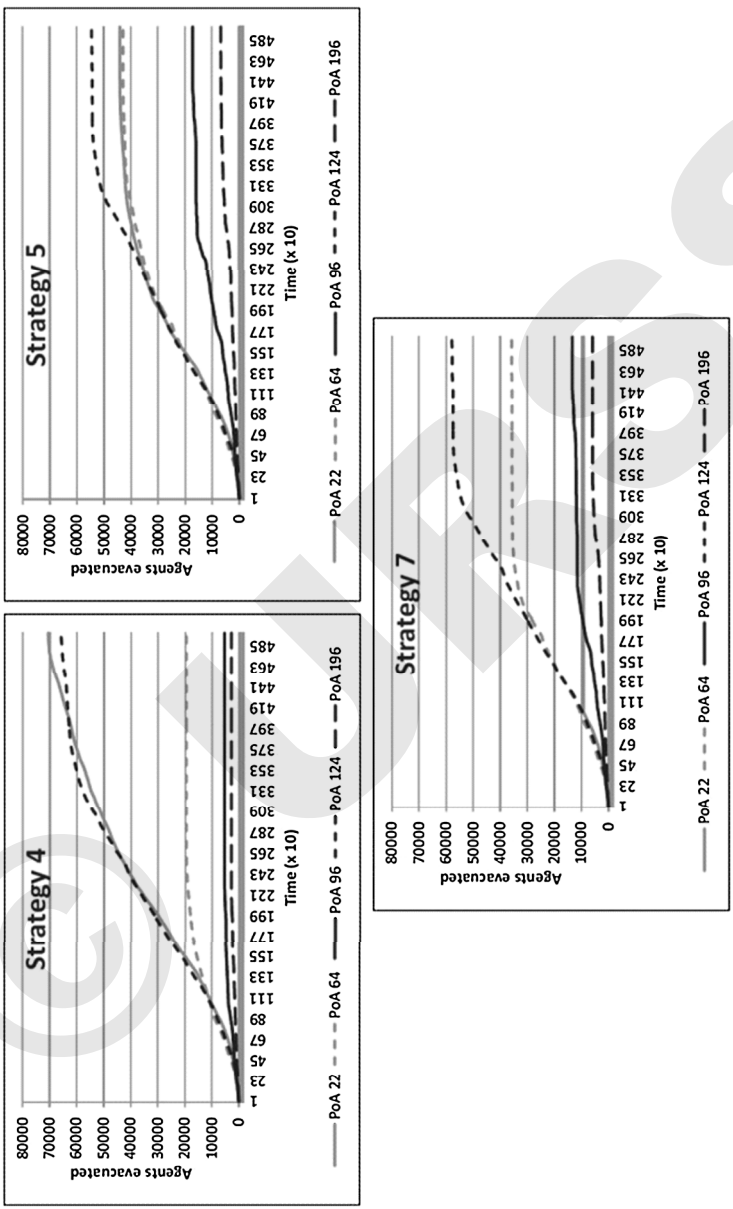


Рис. 2.1.11.16. Результаты применения отобранных стратегий (4, 5 и 7) передвижений для параллельной версии симулятора в рамках масштаба всего города: количество эвакуированных через выходы 22, 64, 96, 124 и 196 агентов. *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Если более подробно рассмотреть результаты по патчам-выходам (рис. 2.1.11.16 и табл. 2.1.11.4), то мы увидим, что при реализации стратегии 5, выходы используются более равномерно. Кроме того, в случае стратегии 4 в определенный момент времени только 2 из 5 выходов остаются активными.

Распределение агентов в рамках 5 процессов, содержащих патчи-выходы, показано на рис. 2.1.11.17. Агенты на схемах представлены в виде крошечных точек, а более крупные точки — патчи-выходы.

Таблица 2.1.11.4

Результаты применения восьми стратегий передвижений для параллельной версии симулятора в рамках масштаба всего города: количество эвакуированных через разные патчи-выходы (22, 64, 96, 124 и 196) агентов, в процентах

	Выход 22	**Выход 64**	**Выход 96**	**Выход 124**	**Выход 196**
Стратегия 4	43 %	12 %	3 %	40 %	18 %
Стратегия 5	27 %	26 %	10 %	33 %	4 %
Стратегия 7	8 %	29 %	11 %	47 %	5 %

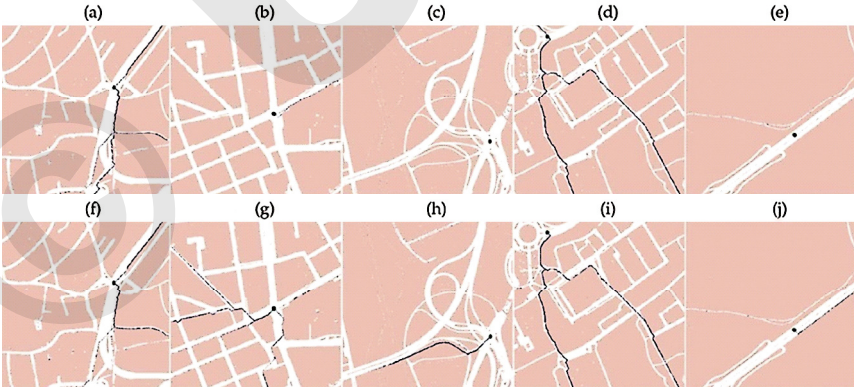


Рис. 2.1.11.17. Визуализация 5 патчей-выходов на середине симуляции (2500-я итерация) для стратегии 4 (a–e) и стратегии 5 (f–j). *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

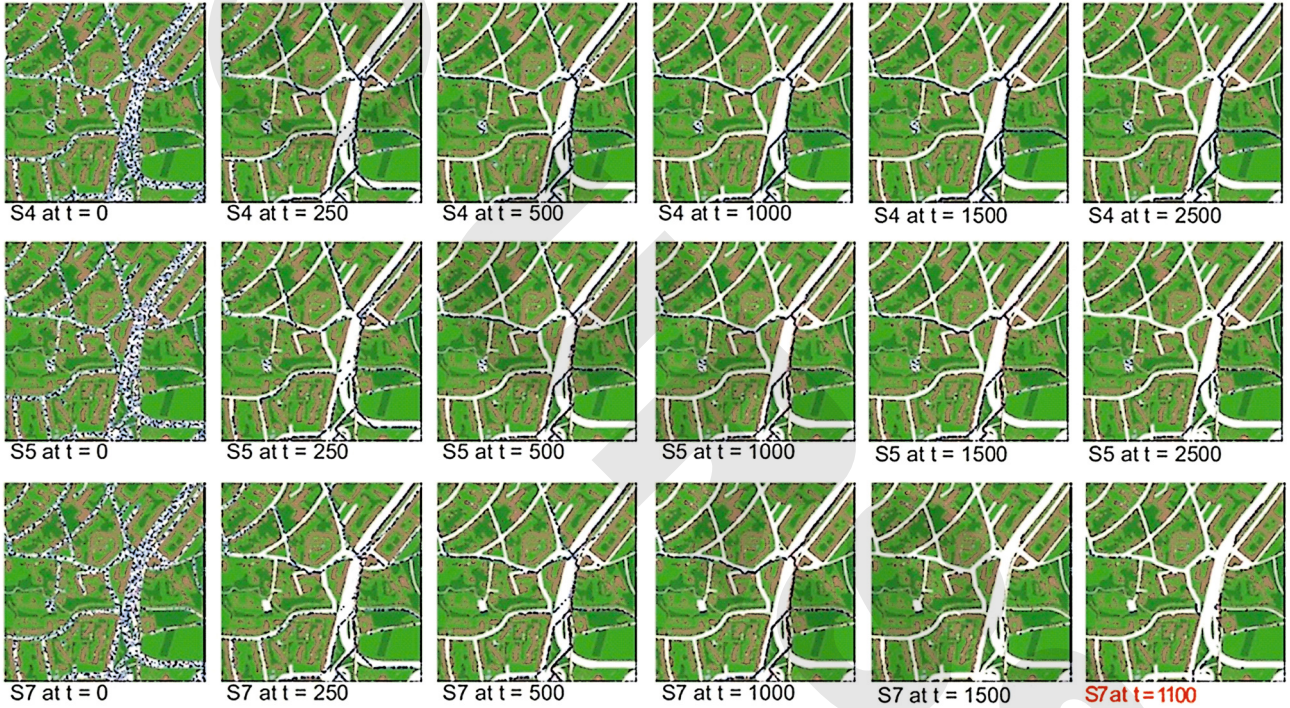


Рис. 2.1.11.18. Визуализация агентов вокруг патча-выхода 22 при реализации стратегии 4 (верхний ряд), стратегии 5 (средний ряд) и стратегии 7 (нижний ряд). *An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation* [15]

Как видно, при реализации стратегии 5 агенты лучшим образом распределяются по улицам (к примеру, если сравнить (b) и (g)). Кроме того, (h) еще используется при реализации стратегии 5, а в стратегии 4 уже нет.

Визуальное сравнение стратегий 4, 5 и 7 приведено на рис. 2.1.11.18. К примеру, видно, что когда патч-выход 22 становится недоступным (на итерации 1100), агенты перемещаются в сторону других выходов.

Общий же вывод по использованным стратегиям следующий. При моделировании систем с большим количеством агентов, несмотря на применение суперкомпьютерных технологий, принципиальное значение имеет детализация модели и степень коммуникации агентов, влияющая на частоту межпроцессорного взаимодействия. Так, упрощение в действиях агентов при применении стратегии 4 (по сравнению со стратегией 1) позволило уменьшить время расчета 5000 итераций симуляции в рамках масштаба всего города до 12 часов (в то время как за 278 часов удалось просчитать только 4 % от общего объема вычислений, необходимых для реализации стратегии 1).

Более подробно про результаты проведенного исследования можно прочитать в работе: «An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation» [15].

2.1.12. POLARIS — программная платформа для построения транспортных симуляторов на базе агент-ориентированного подхода

В Аргоннской национальной лаборатории (национальный исследовательский центр Министерства энергетики США) разработано программное обеспечение для построения агент-ориентированных моделей, в основном используемых для симуляции транспортных потоков.

Основные утилиты разработанного пакета: 1) модуль, отвечающий за параллельную обработку событий; 2) модуль, реализующий межпроцессный обмен; 3) библиотека для визуализации; 4) библиотека для ввода-вывода данных и др.

Программная платформа POLARIS позволяет интегрировать различные процедуры (распределение вычислений транспортных потоков по процессорам, спрос агентов на поездки) в рамках одной модели с общей памятью, обрабатывающей все события, происходящие в процессе симуляции городской системы.

Разработчики проекта отмечают, что до недавнего времени в моделях, рассматривающих транспортные системы, отдельные составляющие, такие как транспортные потоки, выбросы газов, формирование запроса на тот или иной вид городского транспорта и др., одновременно не учитывались. Однако за счет возрастающей сложности моделируемых систем, нелинейного увеличения необходимых для учета взаимодействующих элементов, возникла потребность в адекватных действительности инструментах моделирования дорожного движения, обеспечивающих высокую производительность конечных симуляций.

Таким образом, проект POLARIS, удовлетворяющий сформулированному выше требованию, а также позволяет:

1. разрабатывать транспортные симуляторы на базе агент-ориентированного подхода;
2. быстро создавать программные конструкции, используя функции автодополнения кода;
3. при необходимости использовать язык программирования низкого уровня для низкоуровневой оптимизации и распараллеливания разрабатываемого приложения с целью достижения его высокой производительности.

По своей сути POLARIS — это набор низкоуровневых библиотек, представляющих пользователю удобный программный интерфейс, а также среду выполнения, облегчающую написание программного кода. В разработанных с помощью POLARIS транспортных симуляторах используется большой объем данных, которые часто обрабатываются одновременно различными программными модулями. В этой связи в параллельном режиме используются директивы для программирования многопоточных приложений с общей памятью.

Комплект разработчика программного обеспечения с использованием агентного подхода — POLARIS

Ниже описывается модель, разработанная с использованием агент-ориентированного подхода и платформы POLARIS, которая состоит из двух компонент. Первый — высокоуровневая среда для проектирования агент-ориентированных моделей, а второй — комплект разработчика, позволяющий настраивать и оптимизировать построенные с помощью высокоуровневого компонента модели. Эта платформа представляет собой среду разработки с открытым исходным кодом, а также набор определенных стандартов и протоколов, позволяющих эффективно разрабатывать транспортные модели с возможностью параллельных вычислений. Одна из функциональных особенностей POLARIS — участие в общем проекте нескольких групп программистов в рамках единой моделируемой среды.

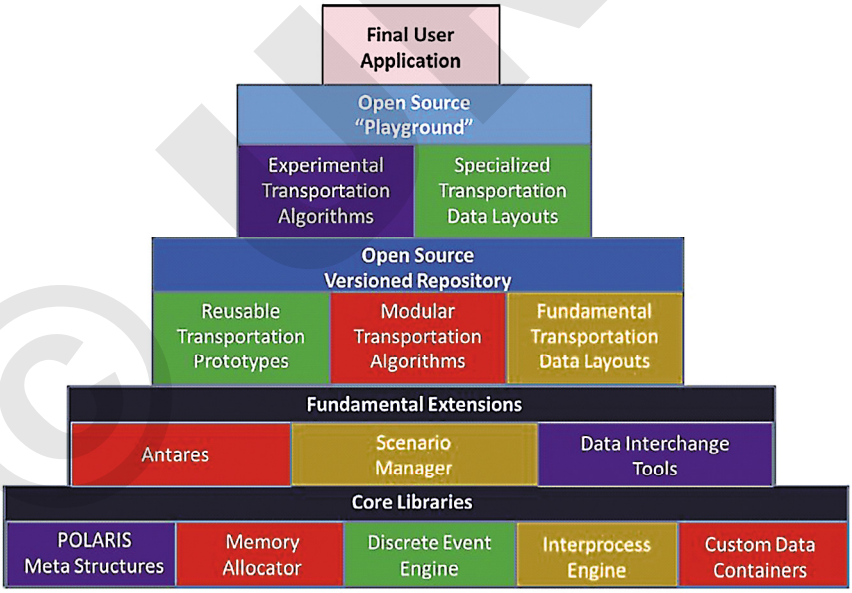


Рис. 2.1.12.1. Схема платформы POLARIS. *POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations* [16]

По мнению разработчиков, для эффективной сборки нескольких транспортных моделей в рамках единой платформы важно правильно ранжировать общие используемые компоненты с тем, чтобы часто используемые модули редактировались реже (рис. 2.1.12.1).

На самом низком уровне находятся основные библиотеки, реализующие дискретные события, механизмы управлению памятью и межпроцессорного взаимодействия, а также стандарты оформления кода. Эти библиотеки жестко контролируются с целью минимизации их модификации. На следующем уровне платформы находятся основные расширения, представляющие собой набор утилит для визуализации, ввода/вывода данных и т. д. Оба уровня являются общими и их можно использовать для любой модели (и не только транспортной). Ниже мы рассмотрим эти библиотеки и утилиты более подробно. Следующий уровень — репозиторий, включающий в себя модули, которые прошли неоднократное тестирование и могут быть использованы в разных моделях — это хорошо отработанные алгоритмы прокладывания маршрутов, интерфейсы для изменения геометрии транспортных сетей и др. Основной функционал данного репозитория — использование отлаженных компонент в составе будущих моделей, а также его постоянное пополнение силами внешних разработчиков. Следующий уровень — пользовательское пространство POLARIS, в рамках которого разработчики могут экспериментировать с новыми расширениями платформы и тестировать узкоспециализированные функции. Используя элементы всех уровней, разработчики могут собрать готовое приложение, и пример такой сборки будет описан ниже.

Распределитель памяти для симуляций

Распределение памяти является важной составляющей высокопроизводительного программного обеспечения и дает значительные возможности для низкоуровневой оптимизации системы. В тоже время, динамическое распределение памяти может негативно отразиться на производительности программы по двум основным причинам. Во-первых, при создании новых объектов тре-

буется постоянно освобождать выделенную ранее память, а в случае некорректного управления памятью возможны так называемые «утечки», при которых память не освобождается, а ее исчерпание может повлечь нестабильность уже операционной системы. Во-вторых, выделение памяти осуществляется блоками — непрерывными фрагментами, и в определенный момент в куче может не быть фрагмента подходящего размера, то есть даже при наличии достаточного количество свободной памяти, операция по ее распределению может быть неуспешной.

Распределитель памяти POLARIS использует методы управления, которые направлены на повышение производительности многоагентного приложения, работающего с использованием дискретно-событийной парадигмы имитационного моделирования. Программный код, разработанный для реализации агентов приложения, отличается от программного кода остальной его части, поскольку содержит в себе инструкции по распределению/освобождению памяти от агентов (к примеру, каждого участника дорожного движения). Кроме того, в этом многопоточном приложении объекты — агенты распределяются/освобождаются нерегулярным образом и поэтому для них требуется механизм управления памятью с минимальным общением между потоками с целью повышения производительности симуляции.

В качестве основного распределителя был выбран TCMalloc (Thread-Caching Malloc), разработанный Google. TCMalloc отличается от традиционных распределителей тем, что специально разработан для многопоточных приложений и он был выбран среди других альтернатив за счет своей производительности и кроссплатформенности.

Подпрограммы распределения/освобождения тесно связаны с процессами составления графиков отдельными агентами. В зависимости от того, выполнил ли в данный момент какой-либо объект определенную операцию, или выполнил ли поток запланированный программный код, для распределения или освобождения памяти используются соответствующие процедуры.

Освобождение блоков осуществляется индивидуально с применением спинлока (циклической блокировки) — примитива низкоуровневой синхронизации, которой реализуется в механизме взаимного исключения (мьютексе). Блоки распределяются только управляющим потоком, за счет чего минимизируется конфликт потоков до того момента, когда вычисления становятся полностью параллельными. В процессе симуляции, в определенные интервалы потоки возвращают основному распределителю незадействованную память, а пользователю предоставляется опционная возможность выбора варианта освобождения памяти — незамедлительно, по расписанию конкретного агента или по мере необходимости. Все упомянутые выше механизмы по управлению памятью существенно повышают производительность конечного приложения.

Модуль для обработки параллельных событий

Одним из центральных элементов платформы POLARIS является модуль для обработки дискретных событий (Discrete EVent Engine, DEVE), оптимизированный для разработки многопоточного кода с использованием распределителя памяти. Он предоставляет собой программный интерфейс, который позволяет создать агента любого типа (человека, автомобиль и т. д.) и описать его деятельность (вид, время начала функционирования, продолжительность и др.).

Агенты проявляют свои намерения посредством «событий», а в свою очередь DEVE организует выполнение функций агентов в установленном разработчиком порядке и вместе с тем упорядочивает потоки для их бесконфликтной работы. Помимо этого, модуль DEVE разрешает проводить различные имитации с использованием программного обеспечения Intel VTune, позволяющего выявить и устранить узкие места в плане производительности.

Дополнительные возможности платформы

Помимо основных компонент (распределения памяти и обработки параллельных событий) в платформе POLARIS реализовано несколько дополнительных возможностей, весьма полезных для

реализации многоагентных транспортных моделей. К ним относятся: 1) графическая библиотека для визуализации интерактивных элементов, 2) интерфейсы для ввода-вывода данных, 3) модуль задания сценариев для проведения множественных экспериментов.

Библиотека Antares предоставляет разработчикам инструменты для 2D- или 3D-визуализации. Сначала создается слой для представления определенного типа объекта (автомобилей, дорог и др.), а затем слой конфигурируется для передачи геометрических фигур (треугольников, квадратов, точек, линий, многоугольников). Кроме того, можно определить отдельное изображение под конкретный объект (к примеру, рисунок автомобиля поверх первоначальной точки).

В основе Antares лежит графическая библиотека OpenGL, определяющая программный интерфейс для написания приложений, использующих двумерную и трехмерную компьютерную графику, а также wxWidgets — кроссплатформенная библиотека инструментов с открытым исходным кодом для разработки приложений с использованием графического интерфейса пользователя. Также отметим, что Antares обеспечивает многопоточность и оптимизацию за счет Intel VTune.

Другой возможностью построения многоагентных моделей, обеспечивающей стандартизацию, является использование унифицированного механизма обмена данными, реализованного в SQLite — компактной встраиваемой СУБД. Библиотеки C++ и Python, доступные как часть проекта POLARIS, позволяют разработчикам считывать все данные из базы данных SQLite, причем реализованная в платформе объектная база данных позволяет использовать язык запросов, написанный на C++ вместо использования SQL.

И, наконец, модуль задания сценариев предоставляет собой набор инструментов для формирования и запуска различных наборов сценариев, описываемых с использованием JavaScript и реализуемых затем в параллельном режиме. При задании сценариев возможно варьировать большим набором параметров (это могут быть погодные условия, дорожные происшествия и др.), а во время реализации в интерактивном режиме можно осуществлять мониторинг

их состояния, включая такие показатели как процент выполнения, оставшееся время работы, загрузку кластеров и др. Менеджер сценариев использует PBS (Portable Batch System) — систему управления распределенными вычислениями, основной функцией которой является запуск задач в вычислительной среде по расписанию.

Многоагентная модель, реализованная в POLARIS

Ниже описывается пример модели, реализованной с помощью платформы POLARIS. В реализующем агента классе прописываются методы, отвечающие за различный функционал (формирование спроса на поездки, поведение во время движения и т. д.). Набор компонент интеллектуальной транспортной системы (ИТС) и центр управления движением (ЦУД), реализованные в модели, контролируют всех агентов, представляющих собой участников дорожного движения в Чикаго.

Спрос на поездки формирует агент — участник дорожного движения, причем у каждого агента есть несколько субагентов, которые осуществляют различные виды деятельности:

- *субагент — обработчик информации* — собирает и обрабатывает внешнюю информацию;
- *субагент-планировщик* — решает вопросы планирования и перепланирования;
- *субагент-диспетчер* — создает и по мере необходимости изменяет ежедневный график поездок;
- *субагент — управляющий перемещениями* — осуществляет имитацию перемещений в системе;
- *субагент-маршрутизатор* — выбирает маршруты в сети.

В рамках модели агент посредством своих субагентов выбирает маршруты и реализует перемещение. Взаимодействие между различными субагентами в процессе симуляции показано на рис. 2.1.12.2. Через определенные временные промежутки агенты проверяют необходимость перепланирования своей деятельности, а при необходимости изменяют график поездок и маршруты.

©URSS

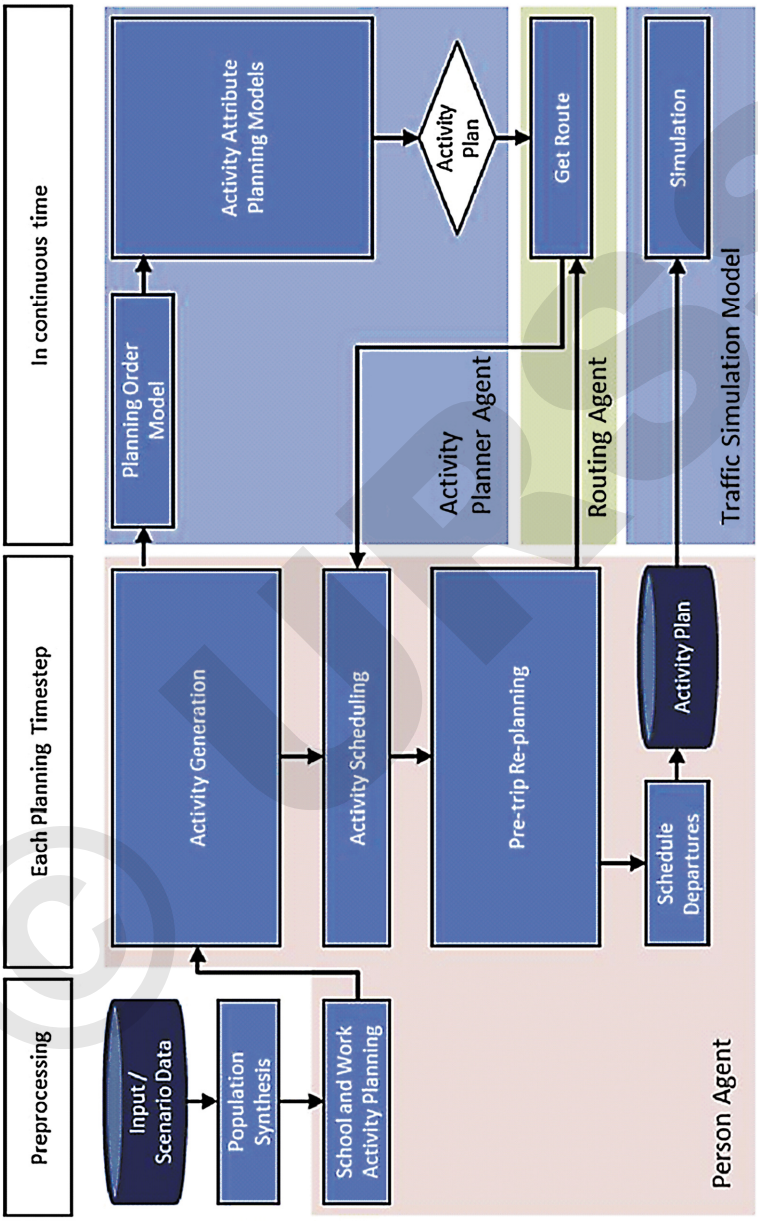


Рис. 2.1.12.2. Платформа многоагентной модели спроса. *POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations* [16]

Действия агентов можно разделить на четыре компоненты: генерирование активности, планирование, составление графиков поездок, перемещения.

Генерирование активности

Генерирование активности в модели обеспечивается набором субмоделей, реализующих механизмы оценки рисков и продолжительности того или иного действия. В зависимости от ряда условий происходит генерация последовательности различных видов деятельности, очередность которых зависит от результатов оценки рисков.

Планирование и перепланирование деятельности

Сразу после генерирования активности, субагент-планировщик осуществляет планирование в зависимости от ряда факторов (напр., занятость агента на работе, наличие свободного времени и др.). Соответствующая модель состоит из взаимосвязанных множественных упорядоченных пробит-регрессий.

Изменение ежедневного графика поездок

Используя модель на основе эвристических правил, субагент-диспетчер составляет графики поездок, нивелируя возможные конфликтные ситуации, связанные с пространственно-временными ограничениями.

Имитация перемещений

Субагент, управляющий данными методами, реализует имитацию перемещений с использованием соответствующего программного интерфейса. К примеру, помещает человека в автомобиль и направляет транспортное средство к определенному участку карты исходя из составленного заранее плана. Причем пока агент перемещается, перечисленные выше субагенты продолжают работать (в случае необходимости перепланируют его деятельность, меняют график и т. д.).

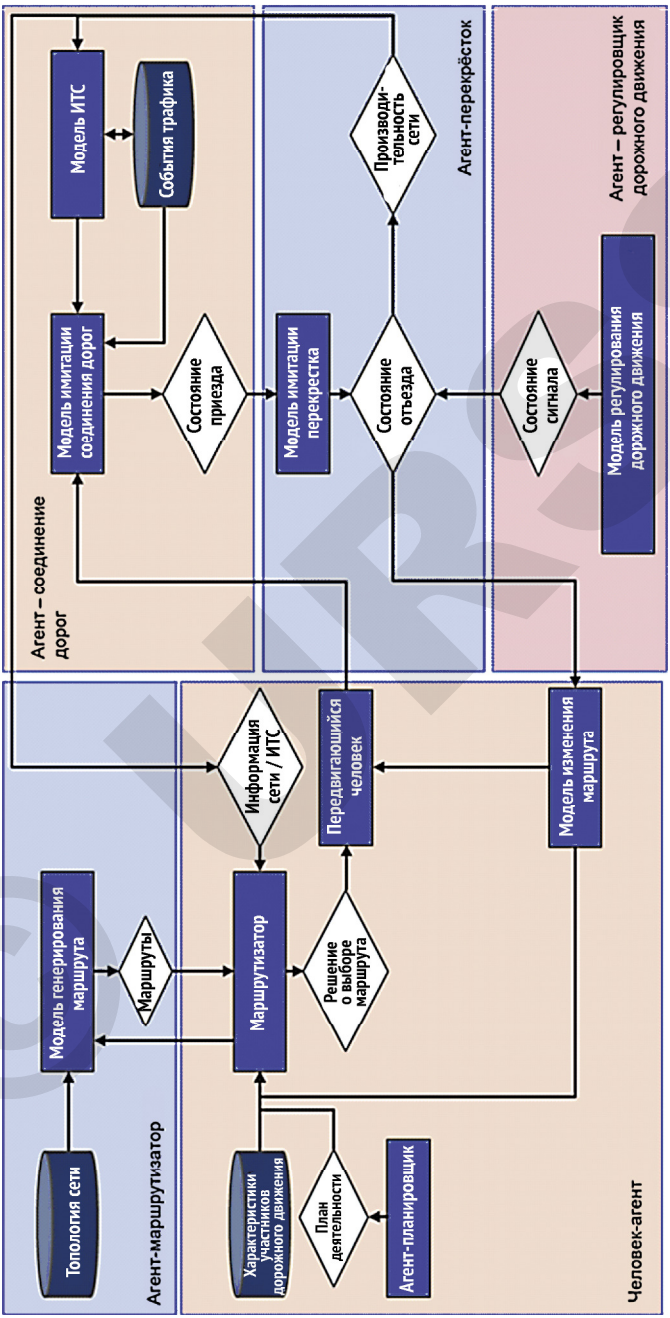


Рис. 2.1.12.3. Схема взаимодействия между агентами в рамках транспортной сети. *POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations* [16]

Модель имитации транспортной сети

В модели агенты выбирают маршруты в зависимости от собственных предпочтений, но также исходя из информации о трафике, получая информацию через общие события модели (это может быть имитация получения информации через Интернет относительно пробок, погодных условий, ДТП и т. д.). Таким образом субагенты-маршрутизаторы вычисляют кратчайшие маршруты, оптимизируя транспортные расходы. Взаимодействие между агентами в рамках сети представлено на рис. 2.1.12.3. Результатом работы модели имитации транспортной сети являются параметры производительности сети, служащие входными данными для модели выбора маршрута.

Модель выбора маршрута

Модель выбора маршрута определяет решения участников движения в части выбора маршрутов, исходя из полученной до и во время поездки информации о ситуации на дорогах.

Модель генерации маршрута

В рамках модели генерации маршрута рассчитываются кратчайшие расстояния от исходной точки до пункта назначения. У каждого субагента-маршрутизатора есть своя копия сетевой топологии, включающая персонифицированную информацию о прогнозировании транспортных расходов исходя из накопленного опыта. Что касается алгоритма поиска кратчайшего маршрута, то он осуществляется с использованием алгоритма поиска «А звезда» (A star, A*), позволяющего найти путь с наименьшей стоимостью от одной вершины (начальной) к другой (конечной). Кроме того, его использование позволяет эффективно распараллелить мультиагентное приложение.

Модель имитации трафика

Совокупное поведение водителей описывается нелинейным дифференциальным уравнением с частными производными пер-

вого порядка. Модель рассчитывает динамику плотности трафика, являющуюся макроскопической характеристикой потока трафика. В POLARIS есть три компоненты, разработанные для вычисления спроса на поездки, работы с сетевыми утилитами и для обработки ряда событий, получаемых от датчиков системы и срабатывающих в процессе симуляции (рис. 2.1.12.4). Последние обеспечивают данными ЦУД, обрабатывающий события, связанные с заторами, ДТП, изменениями погодных условий и т. д.

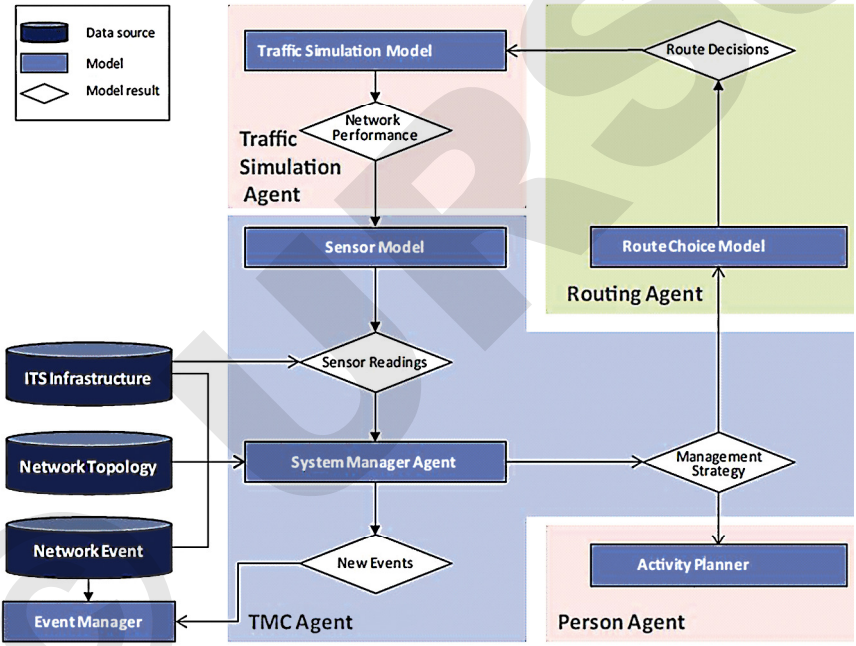


Рис. 2.1.12.4. Компоненты модели управления трафиком. *POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations* [16]

Симуляции и предварительные результаты

Для оценки эффективности платформы POLARIS исследователи провели симуляции для Большого Чикаго (метрополитенский

ареал Чикаго) — городской агломерации, в которую входят 10 млн человек, совершающих в среднем 27 млн поездок. Компоненты модели были откалиброваны с использованием данных подробной матрицы корреспонденций. На рис. 2.1.12.5 приведены усредненные данные тестовых прогонов модели.

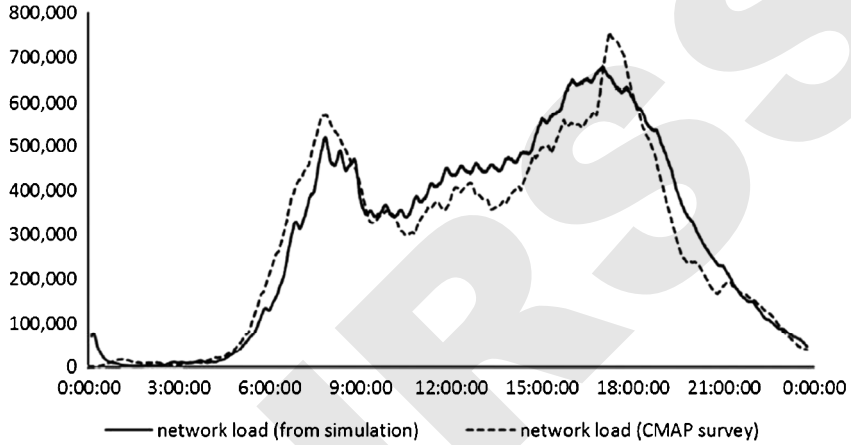


Рис. 2.1.12.5. Количество автомобилей в сети в зависимости от времени суток: пунктирная линия — реальные данные, сплошная линия — данные симуляции платформы POLARIS. *POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations* [16]

Для более тщательного анализа был выделен центральный деловой район Чикаго (Chicago CBD), который содержит примерно 5000 дорожных отрезков, 3000 перекрестков, 6500 зон притяжения для 400 000 агентов — участников дорожного движения. Транспортная инфраструктура также включает в себя 20 информационных табло, расположенных вдоль скоростных автомагистралей по всей территории, с меняющимися сообщениями относительно загрузки дорог. В рамках симуляций рассчитывались последствия трех сценариев: 1) обычный день без происшествий; 2) сценарий с ДТП с отключенной информационной инфраструктурой; 3) сценарий с ДТП и включенной информационной инфраструктурой.

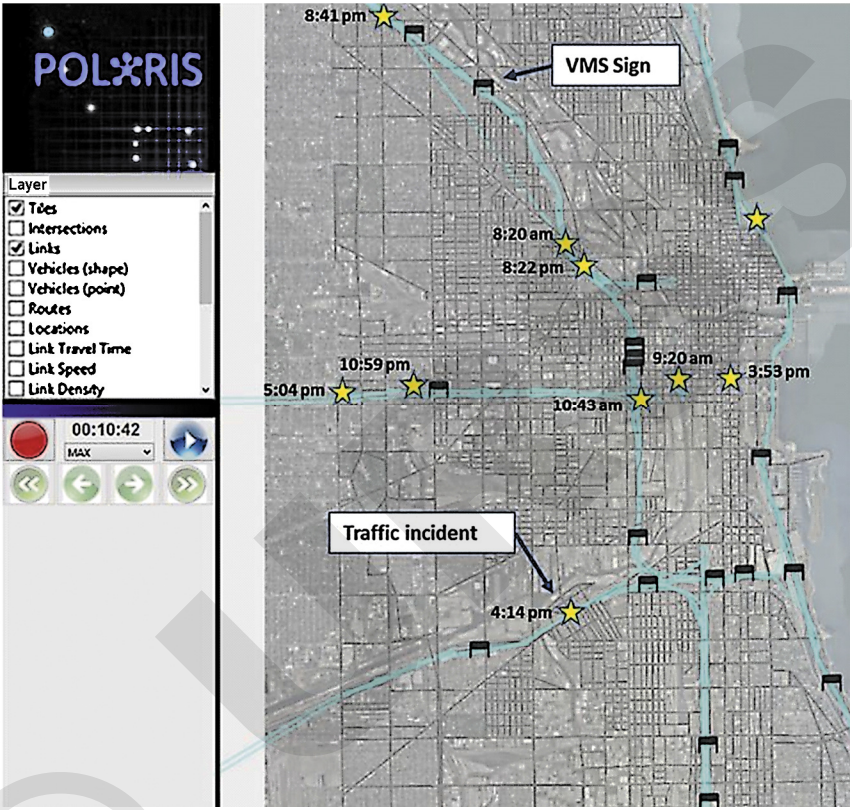


Рис. 2.1.12.6. Дорожная сеть центрального делового района Чикаго с информационными табло, реализованная в платформе POLARIS. Графический пользовательский интерфейс предоставляет возможность интерактивного взаимодействия с симуляцией в процессе ее реализации (к примеру, можно регулировать, какие сообщения должны показываться на табло). Места ДТП помечены желтыми маркерами. *POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations* [16]

ДТП в рамках имитации включает в себя 9 аварийных событий (это среднестатистический показатель для выбранного района).

На рис. 2.1.12.6 приведен скриншот платформы POLARIS применительно к описанной симуляции. В случае ДТП у соответствующих дорожных отрезков снижается пропускная способность и скорость потока. Агенты, получая информацию с табло, или же непосредственно наблюдая затор, могут инициировать запуск соответствующей субмодели для изменения маршрута.

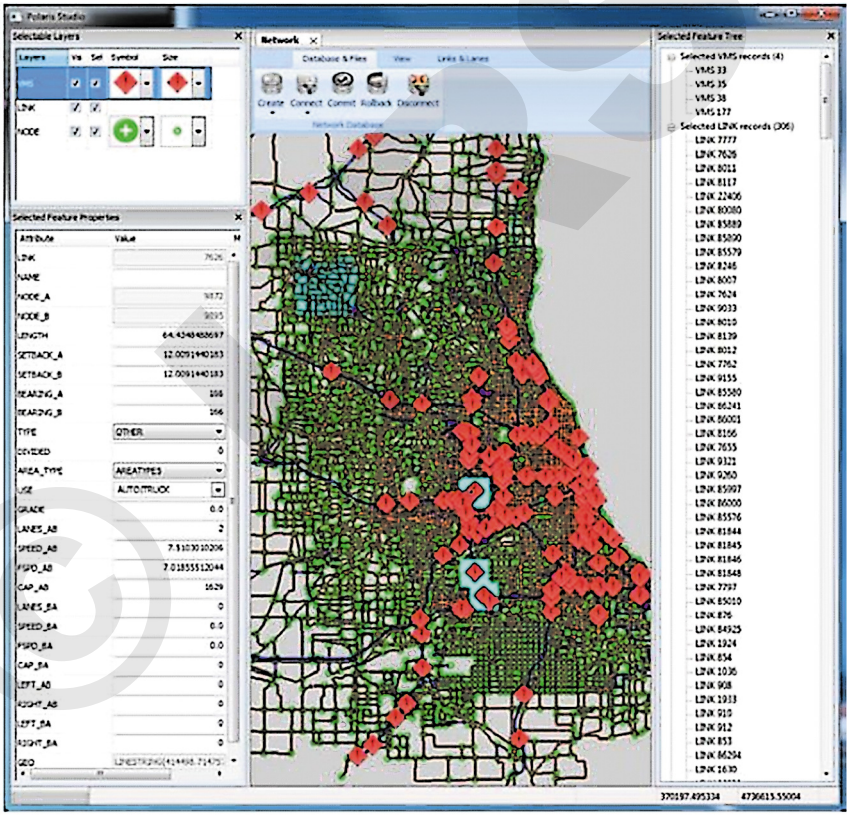


Рис. 2.1.12.7а. Редактор дорожной сети

Отметим, что пользовательский интерфейс POLARIS поддерживает широкий спектр возможностей для редактирования элементов дорожной сети (перекрестков, слоев ГИС с различными метками и т. д.). Скриншоты упомянутого редактора приведены на рис. 2.1.12.7a и b.

На рис. 2.1.12.8 отражено влияние ДТП на эффективность дорожной сети с точки зрения времени задержек автомобилей в заторах. Как видно, информационные табло оказывают положительное влияние на отслеживаемый показатель. Так, их использование снижает общее время задержек на 3462 часа, или 18 %.

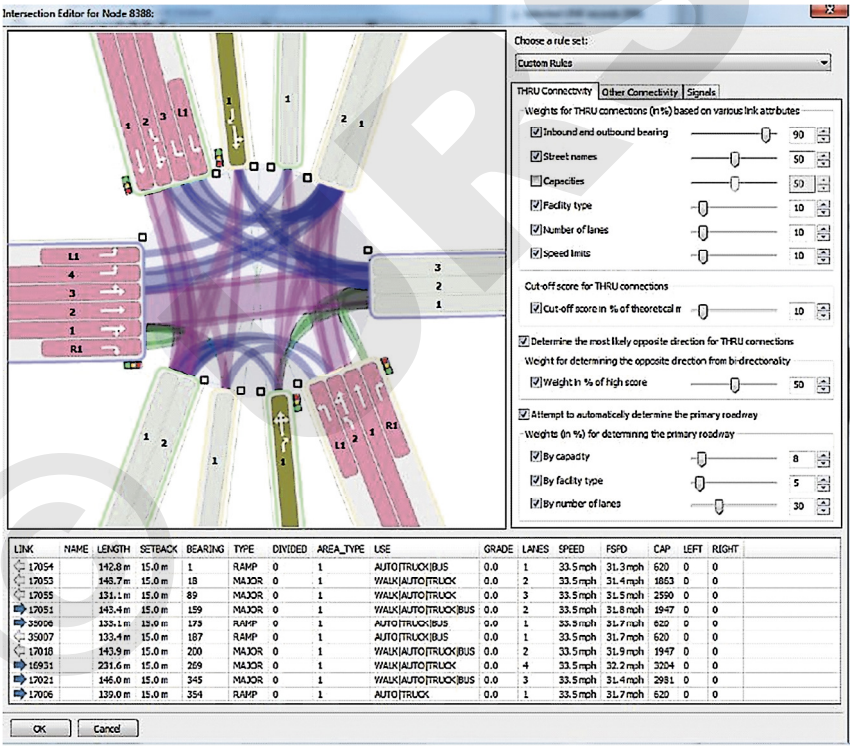


Рис. 2.1.12.7b. Редактор перекрестков. *POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations* [16]

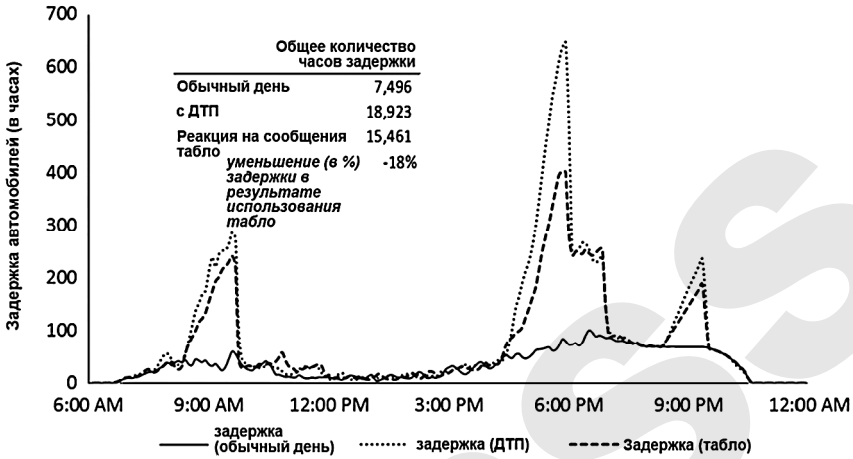


Рис. 2.1.12.8. Результаты симуляций для трех сценариев: 1) обычный день без происшествий; 2) сценарий с ДТП и отключенными информационными табло; 3) сценарий с ДТП и включенными табло, уведомляющими агентов о местах ДТП. *POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations* [16]

Более подробно про POLARIS можно прочитать в статье: «POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations» [16].

2.1.13. Симулятор поведения толпы с использованием агентного подхода, клеточных автоматов и высокопроизводительных параллельных вычислений

В работе ученых Научно-технического университета г. Кракова (AGH University of Science and Technology) рассматривается симулятор, построенный на базе агентного подхода, позволяющий имитировать поведение толпы на различных объектах — от небольших помещений до крупных центров притяжения большого количества людей (стадионов, высотных зданий и т. д.). Использование клеточных автоматов в качестве основы для пространственного рас-

пределения агентов позволяет эффективно распараллеливать программный код.

В классических моделях клеточных автоматов агенты обычно представляются в виде определенного состояния ячейки (рис. 2.1.13.1, a). К примеру, каждому индивиду соответствует ячейка, пропорциональная квадрату со стороной равной 40 см. По мнению исследователей, данный подход довольно грубо распределяет агентов в рамках пространства и поэтому, в качестве альтернативы, был предложен способ реализации их перемещения, основанный на модели социальных сил, а также с использованием мелкозернистой решетки и отображением агентов в виде эллипсов (рис. 2.1.13.1, b).

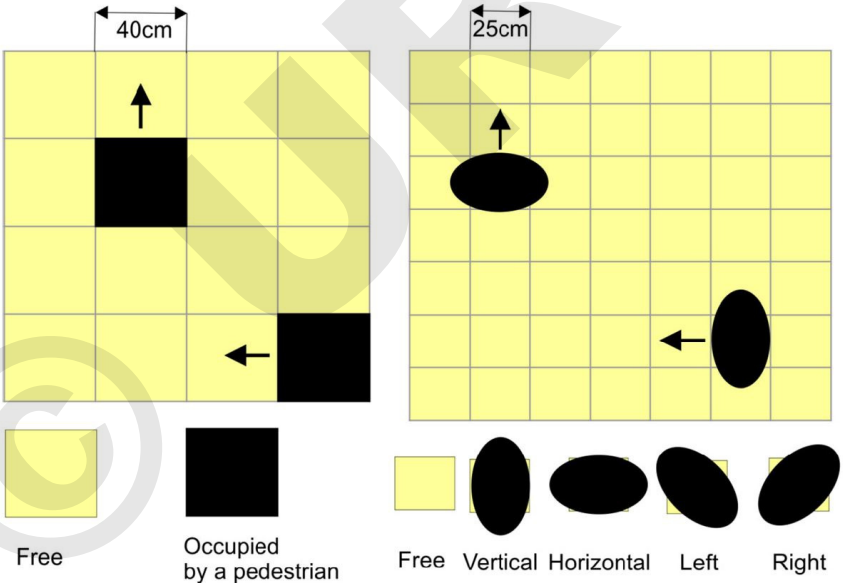


Рис. 2.1.13.1. Физическое представление агентов в классической модели клеточного автомата (a) и в модели социальных сил (b). *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

В предлагаемой модели, рассматривающей процесс эвакуации, клеточный автомат определяется как кортеж вида $CA = (L, S, N, f)$, где L — это квадратная решетка, содержащая различные типы пространств, определяющих возможности перемещения агентов (лестницы, эскалаторы, препятствия и т. д.); S — набор состояний агента, включая его тип и физическое расположение; N — окрестность Мура; f — функция перехода, зависящая от свойств агента и типа занятой ячейки.

Агент в модели определяется следующим набором

$$A_j = (\tau_j, (x_j, y_j), R_j),$$

где τ_j — тип агента j; x_j, y_j — его координаты, а R_j — стратегическая, тактическая и оперативная способности агента.

Большинство моделей, рассматривающих движение агентов, оперируют достаточно простыми правилами их поведения (**оперативного** и изредка **тактического уровня**). В предлагаемой авторами модели используются множество дополнительных факторов, влияющих на процесс принятия решений — распространение информации через систему вещания, инструкции для персонала и др. В зависимости от расположения громкоговорителей, ячейки различаются по степени восприятия силы звука, а звуковые поля (sound fields, *SF*) могут быть определены по следующей упрощенной формуле (2.1.13.1):

$$SF_{x,y} = \{(x', y'): (x' - x)^2 + (y' - y)^2 \leqslant r^2\}, \qquad (2.1.13.1)$$

где x, y — координаты источника звука, а r — всенаправленный радиус сферической волны.

Когда агент попадает в зону звуковых полей, он может услышать сообщение с вероятностью P_{SF} и изменить свое поведение (пойти к другому аварийному выходу, увеличить/уменьшить скорость и т. д.). Другая важная возможность для корректировки принятия решений на **стратегическом уровне** — общение внутри группы агентов (в непосредственной близости или с использованием мобильных устройств связи), определяемой набором

$$(L_i, R_i, S_i),$$

где $L_i = (x_i, y_i)$ — лидер группы i, а x, y — его координаты; $R_i = (r_1, r_2, \ldots, r_n)_i$ — правило группы i; $S_i \in \{2, 3, 4, \ldots,\}$ — размер группы i; r — отдельное правило из числа возможных n; $i \in \{1, 2, 3, \ldots, m\}$ — индекс группы.

В начале симуляции агент может быть отнесен к определенной группе с вероятностью P_{GB}, если расстояние между ним и лидером группы, определяемое следующим образом:

$$D = f(\rho) = \mathrm{Dens}_{\max} / \rho,$$

где ρ — плотность агентов на квадратный метр вокруг лидера группы, меньше некоторого задаваемого значения.

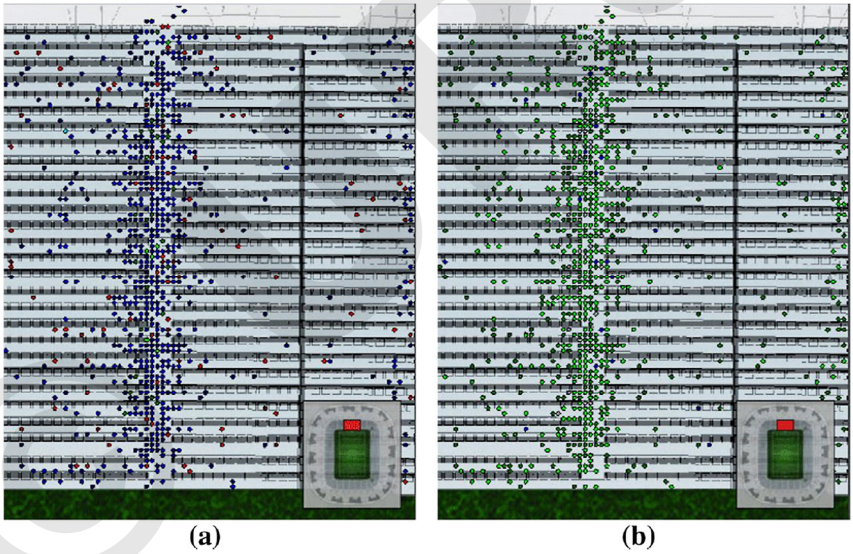


Рис. 2.1.13.2. Пример эвакуационного участка, расположенного на нижнем уровне стадиона. После получения сообщений о проблемах на выходе, агенты снизили скорость перемещения: на рис. (a) темные цвета означают более высокие скорости, а на рис. (b) светлым цветам соответствуют низкие скорости. *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

В процессе симуляции агенты группы придерживаются определенному набору правил, к примеру: (1) правило r_1 — «следуй за лидером», предписывающее агенту пытаться перемещаться к ячейке где находится лидер, или, в крайнем случае, в окрестность Мура порядка 2; (2) правило r_2 — «не обгоняй», предписывающее сохранять скорость, не превышающую скорость лидера группы, и др.

Пример принятия агентами решений на стратегическом уровне приведен на рис. 2.1.13.2, где отображен процесс эвакуации с футбольного стадиона на севере Мюнхена — «Альянц Арена» (Allianz Arena). Во время соответствующей симуляции агентам сообщили о том, что ближайший выход заблокирован и до поступления новой информации им нужно оставаться на текущих позициях, в результате чего средняя скорость агентов снизилась со значения 1,34 до 0,5 м/с.

Тактический и оперативный уровень

Используемый на этом уровне алгоритм состоит из двух основных этапов: принятие решения и разрешение конфликтов. *На первом этапе* агент выбирает следующую для перемещения ячейку $c_{i,j}$, учитывая конфигурацию окружающего пространства, наличия рядом стен и других препятствий в рамках окрестности Мура порядка 1. Горизонт его видимости, визуализированный на рис. 2.1.13.3, вычисляется по следующей формуле:

$$VF_{x,y} = \left\{(x', y') : |x' - x| \le r \wedge |y' - y| \le r \wedge \arccos\left(\frac{a \cdot b}{|a||b|}\right)\right\} \le 90°, \quad (2.1.13.2)$$

где x, y — координаты центральный ячейки, r — радиус окрестности Мура, dx и dy — координаты ячейки с минимальными значениями; a — вектор (x', y'), b — вектор (dx, dy).

Для агентов используется система приоритетов — в зависимости от их местонахождения, принадлежности к определенному типу (зрители, персонал, пожарные) и т. д., влияющих на порядок вычислений, осуществляемых в многопоточном режиме.

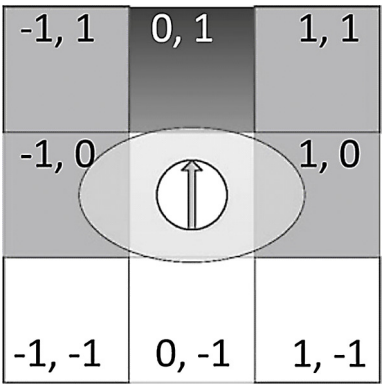


Рис. 2.1.13.3. Агент занимает центральную ячейку с координатами (0, 0), черная ячейка — локальный минимум со значением (*dx*, *dy*) = (0, 1), светлые и темно-серые ячейки — поля видимости, вычисленные по формуле (2.1.13.2). *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

Градиент окружающего пространства указывает выбранный агентом объект (Point of Interest, *POI*) для перемещения, к примеру эвакуационный выход. Таким образом, агент получает информацию о примерном направлении движения, а также об агентах, проделавших этот путь ранее.

Процесс принятия решений агентом в том числе включает выбор ячейки для последующего перемещения. С использованием формулы (2.1.13.3) рассчитывается конкретная ячейка из набора доступных ячеек в рамках горизонта видимости *VF*, а за счет варьирования весовых параметров отдельных агентов, траектории их движения незначительно различаются.

$$\mathrm{Cost}(c_{ij}) = S_{ij}^{a} + \left(\mathrm{dens}(c_{ij})\right) + \alpha \cdot \mathrm{dist}(c_{ij}, POI)) \cdot W \cdot I,$$
$$\mathrm{Dens}(c_{ij}) = e^{\delta \cdot D_{ij}}, \qquad (2.1.13.3)$$

где S^{a}_{ij} — значение в возможном для перемещения статическом поле, a — индекс возможного поля, ведущего к *POI*; $\alpha \in [0, \infty)$ — весовой параметр; dens (c_{ij}) — плотность в окрестности ячейки c_{ij}; D_{ij} — значение динамического поля; $\delta \in [0, \infty)$ — весовой параметр; dist(c_{ij}, *POI*) — расстояние между ячейкой с координатами i, j и ближайшим *POI*; W — параметр, определяющий возможность уклонения от стен и учитывающий расстояние от них; I — параметр инерции, определяющий возможность сохранения направления движения.

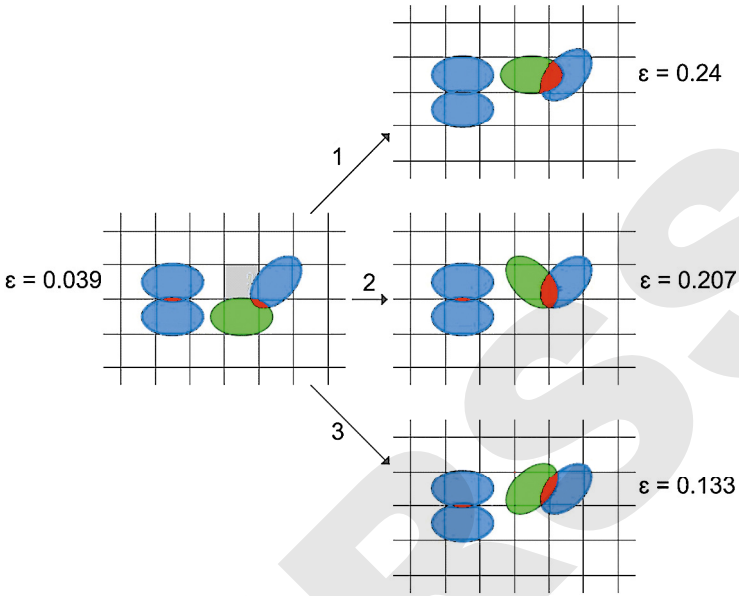
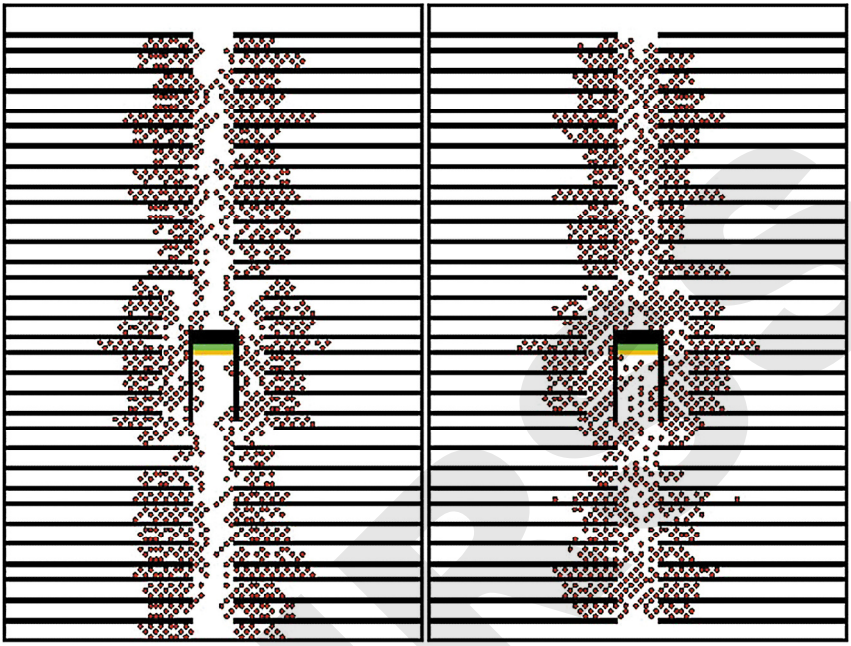


Рис. 2.1.13.4. Возможные сценарии разрешения конфликта. Слева — начальная фаза конфликтной ситуации, справа — возможные варианты разрешения конфликта, подразумевающие изменения положения тела агента. *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

Вторая составляющая алгоритма формализует процесс разрешения конфликтов, когда на одну ячейку претендует более одного агента. В модели предусмотрен коэффициент сжимаемости ε, в зависимости от значения которого меняется расположение агентов относительно друг друга (примеры некоторых возможных конфигураций приведены на рис. 2.1.13.4).

В процессе перемещения агент проверяет возможность занятия выбранной ячейки, и если коэффициент ε не позволяет сделать это непосредственно, а также повернувшись на 45°, то агент остается на месте, но при этом алгоритм разрешения конфликта сразу же запускается вновь относительно других, следующих по значимости для агента ячеек. Визуализация результатов симуляции с учетом этого дополнительного шага, повышающего реалистичность модели, приведены на рис. 2.1.13.5.



(a) (b)

Рис. 2.1.13.5. Пример моделирования одного сектора стадиона «Висла» (г. Краков): без использования дополнительного шага (a) и с его применением (b). *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

Симулятор

Построенный симулятор, одновременно являющийся и фреймворком для разработки аналогичных моделей, реализован с использованием многоуровневой архитектуры, слои которой перечислены ниже:

1. данные и логика работы симулятора (пакет *core*);
2. слой обработки пользовательских запросов и также содержащий служебные утилиты (пакеты *engine* и *tools*);
3. слой, содержащий графический интерфейс пользователя (пакет *gui*).

На рис. 2.1.13.6 приведена схема взаимодействия между слоями и компонентами программных пакетов.

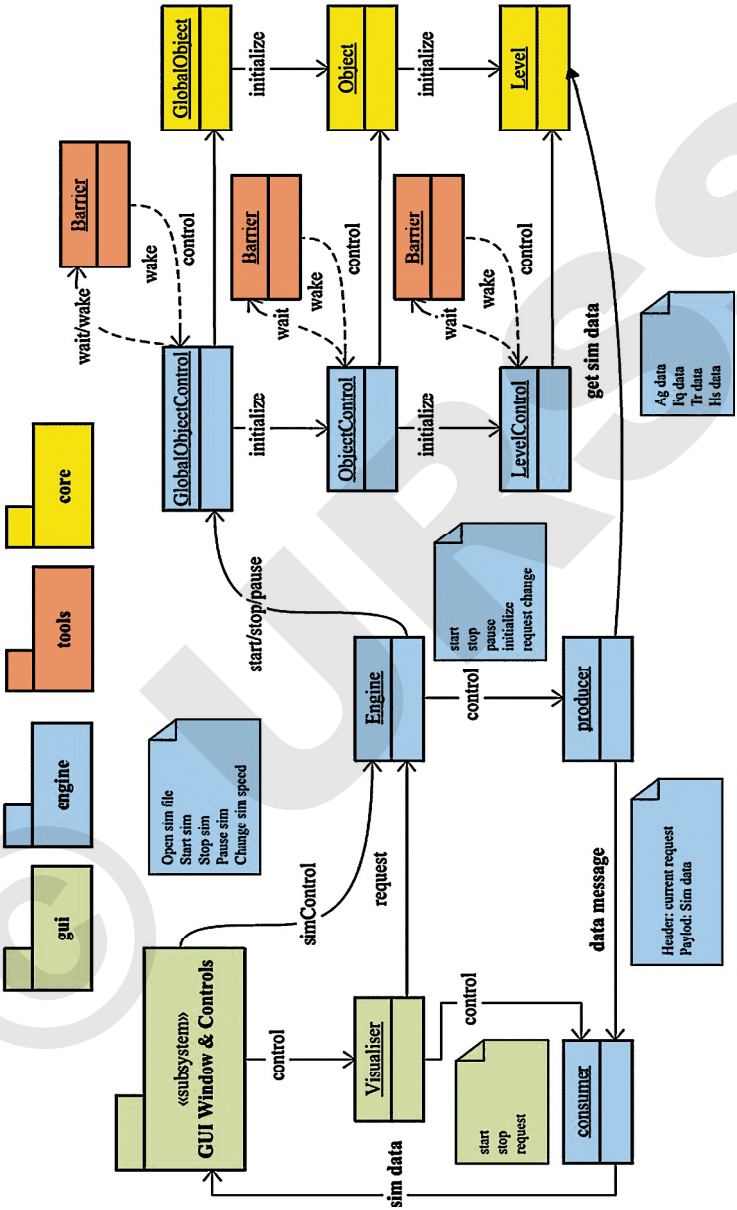


Рис. 2.1.13.6. Архитектура симулятора. *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

На логическом слое симулятора разработчики могут оперировать глобальным объектом (*Global Object*) и уровнями (*Levels*; к примеру, стадионом, отдельными трибунами, секциями и т. п.). В такой нотации стадион «Висла» является глобальным объектом, его четыре трибуны (восток, запад, север и юг), соответствуют объектам, а секции трибун являются уровнями (рис. 2.1.13.7).

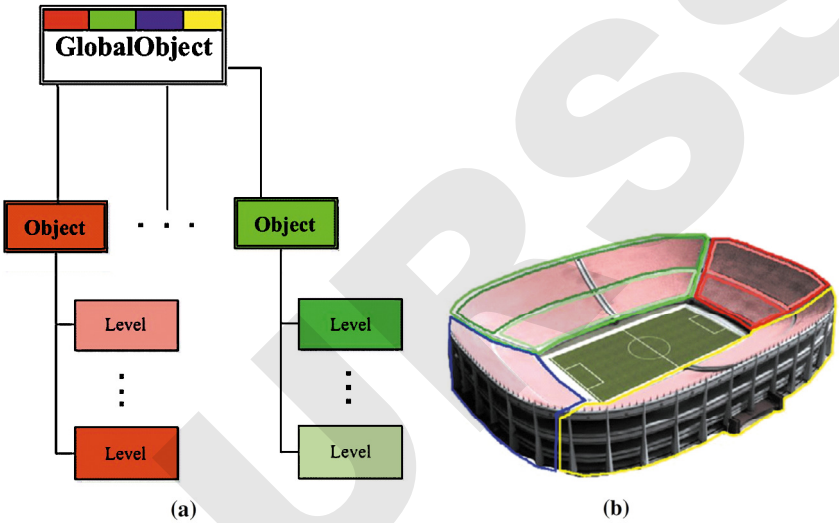


Рис. 2.1.13.7. Иерархия объектов на слое логики и данных (a); разделение стадиона на секции, связанные с логическим слоем (b). *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

Структура симулятора хорошо масштабируема для разных объектов (стадионы, высотные здания и т. д.). Среди важных компонентов, учитываемых в уровнях можно выделить статические и динамические поля, геометрию помещений и настройку симуляций.

Статическое поле определяется трехмерным массивом $S[x][y][POIs]$, позволяющим агентам модели двигаться к *POI* (например, к аварийному выходу). В свою очередь, динамическое поле определяется двумерным массивом $D[x][y]$, меняющимся на каждом шаге моделирования.

Слой, содержащий служебные утилиты, включает в себя директивы для распараллеливания программного кода, а при синхронизации используется авторская методика с применением барьеров памяти для компонентов трех уровней (глобального объекта и объектов двумя уровнями ниже). Обработка событий, связанных с переходом агента на другой уровень (секцию или к *POI*), то есть обращение к памяти после барьера, будет возможна только после выполнения всех инструкций в рамках текущего уровня (рис. 2.1.13.8).

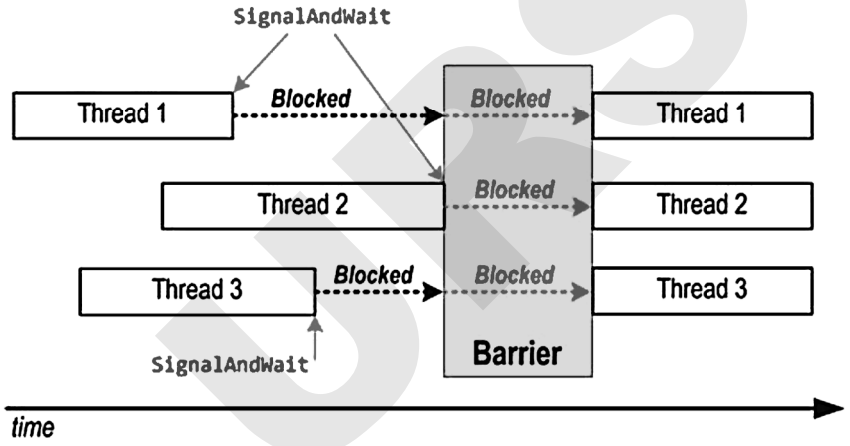


Рис. 2.1.13.8. Синхронизация потоков с использованием барьеров памяти. *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

Слой, содержащий графический интерфейс пользователя, включает в себя диалоговые окна, кнопки, виджеты и др., а результаты симуляции обновляются в соответствующем окне с периодом 1/8 секунды. Для отображения агентов предусмотрено 4 уровня детализации (рис. 2.1.13.9, верхняя часть), а для отслеживания траекторий их движения и сбора статистики реализован специальный функционал (рис. 2.1.13.9, нижняя часть).

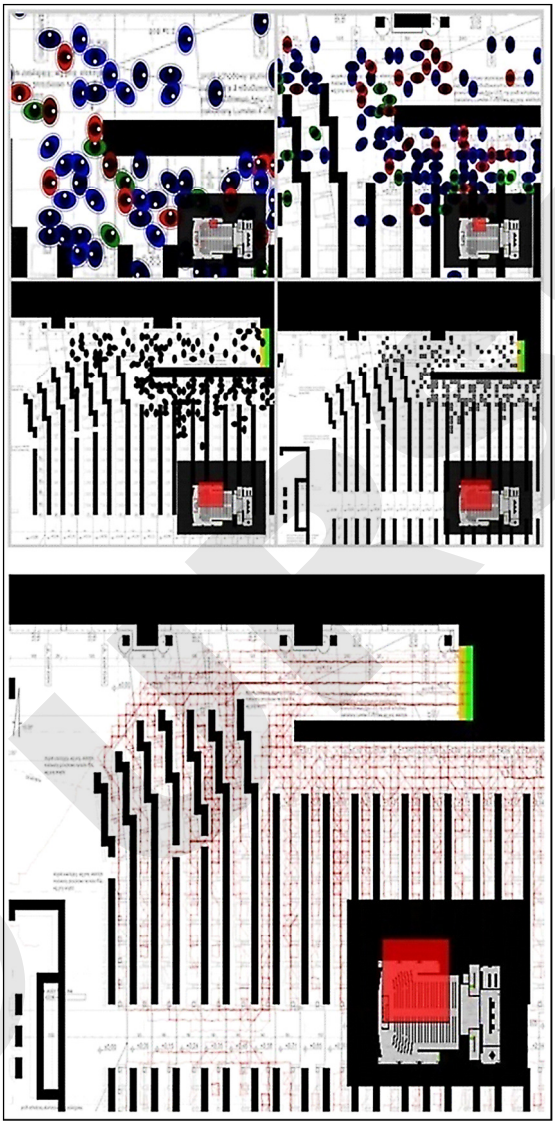


Рис. 2.1.13.9. Отображение агентов с различной степенью детализации (a) и траектория их движения (b). *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

Результаты симуляций

Для тестирования адекватности симулятора сначала были проведены эксперименты по оценке времени эвакуации из лекционной аудитории. Усредненные результаты 50 прогонов показали время эвакуации всего класса за 6,6 с, а эмпирические данные 6,24 с. На рис. 2.1.13.10 приведены полученные значения для 31 участника экспериментов.

Как видно, время эвакуации для всех пассажиров практически одинаковое. Исключение составляют агенты 8, 13 и 18, которые в рамках компьютерной симуляции меняли направление выхода в зависимости от текущей ситуации.

Далее были проведены симуляции для восточной трибуны стадиона «Висла». Общее количество агентов в симуляции — 11 808, время эвакуации для 95 % агентов — 653 с, самая быстрая эвакуация — 3 с, самая медленная — 653 с, средняя скорость перемещения агентов — 0,316 м/с. На рис. 2.1.13.11 приведены дополнительные результаты. Отметим то, что с течением времени средняя скорость снижалась, что связано с заполнением ближайших выходов (рис. 2.1.13.11b). По мнению разработчиков симулятора такой результат указывает на ошибки в проектировании стадиона.

Наблюдение за выходом болельщиков восточной трибуны стадиона «Висла» показало, что результаты моделирования (рис. 2.1.13.5 внизу) повторяют реальный процесс.

Для более крупного стадиона — «Альянц Арена», вмещающего 70 000 зрителей, были проведены симуляции для 58 000 агентов. Усредненное время эвакуации после множества прогонов составило 1117 с, средняя скорость перемещения агентов — 0,726 м/с, самая быстрая эвакуация — 2 с, самая медленная — 1117 с.

На рис. 2.1.13.12 видно, что после первых 300 с от момента начала симуляций средняя скорость агентов стабилизируется на уровне 0,7 м/с и в течение 70 % времени процесс эвакуации осуществляется в стабильном режиме и с довольно высокой скоростью.

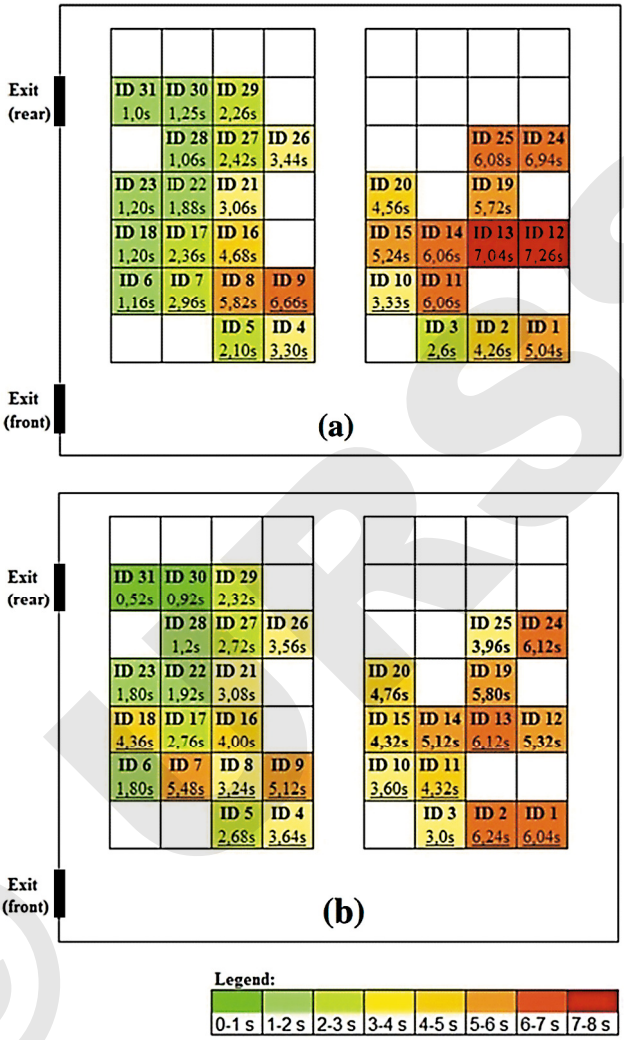


Рис. 2.1.13.10. Первоначальное размещение агентов в аудитории. В верхней части каждой ячейки приведен персональный номер участника, а в нижней — время, затраченное на эвакуацию. В части (a) рисунка — результаты моделирования, в части (b) — эмпирические данные. *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

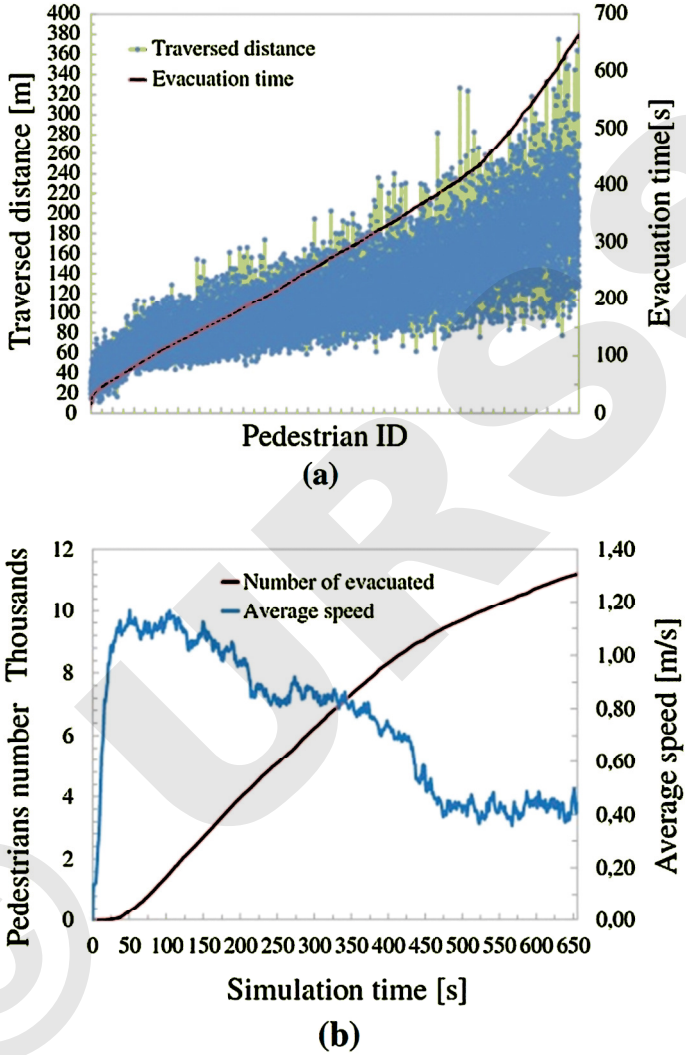


Рис. 2.1.13.11. Результаты эвакуации восточной трибуны стадиона «Висла»: (a) пройденное расстояние и время эвакуации для каждого агента; (b) количество эвакуированных агентов в различные промежутки времени и средняя скорость перемещения. *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

© URSS

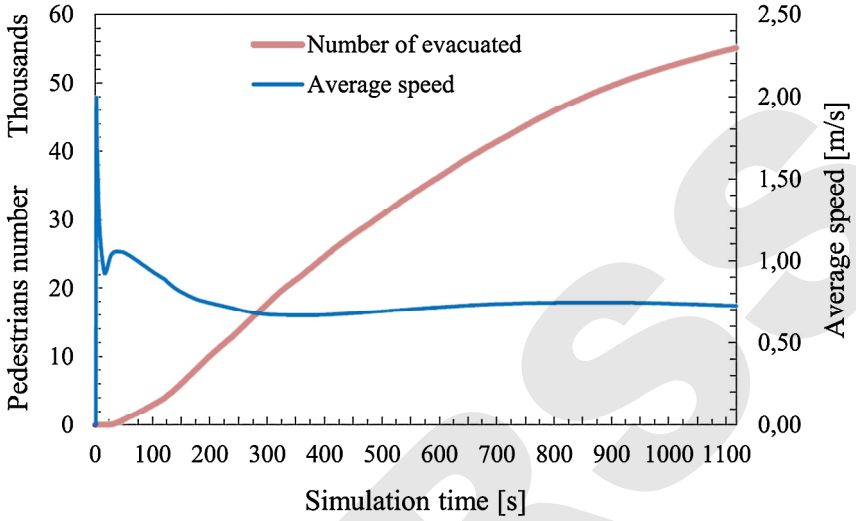
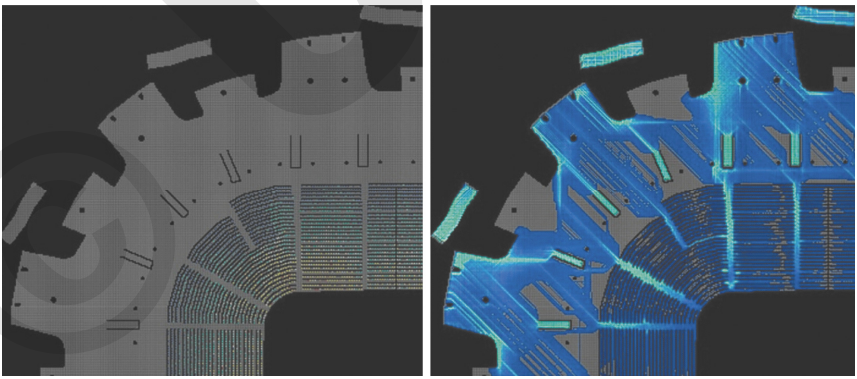


Рис. 2.1.13.12. Количество эвакуированных с течением времени агентов и средняя скорости их перемещения на стадионе «Альянц Арена». *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]



(a) (b)

Рис. 2.1.13.13. Визуализация статистики эвакуации агентов нижней части стадиона «Альянц Арена»: частотная матрица (a), тепловое отображение времени эвакуации (b). *Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior* [17]

На рис. 2.1.13.13 визуализирована статистика нижней секции стадиона. Частотная матрица, составленная на основе анализа движущихся агентов, может быть полезна для обнаружения узких мест или мест с аномально высокой плотностью. В свою очередь тепловое отображение времени эвакуации для каждого агента показывает, что первыми эвакуируются агенты, находящиеся ближе к выходам.

По мнению разработчиков построенный симулятор может быть использован как для выявления узких мест в оцениваемых объектах (стадионах, зданиях и т. д.), так и для интегрирования в системы онлайн-мониторинга территорий массового скопления людей, использующих камеры видеонаблюдения с функцией распознавания лиц, датчики глубины, GPS-трекеры и др. Совместное использование таких систем и описываемых моделей позволит прогнозировать поведение толпы в режиме реального времени и может быть дополнительным аналитическим инструментом поддержки принятий решений для органов, обеспечивающих безопасность во время проведения массовых мероприятий. Более подробно симулятор представлен в работе: «Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior» [17].

2.1.14. D-MASON — масштабируемая распределенная среда построения агент-ориентированных моделей

В статье исследователей из университета Салерно (Италия) рассматривается распределенная версия пакета MASON, разработанная в университете Джорджа Мейсона (США) для реализации программного кода агент-ориентированных моделей в распределенной среде — D-MASON. Большинство разработчиков агентных моделей в первую очередь являются специалистами в определенных предметных областях, а их программистские навыки зачастую ограничиваются знаниями специализированных высокоуровневых пакетов, и в этой связи постоянно растет запрос на системы автоматического распараллеливания программного кода без его пере-

работки, сильно упрощающие построение высокопроизводительных приложений. Ранее мы уже описывали подобного рода фреймворки — RepastHPC, Pandora, Flame и др., а здесь рассмотрим особенности пакета D-MASON, среди которых — эффективное разделение моделируемого пространства (двух и трехмерного), новый механизм согласованности памяти, интеграция с облачными сервисами. Дистрибутив, подробную документацию, учебные пособия и прочие материалы можно скачать по ссылке: https://github.com/isislab-unisa/dmason

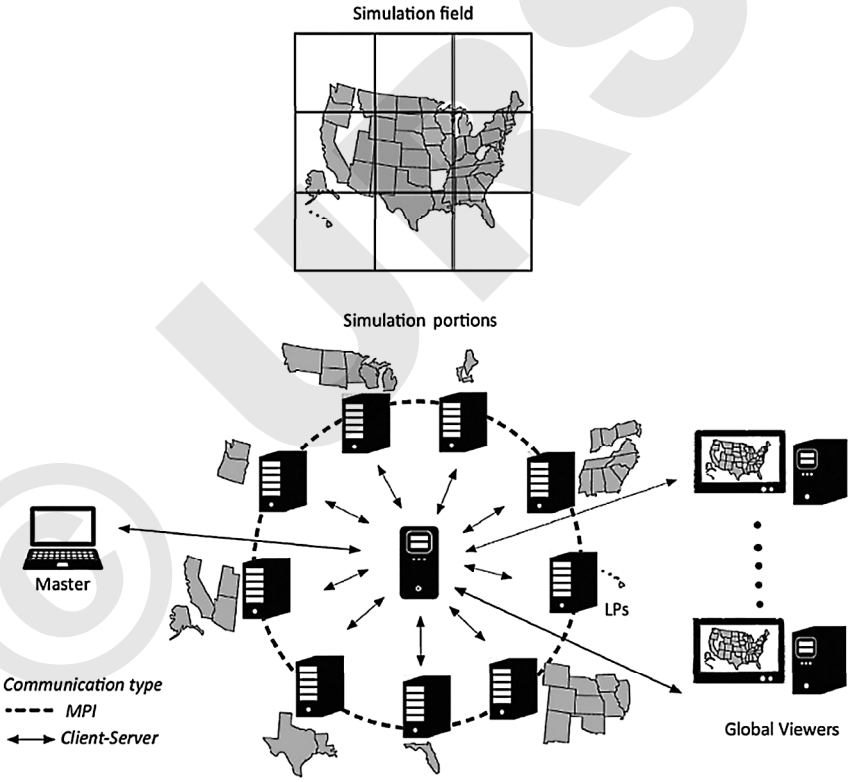


Рис. 2.1.14.1. Архитектура пакета D-MASON. *Distributed MASON: A scalable distributed multi-agent simulation environment* [18]

Пакет D-MASON позволяет реализовывать агент-ориентированные модели в распределенной среде, увеличивая их производительность, при этом обеспечивая обратную совместимость с базовой средой MASON.

Работа D-MASON основана на парадигме master/slave (ведущий — ведомый), при использовании которой главное приложение разделяет моделируемое пространство на части и распределяет рабочую нагрузку по ведомым процессам, каждый из которых задействует один или несколько логических процессоров (Logical Processors, LP) в соответствии с их вычислительными возможностями (рис. 2.1.14.1). Ведущий процесс устанавливает однозначную связь между LP и обслуживаемыми ячейками, при которой каждый LP отвечает за:

- симуляцию агентов, относящихся к соответствующей ячейке;
- обработку события, связанного с миграцией агентов в другие ячейки;
- связь и синхронизацию между соседними ячейками.

Основные задачи, решаемые с помощью D-MASON: распределение выполняемой работы, балансировка нагрузки, связь между процессами, синхронизация и воспроизводимость. Ниже рассмотрим их подробнее.

Распределение работы

Сложной и неоднозначной проблемой при распараллеливании программного кода является декомпозиция общей задачи на отдельные подзадачи и распределение их по всем LP. В случае реализации агент-ориентированной модели, самым простым способом ее решения является равномерное деление агентов по свободным LP. Такой подход, называемый разработчиками как «разделение агентов» (agents partitioning), хотя и обеспечивает сбалансированную нагрузку на процессоры, но создает проблемы при синхронизации в случае высокой степени связности между агентами и необходимости обмена большим количеством сообщений.

Для решения обозначенной проблемы, при разбиении всей совокупности агентов разработчики группируют их в зависимости от

набора связей в рамках определенной фиксированной окрестности, называемой «область интересов» (Area of Interest, AOI) с последующим распределением по LP. Поскольку наиболее интенсивное взаимодействие между агентами происходит в пределах AOI, потери в производительности при синхронизации также снижаются.

Балансировки нагрузки

Поскольку агенты могут менять свою группу общения, а также географическое местоположение, разработчики предусмотрели адаптивную технологию перераспределения агентов по LP, обеспечивающую сбалансированность нагрузки по процессорам и эффективность всего приложения, которая будет рассмотрена ниже.

Связь

В D-MASON связь между LP основана на поведенческом шаблоне проектирования передачи сообщений «Издатель/Подписчик» (Publish/Subscribe, P/S), в котором отправители сообщений напрямую не связаны с подписчиками, равно как и последние с отправителями. В свою очередь сообщения делятся на классы, которые не содержат сведения о подписчиках и издателях.

Первая версия D-MASON была реализована с использованием стандарта Java Message Service (JMS) и сервера Apache ActiveMQ в качестве провайдера JMS. Ниже будет рассмотрено реализованное разработчиками улучшение шаблона с использованием децентрализованной стратегии на базе интерфейса MPI.

Синхронизация

Для корректного выполнения параллельной версии программы, по сравнению с ее последовательным аналогом, каждый LP должен непрерывно собирать информацию обо всех связанных с ним ячейках. В D-MASON выполнение программы осуществляется в виде отдельных шагов, каждый из которых состоит из трех этапов: 1) *симуляция*, 2) *связь* и 3) *синхронизация* (рис. 2.1.14.2). В процессе связи LP отправляет своим соседям информацию об агентах, которые мигрируют к ним, а также об агентах, которые могут попасть в

AOI соседней ячейки. D-MASON использует механизм консервативной синхронизации для последовательной интеграции распределенных симуляций: сначала локально синхронизируется информация, полученная LP с соседних ячеек на шаге $t - 1$, а затем наступает следующий шаг симуляции t.

Процесс локальной синхронизации в виде последовательной диаграммы отображен на рис. 2.1.14.3, где две соседние ячейки ([0–0] и [0–1]) непрерывно обмениваются сообщениями о мигрирующих агентах и агентах, находящихся вблизи границы.

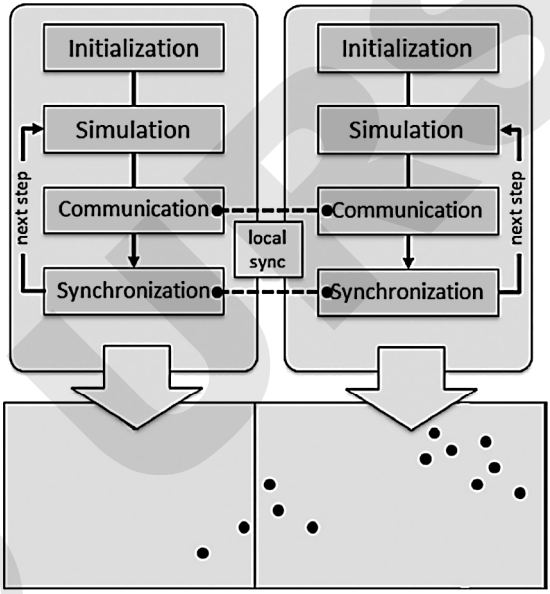


Рис. 2.1.14.2. Синхронизация процессоров LP в пакете D-MASON: два прямоугольника представляют две соседние ячейки, обслуживаемые разными LP; черные точки — агенты модели. После инициализации каждый шаг модельного времени состоит из трех этапов: 1) *симуляция*, когда каждый LP работает самостоятельно, 2) *связь* и 3) *синхронизация* (обмен сообщениями между LP). *Distributed MASON: A scalable distributed multi-agent simulation environment* [18]

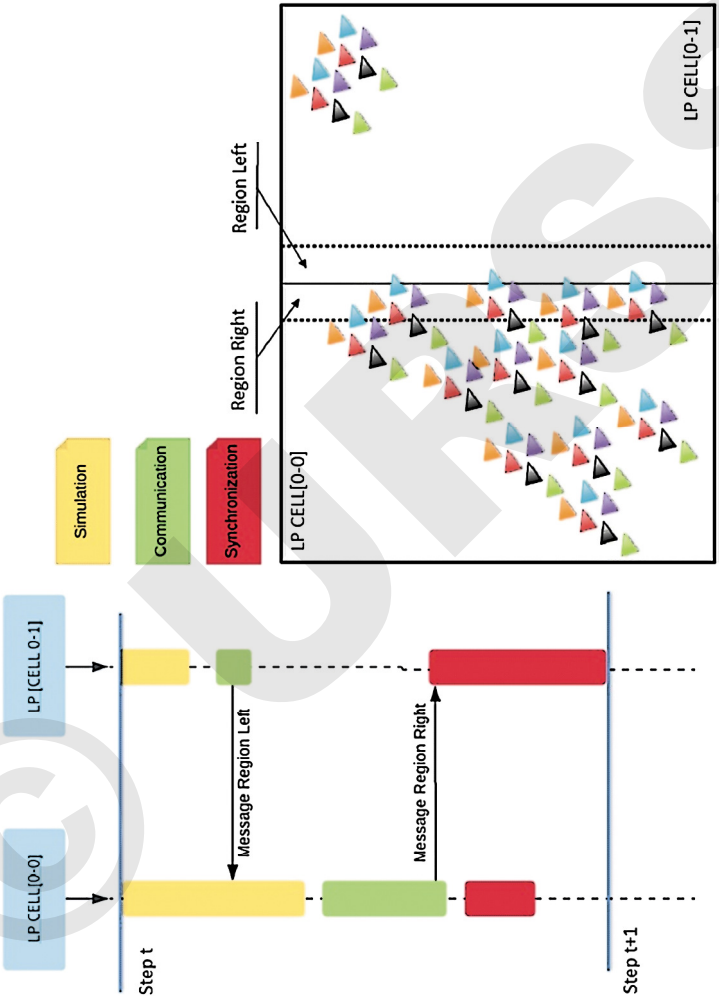


Рис. 2.1.14.3. Диаграмма локальной синхронизации в пакете D-MASON. *Distributed MASON: A scalable distributed multi-agent simulation environment* [18]

Воспроизводимость

Разработчики D-MASON акцентируют особое внимание на том, что порядок, согласно которому вычисляется код агентов, не влияет на результаты симуляций. Таким образом, если реализуемая имитационная модель является полностью детерминированной, то результаты распределенных симуляций с использованием D-MASON идентичны результатам последовательной версии пакета MASON.

В том случае, когда симуляции проводятся с использованием случайных начальных значений некоторых параметров, результаты по-прежнему воспроизводимы при условии, что каждая симуляция выполняется с одинаковым количеством LP. Если же число LP меняется, результаты симуляций также разнятся, но и в этом случае тесты показали, что усредненные оценки по множеству прогонов достаточно стабильны.

Архитектура пакета D-MASON

При разработке архитектуры были установлены следующие требования к пакету D-MASON: (1) эффективность использования аппаратной платформы, (2) возможность разработки с использованием различных парадигм моделирования, (3) удобный для пользователя интерфейс, (4) корректность расчетов. Для удовлетворения этим требованиям D-MASON включает в себя четыре функциональных блока (уровня), отображенных на рис. 2.1.14.4.

1. *Блок управления системой* (*System Management*), предоставляющий удобные средства для регулирования работы симуляций в распределенной системе.
2. *Блок визуализации* (*Visualization*), позволяющий разработчикам создавать продвинутые средства отображения результатов работы агентных моделей.
3. *Блок, обеспечивающий связь* (*Communication*) с другими блоками (уровнями).
4. *Блок распределенного моделирования* (*Distributed Simulation, DS*), обеспечивающий работу системы на множестве вычислительных узлов.

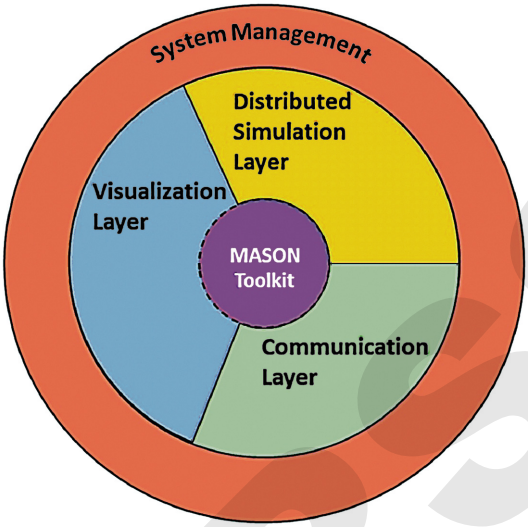


Рис. 2.1.14.4. Архитектура пакета D-MASON. *Distributed MASON: A scalable distributed multi-agent simulation environment* [18]

Блок DS является ядром D-MASON и включает в себя два основных пакета: *Engine* и *Field*, первый из которых состоит из трех объектов:

- *DistributedState* — обеспечивает идентификацию ячеек и миграцию агентов между ними;
- *DistributedMultiSchedule* — отслеживает время симуляции и процесс синхронизации между этапами симуляции;
- *RemoteAgent* — представляет абстракцию распределенного объекта Steppable.

В листинге 2.1.14.1 приведен простой пример реализации объекта DistributedState. При инициализации для каждого LP задаются 100 агентов и в двумерном непрерывном пространстве (DContinuousGrid2D) случайным образом определяются их позиции.

Листинг 2.1.14.1. DSimulation

```
1 public class DSimulation extends DistributedState<Double2D>
2 {
3   public DContinuousGrid2D sim_field;
4   public DSimulation (GeneralParam params, String prefix)
5   {super (params, new DistributedMultiSchedule<Double2D> (), prefix,
            params.getConnectionType ());}
6   public void start ()
7   {super.start ();
8   try {simfield = DContinuousGrid2DFactory.
            createDContinuous2D(10.0/1.5, 200, 200, this, super.AOI,
            TYPE.pos_i, TYPE.pos_j, super.rows, super.columns, MODE,
            ”dfield1”, topicPrefix, true);
9       init_connection ();
10  } catch (DMasonException e) {e.printStackTrace ();}
11  DAgent agent = null;
12  for (int i = 0; i < 100; i++) {
13      agent=new DAgent (this, new Double2D (0, 0),
                  this.random.nextInt ());
14      agent.setPos (sim_field.getAvailableRandomLocation ());
15      sim_field.setObjectLocation (agent, agent.pos);
16      agent.setColor (Color.RED);
17      schedule.scheduleOnce (sim_field);}
18  }
19  public DistributedField2D getField() {return sim_field;}
20  publ i c void addToField (RemotePositionedAgent rm, Double2D loc)
21  {sim_field.setObjectLocation (rm, loc);}
22  public SimState getState() {return this;}
23 }
```

В листинге 2.1.14.2 показан код для очень простого агента, создаваемого с помощью двух объектов: абстрактным классом RemoteDAgent и реализующем агента классом — DAgent. Эти два объекта являются подклассами RemoteAgent и определяют местонахождение агентов в пространстве и их положение в рамках сетевой структуры (без привязки к географическим координатам).

Листинг 2.1.14.2. DAgent

```
1 // абстрактный базовый класс агента
2 public abstract class RemoteDAgent<E> implements Serializable,
        RemotePositionedAgent<E> {
3   public E pos; // местоположение агента
4   public String id; // идентификатор удаленного агента
5   public RemoteDAgent () {}
6   public RemoteDAgent (DistributedState<E> state) {
7       int i = state.nextId ();
8       this.id = state.getType ().toString () + ”–” + i ;
9   }
10  public E getPos () {return pos;}
11  public void s e tPos (E pos ) { this.pos = pos;}
12  public String getId () {return id;}
13  public void setId (String id) { this.id = id;}
14  public boolean equals (Object obj) {// код опущен
15 }
16 // класс агента
17 public class DAgent extends RemoteDAgent<Double2D>{
18  private int val;
19  public DAgent () {} // требуется для сериализации D-MASON
20  public DAgent (String id, Double2D location, Integer val) {
21      this.id = id; this.pos = location; this.val = val;
22  }
23  public Bag getNeighbors (DistributedState<Double2D> sm) {
24      return ((DContinuousGrid2D)sm.getField ()).
            getNeighborsExactlyWithinDistance (pos, 10 , t rue);
25  }
26  public void step (SimState state) {
27      Bag b = getNeighbors ((DistributedState) state);
28      int max=val;
29      for (Object f: b) {
30          DAgent d = (DAgent) f;
31          int dval = d.getVal ();
32          if (max < dval) max = dval;
33      } this.val = max;
34  }
35  public int getVal (DistributedMultiSchedule schedule){ return val;}
36 }
```

В листинге 2.1.14.3 приведен пример кода, реализующий экземпляр брокера сообщений Apache ActiveMQ на локальном компьютере. В тестовом примере выполняется 8 LP.

Листинг 2.1.14.3. DTestLocalMachine

```
1 public class DTestLocalMachine {
2   private static int numSteps = 100, //количество шагов
3   private static int rows = 2; //количество рядов
4   private static int columns = 4; //количество столбцов
5   private static int AOI = 10; //AOI
6   private static int CONNECTION_TYPE=ConnectionType.pureActiveMQ;
7   private static String ip = ”127.0.0.1”;
8   private static String port = ”61616”;
9   private static String topicPrefix = ” toysim”;
10  private static int MODE = DistributedFie1d2D.UNIFORM_PARTITIONING_
    MODE;
11  public static void main (String [] args) {
12      (new ActiveMQStarter ()).startActivemq (); // запускается локальный
        ActiveMQ
13      System.setProperty (”org.apache.activemq.SERIALIZABLE_PACKAGES”, ”*”);
14      class worker extends Thread {
15          private DistributedState <?> ds;
16          public worker (DistributedState <?> ds) {this.ds = ds; ds.start ();}
17          public void run () {
18              int i = 0;
19              while ( i!= numSteps) {
20                  ds. schedule.step(ds); i++;
21              } System.exit (0);
22          }
23      } ArrayList <worker> myWorker = new ArrayList <worker> ();
24      for (int i = 0; i < rows; i++) {
25          for (int j = 0; j < columns; j++) {
26              GeneralParam genParam = new GeneralParam (null, null,
                AOI, rows, columns, null, MODE, CONNECTION_TYPE);
27              genParam.setI (i); genParam.setJ (j);
28              genParam.setIp(ip); genParam.setPort (port);
29              ArrayList <EntryParam <String, Object>> simParams = new
                ArrayList <EntryParam <String, Object >> ();
```

```
30              DSimulation sim = new DSimulation (genParam,
                simParams, topicPrefix) ;
31              worker a = new worker (sim);
32              myWorker.add (a);
33          }
34      } for (worker w: myWorker) w.start ();
35  }
36 }
```

Некоторые новые функции D-MASON

Изначально D-MASON был разработан для использования в слабосвязанной среде с разнородными машинами для их задействования при решении ресурсоемких задач. На данный момент D-MASON может быть развернут в самых различных вычислительных средах — от одного многоядерного компьютера до неограниченного количества машин, доступных в облаке. Помимо этого, в D-MASON реализованы различные рассмотренные ниже механизмы, позволяющие значительно увеличить производительность приложений.

Разделение пространства в D-MASON

D-MASON поддерживает два вида пространств: геометрическое (двух- или трехмерное) и сетевое, в котором агенты распределяются по узлам сети, а отношения между ними описываются ребрами. Для разделения геометрических пространств предлагается два подхода.

- **Равномерное разбиение**, при котором моделируемое пространство делится на ячейки одинакового размера. На рис. 2.1.14.5 приведен упрощенный пример разделения двумерного пространства, представляющего США, на 9 ячеек. Желтая, красная и зеленая зоны — это перекрывающиеся области между ячейками AOI, о которых упоминалось выше. К сожалению, такое разбиение не гарантирует сбалансированного разбиения (например, если предположить, что на рис. 2.1.14.5 серые зоны являются территориями с высокой плотностью агентов, то вычислительная нагрузка на LP будет очень неравномерной).

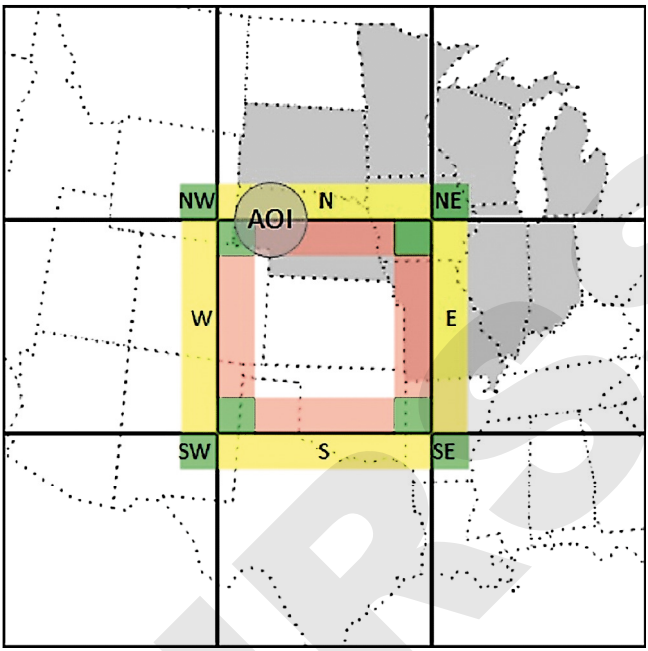


Рис. 2.1.14.5. Равномерное разбиение пространства на 9 ячеек. *Distributed MASON: A scalable distributed multi-agent simulation environment* [18]

- **Неравномерное разбиение**, при котором учитывается пространственное распределение агентов для последующей балансировки нагрузки между LP. Для разделения применяется алгоритм построения дерева квадрантов или квадродерева (Quadtree), в котором каждая его часть делится на четыре сбалансированные области (квадраты, прямоугольники), а точка разделения — центр скопления агентов, взвешенный в соответствии с их вычислительной сложностью. Пример такого разбиения для 9 LP приведен на рис. 2.1.14.6. Как видно, размер ячейки обратно пропорционален ее вычислительной сложности. Ниже будут приведены результаты, показывающие, что такая стратегия улучшает масштабируемость и снижает нагрузку, связанную с синхронизацией.

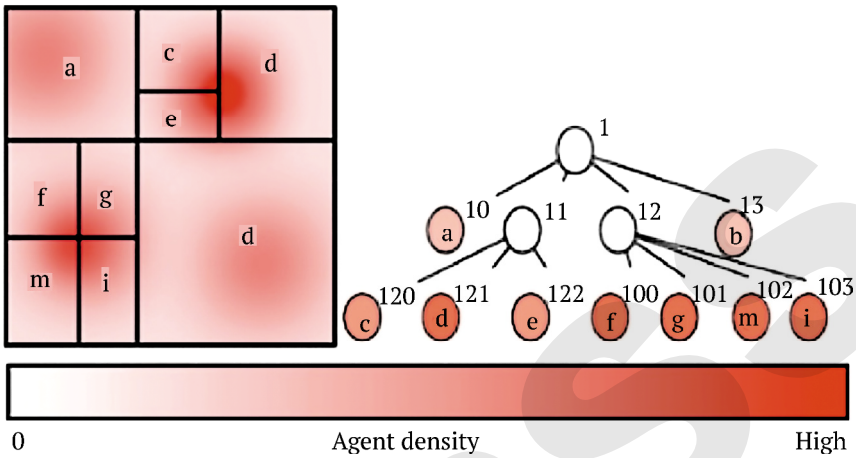


Рис. 2.1.14.6. Неравномерное разбиение пространства с использованием квадродерева для расчетов на 9 LP. Насыщенность цвета определяется плотностью распределения агентов. *Distributed MASON: A scalable distributed multi-agent simulation environment* [18]

Аналогичным образом в D-MASON разбивается трехмерное пространство, но только усложняется процесс синхронизации. Так, для простого примера, приведенного на рис. 2.1.14.5, в случае добавления третьего измерения в прогрессии увеличивается число соседних ячеек. Для 2D-случая их 8, а для 3D — 26.

Разделение распределенной сети

Очень часто изучаемая предметная область, рассматриваемая через призму агентного взаимодействия, может быть структурирована в виде семантической сети. В процессе распараллеливания соответствующий модели, задача состоит в том, чтобы разделить сеть на фиксированный набор компонент таким образом, чтобы они имели приблизительно одинаковый размер, но самое главное — количество соединяющих ее узлы элементов, принадлежащих разным компонентам, должно быть сведено к минимуму (рис. 2.1.14.7). Для решения обозначенной задачи в D-MASON реализован алгоритм графовой декомпозиции METIS.

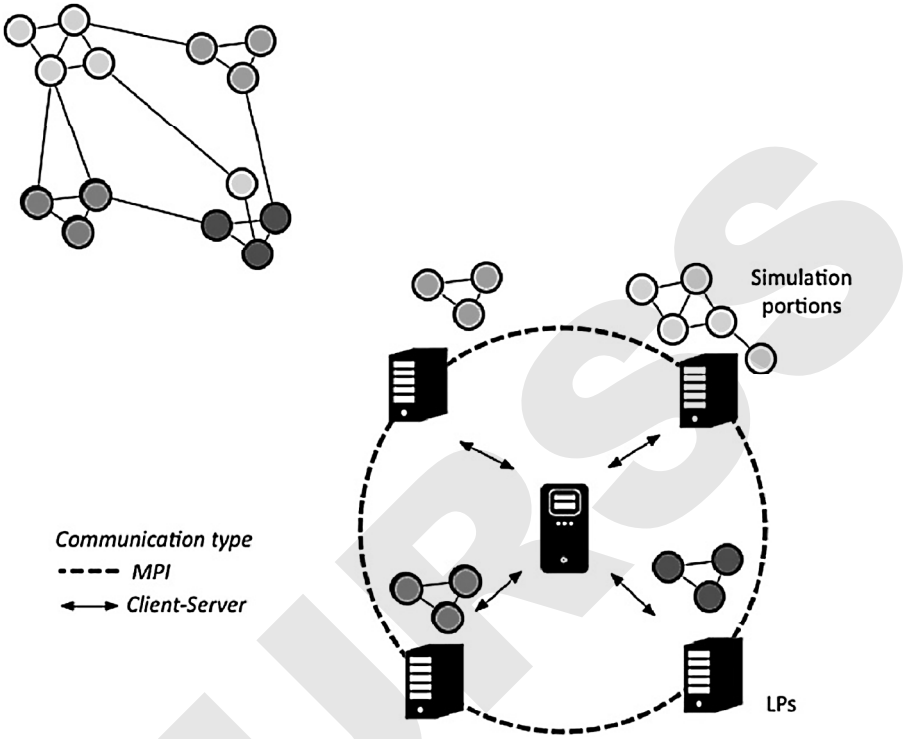


Рис. 2.1.14.7. Разделение сети в пакете D-MASON. *Distributed MASON: A scalable distributed multi-agent simulation environment* [18]

Механизм согласованности памяти

Реализация механизма согласованности памяти довольно проста. Во время очередного шага симуляции при обновлении состояния агента его предыдущее состояние (полученное в конце предыдущего шага) сохраняется в ассоциативном массиве. При выполнении операции чтения сначала происходит проверка о наличии в массиве записи о соответствующем агенте. Если запись имеется, то состояние агента считывается, а в противном случае его текущее состояние остается без изменения. По окончании шага симуляции массив очищается. По сравнению с классической двойной буферизацией, данный подход менее требователен к памяти.

© URSS

Блок, обеспечивающий связь с другими блоками

В D-MASON предусмотрено два вида связи:

- *централизованная* — используется для архитектур общего назначения с применением JMS и сервера Apache ActiveMQ;
- *децентрализованная*, которая разработана для экстремально масштабных вычислений (Extreme-Scale Computing) с использованием программного интерфейса MPI.

Управление системой

Управление D-MASON осуществляется с использованием контейнера сервлетов Jetty, а для пользовательского интерфейса был использован стиль дизайна приложений, разработанный Google. Для пользователя панель управления предоставляет четыре окна со следующими настройками.

1. *Окно мониторинга* для отслеживания доступных вычислительных ресурсов в сети, с информацией о количестве узлов, их загруженности, доступной памяти и т. д., с возможностью выбора конкретных ресурсов для запуска планируемой симуляции.
2. *Окно симуляций* со списком всех выполняемых в текущий момент симуляций, позволяющее управлять ими (останавливать, перезапускать, отменять).
3. *Лог системы*, содержащий информацию о всех выполненных симуляциях, их параметрах, сгенерированных файлах.
4. *Окно параметров*, в котором можно менять конфигурация системы (IP адрес, номер порта и др.).

D-MASON в облаке

В D-MASON реализовано расширение, позволяющее масштабировать агент-ориентированные модели, реализуя среду «Симуляция как услуга» (SIMulation-as-a-Service (SIMaaS)), протестированную на базе облака Amazon Web Services.

© ДМК

Вычислительные эксперименты

В этом разделе будут приведены результаты различных вычислительных экспериментов по оценки масштабируемости D-MASON при реализации классических агент-ориентированных моделей, а также при реализации D-MASON в облачной среде.

Масштабируемость D-MASON для различных агент-ориентированных моделей

В рамках первой серии расчетов оценивалась масштабируемость пакета D-MASON для перечисленных ниже четырех агент-ориентированных моделей, реализованных в 2D-среде.

- *Круг* (*Circle*) — модель, в которой рассматривается набор взаимодействующих частиц (притягивающихся и отталкивающихся) в ограниченном пространстве с определенном радиусом. Через некоторое время система приходит в стабильное состояние.
- *Боиды* (*Boids, Flockers*) — модель, разработанная Крейгом Райнольдсом в 1986 г. в качестве симулятора перемещения стаи птиц, в основе которой лежит простой набор правил: сплоченность (особи стараются держаться ближе к центру масс), разделение (особи избегают столкновений), выравнивание (особи двигаются в направлении соседних особей).
- *Игра «Жизнь»* (*Game of Life*) — клеточный автомат, предложенный Джоном Конвеем в 1970 г. Каждая клетка (особь) может находиться в двух состояниях (быть живой или мертвой), правила перехода клеточного автомата к следующему поколению особей весьма просты и зависят от состояний соседних клеток.
- *Муравьиный алгоритм* (*Ant colony optimization algorithms*) — модель, основанная на поведении муравьев, которые в процессе поиска питания прокладывают тропу между своим гнездом и источником пищи с помощью феромонов.

На рис. 2.1.14.8 приведены полученные результаты для каждой из четырех моделей с использованием 10^6 агентов и 1000 шагов симуляции. Левый график показывает зависимость времени расчетов

от числа используемых LP, а график справа — ускорение. Как видно, игра «Жизнь» и модель «Боиды» демонстрируют очень хорошие результаты: ускорение составляет ≈62 и ≈89 раз соответственно.

Наихудшие показатели у муравьиного алгоритма (≈2), поскольку распределение муравьев по пространству в ходе симуляций постоянно меняется. Помимо расчетов в 2D-среде были проведены эксперименты с моделью боидов, реализованной для трехмерного

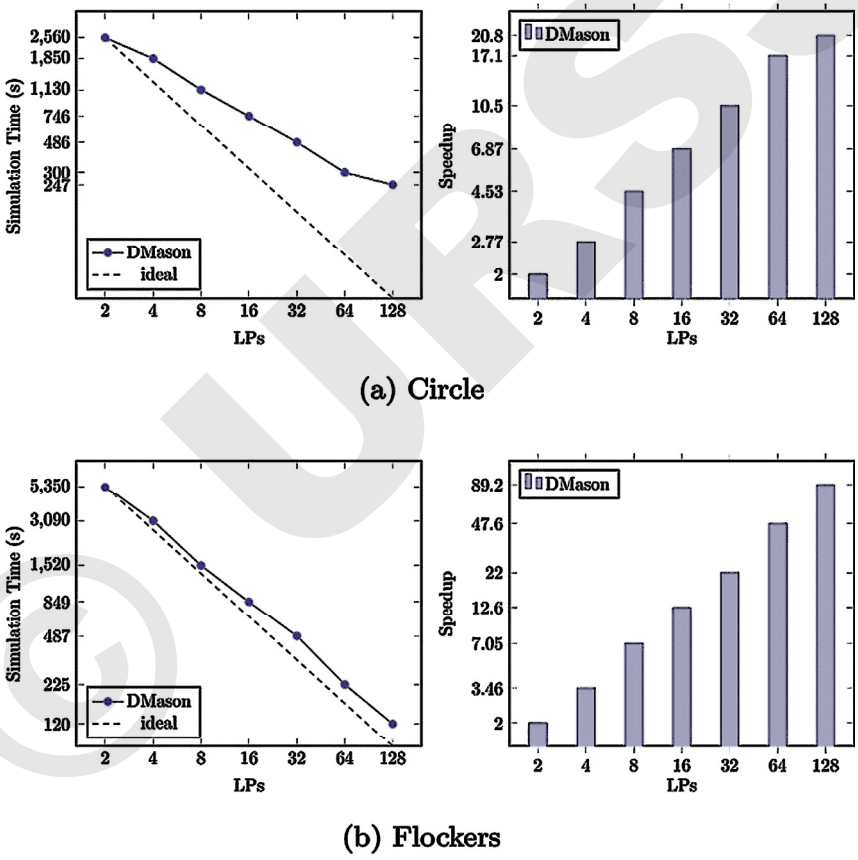
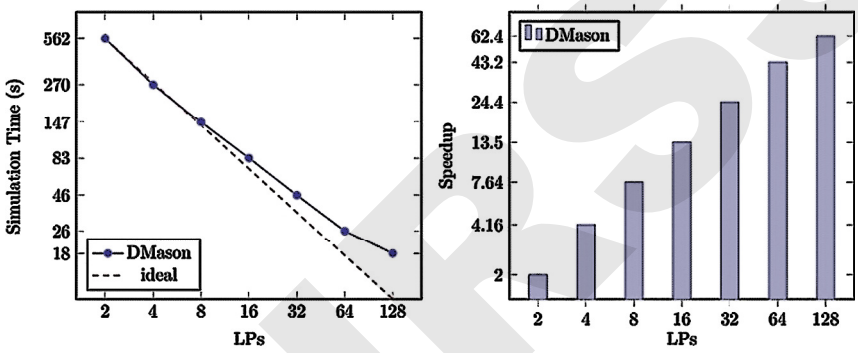


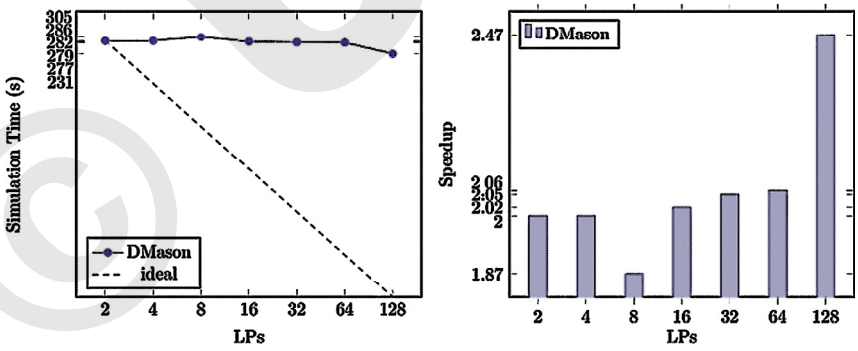
Рис. 2.1.14.8 (a, b). Масштабируемость различных агент-ориентированных 2D-моделей, реализованных в D-MASON, для 10^6 агентов. *Distributed MASON: A scalable distributed multi-agent simulation environment* [18]

пространства размерностью $10^3 \times 10^3 \times 10^3$ с использованием 10^6 агентов.

На рис. 2.1.14.9 приведены результаты симуляций с применением двух видов связи — централизованной и децентрализованной. Как видно, в обоих случаях ускорение очень хорошее (70,7 раз для централизованной и 72,9 раз для децентрализованной связи).



(c) Game-of-Life



(d) Ants

Рис. 2.1.14.8 (c, d)

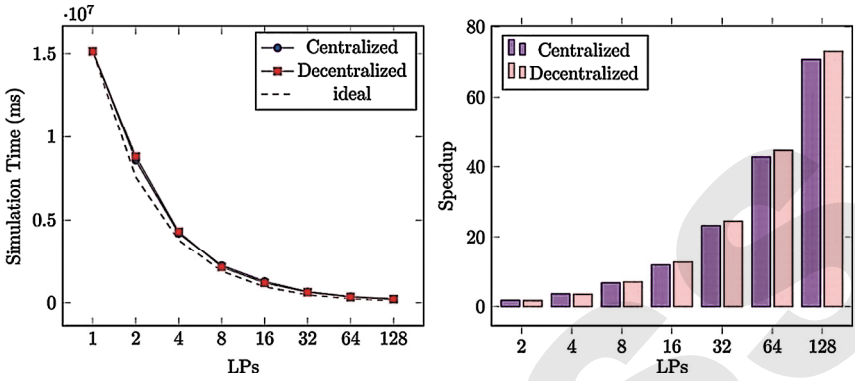


Рис. 2.1.14.9. Масштабируемость 3D-модели в D-MASON для 10^6 агентов (ось абсцисс — число LP, ось ординат — время симуляций в миллисекундах). *Distributed MASON: A scalable distributed multi-agent simulation environment* [18]

D-MASON в облачной среде: оценка производительности и стоимости

В рамках второй серии расчетов оценивалась производительность D-MASON в облаке с использованием модели боидов, реализованной для 2D пространства размерностью 6400 × 6400 с 10^6 агентов. Каждая симуляция выполнялась в течение 15 минут, а в качестве результатов рассматривалось количество выполненных шагов за это время, а также стоимость облачного сервиса. Девять стратегий пространственного разделения (4 × 4, 4 × 8, 6 × 8, 8 × 8, 10 × 8, 12 × 8, 14 × 8, 16 × 8, 12 × 12) соответственно определили 16, 32, 48, 64, 80, 96, 112, 128 и 144 ячеек для эквивалентного количества задействованных LP. Для расчетов был использован веб-сервис, предоставляющий вычислительные мощности в облаке, — Amazon Elastic Compute Cloud (Amazon EC2) с инстансами (instance) c4.4xlarge со следующими характеристиками: процессор Intel Xeon E5–2666 v3 (Haswell) с 16 vCPU и 30 ГБ памяти. Стоимость сервиса на момент проведения расчетов составляла 0,80 долл. в час (по запросу) или 0,14 долл. в час (по свободным ресурсам). Полученные результаты, являющиеся усредненными величинами после 10 симуляций

для каждой из 9 стратегий, приведены в табл. 2.1.14.1. Как видно, D-MASON в облаке хорошо масштабируется, а затраты на проведение симуляций незначительны для всех рассмотренных случаев.

Таблица 2.1.14.1

D-MASON в облачной среде: сравнение производительности и стоимости

Число инстансов	Число LP	Способ разбиения пространства	Количество выполненных шагов симуляции за 15 минут	Общая стоимость сервиса по запросу (долл. в час)	Общая стоимость сервиса по свободным ресурсам (долл. в час)	Общая стоимость сервиса по запросу (долл. за 1000 шагов)
1	16	4 × 4	720	0,80	0,14	0,28
2	32	4 × 8	1251	1,59	0,27	0,32
3	48	6 × 8	1784	2,39	0,41	0,33
4	64	8 × 8	2849	3,18	0,54	0,28
5	80	10 × 8	3817	3,98	0,68	0,26
6	96	12 × 8	4795	4,78	0.81	0,25
7	112	14 × 8	5806	5,57	0,95	0,24
8	128	16 × 8	6645	6,37	1,08	0,24
9	144	12 × 12	7292	7,16	1,22	0,25

Более подробно про эксперименты с D-MASON можно прочитать в работе: «Distributed MASON: A scalable distributed multi-agent simulation environment» [18].

2.1.15. Care HPS — высокопроизводительный инструмент для параллельного и распределенного агентного моделирования

Исследователи из автономного университета Барселоны разработали инструмент для распараллеливания агент-ориентированных моделей — Care HPS (High Performance Simulation),

позволяющий в автоматическом режиме решать задачи распределения выполняемого кода, балансировки вычислительной нагрузки, связи и синхронизации. Ниже кратко описывается предлагаемый фреймворк, а также приводятся результаты экспериментов.

Первая версия Care HPS появилась в 2004 г. как результат развития агент-ориентированной модели для изучения поведения косяков рыб, построенной с использованием технологии MPI и консервативного алгоритма синхронизации. За период 2005–2008 гг. была улучшена масштабируемость приложения, что позволило значительно увеличить число агентов при запуске симуляций. В 2009 г. в Care HPS для выборочной спецификации агентов был реализован механизм нечеткой логики, в 2010–2012 гг. улучшены механизмы распределения выполняемого кода и балансировки вычислительной нагрузки, а в 2013 г. функционал системы был расширен за счет использования массовой синхронной параллели (Bulk Synchronous Parallel). В 2014 г. фреймворк был существенно модернизирован на уровне реализующих агентов классов с целью лучшей масштабируемости в связи с ростом ресурсоемкости задач. Масштабируемость, безусловно, зависит от конкретной задачи, поэтому в Care HPS предлагаются различные механизмы синхронизации.

Архитектура Case HPS

Care HPS поддерживает как интерфейс передачи сообщений MPI, так и технологию OpenMP и содержит в себе несколько компонент, реализованных на языке C++ (рис. 2.1.15.1). Пользователи решают задачу проектирования модели (в том числе с использованием готовых функциональных элементов управления), а всю работу по распределению агентов по процессорам, синхронизации процессов и т. д. выполняет фреймворк.

Далее расскажем про наиболее важные компоненты Care HPS.

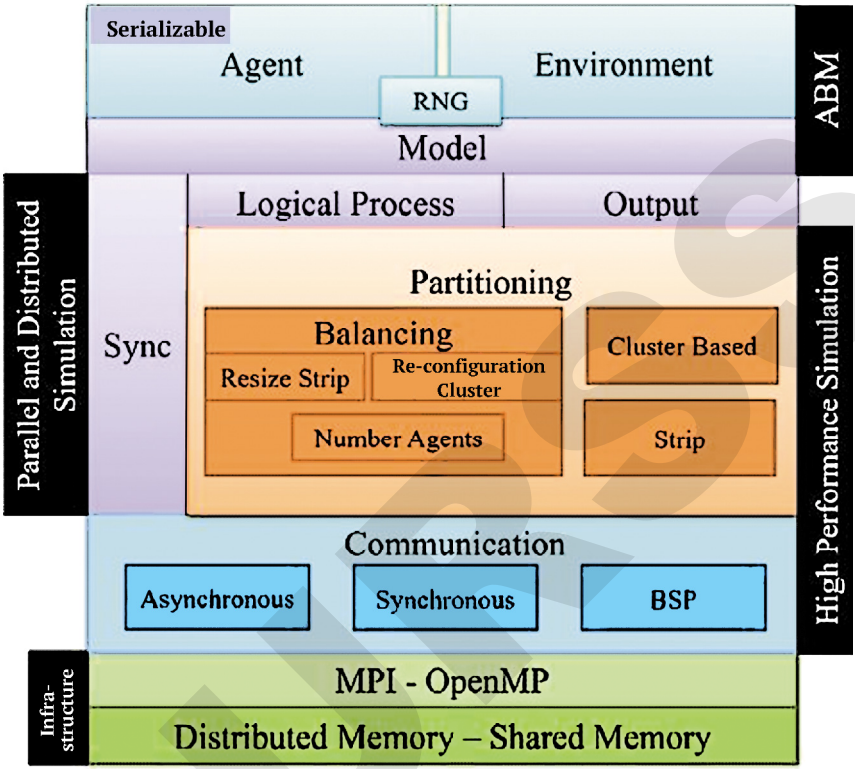


Рис. 2.1.15.1. Архитектура Care HPS. *Care HPS: A high performance simulation tool for parallel and distributed agent-based modeling* [19]

Компоненты агентной модели

Компонент *Agent* имеет встроенный класс *agent*, реализующий поведение агента в двух- и трехмерном пространствах с использованием следующих полей.

1. *position* — определяет положение агента в пространстве.
2. *velocity* — определяет направление агента.
3. *covering radius* — определяет радиус взаимодействия агента с окружающей средой и другими агентами.
4. *member* — используется для сериализации состояния агента (сериализуемый компонент).

5. *ID* — уникальный идентификатор агента.
6. *random_move* — определяет, перемещается ли агент случайно.
7. *alive* — определяет, жив ли агент.

Моделируемая задача может потребовать дополнительных полей, которые должны быть реализованы внутри базового метода *update_agent*, выполняемого каждую единицу времени с момента начала симуляции и до ее завершения.

Важным функционалом класса *agent* является возможность сериализации состояния агента, осуществляемой с использованием компонента *member*. Это необходимо, поскольку агенты могут мигрировать из одного процесса в другой, с выполнением соответствующих операций удаления и создания, но при этом состояния агентов должны сохраняться.

Компонент *Environment* позволяет моделировать окружающее агентов пространство внутри модели, а соответствующий программный класс, являющийся контейнером, содержит методы для взаимодействия с агентами модели.

Часто при запуске симуляций требуется сформировать уникальный набор начальных значений для агентов модели. С использованием компонента *Random number generation* можно определить их посредством нормального, экспоненциального, гамма и равномерного распределений.

Последний компонент *Model* позволяет производить общую настройку симуляции.

Компоненты, отвечающие за параллельные вычисления и распределение программного кода

На этом уровне присутствуют следующие компоненты:

- *Logical process* — создает распределенные процессы, связанные посредством передачи сообщений, а также контролирует ход их выполнения;
- *Sync* — осуществляет синхронизацию;
- *Partitioning* — определяет наиболее эффективный механизм распределения агентов по процессам и реализует его;

- *Balancing* — осуществляет балансировку вычислительной нагрузки (назначение агентов для конкретных ядер, изменение размера разделов и т. д.).

Разработчики Care HPS сравнили свой продукт с некоторыми наиболее известными аналогами — Flame, D-Mason, Pandora и Repast HPC. Каждое из этих программных средств имеет сильные и слабые стороны, связанные с первоначальными целями их создания.

В Care HPS агенты реализуются путем расширения базового класса *Agent*, в пакете Flame посредством настройки X-машины, в D-Mason с помощью расширения абстрактного класса *RemoteAgent*, также позволяющим сериализовать агентов. В Pandora используется общий базовый класс, посредством которого моделируется любая сущность, а в RepastHPC пользователь должен реализовать функционал агентов с помощью методов базового класса и более того, каждый агент должен реализовать интерфейс *Agent*.

Для синхронизации в Care HPS используется метод обратного вызова, во Flame — «доска сообщений», гарантирующая, что все агенты будут видеть один и тот же набор сообщений, в D-Mason применяется пошаговая локальная синхронизация, в Pandora — система планирования, а в RepastHPC — дискретный планировщик событий.

Care HPS предлагает три стратегии распределения агентов: кластерное, чистое MPI и гибридное, а для экспертов в области высокопроизводительных вычислений возможна реализация других алгоритмов. Во Flame применяется алгоритм распределения нагрузки с использованием перебора и упорядочения агентов по круговому циклу (Round Robin Partitioning). В D-MASON главное приложение разделяет моделируемое пространство на части и распределяет рабочую нагрузку по ведомым процессам, каждый из которых задействует один или несколько логических процессоров в соответствии с их вычислительными возможностями. В Pandora реализовано пространственное разделение, при котором каждый

вычислительный узел получает часть моделируемой среды и агентов, расположенных в его границах. Затем каждая часть делится на четыре раздела, обрабатываемых последовательно. RepastHPC реализует разбиение с помощью концепции проекции, в рамках которой пользователь может выбрать один из трех ее типов: решетка, непрерывное пространство и сеть.

Связь между процессами в Care HPS реализуется в следующие моменты: 1) миграция агентов между процессами; 2) синхронизация; 3) инициализация модели; 4) балансировка нагрузки. В свою очередь, во Flame связь осуществляется через «доску сообщений», а возможности прямого обмена сообщениями между агентами не существует. В D-MASON связь между процессами основана на поведенческом шаблоне проектирования передачи сообщений «Издатель/Подписчик», в котором отправители сообщений напрямую не связаны с подписчиками, равно как и последние с отправителями. В свою очередь сообщения делятся на классы, которые не содержат сведения о подписчиках и издателях. Pandora использует OpenMP и MPI таким образом, что локальные взаимодействия осуществляются посредством интерфейса OpenMP, а на каждом шаге соседние узлы обмениваются информацией посредством директив MPI.

Приведем два самых простых примера использования Care HPS: 1) агент-ориентированная модель перемещения рыб с препятствиями; 2) расширение возможностей пакета путем использования новой стратегии распределения нагрузки.

Движение рыбы в модели зависит от ее текущего положения и наличия препятствий, а также от поведения соседних рыб. Код в листинге 2.1.15.1 проверяет, происходит ли столкновение, и если да, то агент отталкивается и меняет направление движения.

Листинг 2.1.15.1. Проверка наличия препятствий

```
void fish :: check_environment (void) {
vector <object*> obj_env = ENV -> getObjects ();
```

```
for (vector <object* > :: iterator ob = obj_env.begin ();
    ob!=obj_env.end ();
    ob++)
if ((*ob) -> check_collision (this -> get_position (),
                              this -> get_velocity (),
                              MAXIMUM_VISION_RANGE))
    this -> repulsion (*this);
}
```

Рассмотрим также пример разработки новой стратегии распределения вычислительной нагрузки. На первом этапе создается начальное количество потоков следующим образом: threads = number_agent/total_number_agents · number_cores, где number_agent — количество агентов внутри раздела, total_number_agents — общее количество моделируемых агентов, а number_cores — количество процессов MPI, поделенное на 4 (выбор числа 4 связан с архитектурой используемых ЦП).

На втором этапе потоки создаются динамически, и пример такого автоматического распараллеливания приведен в листинге 2.1.15.2.

Листинг 2.1.15.2. Реализация стратегии распределения нагрузки

```
void partitionig_strip_hybrid :: execute () {
    long thread = number_agent / total_number_agents*number_cores ;
    if (abs (num_thread) >=1){
        if (abs (thread )==1)
            omp_set_num_threads (2);
        else if ((abs(thread) >= 2) and (abs(thread) <= 8))
            omp_set_num_threads (abs (thread));
        else if (abs(thread) > 8)
            omp_set_num_threads (8);
        #pragma omp parallel default(none) firstprivate(i)
            shared(_bucket)
```

```
        {
            unsigned long j ;
            #pragma omp for private (j) schedule (dynamic) nowait
            // для каждого агента выполняется следующее
            for (j =0; j < _bucket -> size (); j++)
                _bucket->at (j) -> update_agent();
        }
    }
    else {
    // выполнить цикл
    }
};
```

Результаты вычислительных экспериментов

Ниже описываются результаты четырех экспериментов, проведенных с тремя агент-ориентированными моделями: 1) перемещения рыб, 2) муравьиной колонии и 3) покупателей.

Первый эксперимент

В реализованной модели у агента есть список необходимых к покупке товаров, которые распределены по множеству магазинов. Каждый из магазинов продает только один вид товара, а цель агента — купить весь перечень из списка.

В рамках расчетов было реализовано три сценария: 1) агенты перемещаются случайным образом, чтобы найти магазин для покупки товара из списка, 2) агенты могут видеть магазины в пределах определенного диапазона, 3) некоторые агенты являются «умными», то есть вычисляют оптимальный маршрут и, более того, агенты модели обмениваются информацией об известных им магазинах.

В табл. 2.1.15.1 приведены результаты симуляций в сравнении с версией модели с идентичными параметрами, но реализованной в пакете Netlogo. Как видно, применение алгоритмов Care HPS в случае интенсивного обмена информацией более эффективно.

© URSS

Таблица 2.1.15.1

Сравнение производительности двух реализаций одной модели для трех сценариев (цифры означают количество шагов симуляций, необходимых для достижения поставленной цели)

	Сценарий 1	**Сценарий 2**	**Сценарий 3**
Netlogo	14 310	6983	2000
Care HPS	19 082	1361	348

Второй эксперимент

В этом эксперименте имитировалась самоорганизация косяка рыб, возникающая в результате локальных взаимодействий отдельных особей, поведение которых зависит от действий соседей. Ни рис. 2.1.15.2 показан результат симуляций для 8192 агентов-рыб с использованием двух процессоров. Трехмерное пространство представляет собой море, а желтая решетка — препятствие, изменившее траекторию движения косяка, который через последовательность шагов проплыл мимо преграды.

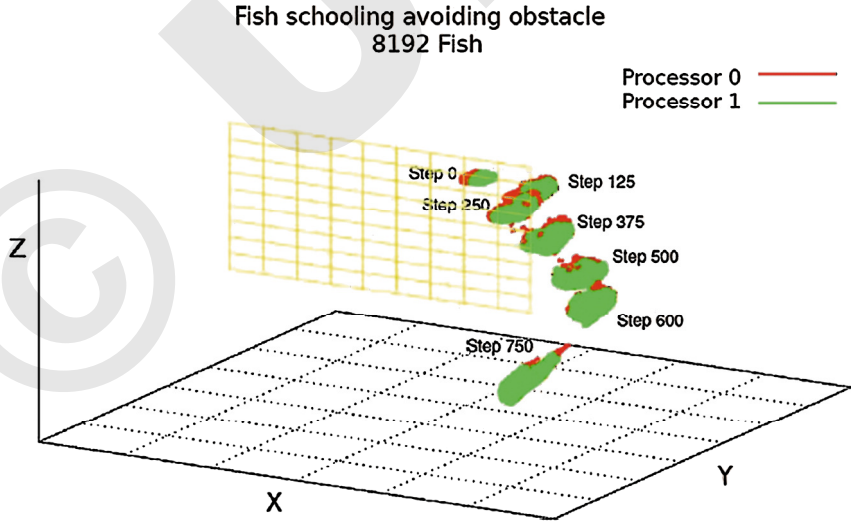


Рис. 2.1.15.2. Поведение косяка рыб, избегающих столкновения. *Care HPS: A high performance simulation tool for parallel and distributed agent-based modeling* [19]

Третий эксперимент

С использованием модели муравьиной колонии были проведены расчеты для 1000, 2500, 5000 и 10 000 агентов с применением двух стратегий разделения вычислительной нагрузки: чистого разделения с использованием директив MPI и гибридного. На рис. 2.1.15.3 приведены результаты для обеих стратегий, из которых следует, что гибридный механизм существенно эффективнее.

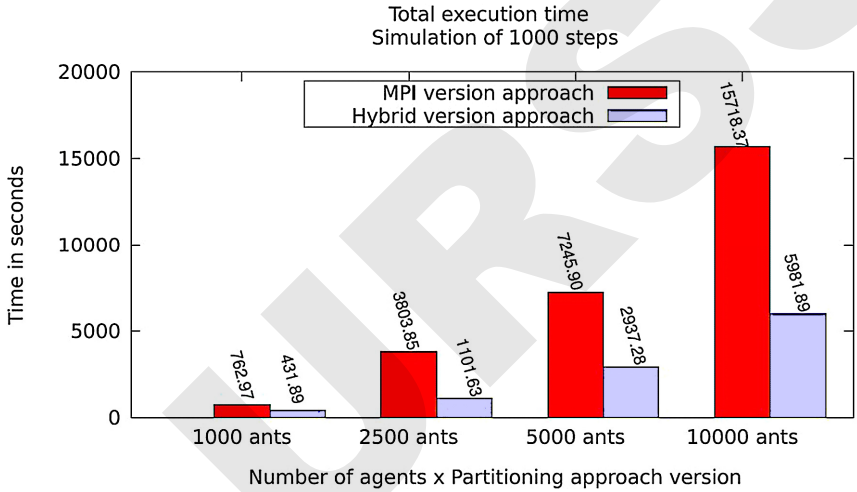


Рис. 2.1.15.3. Общее время выполнения алгоритма для чистого MPI и гибридного разделения нагрузки. *Care HPS: A high performance simulation tool for parallel and distributed agent-based modeling* [19]

Четвертый эксперимент

В рамках последнего эксперимента оценивалась масштабируемость агент-ориентированной модели покупателей, реализующей третий сценарий первого эксперимента.

Ни рис. 2.1.15.4 и 2.1.15.5 приведены результаты для 10 000 и 50 000 агентов соответственно, которые осуществляют 50 000 шагов симуляции и должны приобрести 10 товаров. Как видно, масштабируемость приложения весьма хорошая.

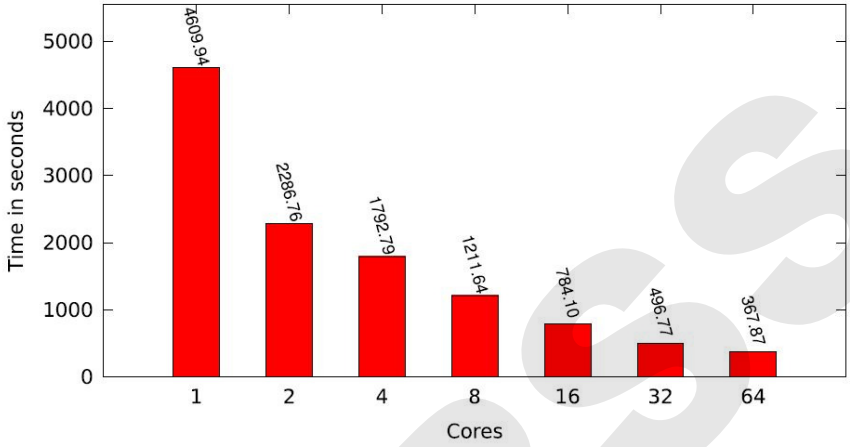


Рис. 2.1.15.4. Масштабируемость модели с 10 000 агентами, выполнившими 50 000 шагов. *Care HPS: A high performance simulation tool for parallel and distributed agent-based modeling* [19]

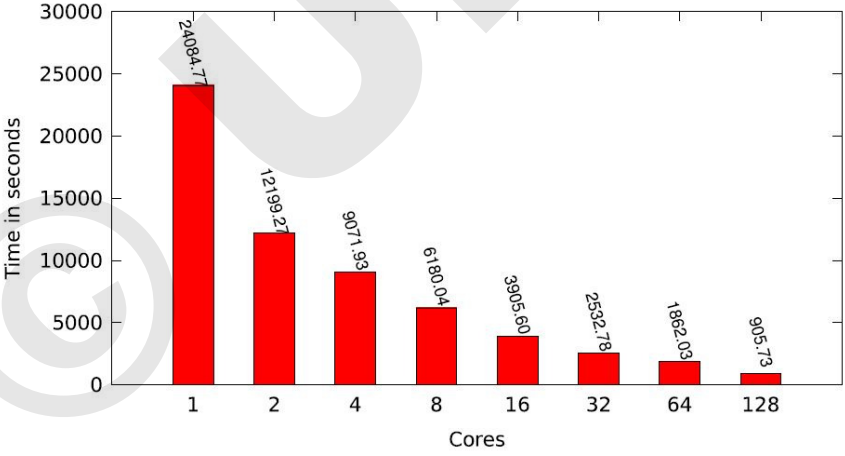


Рис. 2.1.15.5. Масштабируемость модели с 50 000 агентами, выполнившими 50 000 шагов. *Care HPS: A high performance simulation tool for parallel and distributed agent-based modeling* [19]

Далее тестировалась способность Care HPS эффективно работать с моделями большой размерности. На рис. 2.1.15.6 и 2.1.15.7 видно, что для 200 000 и 250 000 агентов масштабируемость также приемлемая.

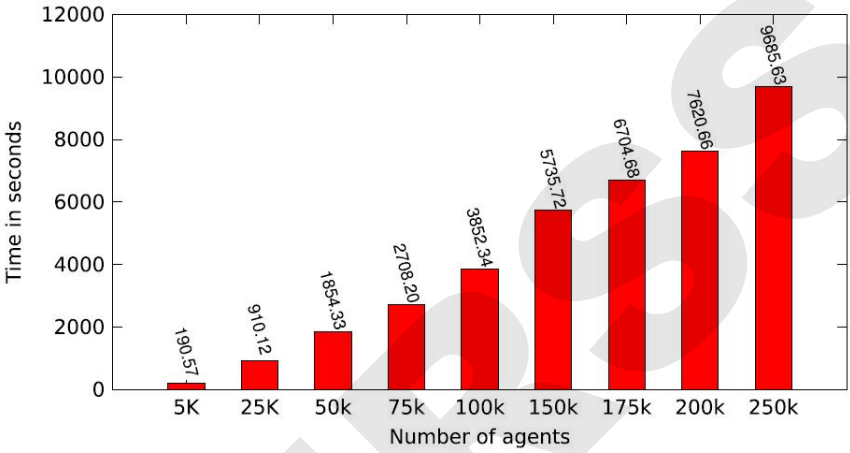


Рис. 2.1.15.6. Масштабируемость модели для 50 000 шагов, выполненных с использованием 64 ядер. *Care HPS: A high performance simulation tool for parallel and distributed agent-based modeling* [19]

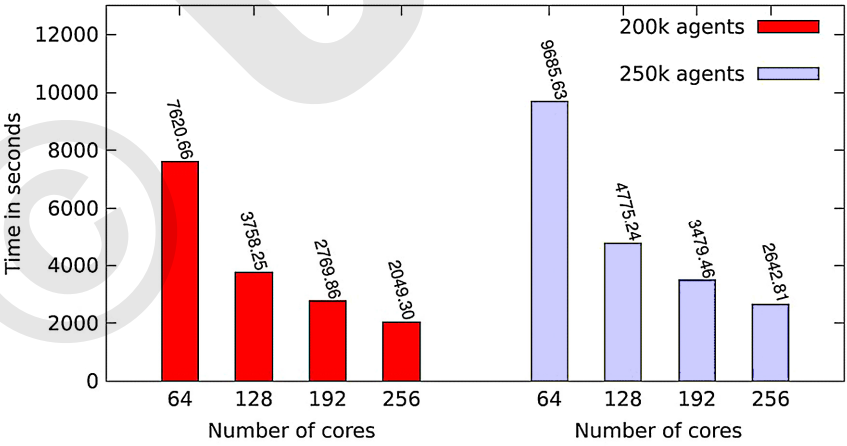


Рис. 2.1.15.7. Масштабируемость модели для 200 000, 250 000 агентов и 50 000 шагов, выполненных с использованием 64, 128, 192 и 256 ядер. *Care HPS: A high performance simulation tool for parallel and distributed agent-based modeling* [19]

В настоящее время разработчики Care HPS используют этот фреймворк в проекте, направленном на прогнозирование распространения лихорадки денге. Более подробно про Care HPS можно прочитать в статье «Care HPS: A high performance simulation tool for parallel and distributed agent-based modeling» [19].

Далее мы переходим к описанию наиболее интересных международных проектов, в рамках которых разрабатываются высокопроизводительные агентные модели.

2.2. Построение цифровых агент-ориентированных моделей и их применение в социально-экономических исследованиях

Агент-ориентированное моделирование прочно вошло в экономическую науку широким фронтом (на всех уровнях — от микро до макро). Для микроуровня предпринимаются большие усилия по разработке методологических принципов построения агентов следующего поколения — более реалистичных, когнитивных, ограниченно рациональных (в противовес традиционным агентам классической экономической теории). Эти принципы частично основаны на некоторых результатах исследований в поведенческой экономике — игнорировании базовой вероятности (base rate neglect), теории перспектив (prospect theory), эффектах фрейминга (framing effects), эффектах привязки (anchoring), конформности (conformity effects), склонности к подтверждению своей точки зрения (confirmation bias), нарушению некоторых принципов предпочтения (как в игре «Ультиматум») и др. Кроме того, нейробиологи, занимающиеся нейроэкономикой, выявляют причины подобных отклонений от рациональности [20], а разработчики агент-ориентированных моделей затем встраивают соответствующие когнитивные механизмы в поведенческие функции агентов следующего поколения [9].

В этом разделе мы рассмотрим наиболее известные международные проекты по моделированию социально-экономических систем с использованием агентных моделей и суперкомпьютерных технологий.

2.2.1. EURACE — крупномасштабная модель европейской экономики

В сентябре 2006 г. стартовал проект по разработке крупномасштабной агентной модели европейской экономики — EURACE, то есть «Europe ACE» (Agent-based Computational Economics), с большим количеством автономных агентов, взаимодействующих в рамках социально-экономической системы. В проект вовлечены экономисты и программисты из восьми научно-исследовательских центров Италии, Франции, Германии, Великобритании и Турции, а также консультант из Колумбийского университета США — нобелевский лауреат Джозеф Стиглиц [21.]

Чтобы преодолеть ограничения широко распространенных моделей, рассматривающих агрегированных агентов, а также предполагающих их рациональное поведение и состояние равновесия, в основу исследования была положена методология ACE (Agent-based Computational Economics).

Для модели используется географическая информационная система, охватывающая широкий перечень объектов — предприятия, магазины, школы, транспортные сети и т. д.

Как отмечают разработчики, практически все существующие агент-ориентированные модели рассматривают либо отдельную отрасль, либо относительно небольшой географический район и, соответственно, небольшую популяцию агентов, а в EURACE представлен весь Европейский союз, так что по масштабам и сложности эта модель является уникальной, а ее численное разрешение требует использования суперкомпьютеров, а также специального программного обеспечения.

Для наполнения модели статистической информацией использовались данные (в виде геоинформационных карт) статистиче-

ской службы Европейского союза уровня NUTS-2, представляющие сведения о 268 регионах 27 стран.

Отметим, что NUTS (фр. «Nomenclature des unités territoriales statistiques» — «Номенклатура территориальных единиц для целей статистики»), представляющая собой стандарт территориального деления стран для статистических целей, разработанный Европейским союзом и охватывающий входящие в него страны. Существуют NUTS-единицы трех уровней, при этом второй уровень (NUTS-2) соответствует административным округам в Германии, графствам в Великобритании и т. д.

В модели три типа агентов: домашние хозяйства (около 10^7), предприятия (около 10^5) и банки (около 10^2). Все они имеют географическую привязку, а также связаны друг с другом посредством социальных сетей, деловых отношений и т. д.

EURACE реализована с помощью гибкой масштабируемой среды для моделирования агентных моделей — FLAME (Flexible Large-scale Agent Modeling Environment), разработанной Симоном Коакли

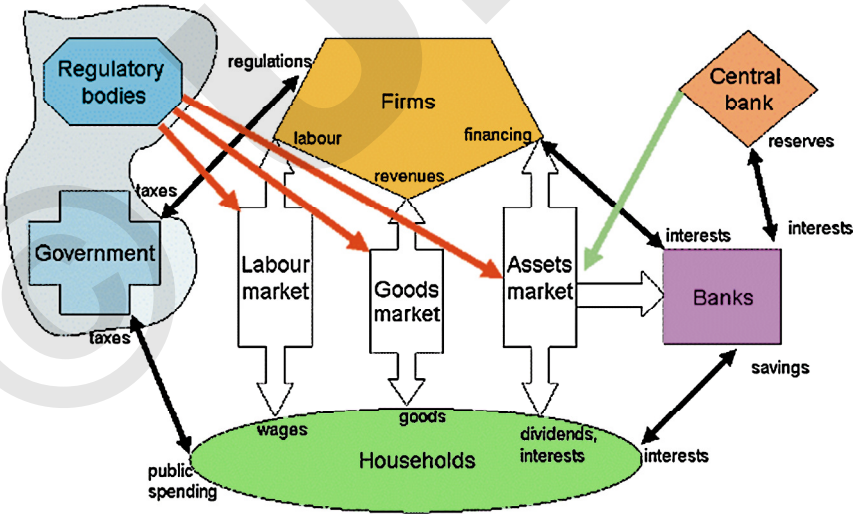


Рис. 2.2.1.1. Концептуальная схема работы модели.
EURACE: A Massively Parallel Agent-Based Model of the European Economy [21]

и Майклом Холкомбом первоначально для моделирования роста клеток, выращиваемых в различных условиях (более подробно см.: www.flame.ac.uk). Используемый в FLAME подход основан на так называемых X-машинах, напоминающих конечные автоматы, но отличающихся от них тем, что у каждой X-машины есть набор данных (своего рода память), а также тем, что переходы между состояниями являются функциями не только состояния, но и набора данных памяти.

Итак, каждый агент в системе FLAME представлен X-машиной, причем предусмотрено общение между агентами посредством передачи сообщений. При работе в параллельном режиме на суперкомпьютере обмен сообщениями между агентами требует больших вычислительных затрат, в связи с чем агенты изначально были распределены по процессорам в соответствии с их географическим положением. Тем самым разработчики программы минимизировали вычислительную нагрузку, исходя из предположения, что в основном общение между агентами происходит в рамках небольшой социальной группы, локализованной в определенной местности. Таким образом, весь модельный ландшафт был поделен на небольшие территории и распределен между узлами суперкомпьютера.

С помощью разработанной модели был проведен ряд экспериментов с целью исследования рынка труда. Не рассматривая детально полученные числовые результаты, отметим, что, по мнению авторов, основной вывод исследования заключается в том, что макропоказатели двух регионов со схожими условиями (ресурсы, развитие экономики и т. д.) в течение продолжительного периода (10 лет и более) могут значительно разойтись за счет первоначальной неоднородности агентов.

Результаты расчетов с базовым модулем EURACE также позволили оценить степень влияния механизма количественного смягчения на длительность экономического спада и эффективность использования этого метода кредитно-денежной политики в сочетании с налогово-бюджетными мерами [22, 23].

2.2.2. Система мониторинга состояния планеты, реализуемая на суперкомпьютерах

FuturICT — беспрецедентный междисциплинарный проект по моделированию технологической, социальной и экономической систем мира — стартовал в 2012 г. и вовлек ученых практических из всех развитых стран. Срок его реализации — 10 лет, а начальное финансирование — 1 млрд евро. Руководители проекта делают особый акцент на использовании самых передовых информационных технологий.

Финансовый кризис, международные конфликты, терроризм, распространение заболеваний и киберпреступность, а также демографические, технологические и экологические изменения свидетельствуют о том, что человечество сталкивается с серьезными проблемами, которые только нарастают, а перспективы туманны и спрос на научные исследования в этой области явно не удовлетворяется.

В силу того, что технологические, социальные и экономические системы становятся все более сложными, используемые в повседневной практике инструменты не позволяют получать реалистичные прогнозы и проводить мониторинг состояния этих систем.

В свою очередь выявление скрытых законов и процессов, лежащих в основе функционирования сложных систем, является одной из наиболее актуальных научных проблем XXI в.

Использование новейших достижений в области естественных наук (математики, физики, химии и др.) и общественных наук (экономики, социологии, психологии, истории и др.) позволит разработать систему мониторинга состояния глобальной социально-экономической системы мира, позволяющую прогнозировать:

- развитие социально-экономических систем (в разрезе стран);
- общественные настроения;
- миграционные процессы;
- демографическую ситуацию;
- состояние окружающей среды

  и многое другое.

По мнению одного из руководителей проекта — Дирка Хелбинга (Dirk Helbing), несмотря на актуальность разработки таких многоуровневых систем и существование некоторых их компонент, комплексного продукта до сих пор нет по причине институциональных барьеров и нехватки ресурсов. В этой связи FuturICT обещает стать первым в своем роде (рис. 2.2.2.1).

FuturICT организован как сеть национальных центров, каждый из которых представлен несколькими научными учреждениями в рамках одной страны. Помимо такой сети, в проекте есть еще проблемно-ориентированная сеть, включающая головные центры нескольких стран и направленная на решение отдельных научных направлений (об этом будет сказано ниже). Таким образом, FuturICT интегрирует организации как на институциональном, так и на проблемном уровнях (рис. 2.2.2.2).

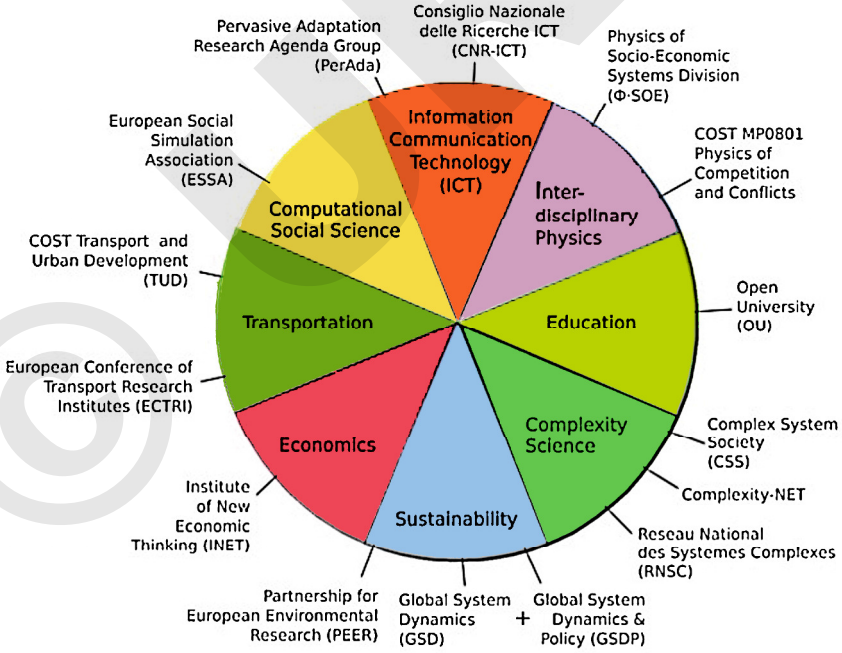


Рис. 2.2.2.1. Интеграция различных научных дисциплин в рамках проекта FuturICT. *https://futurict2.eu*

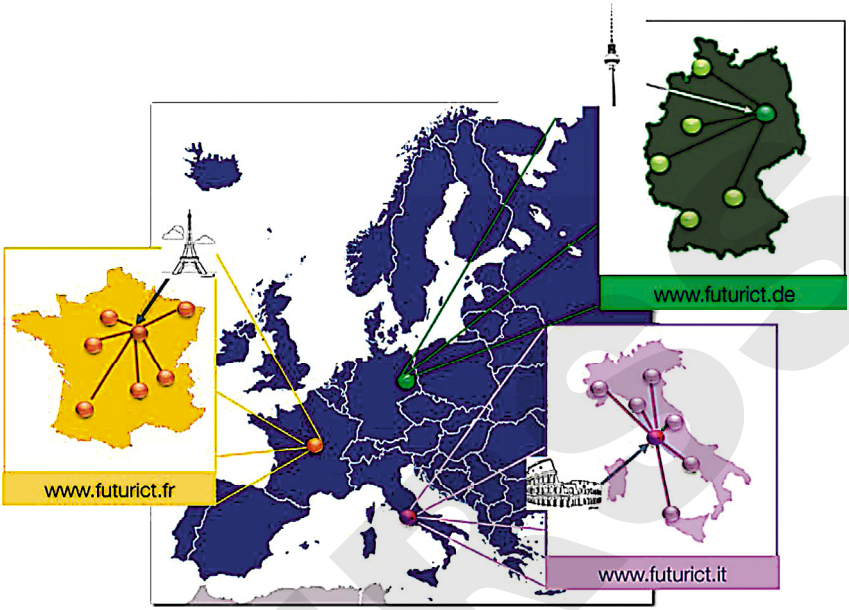


Рис. 2.2.2.2. Организация взаимодействия в FuturICT. *https://futurict2.eu*

Каждое научное сообщество на уровне страны имеет определенную степень автономии, но вместе с тем и набор обязательств (рис. 2.2.2.3).

Разработчики предлагают платформу FuturICT, включающую в себя три составляющие, названные следующим образом.

1. Нервная система планеты (Planetary Nervous System).
2. Симулятор живой планеты (Living Earth Simulator).
3. Глобальная объединенная платформа (Global Participatory Platform).

Набор моделей, формирующих «симулятор живой планеты», посредством «обсерваторий» (observatories) позволит обнаруживать кризисы и выявлять решения для смягчения последствий. Эти модели будут верифицироваться и калиброваться с помощью данных, собираемых в режиме реального времени с помощью «нервной системы планеты». В конечном счете принимающие решения лица

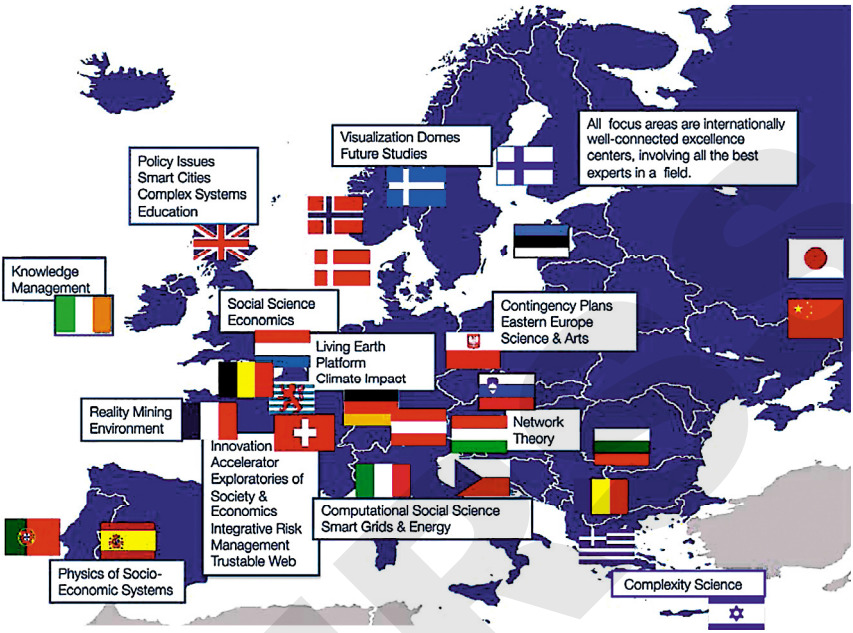


Рис. 2.2.2.3. Направления деятельности национальных научных центров. *https://futurict2.eu*

будут взаимодействовать с «глобальной объединенной платформой», сводящей результаты работы двух первых частей FuturICT.

Различного рода «обсерватории» (финансовые, экономические, энергетические, транспортные) объединяются в четыре группы основных направлений исследований в области общественных отношений, технологического и экономического развития, а также в области мониторинга состояния окружающей среды (рис. 2.2.2.4).

«Нервную систему планеты» можно представить в виде глобальной сети «сенсоров», собирающих информацию о социально-экономической, технологической и экологической системах мира. Для ее построения, координаторы проекта FuturICT тесно сотрудничают с командой Сэнди Пентленда (Sandy Pentland) из Массачусетского технологического института (MIT) с целью «подключения сенсоров» к современным гаджетам.

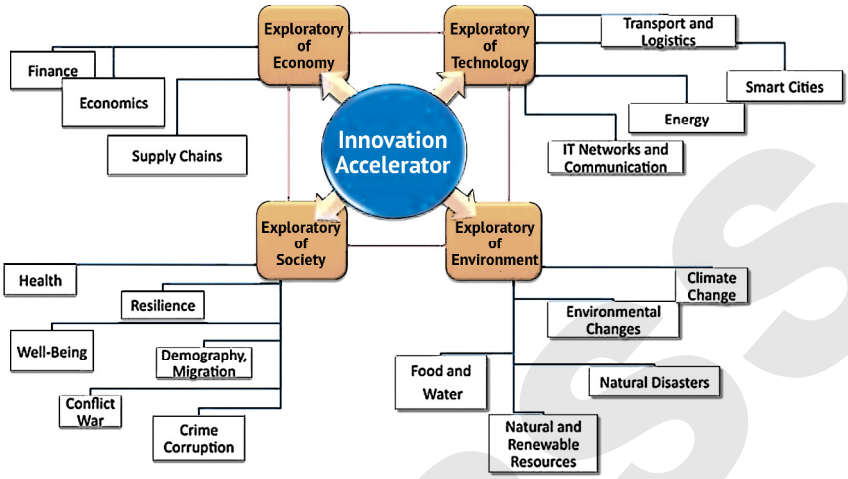


Рис. 2.2.2.4. Направления исследований в области общественных отношений, технологического и экономического развития, а также в области мониторинга состояния окружающей среды. Как видно, область общественных отношений включает в себя «обсерватории» социального благополучия, здоровья, преступности, коррупции, демографии, миграции и др. *https://futurict2.eu*

В рамках «симулятора живой планеты» будет реализована открытая программная платформа, которая, по мнению инициаторов проекта, будет напоминать известный App Store в том смысле, что ученые, разработчики и интересующиеся люди смогут загружать свои и скачивать другие модели, относящиеся к различным частям планеты. При этом основной подход к моделированию будет основан на агент-ориентированной парадигме. Унифицированные модельные компоненты в дальнейшем предполагается одновременно реализовывать с использованием суперкомпьютерных технологий.

«Глобальная объединенная платформа» в том числе будет открытой площадкой для обсуждения получаемых с помощью FuturICT прогнозов развития мира со стороны граждан, представителей власти и бизнес-сообщества.

Более подробно про проект можно прочитать на сайте: https://futurict2.eu/

2.2.3. Компьютерная модель мира от Microsoft

В 2010 г. компания Microsoft анонсировала ни много ни мало **компьютерную модель мира** с использованием в том числе агент-ориентированного подхода, то есть, по сути, создание виртуальной реальности, описывающей текущее состояние социальной и экономической системы всего мира.

Ниже приведен текст, озвученный в промо-роликах топ-менеджером Microsoft Бобом Маглиа (Bob Muglia), экс-президентом подразделения серверов и инструментов бизнеса в компании:

> «На протяжении своей истории человечество стремилось к новым знаниям и пыталось ответить на самые важные для себя вопросы. При этом было собрано огромное количество самых разных данных, которые наши ученые и специалисты в области информационных технологий будут обрабатывать в течение ближайших пяти лет. Мы будем использовать весь потенциал наших технических наработок с целью моделирования окружающего нас мира, что позволит, по нашему мнению, перейти на другой уровень понимания наиболее актуальных проблем нашего времени для их последующего разрешения. В этой связи мы считаем, что наука находится в точке бифуркации: новейшие достижения в области суперкомпьютерных технологий, а также в области технологий интеллектуальной обработки информации позволят нам разрабатывать сложные модели, возможности которых соответствуют тем скоростям, с которыми меняется современный мир. Мы считаем, что создаваемые нами модели позволят найти пути трансформации для ключевых отраслей, оказывающих влияние на все население планеты, — для энергетики и здравоохранения. Кроме того, мы предполагаем, что инструментарий окажет огромное влияние на бизнес-сообщество и отрасль инженерии знаний, что может привести к созданию принципиально новых продуктов, предприятий и отраслей промышленности. Мы приглашаем профессионалов разных областей (инженеров, экономистов, бизнесменов, политиков и др.) к сотрудничеству и совместному поиску решений для ЛЮДЕЙ и ПЛАНЕТЫ».

Несомненный интерес крупнейших игроков ИТ-рынка к агентным моделям лишний раз доказывает перспективность инструмента и его большое будущее.

2.2.4. Параллельный крупномасштабный симулятор транспортных потоков с использованием агент-ориентированного подхода и языка X10

Специалисты исследовательской лаборатории компании IBM в г. Токио совместно с учеными Токийского технологического института разработали платформу для построения крупномасштабных симуляторов транспортных потоков с использование нового языка для параллельного программирования X10. Эксперименты с разработанными моделями для более 100 городов всего мира продемонстрировали практически линейный прирост производительности в зависимости от числа используемых процессорных ядер.

Помимо упомянутой платформы был разработан симулятор транспортных потоков, называемый «Megaffic» (от IBM Mega Traffic Simulator), в котором также используется язык программирования X10, разработанный для параллельного программирования, по сути представляющий собой расширение языка Java с дополнительной поддержкой массивов и процессов, а также разделяемого глобального адресного пространства.

Компоненты симулятора Megaffic представлены на рис. 2.2.4.1. Для моделирования транспортных потоков используется агент-ориентированная парадигма, в которой каждое транспортное средство является отдельным агентом. В процессе инициализации модели вычисляются место отправления и пункт назначения для всех агентов, а в процессе симуляции агенты взаимодействую друг с другом, что влияет на выбор маршрута, изменение скорости и полосы движения. После достижения агентом пункта назначения, он уничтожается из текущей симуляции.

Одна из компонент платформы предоставляет возможность редактирования объектов дорожной сети, а также возможность изменения конфигурации дорог, что позволяет проводить сценарный анализ последствий этих модификаций.

Программная инфраструктура на основе X10 с использованием агентного подхода (X10-based Agent eXecutive Infrastructure for Simulation, XAXIS) — платформа для имитационного моделирования

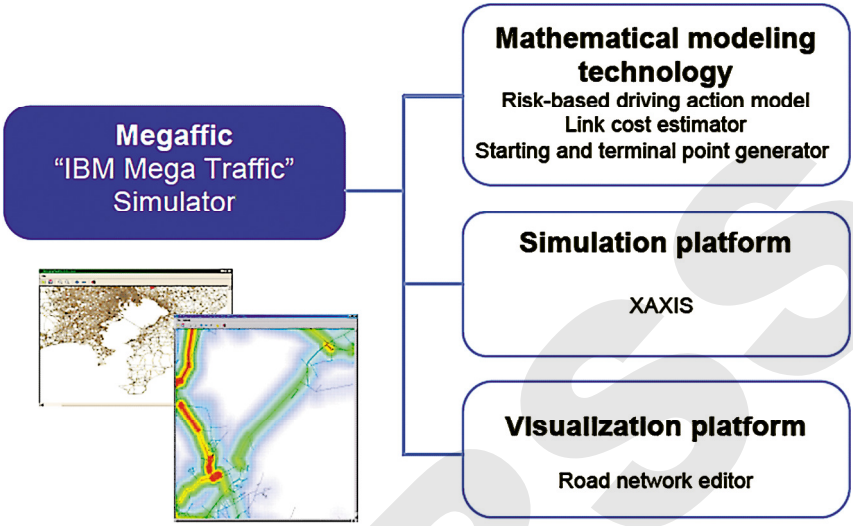


Рис. 2.2.4.1. Конструкция симулятора Megaffic. *https://researcher.watson.ibm.com*

с использованием крупномасштабных агентных моделей и нового языка для параллельного программирования X10. Этот язык — разработка исследовательской лаборатории компании IBM, позволяющий конструировать высокопроизводительные приложения в распределенных системах с различной конфигурацией. В X10 используется разделенное глобальное адресное пространство (Partitioned Global Address Space, PGAS), и, таким образом, вся память вычислительной системы делится на логические разделы, каждый из которых локален для какого-то процесса или потока. В X10 используется концепция, при которой выполняемая задача может порождать подзадачи, которые, в свою очередь, также могут порождать подзадачи. При этом, подзадачи не могут ожидать завершения работы от породивших их задач, но возможно обратное ожидание. Такая концепция оптимизирует время простоя системы.

Архитектура XAXIS

На рис. 2.2.4.2 показана архитектура XAXIS. Менеджер агентов управляет выполнением программного кода агентов в асинхронном

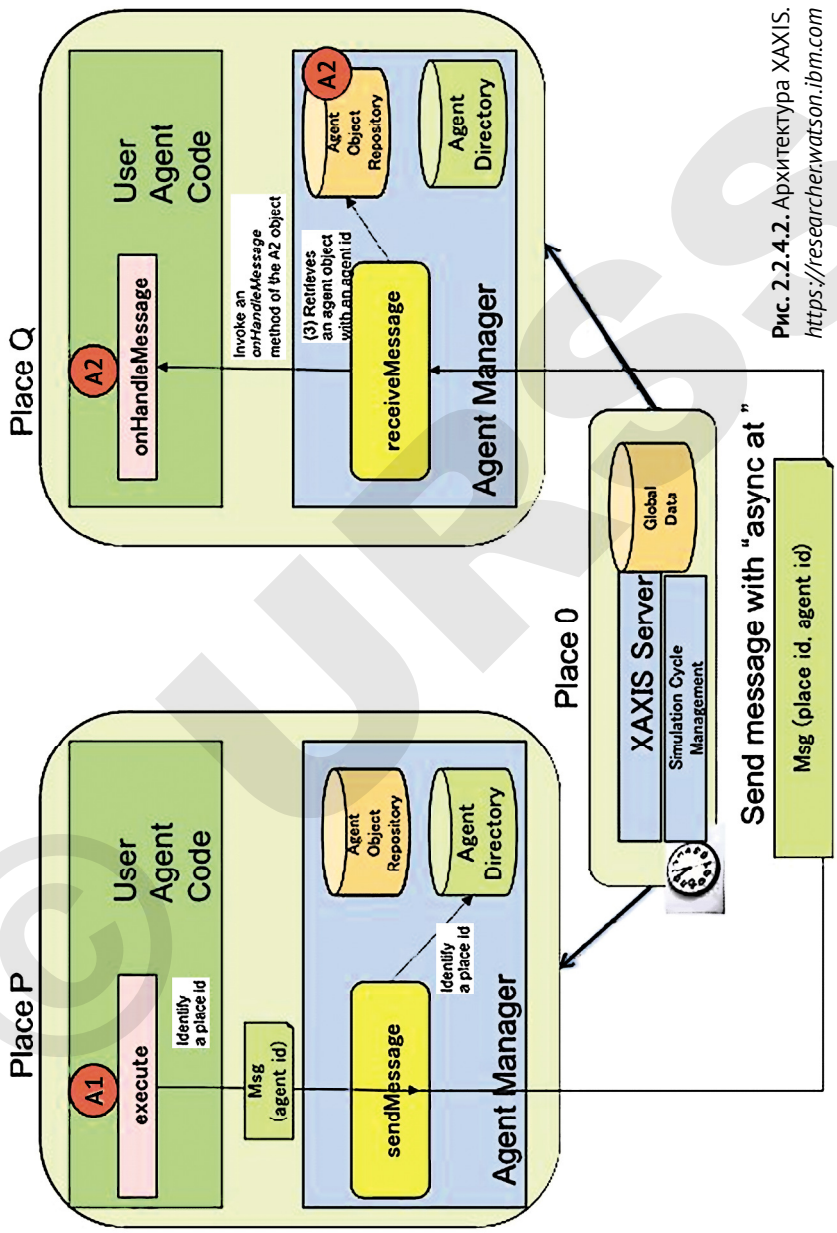


Рис. 2.2.4.2. Архитектура XAXIS. *https://researcher.watson.ibm.com*

режиме. Разработчик, используя XAXIS, описывает логику работы агентов и их взаимодействие на высоком уровне абстракции, а низкоуровневые коммуникации, связанные с распределением агентов и выполняемого кода по узлам суперкомпьютера, осуществляются программной средой в автоматическом режиме.

Практическая реализация транспортных агентных моделей на суперкомпьютере

Симулятор транспортных потоков XAXIS реализуется в основном на TSUBAME 2.0 — суперкомпьютере петафлопсного класса с гетерогенной архитектурой, находящемся в Международном научно-информационном вычислительном центре Токийского технологического института (Global Scientific Information and Computing Center, Tokyo Institute of Technology). Также отметим, что данный суперкомпьютер постоянно находится в топе мирового рейтинга «зеленых» суперкомпьютеров (то есть наиболее энергоэффективных).

Рис. 2.2.4.3. Визуализация симуляций для Японии.
https://researcher.watson.ibm.com

© URSS

Виртуальная проекция дорожной сети Японии включает в себя 2 552 160 дорожных отрезков и 993 731 перекрестков (рис. 2.2.4.3).

В свою очередь TSUBAME 2.0 содержит 52 ГБ на 1 узел, что позволяет хранить всю дорожную сеть в памяти.

Для увеличения производительности симулятора дорожная карта Японии, с помощью сторонней утилиты (METIS), была разделена на 100 областей с равномерным количеством перекрестков (в среднем по 10 000), а при тестировании приложения использовалось 100 узлов суперкомпьютера с 1200 ядрами. В рамках одной симуляции осуществлялось 3600 шагов (условно соответствующих одному часу) и около 180 000 поездок. На рис. 2.2.4.4 показано время выполнения симуляций для всех 100 областей, где среднее время выполнения составило 269,95 секунд. Несмотря на то, что число перекрестков для всех областей было примерно одинаковым, для некоторых из них, к примеру для № 5 (Осака), № 12 (юго-восточная часть г. Нагоя и префектура Айти), № 46 (Хоккайдо с административным центром — Саппоро), № 72 (префектура Фукуока с одноименным центром), потребовалось заметно больше времени для

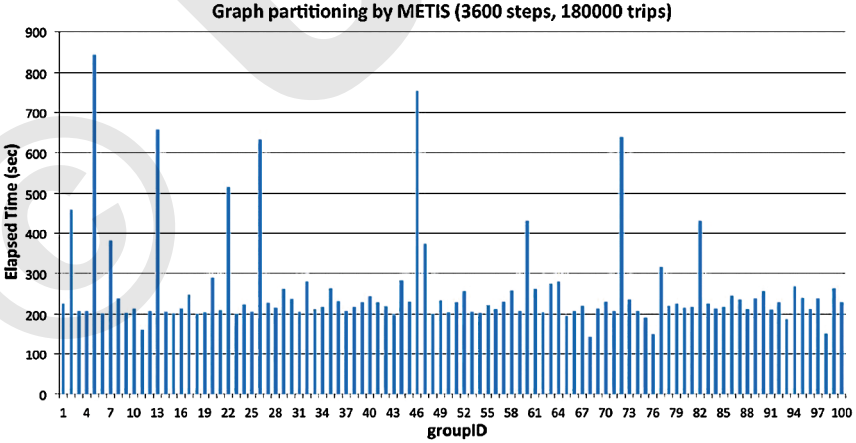


Рис. 2.2.4.4. Время выполнения симуляций для 100 областей дорожной сети Японии, в секундах. *https://researcher.watson.ibm.com*

исполнения. Это связано с более плотным трафиком, и, как следствие, с большим числом агентов в данных областях. Решением проблемы может быть дополнительное разделение сложных областей на подобласти, используя METIS.

Далее были проведены симуляции для дорожных сетей наиболее крупных городов мира (Сингапур, Лондон, Мехико и др.) с использованием карт проекта OpenStreetMap. На рис. 2.2.4.5 приведены результаты для 1000 шагов и 50 000 поездок, где на оси абсцисс отложены ID городов, а на оси ординат — время в секундах. Отметим, что размерности виртуальных дорожных сетей варьируются от десятков тысяч до двух миллионов перекрестков.

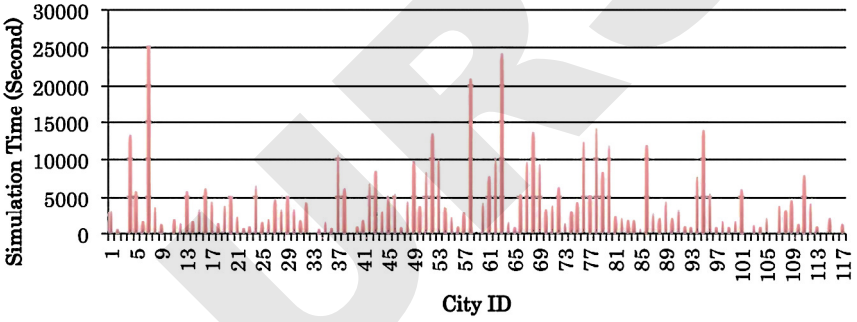


Рис. 2.2.4.5. Время выполнения симуляций для рассматриваемых городов мира. *https://researcher.watson.ibm.com*

А на рис. 2.2.4.6 показаны примеры визуализации симуляций для некоторых из этих мегаполисов.

Следует отметить явный интерес одного из крупнейших в мире производителей и поставщиков аппаратного и программного обеспечения — компании IBM к агент-ориентированному подходу и к разработке фреймворков для построения моделей для суперкомпьютеров. Более подробно про реализуемый проект можно прочитать на сайте корпорации:

https://researcher.watson.ibm.com/researcher/view_group.php?id=536

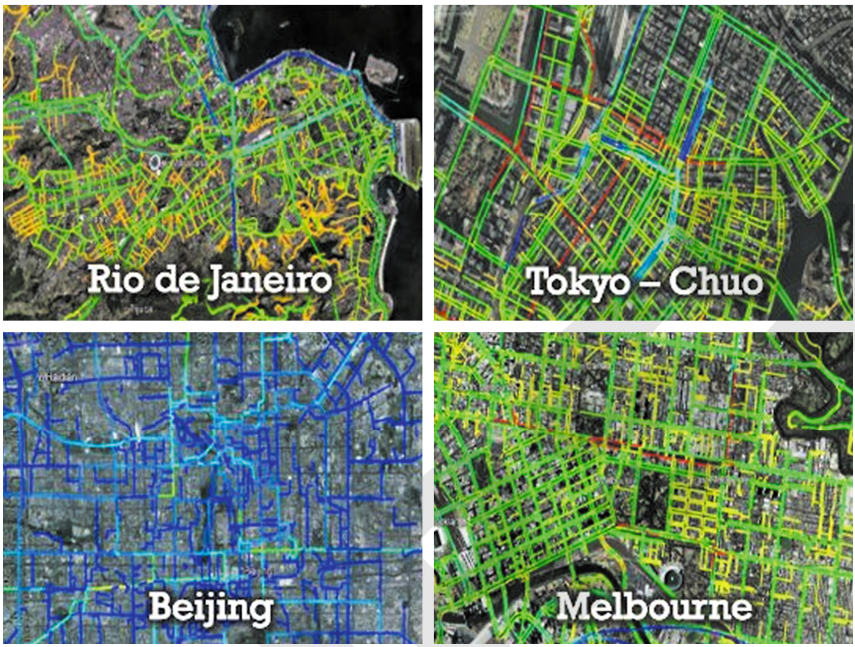


Рис. 2.2.4.6. Визуализация симуляций для некоторых мегаполисов. *https://researcher.watson.ibm.com*

2.2.5. Другие проекты в сфере экономики

Известный немецкий социолог Никлас Луман (Niklas Luhmann), посвятивший всю жизнь изучению человеческого общества, рассматривал экономику как часть более сложной социальной системы и выдвинул известную гипотезу о том, что экономические системы с целью сохранения усиливают исходное неравенство [24], с. 112). Его идеи были частично реализованы в виде агент-ориентированной модели Флейшманна, в которой рассматривается простая экономическая система. Основной результат проведенных с ее помощью экспериментов состоит в том, что неравенство в обществе всегда возникает независимо от начального состояния, имеющихся различий между агентами и модификаций правил взаимодействия между ними [25]. Автор поддерживает гипотезу Лумана и пишет,

что «потребовался длительный период модельного времени (для 36 агентов, 3 видов товара общим количеством 900 и 150 000 торговых операций), чтобы подтвердить гипотезу о том, что экономика производит неравенство из неравенства». Флейшманн также отмечает в работе несколько важных предположений и предлагает расширения модели для их проверки. Это обстоятельство иллюстрирует, что агентный подход используется не только для генерирования гипотез, но и для их оспаривания.

В отдельное направление следует выделить агент-ориентированное моделирование финансовых рынков. Анализ библиографических баз Web of Science и Scopus показал, что к настоящему моменту разработано сотни агентных моделей, связанных с моделированием финансовых рынков. Использование этого инструмента для обозначенных целей вполне оправдано. Как отмечается в тематической публикации самой крупной в мире аудиторской и консалтинговой компании Deloitte, посвященной моделированию финансовых рынков, применение агент-ориентированных моделей позволяет имитировать самые разнообразные типы поведения участников рынка. В отличие от традиционных математических методов, в которых зачастую формализуются зависимости, построенные на основе субъективных предположений, агент-ориентированный подход позволяет получать новые знания о механизмах работы финансовых рынков за счет проведения многочисленных симуляций.[1]

Один из наиболее цитируемых исследователей финансовых рынков с использованием агентного подхода — профессор Кильского университета Томас Люкс в своей открытой лекции «*Agent-Based Models in Finance: Foundations, Explanatory Power and Application*», прошедшей 3 февраля 2021 г., отметил, что за последнее десятилетие агент-ориентированные модели проделали большой путь в

[1] Agent-based modeling for central counterparty clearing risk. 2020. April. URL: https://www2.deloitte.com/uk/en/pages/audit/articles/agent-based-modelling-for-central-clearing-risk.html

области имитации работы финансовых рынков за счет возможности программной формализации настроения их участников[1].

В работах Лебарона Блейка (Lebaron Blake) рассматривается агент-ориентированный подход применительно к фондовому рынку США. Автор получил интересные и принципиальные результаты, заключающиеся в том, что при расширении доступного для агентов набора стратегий может значительно меняться траектория биржевых индексов [26, 27].

В 2016 г. профессор Университета Джорджа Мейсона (George Mason University) Роберт Акстелл опубликовал модель, в которой достаточно простые агенты (около 120 млн) объединяются в группы и этот механизм воспроизводит процесс образования фирм в США (около 6 млн фирм, см. [28]). Это работа основана на более раннем исследовании — агент-ориентированной модели, симулирующей динамику образования фирм, которая весьма реалистично сгенерировала существовавшую в то время совокупность фирм в США в соответствии с их размером. Полученные результаты были опубликованы в журнале «Science» [29]. В противовес классическим моделям, предполагающим, что экономика находится в состоянии равновесия, в реальности наблюдается значительная динамика практически во всех сферах. К примеру, автор отмечает, что в США (на момент его публикации) около 3 млн работников ежемесячно находят новую работу, то есть примерно 1 из 40 работников меняет работодателя и 1 из 60 фирм закрывается. Соответственно, агенты модели постоянно находятся в движении, приспосабливаются к возникающим обстоятельствам и предопределяют заметные флуктуации на рынке труда. Один из результатов заключается в том, что на рынке труда не получается устойчивых равновесных состояний [30]. Тем не менее, несмотря на микроуровневую турбулентность, ассиметричное распределение Ципфа для размеров фирм получилось стационарным.

[1] URL: https://wilmott.com/agent-based-models-in-finance-foundations-explanatory-power-and-applications

Один из инструментов эконофизики — статистическая механика, в частности, была использована для понимания механизмов распределения денег среди взаимодействующих агентов. В работе «*Colloquium: Statistical mechanics of money, wealth, and income*» это хорошо продемонстрировано [31]. Анализ данных результатов экспериментов по распределению богатства и доходов показал образование двух классов. Большинство населения принадлежит к низшему классу, характеризующимся экспоненциальным распределением, а небольшая часть высшего класса — степенным. При этом низший класс стационарен, а высший находится в постоянной динамике и неравновесных состояниях.

Весьма интересное исследование было проведено Робертом Пекхэмом (Robert Peckham) — ученым из Гонконгского университета. Им были исследованы вопросы распространения различного рода инфекций (H1N1, H5N1, SARS, HIV/AIDS и др.) во взаимосвязи с развитием финансовых кризисов. Автор не только идентифицирует причинно-следственную связь между этими процессами, но и отмечает схожесть механизмов их развития. По его мнению проблемы на финансовых рынках, как и сообщения о распространения пандемии, вызывают панику и экономическую нестабильность, и в этой связи экономисты все чаще обращаются к эпидемиологии с целью поиска эффективного «антидота». В этом смысле инфекционные заболевания переопределяют финансовую нестабильность как патогенную форму, оказывающую негативное влияние на всю социально-экономическую систему страны, а инициаторов кризиса (спекулянтов и пр.) как возбудителей заражения [32].

По словам известного физика Жана-Филиппа Бушо (Jean-Philippe Bouchaud), руководителя исследовательского отдела Capital Fund Management (Париж, Франция), «изучение сложных физических систем за последние 40 лет значительно продвинулось вперед, появились новые идеи и методы. Это актуально и для экономики, поскольку с помощью новых инструментов удалось построить симуляторы, позволяющие выявлять точки бифуркации, вызывающие кризисы. К таким инструментам в первую очередь относятся

агент-ориентированные модели, являющиеся более реалистичными и гибкими, нежели классические модели экономики. В отличие от последних, в агентных моделях возможно реализовывать приводящие к панике человеческие эмоции. Страны с развитой экономикой до сих пор борются с последствиями кризиса, и экономисты не могут прийти к согласию по поводу его фундаментальных причин. С использованием новых инструментов мы, безусловно, может добиться большего»[1].

Нельзя не упомянуть про новое направление в прикладной экономике — агент-ориентированную вычислительную экономику (Agent Based Computational Economics, ACE). Один из идеологов этого направления — профессор Университета штата Айова Ли Тесфатси (Leigh Tesfatsion), определяет процесс построения моделей в соответствии с методологией ACE как лабораторный биологический эксперимент с использованием чашки Петри. Предполагается, что разработчик задает начальные условия для моделируемой системы в соответствии с некоторой целью ее функционирования, а в дальнейшем динамика этого виртуального мира определяется без участия извне, а под воздействием его составных частей. Тесфатси подчеркивает, что парадигма ACE разработана в качестве дополнения к современной экономической теории, предполагающей рациональность и оптимальность ее участников, а также существование равновесия [33]. В свою очередь, ACE предполагает большую реалистичность агентов, непредсказуемость поведения агентов по отношению друг к другу и некую автономность виртуальных миров, в которых они существуют. В проект по созданию таких миров вовлечено много исследователей, а результаты выложены на сайте: http://www2.econ.iastate.edu/tesfatsi/ace.htm.

После критики Аланом Кирманом [34] моделей репрезентативного агента макроэкономической теории, предпринимались попытки разработать макроэкономическую теорию с использованием разнородных индивидов. Одна из них заключалась в расширении

[1] URL: https://www.ft.com/content/f653a4ee-2dce-11e8-a34a-7e7563b0b0f4

методологии построения динамических стохастических моделей общего равновесия (Dynamic Stochastic General Equilibrium, DSGE) путем включения в них гетерогенных агентов, но самая главная — использование агент-ориентированных моделей. Большое количество успешных примеров в этих направлениях собрано в работе «*Computational Economics: Heterogeneous Agent Modeling*» [35].

В последние годы этот инструментарий прочно вошел в арсенал средств, используемых научными организациями, аналитическими центрами, корпорациями, органами государственного управления и др. применительно к вопросам моделирования и прогнозирования экономических процессов. Так, в октябрьской статье 2003 г. «Agents of creation»[1] журнала «The Economist» агент-ориентированные модели позиционировались как новый инструмент моделирования сложных систем. В 2010 г. в этом же издании в статье «Agents of change»[2] они рассматривались как альтернатива динамическим стохастическим моделям общего экономического равновесия и с тех пор постоянно фигурируют в качестве доказавшего свою эффективность программного средства.

В редакционной статье журнала «Nature» агентное моделирование рекомендовалось в качестве перспективного и многообещающего инструмента исследования социально-экономических процессов [36].

Агент-ориентированное моделирование бизнес-процессов

Агентный подход также используется при моделировании бизнес-процессов. Так, в работе «*Agent-based modeling in marketing: Guidelines for rigor*» отмечается, что «Агент-ориентированное моделирование позволяет понять как сложные маркетинговые решения возникают из достаточно простых действий, осуществляемых на микроуровне. Такие решения слишком сложны для традиционных ана-

[1] Статья журнала «The Economist», вышедшая в 2003 г. URL: https://www.economist.com/science-and-technology/2003/10/09/agents-of-creation

[2] Статья журнала «The Economist», вышедшая в 2010 г. URL: https://www.economist.com/finance-and-economics/2010/07/22/agents-of-change

литических подходов, поэтому агентное моделирование, применяемое для имитации отдельных потребителей и организаций, часто используют для обозначенной проблемы» [37]. В качестве примера авторы реализовали расширенную модель распространения инноваций Басса с использованием агентного подхода.

Исследователи из Силезского университета (Опава, Чехия) в коалиции с коллегами из Дании разработали среду моделирования MAREA (Multi-agent REA framework), интегрированную с ERP (Enterprise Resource Planning) системой. В работе «*Is there a Need for Agent-based Modelling and Simulation in Business Process Management?*» представлены результаты, полученные с ее помощью на примере модели, имитирующей работу торговой компании по продаже компьютерных кабелей. Конкретно оценивалось влияние изменений в наборе различного рода ресурсов на финансовый результат компании. Расчеты показали, что даже небольшие низкоуровневые изменения оказывают статистически значимое влияние на результаты бизнес-процессов. По мнению разработчиков, агент-ориентированное моделирование в ближайшее время получит большее распространение в управление бизнес-процессами [38].

Программно-аналитический комплекс «МЁБИУС»

В течение многолетней работы в ЦЭМИ РАН были разработаны методологические принципы построения сложных программно-аналитических комплексов, сочетающих в себе различные подходы к моделированию социально-экономических процессов, и в первую очередь агент-ориентированный. Эволюционным путем мы построили модельный комплекс «МЁБИУС»[1], который вобрал в себя преимущества разных подходов и состоит из нескольких блоков (рис. 2.2.5.1).

[1] Название программно-аналитического комплекса связано с двумя обстоятельствами. Во-первых, лента Мёбиуса является главным символом ЦЭМИ РАН, а его горельеф украшает здание института. Во-вторых, этот знак является физическим воплощением бесконечности, что соответствует масштабам вычислительных возможностей одного из модулей — «МЁБИУС-суперкомпьютер», способного технически реализовывать модели с численностью порядка 10^9 агентов на самых производительных в мире системах (в том числе на суперкомпьютерах экзафлопсного уровня).

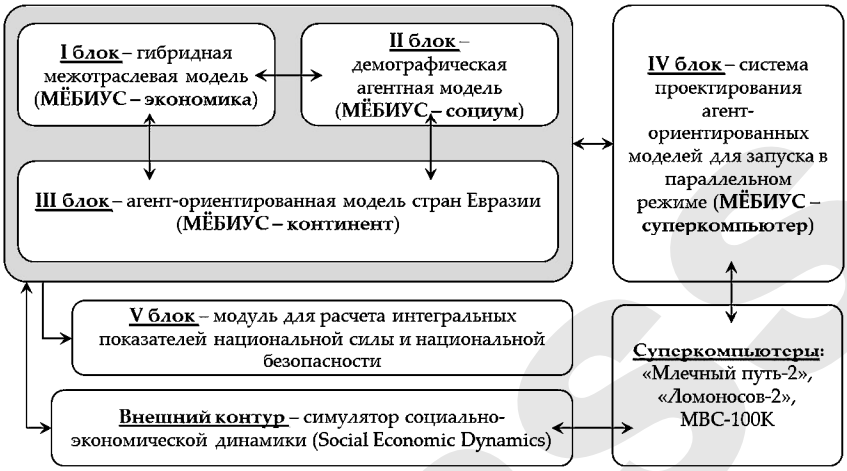


Рис. 2.2.5.1. Схема программно-аналитического комплекса «МЁБИУС». *Программно-аналитический комплекс «МЁБИУС» — инструмент планирования, мониторинга и прогнозирования социально-экономической системы России* [39]

Все блоки комплекса «МЁБИУС», хотя и были изначально разработаны отдельно, тесно связаны друг с другом и представляют собой единое целое. Демографическая агент-ориентированная модель является базовым элементом для агентной модели Евразийского континента, а также определяет динамику численности домашних хозяйств для межотраслевого блока «МЁБИУС — экономика», который, в свою очередь, также связан с моделью стран Евразии. Для ускорения расчетов к этим инструментам может быть подключена система «МЁБИУС — суперкомпьютер», которая в автоматическом режиме перераспределяет исполняемый программный код по задаваемому числу процессоров для выполнения в параллельном режиме [39][1].

С использованием программного комплекса был проведено множество расчетов, к примеру:

[1] Для нас доступны суперкомпьютеры «Млечный путь-2» (Национальный суперкомпьютерный центр КНР, г. Гуанчжоу), «Ломоносов-2» (МГУ имени М. В. Ломоносова), МВС-100К (Межведомственный суперкомпьютерный центр РАН).

- прогноз изменений возрастно-половой структуры населения регионов России [40];
- демографические изменения в Европейском союзе с учетом важных факторов, таких как внутренние установки людей, влияющие на их стратегию репродуктивного поведения, а также миграционный приток извне [41];
- динамику трудовой миграции между Россией и Китаем [42];
- уровня загрязнений в некоторых регионах РФ с учетом пространственного размещения производств и стационарных источников выбросов [43];
- прогноз основных макроэкономических показателей России и ее субъектов в результате изменения ставок основных налогов, бюджетных субсидий, объема денежной массы, ключевой ставки, цен на основные энергоносители, курса доллара и др. [44].

Отдельным блоком выступает модуль для расчета интегральных показателей национальной силы и национальной безопасности для 193 стран — членов ООН. Веса выбранных факторов (их несколько десятков) были рассчитаны с использованием методов многомерного статистического анализа.

Помимо разработанных в ЦЭМИ РАН блоков программно-аналитического комплекса, совместно с IT компанией Guangzhou Milestone Software Co., Ltd, специалистами академии общественных наук Китая и национальным суперкомпьютерным центром КНР был разработан симулятор социально-экономической динамики для более 100 стран мира. Этот инструмент используется в составе «МЁБИУСа» следующим образом — симулятор является внешним для него контуром, и таким образом получаемые с его помощью данные являются входными для первых трех блоков комплекса, равно как и наоборот (Макаров, Ву, Ву и др., 2020).

В следующем разделе мы опишем построенные в ЦЭМИ РАН мультиагентные демографические модели, различающиеся уровнем детализации при имитации репродуктивного поведения людей. Далее, с использованием разработанной авторами технологии поддержки агент-ориентированных моделей для суперкомпьютеров — STARS

(Supercomputer Technology for Agent-oRiented Simulation), проанализируем этапы и методы эффективного отображения счетного ядра мультиагентной системы на архитектуру современного суперкомпьютера.

2.3. Разработка на основе агент-ориентированных моделей и суперкомпьютерных технологий возможных сценариев развития социально-экономической системы России[1]

Одна из задач нашего исследования заключалась в рассмотрении различных методов распараллеливания мультиагентных систем для запуска их на суперкомпьютерах, а также в сопоставлении показателей эффективности этих методов распараллеливания в зависимости от архитектуры мультиагентной системы, численности популяции агентов и числа используемых ядер вычислительных кластеров.

Таким образом, на первом этапе нашего исследования необходимо было разработать такие мультиагентные модели, которые обладали бы следующими свойствами:

- моделировали реально происходящие в социально-экономических системах процессы, в которых действуют самостоятельные различающиеся между собой акторы;
- при этом моделируемые процессы должны быть массовыми и требовать для адекватного представления в модели создания популяции агентов очень большой численности, а значит, естественным образом нуждаться в использовании суперкомпьютеров для проведения численных экспериментов;
- модели должны отличаться друг от друга какими-либо особенностями архитектуры, которые имеют существенное значение для выбора методов их распараллеливания и реализации на суперкомпьютере.

[1] Данные модели были разработаны при непосредственном участии ведущего научного сотрудника, кандидата экономических наук Е. Д. Сушко, а также специалиста компании «Яндекс» Г. Б. Сушко.

На втором этапе следовало реализовать и апробировать разработанные конструкции моделей на обыкновенном персональном компьютере для популяции агентов с максимально возможной численностью.

И, наконец, уже на третьем этапе — каким-либо образом преобразовать эти модели для переноса их на суперкомпьютер с использованием различных методов распараллеливания их работы и с возможностью на основе этого значительного увеличения популяции агентов. При этом первоначальные модели и результаты проведенных на них экспериментов, во-первых, играют роль контрольных примеров, позволяющих существенно сократить время на отладку и верификацию суперкомпьютерных моделей, а во-вторых, служат базой для сравнения показателей эффективности выбранных методов распараллеливания.

Классическим примером массовых социально-экономических процессов, в которых каждый из участников под влиянием своих интересов и в соответствии со своими возможностями может действовать самостоятельно, следует признать процессы демографические. Поэтому не случайно в литературе можно найти описания многочисленных мультиагентных систем, посвященных рассмотрению самых разнообразных аспектов основных демографических процессов, таких как смертность, рождаемость и миграция [45, 46, 47, 48].

2.3.1. Концепция разработанных демографических моделей

Разработанные нами демографические мультиагентные модели посвящены рассмотрению такого известного явления как демографический переход и связанных с ним структурных изменений в обществе. Демографическим переходом принято называть переход от традиционного типа воспроизводства населения с высокими уровнями показателей смертности и рождаемости к современному — с низкими уровнями этих показателей. Особенности и этапы этого феномена подробно рассматривались в работах многих известных

демографов, как зарубежных, так и отечественных, здесь достаточно сослаться на известную работу А. Г. Вишневского [49]. Наблюдаемыми признаками демографического перехода являются существенные изменения в показателях смертности и рождаемости. Так, с развитием общества, повышением его благосостояния и улучшением системы здравоохранения показатели смертности (особенно младенческой) снижаются.

Происходящее одновременно изменение уровня рождаемости вызвано изменением стиля жизни, особенно стиля жизни женщин — их эмансипацией, повышением уровня образования, появлением личностных целей, не связанных с семьей. Одним из ярких симптомов этого может служить набирающее силу в развитых и благополучных в экономическом отношении странах движение «Child-free» — полный отказ от рождения детей. В итоге в современном обществе суммарный коэффициент рождаемости (среднее число детей, рождаемых женщиной в течение репродуктивного возраста) снижается зачастую до уровня, не обеспечивающего простое воспроизводство населения.

Следует, однако, подчеркнуть, что изменения уровней смертности и рождаемости происходят хотя и параллельно, но несинхронно, причем, снижение рождаемости отстает от снижения смертности на одно-два поколения. Все эти процессы вызывают не только снижение общей численности населения (депопуляцию), но и кардинальные изменения в его возрастной структуре — так называемое старение (увеличение доли старших возрастных когорт). Обычно такого рода процессы рассматриваются в связи с междустрановыми сопоставлениями, однако они могут наблюдаться и внутри одной страны, если отдельные группы ее населения придерживаются разных репродуктивных стратегий. Так, различия в поведения людей, придерживающихся стратегий, присущих традиционному и современному обществам, наблюдаются в выборе времени вступления в брак, в среднем числе рождаемых каждой женщиной детей, а также в распределении времени рождения детей на протяжении фертильного периода женщин.

Особенности этих процессов в нашей стране хорошо представлены, например, в сборнике [50], посвященном результатам уникального социально-демографического обследования «Родители и дети, мужчины и женщины в семье и обществе». Это исследование проводилось в рамках международной программы «Поколения и гендер», объединившей исследователей-демографов из более 30 стран мира. Репродуктивные стратегии женщин современной России изучены также в работе [51].

Использование результатов этих работ позволяет перенести фокус моделирования демографических процессов на микроуровень и применить агент-ориентированный подход. Кроме того, указанные данные позволяют создать достаточно интеллектуальных, интенциональных агентов [52], то есть агентов, наделенных собственными механизмами мотивации. Для таких агентов каким-либо образом моделируются внутренние убеждения, желания, намерения и мотивы, порождающие цели, которые и определяют действия агентов. В наших моделях роль таких «убеждений» будут выполнять желаемое максимальное число детей, а также распределение рождения этих детей в течение репродуктивного периода женщины.

Для моделирования таких демографических процессов, как смертность и рождаемость населения (миграция в модели не учитывается), и перспективного исчисления численности населения мы воспользовались методом передвижки возрастов [53]. Этот метод заключается в том, что каждый год выжившие в соответствии с половозрастными коэффициентами выживаемости ежегодно становятся на год старше и участвуют в репродуктивном процессе (в соответствии с данными о рождаемости для женщин фертильного возраста).

На основе такого подхода и с использованием реальных статистических данных в ЦЭМИ РАН были разработаны две мультиагентные демографические модели, различающиеся уровнем детализации при имитации репродуктивного поведения людей. Причем направление усложнения поведения агентов было выбрано, в том числе, и с учетом значимости вносимых изменений для выбора

алгоритмов распараллеливания моделей при их переносе на суперкомпьютер. С этой точки зрения принципиальным отличием является то, что в первой модели агенты не образуют супружеские пары (нет взаимодействия с другими агентами), а просто агенты-женщины репродуктивного возраста самостоятельно определяют число желаемых детей и время их рождения (создания новых агентов). В то время как во второй модели агенты-люди между собой взаимодействуют — образуют пары и согласовывают рождение общих детей.

Модели представляют собой программные продукты, реализованные в среде AnyLogic[1] и обладающие интерфейсом для демонстрации пользователю происходящих во времени изменений

Рис. 2.3.1.1. Общая схема работы демографической модели

[1] *AnyLogic* — инструмент имитационного моделирования, который поддерживает все подходы к созданию имитационных моделей: процессно-ориентированный (дискретно-событийный), системно динамический и агентный, а также любую их комбинацию. Более подробно на сайте: http://www.anylogic.ru

основных демографических характеристик популяции агентов условного региона. Общая схема работы моделей дана на рис. 2.3.1.1.

Представим вначале конструкцию первой модели, а затем покажем изменения отдельных элементов конструкции, произведенных во второй модели.

2.3.2. Конструкция первой разработанной модели

Агенты разделены на две группы, различающиеся репродуктивными стратегиями. Агенты первой группы придерживаются традиционной стратегии, для которой характерна высокая рождаемость, а агенты второй группы — современной, при которой рождаемость существенно ниже.

Компоненты агентов-людей:

- тип репродуктивного поведения;
- пол;
- возраст;
- максимальное (желаемое) число детей;
- фактическое число детей;
- родственная связь с агентом-матерью;
- процедура включения в коллекцию женщин, желающих родить ребенка.

Компоненты среды:

- текущий год;
- общая численность популяции агентов;
- доля агентов традиционного типа;
- половозрастная структура популяции агентов;
- доля агентов традиционного типа в каждой возрастной когорте;
- коэффициенты смертности, дифференцированные по полу и возрасту;
- суммарные коэффициенты рождаемости для двух типов агентов;
- параметры вероятностных распределений, с помощью которых случайным образом определяется желаемое число детей для агентов-представительниц каждого типа;

© URSS

- распределение рождений по возрасту матери для двух типов агентов;
- коллекции агентов-женщин, желающих родить ребенка (отдельно для каждого типа агентов и для каждой возрастной когорты, соответствующей репродуктивному периоду женщин — от 15 до 49 лет);
- процедура имитации смертности;
- процедура имитации рождаемости;
- процедуры связи с внешними файлами (чтение исходных данных и запись результирующих показателей).

При имитационном моделировании большое значение имеет не только правильная организация процессов, происходящих в течение модельного времени, но и точное воссоздание в начале работы модели стартового состояния системы.

Установка стартового состояния (блоки A2–A5 на схеме)

В начале работы модели из базы данных (таблиц формата Excel) считывается массив исходной информации. Для воссоздания структуры популяции агентов используется исходная информация, представленная в виде двух стандартных возрастно-половых пирамид, соответствующих двум типам воспроизводства (для популяции агентов традиционного типа характерна относительно бóльшая численность новорожденных и детей, поэтому пирамида имеет более широкое основание).

Параметрами модели служат общая численность популяции агентов, а также доля в этой численности агентов традиционного типа. В соответствии со значениями параметров и соотношениями, заданными возрастно-половыми пирамидами, создается множество агентов и распределяются такие их характеристики как тип репродуктивного поведения, пол и возраст. То есть каждому агенту присваиваются такие значения возраста, пола и типа, чтобы структура создаваемой популяция точно воспроизводила рассчитанную на основе исходных данных.

Каждому агенту-женщине присваивается также выбранное случайным образом с помощью β-распределения желаемое число детей в интервале от минимального до максимального для ее типа.

Затем устанавливаются связи «мать — ребенок» между созданными агентами, для чего используются данные о распределении рождений по возрасту матери. Фактически, для каждого агента моложе двадцати лет случайным образом с помощью β-распределения определяется возраст его матери, а затем из когорты полученного возраста выбирается агент-женщина того же типа, желаемое число детей которой больше числа рожденных. То есть считается, что тип ребенка совпадает с типом матери. Агент-ребенок и агент-мать запоминают друг друга, а кроме того, к числу детей агента-женщины добавляется единица. Параметры используемых здесь β-распределений подобраны таким образом, чтобы получаемые частоты выпадения возрастов соответствовали эмпирическим данным о распределении рождений по возрасту матери для двух типов агентов, представленным на рис. 2.3.2.1.

Анализ текущего состояния популяции агентов (блоки B2–B4)

Здесь собирается статистика по популяции агентов — общая численность агентов, численность агентов-мужчин и агентов-женщин по возрастным группам и типам. Собранная статистика сохраняется в таблицах выходного файла формата Excel для последующей обработки и анализа.

Интерфейс модели демонстрирует пользователю «фактическое» состояние популяции в текущем году, а также графики, показывающие изменение отслеживаемых результатов работы модели в течение модельного времени. После отображения на экране текущего состояния программа ждет решения пользователя о прекращении или продолжении симуляции. В первом случае происходит остановка работы модели, а во втором — переход к следующему шагу ее работы.

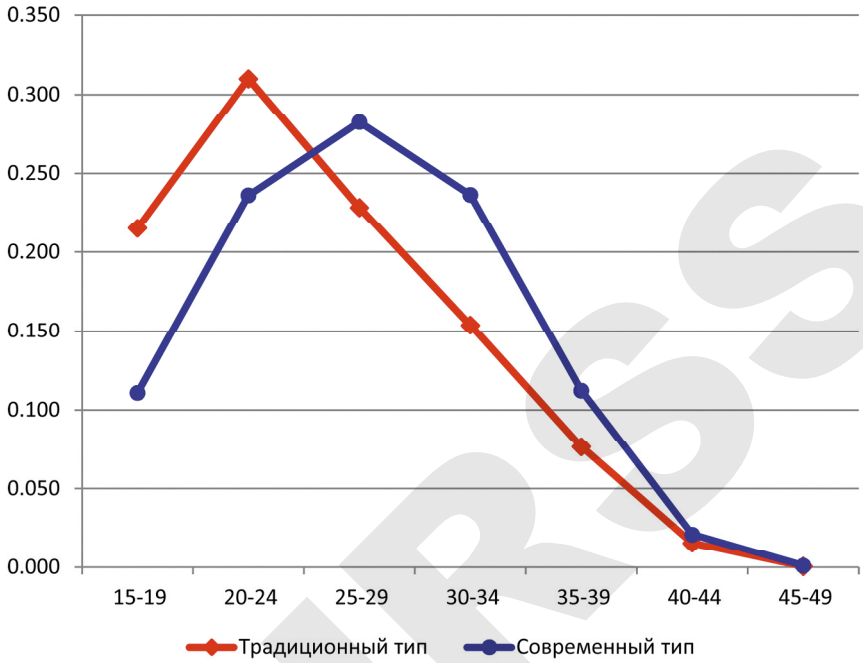


Рис. 2.3.2.1. Доли от общей численности новорожденных для матерей из разных возрастных групп

Действия, выполняемые на каждом шаге работы модели (блоки С1–С6)

К значению переменной года прибавляется единица.

Имитация смертности (*блок С2*). На основе дифференцированных по полу и возрасту коэффициентов смертности рассчитывается вероятность умереть для каждого агента популяции, после чего вероятностным образом определяется его судьба. Агенты, которым выпало умереть, уничтожаются, а оставшиеся становятся на год старше. (Коэффициенты смертности для агентов двух типов приняты одинаковыми и постоянными на протяжении всего периода симуляции.)

Имитация рождаемости (*блоки С3–С6*). Агенты-женщины репродуктивного возраста в соответствии со своим возрастом и типом пополняют коллекции желающих родить, если их желаемое число

детей больше фактического. Именно этот выбор и представляет собой действие, доступное агентам в модели.

Затем для каждого типа отдельно рассчитывается общее число агентов-женщин репродуктивного возраста и общее число младенцев, которых они должны родить в текущем году, соответствующее заданным суммарным коэффициентам рождаемости.

После чего для каждого типа создается рассчитанное количество агентов нулевого возраста, а пол каждого новорожденного агента определяется случайным образом (женский — с вероятностью 0,488 [54]).

И наконец, каждому новорожденному агенту подбирается мать из коллекций агентов-женщин репродуктивного возраста, желающих в этом году родить ребенка. Если при этом число рожденных агентом-матерью детей сравняется с числом желаемых, то она удаляется из коллекции желающих родить и больше не будет участвовать в репродуктивном процессе.

2.3.3. Результаты апробации первой модели

С использованием модели были проведены эксперименты по прогнозированию изменения численности популяции агентов, населяющих условный регион, возрастной структуры этой популяции, а также соотношения численности агентов разных типов для основных возрастных групп и для популяции в целом. Были проведены двадцать экспериментов при следующих значениях параметров:

- общая численность популяции агентов — 100 000;
- доля агентов традиционного типа — 10 %;
- показатели смертности для агентов двух типов соответствуют данным по России в целом;
- суммарный коэффициент рождаемости для агентов-женщин традиционного типа — 3,0, а для современного типа — 1,4;
- минимальное число желаемых детей для агентов традиционного типа — 2, а максимальное — 10; для агентов современного типа — 0 и 3 соответственно;
- базовый год — 2007;
- период симуляции — 20 шагов (лет).

В качестве прогнозных значений характеристик популяции были взяты их соответствующие средние по всем экспериментам, при этом модель продемонстрировала высокую устойчивость. Так, для общей численности популяции отклонения от среднего значения на протяжении 20 лет располагались в интервале от –0,23 % до 0,16 %, что можно считать очень хорошим результатом.

Анализ результатов экспериментов показал, что модель адекватно имитирует такие наблюдаемые в реальной жизни процессы как снижение общей численности населения (депопуляция), а также его старение. Снижение численности агентов на рис. 2.3.3.1 заметно, начиная с 2016 г. Там же показаны максимальные и минимальные значения численности агентов по годам. Старение населения видно на рис. 2.3.3.2, на котором можно заметить, что численность агентов моложе 50 лет неуклонно снижалась на протяжении всего периода, а численность агентов старше 80 лет возрастала.

Помимо этого, модель демонстрирует также и еще один эффект, наблюдаемый в реальности, а именно то, что соотношение численности агентов двух типов будет неодинаковым в разных

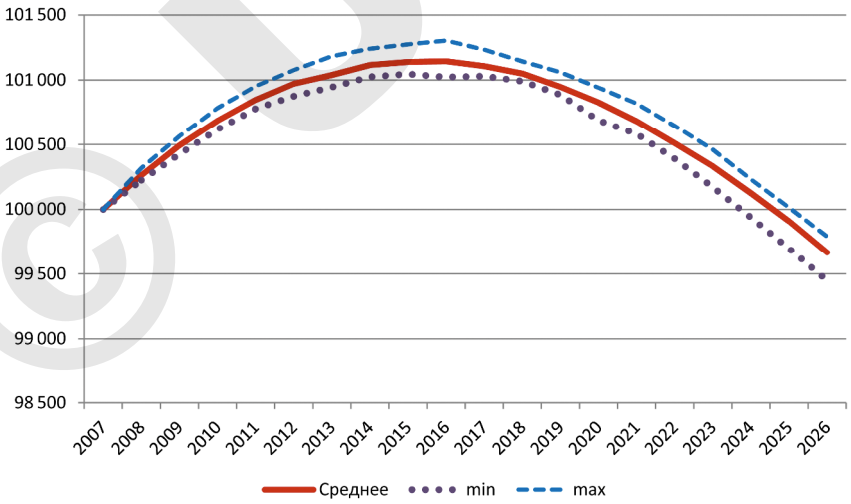


Рис. 2.3.3.1. Динамика численности популяции агентов

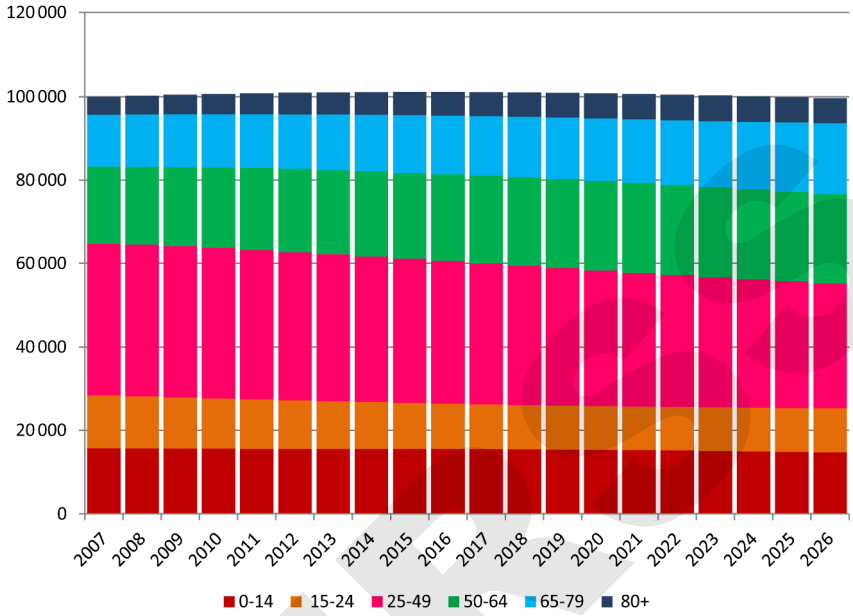


Рис. 2.3.3.2. Динамика возрастной структуры популяции агентов

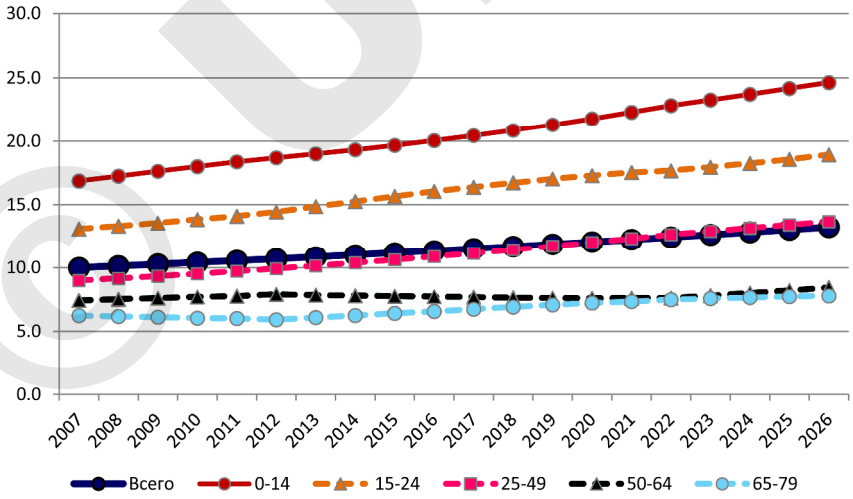


Рис. 2.3.3.3. Доли агентов традиционного типа по популяции агентов в целом и по основным возрастным группам, %

возрастных группах. Этот эффект показан на рис. 2.3.3.3, на котором видно, что при общем небольшом росте доли агентов традиционного типа на каждом шаге моделирования, эта доля, к тому же, заметно различается в разных возрастных группах — она тем выше, чем возрастная группа моложе.

Более того, скорость роста этой доли также тем выше, чем моложе агенты. Так, если в среднем по популяции доля агентов традиционного типа за 20 лет эксперимента выросла на 3,2 %, то в группе самых младших агентов (от 0 до 14 лет) эта доля выросла на 7,7 %, а в следующей возрастной группе агентов (от 15 до 24 лет) — на 5,8 %.

2.3.4. Отличия второй модели

Исходные данные для второй модели те же, что и для первой.

Общая последовательность имитируемых демографических процессов также совпадает со схемой первой модели. С содержательной точки зрения отличие новой модели в более детализированном и реалистичном описании процессов воспроизводства населения. Так, рождению детей предшествует образование семейных пар, которые «обсуждают» число желаемых в будущем детей, при этом процедура поиска партнера полностью контролируется самим агентом и зависит от его принадлежности к тому или иному типу репродуктивного поведения. Это «обсуждение» в модели реализуется в виде обмена сообщениями между агентами, ищущими себе партнера. Взаимодействие агентов при поиске партнеров является именно тем отличием новой модели, которое существенным образом влияет на организацию распараллеливания ее работы для запуска на суперкомпьютере.

Итак, отличия в конструкции основных блоков модели.

1. Новые компоненты агентов и среды

Компоненты агентов:

- место жительства;
- все родственные связи;
- процедура поиска партнера;
- процедура рождения ребенка.

Компоненты среды:

- коллекции одиноких агентов — мужчин и женщин, желающих создать семью;
- модифицированная процедура имитации рождаемости.

2. Установка стартового состояния системы

- Вся территория, на которой расселяются агенты в рабочем окне модели, делится на кварталы, и каждому агенту присваивается его «адрес» — номер квадрата.
- Желаемое число детей присваивается всем агентам, независимо от пола.
- Агенты-дети наделяются обоими родителями.
- Устанавливаются все родственные связи агентов: партнер, родители, дети, братья и сестры.
- Для каждой возрастной когорты в интервале от 15 до 48 лет для женщин (репродуктивный период) и от 18 до 65 лет для мужчин формируются коллекции (списки) одиноких мужчин и одиноких женщин (отдельно), желающих создать семью.
- Агенты из коллекций одиноких женщин пытаются найти одинокого мужчину для образования пары;
- Агенты-женщины репродуктивного возраста, имеющие партнера и еще не родившие столько детей, сколько они планировали, пополняют коллекции женщин, желающих родить ребенка.

3. Действия на шаге работы модели — модифицированная процедура имитации рождаемости

- Рассчитывается общее количество агентов-женщин репродуктивного возраста каждого типа, а также их количество для пятилетних возрастных когорт.
- Рассчитывается общее количество агентов-новорожденных каждого типа с учетом соответствующих общих коэффициентов рождаемости.
- Рассчитывается количество агентов-новорожденных, приходящихся на группы агентов-женщин из пятилетних возрас-

© URSS

тных когорт (с учетом заданного распределения новорожденных по возрасту матерей каждого типа).

- Для каждой возрастной когорты и каждого типа рассчитываются вероятности рождения ребенка у агентов из коллекций женщин, желающих родить.
- Для каждого агента из указанных коллекций вероятностным образом решается его судьба, и если ответ положительный, то выполняется внутренняя процедура рождения ребенка этим агентом.

4. Алгоритм действий агента — процедура поиска партнера

- Для каждой одинокой женщины-агента случайным образом определяется возраст будущего претендента (максимальная разница в возрасте — двадцать лет).
- В коллекции одиноких агентов-мужчин данного возраста и того же типа, что и агент-женщина, находится такой, который проживает в том же квартале и не является ее родственником.
- С этим агентом-претендентом агент-женщина обменивается сообщениями о желаемом числе детей.
- Если разница между этими числами не превышает двух, то образуется пара. В таком случае образуются партнерские связи, родственные связи с уже имевшимися у супругов детьми (если они были). Кроме того, оба партнера удаляются из коллекций одиноких агентов, а агент-женщина добавляется в соответствующую коллекцию женщин, желающих родить (если она еще не родила столько детей, сколько хотела).

С использованием второй модели были повторены те же эксперименты, что и с первой, и были получены аналогичные результаты. Таким образом, можно заключить, что разработанные агентные модели, несмотря на очевидное упрощение действительности, верно воспроизводят как начальное состояние населения условного региона, включая его половозрастную и социальную структуру, так и динамику основных характеристик этого населения.

© URSS

2.3.5. Адаптация моделей для запуска на суперкомпьютере

Учет специфики среды исполнения современных суперкомпьютеров

Далее мы опишем разработанную авторами (совместно со специалистами МГУ — В. А. Васениным, В. А. Рогановым, И. А. Трифоновым) технологию поддержки агентных симуляторов для суперкомпьютеров STARS (Supercomputer Technology for Agent-oRiented Simulation) на примере разработанных демографических моделей.

Важно отметить, что кроме общей схемы распараллеливания, которая определяет способ преобразования исходного кода модели, необходимо удовлетворять базовым требованиям современных суперкомпьютеров, для которых де-факто стандартом является ориентация на MPI. К сожалению, рассчитывать ни на общую память у узлов суперкомпьютера, ни на поддержку стандарта OpenMP, в чем авторы убедились еще несколько лет назад, не приходится. Также не приходится рассчитывать ни на установленную на суперкомпьютер среду моделирования AnyLogic, ни на прочие возможности типа библиотек для работы с файлами в формате MS Excel. Нежелательным является и использование стандартной среды исполнения Java, поскольку она редко присутствует на суперкомпьютере.

Поэтому вся специфика, свойственная исходной среде исполнения AnyLogic и стандартному Java-окружению, должна быть адаптирована к суперкомпьютеру, а взаимодействия между вычислительными узлами основываться на стандарте MPI.

Учет имеющегося опыта и анализ новых программных технологий

Уже опробованная авторами при распараллеливании демографической модели России в 2011 г. библиотека для многоагентного моделирования ADEVS показала себя достаточно хорошо [55]. В последних версиях ADEVS имеется также некоторая поддержка Java, что также является плюсом. Однако разработчики ADEVS до сих пор

не реализовали параллельную работу на суперкомпьютере (за исключением технологии OpenMP для мультипроцессоров, что в предыдущей нашей работе потребовало значительных доработок в части поддержки MPI).

Также при распараллеливании предыдущей достаточно простой модели она была переписана на C++ целиком, что является избыточным: пред- и постобработка данных, а также создание начального состояния многоагентной среды не является критическими по времени операциями. Для суперкомпьютера обычно достаточно распараллелить лишь вычислительное ядро алгоритма, то есть в данном случае — фазу пересчета состояния популяции. В текущей реализации моделей фаза пересчета не использует планировщик AnyLogic, так как требуется обеспечить строго определенную очередность обработки состояния агентов.

Анализ новейших программных технологий показал, что в последнее время активно развиваются встраиваемые средства для исполнения Java программ, которые используют так называемую AOT (Ahead-Of-Time) компиляцию. При этом результатом работы AOT-компилятора является обычный автономный исполняемый модуль, содержащий машинный код для целевой платформы. Интересно отметить, что подобный подход используется в новых версиях операционной системы Android, что, на наш взгляд, не случайно — и для встраиваемых систем, и для суперкомпьютера эффективность исполнения кода является основным фактором. Эксперименты с одним из подобных продуктов — AOT-компилятором Avian — позволили сделать следующие выводы.

1. Avian позволяет получить автономный исполняемый модуль в виде MPI-приложения для суперкомпьютера, при этом на C++ легко реализуется произвольный дополнительный код, включая инициализацию и привязку к коммуникационной библиотеке MPI.
2. Скорость работы полученного программного модуля (к примеру, на классической игре Д. Конвея «Жизнь») приблизительно соответствует скорости работы ADEVS.

Это позволило переложить значительную часть работы на AOT-компилятор и реализовать на C++ только самое необходимое, оставив для ADEVS нишу поддержки ускоренных стадий со сложным межагентным взаимодействием.

Выбор технических решений для распараллеливания модели

В качестве программного средства для организации межузлового взаимодействия выбрана парадигма активных сообщений (мобильных сериализованных объектов). Она широко и с успехом применяется в этом качестве и удобна прежде всего тем, что избавляет программиста от необходимости анализировать поступающие сообщения. Достигается это тем, что сообщения пересылаются в виде сериализованных объектов, при приеме и восстановлении объектного вида которых просто вызывается их метод «handle». То есть о том, что должно происходить при приеме сообщения, заботится исключительно формирующая и посылающая его сторона.

На базе активных сообщений для встраиваемой среды исполнения Avian + MPI был реализован программный каркас, в котором выделенный (первый по счету) узел кластера выдает задания (так называемые стадии расчета) и собирает результаты со всех участвующих (включая его самого) в расчете узлов. Одному шагу моделирования (году, в терминологии текущей модели) может соответствовать несколько стадий, так как при межузловом межагентном взаимодействии может возникнуть потребность в согласовании определенных свойств у агентов (например, согласование желаемого количества детей у партнеров).

Для формата ALP, используемого средой AnyLogic для хранения описания моделей был разработан конвертер, в котором для каждой секции входного файла реализован обработчик, собирающий существенную информацию о модели и генерирующий Java-классы. Все эти обработчики используются конвертером на основе программной специфики модели (то есть набора правил трансляции). Например, инициализация поля главного класса модели Population обрабатывается так, чтобы получить количество агентов

из окружения. Это позволяет задавать это значение в виде опций командной строки.

Эмуляция необходимых функций AnyLogic (взаимодействия; генераторы случайных чисел, обеспечивающие используемые в модели вероятностные распределения; чтение и запись Excel-файлов) была реализована в виде отдельного программного пакета RT (RunTime).

C++ код для работы с MPI и низкоуровневыми примитивами размещается в отдельном программном модуле, производит инициализацию и вызывает входную точку модели, предоставляя некоторые примитивы, а также полезные средства для отладки.

Технология получения и запуска параллельной версии

Исходное, разработанное в среде AnyLogic описание модели представляет собой XML-файл с расширением .ALP (XML-файл в стандарте AnyLogic), в котором содержится описание агентных сущностей модели, их параметры, а также правила пересчета состояния агентов во время эволюции. Кроме ALP-описания к модели прилагается файл с данными в формате Excel, в котором указаны численные параметры, используемые как на стадии формирования населения, так и при пересчете состояния популяции.

Весь процесс создания распараллеленного суперкомпьютерного варианта программы происходит в несколько стадий, которые перечислены ниже.

1. Входной ALP-файл с описанием модели считывается конвертором, который реконструирует объектное представление для всех описанных сущностей и правил, и, проведя необходимую их обработку, формирует набор программных модулей (Java-классов общим объемом в несколько тысяч строк), содержащих всю существенную информацию. При этом переменные агентов преобразуются в поля классов, а правила реакции агентов на события — в соответствующие методы.
2. Excel-файл с входными параметрами модели преобразуется в исходный текст на языке Java (также объемом в несколько ты-

сяч строк), чтобы обеспечить оперативный доступ к параметрам модели на каждом узле суперкомпьютера. Иными словами, внешний набор параметров модели становится составной частью исполняемого модуля для суперкомпьютера.

3. Сформированные программные модули совместно с разработанным кодом эмуляции используемых функций среды AnyLogic компилируются и происходит их компоновка в машинный код для целевого суперкомпьютера. Исполняемый модуль при этом является полностью автономным и при запуске принимает лишь несколько ключевых параметров в командной строке. Например, запуск модели с миллионом агентов для 20 лет на 12 вычислительных узлах производится следующей командой:

```
$ mpirun -np 12 -x Population=1000000 rt.Main
```

Выходные данные при этом собираются с вычислительных узлов в процессе счета, ключевые характеристики популяции и общее потраченное на расчет время сохраняются в журнале, а также печатаются по окончании запуска:

```
[0]: totalNumberPeople = 990221
[0]: * * * Total stages time: 21.173s * * *
```

Сравнительные результаты скорости работы моделей

Для первых экспериментов с параллельной версией упрощенной демографической модели использовался фрагмент лабораторного вычислительного кластера с 12 счетными ядрами и общим объемом оперативной памяти 96 ГБ. При такой конфигурации в оперативной памяти без труда помещаются 60 млн агентов, что позволяет провести моделирование динамики роста населения в масштабах небольшой страны.

Параллельная версия также испытывалась на многоядерном процессоре. Результаты замеров времени работы оригинальной и параллельной версии модели представлены в нижеследующих таблицах (табл. 2.3.5.1–2.3.5.3).

Таблица 2.3.5.1

Оригинальная версия, 1 вычислительное ядро (8 ГБ оперативной памяти)

Агенты	Секунды
25 000	2,5
50 000	4,7
100 000	17,6
200 000	52,7
400 000	194
800 000	766

Таблица 2.3.5.2

Параллельная версия, 4 вычислительных ядра

Ядра	Кварталы	Агенты	Секунды
1	12	1 000 000	210
4	12	1 000 000	50
4	40	1 000 000	30
4	40	2 000 000	75
4	20	4 000 000	344

Таблица 2.3.5.3

Фрагмент вычислительного кластера
(12 счетных ядер Core i7, 96 ГБ оперативной памяти)

Ядра	Кварталы	Агенты	Секунды
12	12	1 000 000	21
12	24	1 000 000	12
12	24	10 000 000	516
12	60	10 000 000	303
12	300	10 000 000	132
12	300	30 000 000	585
12	300	50 000 000	1371
12	300	60 000 000	1833

© URSS

Как видно из приведенных таблиц, скорость работы и вместимость по числу агентов суперкомпьютерной версии значительно выше тех же показателей оригинальной модели. Также видно, что при повышении локальности межагентных взаимодействий (уменьшения площади кварталов) производительность счета повышается.

Таким образом, для моделирования динамики развития населения России или Европы достаточно небольшого Blade-сервера, при этом время одного расчета не превысит нескольких минут.

Для проведения более масштабных экспериментов с использованием разработанной модели потребуется уже более вместительный и производительный суперкомпьютер.

Выводы и перспективы развития разработанной технологии

Главным положительным моментом разработанного подхода к распараллеливанию моделей, разработанных в среде AnyLogic, является автоматизация создания их суперкомпьютерных версий. Это значительно упрощает разработку, так как в большинстве случаев после несущественной модификации исходной модели не требуется доработка правил преобразования в исполняемый модуль для суперкомпьютера.

Данный подход является расширяемым в части используемого целевого языка и программно-аппаратной платформы. Кроме уже успешно опробованных платформ исполнения Avian и ADEVS можно разработать и другие, более низкоуровневые средства для ускорения фазы пересчета состояния агентов, а в перспективе рассмотреть вопрос об использовании таких аппаратных ускорителей, как Xeon Phi и Nvidia CUDA.

Использование более эффективных низкоуровневых средств желательно и для ускорения работы некоторых примитивов. Как выяснилось в процессе разработки, на итоговую скорость работы модели весьма существенное влияние оказывает эффективность используемых генераторов случайных чисел. Поэтому их оптимизация является приоритетной задачей. Перспективным подходом здесь может

быть генерация случайных чисел с нужными параметрами используемых вероятностных распределений в отдельных счетных потоках.

Примененная технология межузлового взаимодействия при помощи активных сообщений позволяет, в случае необходимости, легко реализовать как интерактивное моделирование, так и интерактивную визуализацию процесса моделирования в масштабе расчетного времени. Однако это возможно лишь в случае, если суперкомпьютер будет доступен в монопольном режиме, — например, будет использоваться компактный персональный суперкомпьютер.

Основным вопросом, который сохраняет актуальность, является вопрос о максимально достижимой эффективности распараллеливания в случае массового взаимодействия агентов, находящихся на разных вычислительных узлах суперкомпьютера.

Достаточно очевидно, что если каждый агент активно взаимодействует со всеми остальными агентами, производительность будет низкой в виду объемного межузлового трафика. Тем не менее даже в таком неблагоприятном случае суперкомпьютер позволяет существенно ускорить моделирование, если модель запускается или многократно (для набора статистики), или с разными параметрами. В другом крайнем случае — когда почти все взаимодействия локализованы по географическому расположению агентов — эффективность распараллеливания будет хорошей.

Таким образом, эффективность параллельной версии напрямую зависит от той доли межагентных взаимодействий, которая потребует пересылки больших объемов данных между вычислительными узлами.

2.3.6. Альтернативный способ реализации агентных моделей на суперкомпьютерах

Далее описывается альтернативная реализация системы для проведения многоагентных симуляций, изначально построенная для запуска на суперкомпьютерах и использования большого числа процессорных ядер. Эта реализация совмещает использование языка программирования высокого уровня C# для написания кода

логики модели и низкоуровневого языка C++ для привязки к системной среде обмена сообщениями MPI. Такой подход позволяет минимизировать трудозатраты на разработку агентных моделей для суперкомпьютеров, при этом не происходит заметной потери эффективности работы программы. Использование технологии MPI позволяет запускать программу как на многоядерных рабочих станциях, так и на кластерных суперкомпьютерах, что нереализуемо с использованием технологии многопоточности и OpenMP.

Важным отличием данной модели от технологии STARS является необходимость явного описания всех взаимодействий в системе через посылку сообщений, что требует дополнительных усилий при разработке, но позволяет лучше структурировать программу и делает ее более эффективной.

Описание технологии построения многоагентных симуляций, предназначенных для запуска на суперкомпьютерах

Для проведения детальных многоагентных симуляций в масштабе больших регионов с вычислительной точки зрения модель должна эффективно масштабироваться до 10^9 агентов. Чтобы проводить симуляцию такой системы, программа должна быть адаптирована к запуску на современных суперкомпьютерах. Для упрощения разработки и последующей модификации кода модели был использован язык высокого уровня C# и среда выполнения Microsoft .NET, которая недавно получила поддержку запуска в среде Linux, обычно применяемой на суперкомпьютерах.

Наиболее стандартная конфигурация современного суперкомпьютера — кластер, состоящий из многоядерных серверов (узлов), связанных высокоскоростной сетью с низкими задержками при передаче данных. Для того чтобы эффективно работать на таком суперкомпьютере, программа должна быть разбита на несколько процессов, обменивающихся сообщениями, каждый из которых содержит часть данных решаемой задачи и будет работать на отдельном ядре кластера. Стандартный способ написания таких программ — использование языков программирования C++ или Fortran и библиотеки MPI, доступной на всех суперкомпьютерах.

В качестве платформы для проведения расчетов мы использовали комбинацию базового уровня, написанного с использованием низкоуровневого языка программирования C++ и библиотеки MPI, предоставляемой системой, и платформы Microsoft .NET в качестве среды исполнения основного кода модели, написанного на высокоуровневом языке программирования C#. Большая часть современных суперкомпьютеров работает под управлением операционной системы Linux, поэтому в качестве реализации платформы .NET были использованы Microsoft .NET Core и MONO[1], доступные для этой операционной системы.

Выбор этих технологий определялся следующими критериями.

1. Система должна быть масштабируемой на множество узлов вычислительного кластера (то есть использовать ресурсы нескольких узлов для ускорения счета), поэтому многопоточная модель расчета оказалась неподходящей, так как она ограничена одним узлом кластера.
2. Модель должна быть легкой в разработке и при дальнейшем развитии, поэтому использовался язык программирования высокого уровня C#.
3. Система должна быть эффективной и использовать системную библиотеку MPI вместо стандартного механизма сетевого взаимодействия TCP/IP или библиотек .NET, таких как Windows Communication Foundation, поскольку эти технологии не оптимальны для суперкомпьютеров и приложений HPC. Библиотеки MPI, установленные на каждом кластере, обычно настраиваются на конкретную сетевую систему кластера, такую как Infiniband, и обеспечивают максимальную пропускную способность и минимальные задержки передачи данных. Кроме того, библиотека MPI доступна и для использования на обычных персональных компьютерах, что позволяет применять одну технологию и при разработке, и при проведении расчетов с использованием суперкомпьютеров.

[1] Подробнее об этом на странице: http://www.mono-project.com

На уровне отдельных агентов поддерживаются моделирование эволюции внутреннего состояния агента, формирование постоянных и временных связей между агентами, обмен сообщениями, а также появление и исчезновение агентов в системе. Чтобы эффективно проводить моделирование, системе необходимо реализовать динамический механизм балансировки нагрузки между вычислительными процессами, то есть переносить агентов с более загруженных процессов на другие с сохранением их состояния и связей. Правильное описание и учет связей агентов в процессе декомпозиции задачи имеет решающее значение для уменьшения сетевого обмена, необходимого для линейного масштабирования модели до сотен и тысяч процессоров.

Для балансировки нагрузки между процессами были выбраны алгоритмы METIS/ParMETIS [56], которые обычно используются для декомпозиции больших графов (до 10^9), расчетных сеток и матриц. Эти алгоритмы позволяют разделить граф агентов и связи между ними на части сходного размера с наименьшим возможным числом связей между частями. Алгоритм можно применять рекурсивно для эффективного расчета иерархического распределения графа. Использование алгоритма позволяет строить как начальную декомпозицию (распределение) системы, так и проводить уточнение декомпозиции в процессе расчета, что необходимо для поддержания балансировки нагрузки по мере добавления новых агентов в систему и удаления части старых агентов. Динамическая декомпозиция и перераспределение агентов должны позволять нам эффективно использовать до 1000 процессорных ядер современных суперкомпьютеров.

Реализация платформы для агентного моделирования

Реализация платформы для агентного моделирования требует правильного определения классов для агентов, сообщений, модели, шагов по времени и служебных классов для входных файлов и вычисления характеристик.

Эффективный механизм обмена сообщениями между агентами был реализован посредством очереди сообщений и коллективных операций на основе MPI. Очередь сообщений накапливает буфер сообщений для разных процессов, а затем использует операцию обмена MPI-сообщениями AllToAll («каждый с каждым») для доставки содержимого сообщений на другие узлы кластера. Эта операция доставляет буферы произвольного размера из каждого процесса MPI ко всем другим процессам наиболее эффективным способом, разбивая буфер на блоки оптимального размера для передачи по сети и скрывая латентность сетевых операций, выполняя одновременно несколько операций отправки и получения.

Для того чтобы использовать платформозависимые MPI-операции из изолированной среды исполнения .NET, были написаны операции бинарной сериализации (упаковки данных в массив байт) и десериализации (распаковки) объектов и C# функции-обертки.

Реализация native-обертки для C++ библиотеки выполнялась с использованием библиотеки InteropServices и аннотаций «DllImport» в управляемом C#-коде. На рис. 2.3.6.1 показана схема взаимодействия модельных агентов через native-библиотеку и операцию обмена сообщениями MPI.

Рис. 2.3.6.1. Код агентов модели реализован с использованием языка программирования C# и исполняется на виртуальной машине. Связь между экземплярами виртуальной машины на разных узлах кластера реализована через низкоуровневую библиотеку и вызовы системной библиотеки MPI

Описание тестовой модели

Для того чтобы проверить разработанный механизм доставки сообщений, была реализована тестовая модель и был проведен ряд тестовых симуляций для оценки масштабируемости спроектированной модели. Тестовая модель состояла из одинаковых агентов, обменивающихся сообщениями на каждом шаге по времени. Модель характеризуется следующими параметрами:

- общее число агентов N;
- доля агентов, участвующих в обмене сообщениями S;
- интенсивность обмена сообщениями I;
- общее время моделирования T;
- количество процессов MPI U.

Традиционно для программ, использующих MPI, программа запускается как набор отдельных процессов, выполняющих одну и ту же программу с разными значениями входных параметров. На начальном этапе модели создается исходный набор из N агентов. Каждый агент имеет единственный числовой параметр — количество сообщений, которое он должен отправить на каждом шаге моделирования (A). Множество агентов распределяется по всем процессам MPI, то есть каждый процесс MPI создает только N/M агентов согласно своему номеру процесса. Процесс с индексом 0 создает агентов $0, \ldots, N/M$, процесс с индексом 1 создает агентов $N/M + 1, \ldots, 2 \times N/M$ и т. д.

На каждом шаге моделирования для каждого агента системы генерируется случайное число, которое определяет, будет ли агент участвовать в процессе отправки сообщения (вероятность события равна S/N). Если агент отправляет сообщения на этом этапе моделирования, то случайное число сообщений A генерируется с равномерным распределением вероятности между 1 и I. После этого агент отправляет A сообщений агентам со случайными номерами, а затем получает ответы от них. Каждый шаг моделирования состоит из пяти этапов:

1. Цикл по всем агентам в текущем процессе и вызов метода performStep для каждого агента, в котором генерируется слу-

чайное число сообщений в соответствии с вероятностями отправки сообщений. Все сообщения помещаются в очередь исходящих сообщений.

2. После создания всех исходящих сообщений вызывается метод sendReceiveMessages, который инициирует обмен частями очереди сообщений между всеми MPI-процессами, используя коллективную операцию AllToAll. Каждый процесс отправляет буферы данных всем остальным процессам и получает соответствующие буферы от всех других процессов. На этом этапе сообщения в очереди сериализуются в двоичные массивы, эти массивы передаются в C++-библиотеку, которая использует библиотеку MPI для выполнения обмена, после чего новые буферы передаются в C#-часть программы, и сообщения десериализуются (переводятся в формат C#-объектов).
3. После обмена блоками данных и десериализации всех сообщений выполняется доставка сообщений соответствующим агентам в каждом MPI-процессе, что приводит к генерации нового набора ответных сообщений.
4. Обмен частями очереди сообщений между всеми MPI-процессами, с использованием коллективной операции AllToAll. Каждый процесс отправляет блоки данных всем остальным процессам и получает соответствующие блоки от всех других процессов.
5. Доставка сообщений заданным агентам в каждом MPI-процессе и вызов соответствующих методов агентов.

Использование отложенной доставки сообщений через очередь позволяет оптимизировать обмен сообщениями, который в этом случае ограничен не задержками сети, а ее пропускной способностью.

После обработки всех агентов популяции вычисляются и записываются в текстовый файл следующие выходные характеристики модели:

- номер шага;
- среднее число получателей сообщений для агентов, участвующих в обмене сообщениями;
- общее время расчета.

Процедура обмена сообщениями

Очередь сообщений, реализованная в программе, представляет собой объект C#, который получает объекты абстрактного класса Message от агентов, каждый процесс MPI содержит один экземпляр объекта очереди сообщений. Имитационная модель состоит из агентов разных типов, и сообщения могут также иметь разные типы, наследованные от класса Message. Для каждого типа сообщения реализованы операции чтения и записи в бинарный поток (сериализация и десериализация), которые используют классы BinaryReader и BinaryWriter стандартной библиотеки C#. Эти операции кодируют и декодируют тип объекта сообщения, номера объектов отправителя и получателя и все дополнительные поля данных сообщения в двоичном формате.

Каждая очередь сообщений содержит набор объектов BinaryWriter, которые используются в качестве буферов для исходящих сообщений. Для каждого исходящего сообщения, помещенного в очередь, определяется номер процесса адресата, и сообщение записывается в соответствующий буфер.

Очередь сообщений реализует метод sendReceiveMessages, который выполняет доставку всех сообщений соответствующим агентам. Для доставки сообщений объекты очереди сообщений всех процессов MPI обмениваются соответствующими буферами сообщений, для чего все объекты BinaryWriter записываются в объект MemoryStream, который генерирует исходящий массив байтов. Для доставки исходящего массива байтов сначала используется функция MPI_AllToAll для обмена размерами всех буферов сообщений между всеми процессами. После этого функция MPI_Alltoallv используется для доставки частей исходящего массива байтов в соответствующие процессы.

После обмена буферами сообщений каждая очередь сообщений имеет отдельные буферы с сообщениями от всех других процессов, то есть процесс 0 имеет K_1 байт от процесса 1, K_2 байт от процесса 2 и т. д. Для всех этих буферов создаются объекты BinaryReader и происходит десериализация. Программа считывает сообщения из

всех входящих буферов и генерирует массив объектов типа Message. Для каждого из этих сообщений определяется индекс агента получателя, а для соответствующего агента вызывается метод уведомления с соответствующим объектом сообщения, переданным в качестве аргумента. Процесс доставки сообщения проиллюстрирован на рис. 2.3.6.2.

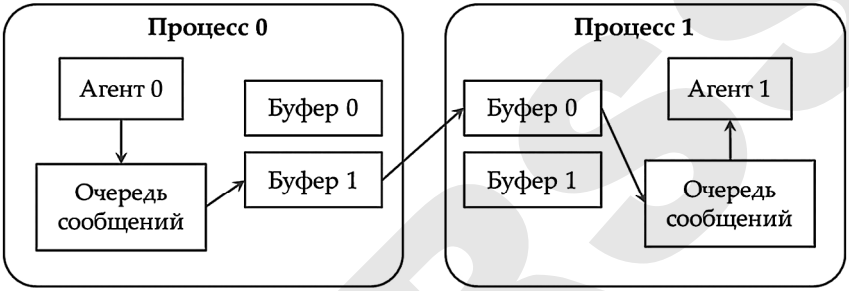


Рис. 2.3.6.2. Процедура отправки сообщения. Сообщение от агента № 0 передается в очередь сообщений процесса № 0, затем сообщение сериализуется и помещается в исходящий буфер для отправки на процессор № 1. После обмена сообщениями этот буфер становится входящим на процессоре № 1, затем сообщение десериализуется и направляется агенту № 1 через очередь сообщений

Численные эксперименты

Для того чтобы оценить влияние параметров модели на параллельную масштабируемость и эффективность, были проведены численные эксперименты. В ходе экспериментов для каждого набора значений параметров модели проводилось несколько расчетов, а затем результаты усреднялись.

Для проведения тестовых расчетов и исследования скорости работы и масштабируемости модели был использован тестовый кластер, состоящий из 4 двухпроцессорных серверов на основе процессоров AMD Opteron 6172 (12 ядер). Полное число ядер в каждом сервере составляло 24, суммарно в кластере было доступно 96 ядер, это же число MPI-процессов использовалось при запуске модели. Сервера в кластере соединялись с использованием высокоскоростной сети QDR Infiniband.

Кроме того, тестирование проводилось также на суперкомпьютере Tianhe-2 (Тяньхэ-2). Суперкомпьютер Тяньхэ-2 (или «Млечный путь-2»), спроектированный Оборонным научно-техническим университетом Народно-освободительной армии Китайской Народной Республики и компанией Inspur, находится в Национальном суперкомпьютерном центре г. Гуанчжоу. Его производительность составляет 34 Пфлопс, а теоретическая пиковая — около 55 Пфлопс и до 2016 г. он являлся самым производительным суперкомпьютером в мире. В ноябре 2016 г. он уступил первое место другому китайскому суперкомпьютеру Sunway TaihuLight, производительность которого составляет приблизительно 93 Пфлопс (пиковая около 125 Пфлопс). С другой стороны, планы развития «Млечного пути-2» предусматривают доведение его производительности до эксафлопсного уровня и вновь занятия лидирующего положения. На данный момент этот суперкомпьютер также из двухпроцессорных узлов, но оборудован 12-ядерными процессорами Intel Xeon E5-2692 (всего 16 000 узлов, то есть 384 000 ядер).

Тестирование модели на этом суперкомпьютере показало схожие параметры масштабируемости, но выявило несовместимость платформы Microsoft .NET Core со старыми версиями ОС Linux. Вместо этого использовалась менее производительная свободная реализация .NET mono (http://www.mono-project.com).

Результаты численных экспериментов

Для изучения параллельной эффективности программы использовался следующий набор параметров:

N = 10 000 000 (число агентов);

S = 200 000 (число агентов, отправляющих сообщения на каждом шаге моделирования);

I = 10 (максимальное количество сообщений для одного агента на каждом шаге);

T = 3000 (количество шагов моделирования).

Моделирование проводилось для числа MPI-процессов M = 1, 24, 48, 96. Запуски с разным числом MPI-процессоров сравнивались

в терминах параллельного ускорения (отношение общего времени вычисления для параллельных и последовательных случаев) и параллельной эффективности (ускорение, деленное на количество использованных ядер процессора).

На рис. 2.3.6.3 показана зависимость ускорения от числа ядер процессора. Увеличение числа ядер процессора приводит к почти линейному ускорению вычислений. Использование 96 ядер приводит к ускорению вычислений в 60 раз, что означает 65 % эффективность использования кластера. На рис. 2.3.6.4 показана зависимость параллельной эффективности от числа ядер процессора.

Для оценки влияния параметров модели на масштабируемость были проведены тестовые симуляции со значениями параметров $N = 10000000$, $S = 200000$, $T = 3000$ с использованием $M = 96$ MPI-процессов при разных значениях интенсивности обмена сообщениями $I = 30, 50, 100, 200, 1000$.

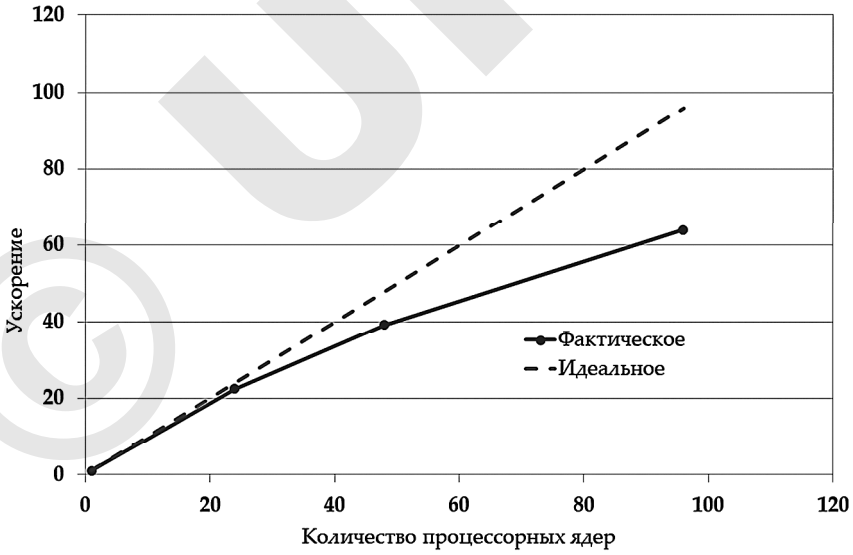


Рис. 2.3.6.3. Зависимость параллельного ускорения тестовой модели от числа используемых процессорных ядер

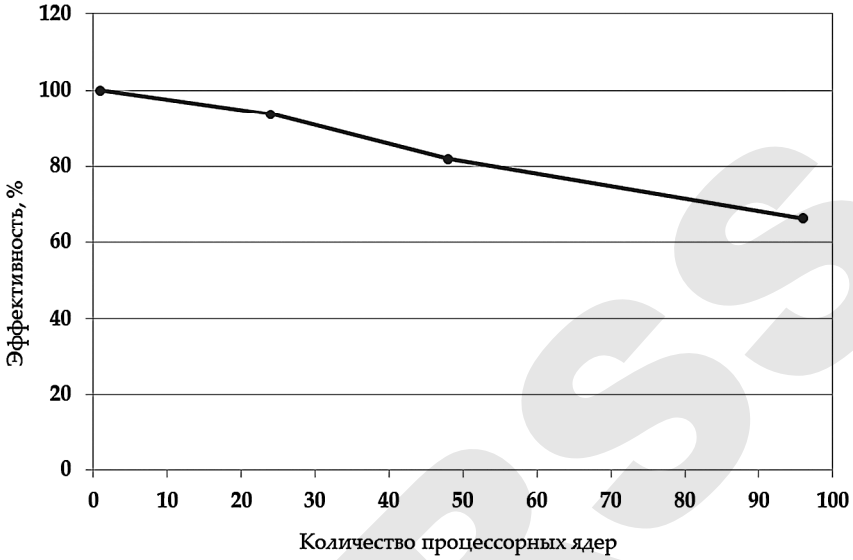


Рис. 2.3.6.4. Зависимость параллельной эффективности программы от числа MPI-процессов

Увеличение интенсивности обмена сообщениями приводит к линейному увеличению сетевого трафика на каждом шаге моделирования, а также к увеличению вычислений (генерации случайных чисел). На рис. 2.3.6.5 зависимость ускорения от интенсивности обмена сообщениями показывает, что эти факторы сбалансированы, а увеличение интенсивности не приводит к ухудшению параллельной эффективности.

Для изучения влияния параметра S параметр I был зафиксирован (I = 10), а расчеты были выполнены для значений S = 300, 500, 1 000 000, 10 000 000. Увеличение параметра S также приводит к линейному увеличению и сетевого трафика и количества вычислений случайных чисел, что суммарно не должно сильно влиять на масштабируемость из-за сбалансированного роста этих параметров. В случае малых значений параметра S размер данных является довольно небольшим, а ускорение определяется больше латентностью сети. Этот эффект приводит к снижению эффективности при

низких значениях *S* и значительно лучшей эффективности при больших значениях параметра.

Следующая серия экспериментов проводилась *для исследования влияния межпроцессорного обмена данными.*

Во втором варианте тестовой модели агенты делились на группы размера *G*, и обмен сообщениями осуществлялся только между агентами внутри группы. Для этого каждый агент получил дополнительный параметр — номер группы N_G, который определялся в начале моделирования для установления равномерного распределения агентов между группами. Использование таких групп соответствует случаю идеальной декомпозиции графа агентов, в котором исчезает большинство связей агентов на разных процессорах.

Для изучения этого эффекта был выполнен набор тестовых расчетов и построена зависимость ускорения от количества участвующих агентов *S*. Результаты этих расчетов показаны на рис. 2.3.6.6. На графике видно, что влияние наличия блоков значительно выше

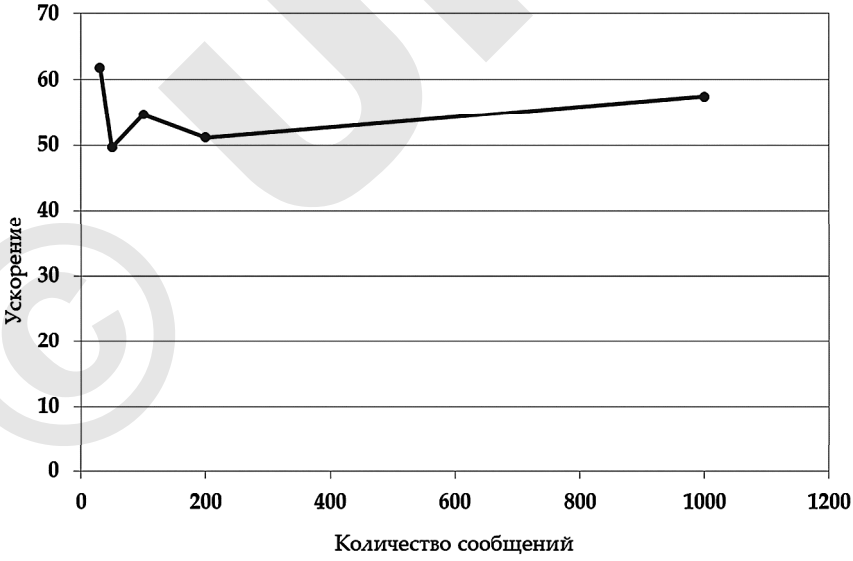


Рис. 2.3.6.5. Влияние интенсивности обмена сообщениями (*I*) на масштабируемость программы

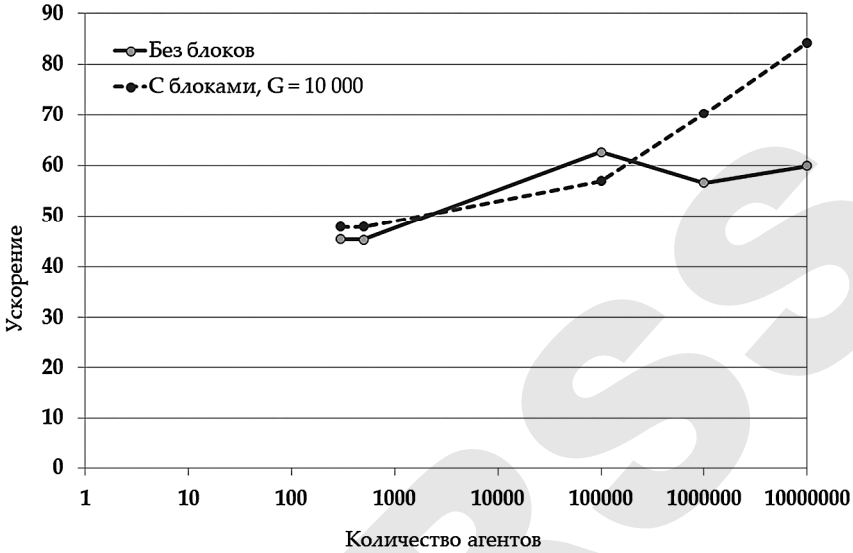


Рис. 2.3.6.6. Зависимость параллельного ускорения от числа агентов, участвующих в обмене сообщениями (*S*). Количество MPI-процессов *M* = 96

при высоких значениях параметра *S,* то есть при высокой доле агентов, участвующих в обмене сообщениями.

Представленная технология агентного моделирования для запуска на суперкомпьютерах была применена при реализации крупномасштабной агентной модели стран Евразии, имитирующей основные процессы движения населения этих стран и их экономики, а также последствия реализации крупных инфраструктурных проектов как результат действий множества самостоятельных агентов.

В модели присутствуют агенты двух типов, образующие иерархию.

а) Страны, которые заинтересованы в расширении экономических связей с другими странами и экономическом росте, чему способствует участие в больших инфраструктурных проектах (например, создание «Нового шелкового пути»). Поэтому страны в модели оценивают выгоду от возможной реализации того или иного инфраструктурного проекта и лоббируют или блокируют его запуск в соответствии со своими интересами.

©URSS

Адекватность имитации этого процесса была подтверждена экспериментально. Так, в ходе экспериментов агентами-странами осуществлялся выбор между двумя маршрутами прохождения «Нового шелкового пути». При этом учитывалось то, что страны входят в различные международные объединения — ЕАЭС, Европейский союз, НАТО — и согласовывают свои позиции внутри этих союзов (то есть интенсивно обмениваются сообщениями внутри соответствующих групп стран). Причем мнение разных стран при выработке общего решения внутри союзов учитывается в модели в соответствии с их экономическим «весом», в роли которого выступает ВВП.

б) Люди — жители этих стран. В модели они могут объединяться в семьи и рожать детей (создавать новых агентов), работать в какой-то отрасли, менять вид деятельности и/или место жительства также в соответствии со своими экономическими интересами. При имитации процесса воспроизводства населения агенты обмениваются сообщениями при поиске партнера для создания семьи. А при имитации процесса миграции обмен сообщениями используется для сбора информации о вакансиях в разных регионах и/или странах Евразии для определения направления миграции. При этом процессы взаимосвязаны, и в них играют роль разные типы связей агента, такие как семейные отношения и пространственное соседство.

Такой подход позволил максимально точно имитировать в модели воспроизводство населения стран Евразии с учетом не только начальной структуры их населения, но и поведенческих особенностей жителей разных стран (например, различные представления о желаемом числе детей в семье). Адекватность имитации миграционных процессов оценивалась на примере трудовой миграции из Китая в Россию. К региональным особенностям, которые были учтены в модели, относятся: а) высокая безработица среди сельского населения Китая; б) образовательная структура населения Китая и в) существенно зависящая от уровня образования тактика сбора информации для принятия миграционного решения. Так, основным источником информации для людей с низким образовательным уровнем являются их близкие родственники и знакомые (в модели соответствующие

агенты обмениваются сообщениями). С использованием модели были поставлены эксперименты по оценке мощности и структуры миграционного потока из китайской провинции Хэйлунцзян в регионы Дальневосточного федерального округа при изменении курса юаня по отношению к рублю и при повышении информированности населения Китая об условиях рынка труда в России.

Исходные условия экспериментов представлены в табл. 2.3.6.1.

Таблица 2.3.6.1

Распределение китайских мигрантов по видам деятельности, а также соотношение их среднемесячной заработной платы в ДФО и в провинции Хэйлунцзян

Отрасли	**Доля занятых, %**	**Соотношение заработной платы при разных значениях курса юаня**		
		6,264	**9,74**	**10,466**
Строительство	8,5	1,86	1,20	1,11
Промышленность	8,5	1,28	0,82	0,76
Сельское хозяйство	8,5	1,60	1,03	0,96
Торговля	59,5	1,84	1,18	1,10
Услуги своим соотечественникам	10,0	1,50	1,20	1,20
Научные исследования	5,0	3,75	3,75	3,75

В ходе эксперимента по варьированию курса валют наблюдалось ожидаемое снижение общего миграционного потока, которое не коснулось группы китайских специалистов, работающих в России. Так, при курсе 9,74 происходило снижение общей численности агентов-мигрантов не менее чем на 10,9 % относительно базового варианта, а при курсе 10,466 — уже более чем на 25 %. Причем в последнем случае мигрантами, в основном, становились безработные [42]. Эксперимент по повышению информированности агентов (когда параметры рынка труда сообщались всем агентам) также продемонстрировал адекватную реакцию модели: при огромной разнице в численности населения рассмотренных регионов происходило активное вовлечение в процессы трудовой миграции из Китая

в Россию сельских жителей, заполнявших вакансии работников низкой квалификации.

Взаимодействие агентов двух уровней реализуется следующим образом: страна создает условия для жизни и работы агентов (ВВП на душу населения, рабочие места и уровень заработной платы в различных отраслях, уровень безопасности, и т. д.) и сообщает агентам-людям эти параметры среды. Для оценки уровня безопасности в модели используются данные глобального индекса терроризма [57].

С началом реализации выбранного варианта инфраструктурного проекта (маршрута) изменяются условия для деятельности людей: появляются новые рабочие места на территориях стран, через которые проходит данный маршрут, а появление новых рабочих мест и изменение уровней зарплат вызывает миграцию агентов-людей. В свою очередь, деятельность людей влияет на экономические показатели отраслей и стран в целом.

Очевидно, что такая иерархия агентов в модели создает дополнительные сложности при распараллеливании ее работы, так как вместо одинаковых по величине условных «кварталов» и соответствующих групп агентов, к ним принадлежащих, появляются страны с очень большой разницей в численности населяющих их агентов-людей. Притом, что между жителями одной страны обмен сообщениями явно интенсивнее.

Сравнение скорости работы агентной модели на суперкомпьютерах Тяньхэ-2 и МВС-100К

Последняя серия расчетов была реализована с целью сравнения скорости работы агентной модели на двух суперкомпьютерах: Тяньхэ-2 и МВС-100К (суперкомпьютер Межведомственного суперкомпьютерного центра РАН).

Расчеты проводились при следующих параметрах:

- полное число агентов — 1 000 000;
- число агентов, посылающих сообщения на каждом шаге — 200 000;
- число сообщений, рассылаемых агентом — 10.

Сравнение проводилось как между кластерами, так и между C++ и C#-версиями программы. В качестве среды исполнения C#-версии программы использовалась среда mono. Эта среда отличается большей кроссплатформенностью и доступна для старых версий ОС Linux, установленных на кластерах.

Результаты приведены в табл. 2.3.6.2 и рис. 2.3.6.7.

Таблица 2.3.6.2

Время расчета итерации двух версий агентной модели для различных кластеров, в секундах

Число использованных процессорных ядер	МВС-100К (mono)	Тяньхэ-2 (mono)	Тяньхэ-2 (C++)
1	1,328		
8	0,352		0,113
16	0,180		0,064
24	0,146	0,078	0,044
48	0,086	0,038	0,035
96	0,076	0,026	

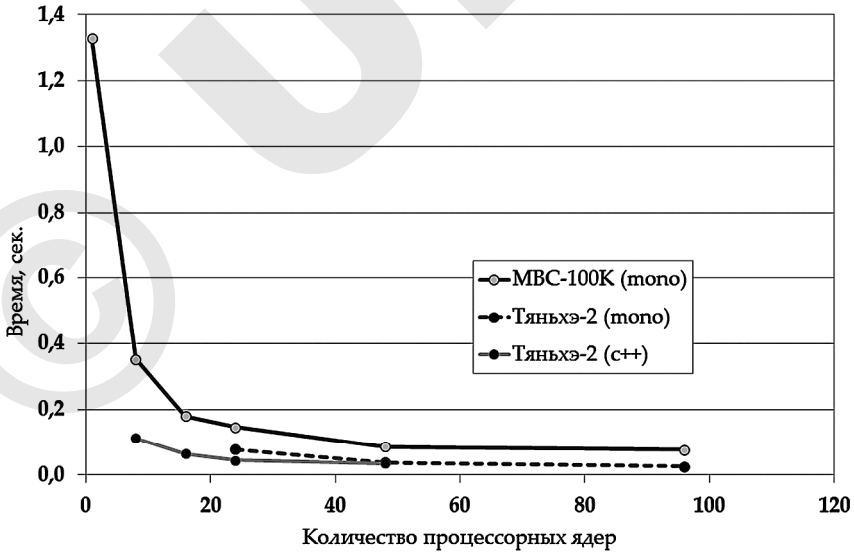


Рис. 2.3.6.7. Время расчета итерации двух версий агентной модели для различных кластеров, в секундах

По результатам можно сказать, что суперкомпьютер Тяньхэ-2 несколько быстрее за счет использования более новых процессоров. Также можно сказать, что C#-версия программы работает примерно в 2 раза медленнее, чем C++-версия, но разница между ними может снижаться при большом количестве процессоров. Возможно, что скорость работы C#-версии удастся повысить за счет использования более быстрой среды исполнения .NET, но это вряд ли будет применимо ко всем доступным кластерам.

2.3.7. Развитие конструкции агент-ориентированной демографической модели

Разработанная модель получала в качестве исходных данные о Федеральной программе материнского капитала по годам, такие как условия ее получения (число детей в семье) и сумма в рублях. В ходе работы модели имитировалось влияние этой программы на вероятность рождения ребенка для тех женщин, которые еще не родили столько детей, сколько желали, и которые не получали материнский капитал ранее. То есть важную роль в имитации процесса рождаемости играла такая характеристика агентов как желаемое число детей в семье, при распределении которой на популяции агентов были использованы данные исследования [58]. Таким образом, если в текущем году сумма материнского капитала была больше нуля, для агентов-женщин, удовлетворявших данным условиям, возрастала рассчитанная для них на текущий год вероятность рождения ребенка, зависящая от их возраста и региона проживания (учитывались региональные тенденции изменения суммарного коэффициента рождаемости). Причем размер прибавки к вероятности рождения ребенка зависел от субъективной значимости для каждого агента-женщины этой суммы, а именно, от величины материнского капитала в стоимости 1 квадратного метра жилья в ее регионе (чем больше значимость, тем больше прибавка).

Для введения в модель имитации влияния региональных программ материнского капитала как единовременной выплаты при рождении ребенка было сделано обобщение данной конструкции и

создан новый класс — «Проект», представляющий в модели региональные меры поддержки семей. При этом было учтено то, что в некоторых регионах для получения поддержки семьи с детьми должны были удовлетворять дополнительному условию на величину среднедушевого дохода. Таким образом, характеристиками класса «Проект» стали:

- индекс региона, в котором вводится мера поддержки семьи;
- индивидуальный номер проекта;
- год начала и год завершения реализации проекта;
- минимальное число детей в семье, дающее право на получение поддержки;
- пороговое соотношение среднедушевого дохода семьи и регионального прожиточного минимума, дающее право на получение поддержки;
- сколько раз семья может получить поддержку;
- суммы поддержки по годам в рублях (с 2002 по 2018 гг.).

Введение в рассмотрение дополнительного условия для получения регионального материнского капитала создало совершенно новую ситуацию, поскольку здесь речь шла о проверке уже не индивидуальных характеристик агента-женщины, претендующей на финансовую помощь, а о рассмотрении характеристик объекта другого типа — семьи, частью которого является агент-женщина. Таким образом, задача потребовала введения в демографическую модель семей как объектов нового типа, главные свойства которого:

- численность семей велика (того же порядка, что и численность популяции агентов-людей);
- семьи и агенты-люди связаны иерархически: агенты-люди являются членами семей;
- семьи, так же, как и агенты-люди: а) пространственно распределены; б) в ходе работы модели могут появляться и исчезать; в) должны поддерживать связи с агентами-людьми в процессе установления и поддержания родственных связей.

С учетом этих свойств и было принято решение о добавлении в модель класса не просто объектов нового типа, а класса агентов нового

© URSS

типа, способных к самостоятельному поведению, причем агенты этих двух типов связаны в модели иерархией: агенты-люди → агенты-семьи. Поскольку класс агентов-семей образован от более общего класса «Агент», он унаследовал от него такие свойства, как:

- индивидуальный адрес (номер пространственной ячейки плюс порядковый номер в ячейке);
- коллекция (список) адресов социальных связей (список рассылки сообщений);
- пометка на удаление.

Кроме того, характеристиками класса «Семья» стали:

- индекс региона, в котором проживает семья;
- суммарный доход;
- коллекция агентов-людей — членов семьи, которая и становится списком рассылки сообщений;
- коллекция программ поддержки, которые семья получала;
- процедура удаления — семья помечается на удаление, если в ней остается только один агент.

Отметим, что понятие семьи в модели отличается от общепринятого понятия домохозяйства и связано это с тем, какие именно семьи могут претендовать на получение материнского капитала. Так, в модели семья не может содержать агентов-родственников трех поколений — семьи образуются агентами, связанными партнерскими (супружескими) отношениями и/или отношениями мать — ребенок (дети), если эти дети не имеют собственных партнеров или детей.

Добавление в модель нового типа агентов повлекло существенные изменения в конструкции демографической агент-ориентированной модели России. Так, в разные классы модели были добавлены:

На главном уровне:

- численность семей;
- процедуры ввода новых исходных данных по регионам (средние зарплата и пенсия, прожиточный минимум);
- процедура ввода данных о региональных проектах материнского капитала;

- в процедуре установке стартового состояния добавлены блоки формирования всех родственных связей между агентами-людьми и образования семей;
- блок стадий, связанных с удалением семей, который выполняется на каждом шаге имитации, соответствующем одному году, после блока стадий, обеспечивающих добавление (рождение) агентов-людей;
- сбор статистики по численности семей.

На уровне региона:

- численность семей;
- средняя заработная плата по годам в рублях (с 2002 по 2018 гг.);
- средняя пенсия по годам в рублях (с 2002 по 2018 гг.);
- прожиточный минимум по годам в рублях (с 2002 по 2018 гг.);
- коллекция всех региональных проектов поддержки семей;
- коллекция действующих в текущий год региональных проектов, которая актуализируется на каждом шаге.

На уровне агентов-людей:

- переменная «доход» (для агентов трудоспособного возраста она рассчитывается исходя из средней заработной платы в регионе, а для агентов старше трудоспособного возраста — исходя из средней пенсии);
- коллекции социальных связей — семьи, партнера, братьев и сестер, а также соответствующее расширение списка рассылки сообщений;
- процедура увеличения вероятности рождения ребенка при наличии действующих проектов в регионе проживания агента-женщины, если она может претендовать на получение помощи — полностью аналогичная механизму влияния федерального материнского капитала;
- процедура поиска партнера для агента-женщины — агента-мужчины, жителя того же региона с разницей в возрасте не более 10 лет, не родственника, у которого нет партнерши;
- процедура установления партнерских отношений и образования новой семьи при появлении первого ребенка.

Внесенные в модель изменения важны с точки зрения требований модели к вычислительным ресурсам. Так, появление популяции нового типа агентов, сопоставимой по численности с популяцией агентов-людей, потребует использования большего числа процессоров. Очевидно также, что должно заметно возрасти время установки стартового состояния модели для воссоздания структуры населения и структуры семейных связей. Кроме того, добавление новых стадий имитируемых процессов (и соответствующих новых точек синхронизации) неизбежно вызовет увеличение времени симуляции каждого шага. Последнее обстоятельство, также как и рост трафика сообщений, которыми обмениваются агенты, в том числе, при образовании семей и установлении родственных связей, может отразиться на эффективности работы суперкомпьютерной модели.

Апробация новой конструкции агент-ориентированной демографической модели

Была рассмотрена динамика полученных ранее результатов моделирования численности населения с учетом влияния федерального материнского капитала и миграционного прироста, а также отклонение полученных значений от данных Росстата за период 2003–2018 гг. [40]. В итоге были выявлены семнадцать регионов, для которых в течение всего периода наблюдалось отклонение прогноза от факта в меньшую сторону, а к 2018 г. модельные данные отставали от фактических более чем на 2 %. Было очевидно, что модель не учитывала влияние некоторых факторов, повышавших в этих регионах рождаемость в указанный период. Очевидно также, что таких факторов может быть множество, нам же для эксперимента важно было выбрать те регионы, в которых можно было проследить влияние именно регионального материнского (семейного) капитала.

Поэтому из полученного перечня регионов мы удалили те, в которых региональные меры поддержки семей не включали программы материнского капитала (например, г. Москва и Республика

Татарстан), а также те, для которых отклонение от фактических значений численности населения увеличивалось равномерно на протяжении всего рассматриваемого периода (то есть отклонение было вызвано постоянно действующими факторами, а не реализацией какого-либо проекта). В итоге для апробации работы новой конструкции модели были выбраны девять регионов, в которых: а) отклонение результатов моделирования от факта заметно возрастало, начиная с 2012 г. и б) были введены региональные программы материнского капитала. Группу таких регионов составили: Ямало-Ненецкий автономный округ (отклонение прогноза от факта в 2018 г. –7,8 %); Ханты-Мансийский автономный округ (–6,8 %); Краснодарский край (–2,9 %); г. Санкт-Петербург (–6,3 %); Тюменская область (–4,9 %); Ставропольский край (–3,2 %); Новосибирская область (–2,4 %); Республика Саха (Якутия) (–2,1 %); Ростовская область (–1,7 %). Особенная ситуация наблюдались в Ставропольском крае. Так, заявленный старт программы регионального материнского капитала с 1 января 2011 г. фактически не состоялся, а в 2015 г. был принят новый Закон № 127-КЗ, который с 1 января 2016 г. и вовсе отменил региональный материнский капитал. Тем не менее, на наш взгляд, анонсированная поддержка семей с детьми могла повлиять на репродуктивное поведение людей, поскольку получить эту поддержку можно было лишь спустя три года с момента появления в семье ребенка (то есть первые выплаты должны были начаться в 2014 г.). Поэтому и данный регион был в списке оставлен, тем более что в указанный период в Ставрополье наблюдался некоторый рост суммарного коэффициента рождаемости (СКР).

Отдельно следует остановиться на ситуации в Ростовской области, так как там право на получение регионального материнского капитала, введенного с 1 января 2012 г., получили при рождении третьего ребенка лишь малоимущие семьи, под которыми понимались семьи, чей среднедушевой доход не превышает величину регионального прожиточного минимума. Что, очевидно, сузило область применения данной меры поддержки семей и должно было снизить эффект от ее введения. Для нас же данный регион пред-

ставляет собой удачный пример, позволивший апробировать более сложную систему стадий имитации, так как здесь семья выступает как единое целое, получающее право на материнский капитал при выполнении заданного условия.

Правомерность выбора данных регионов подтверждается еще и характером изменений в этих регионах суммарного коэффициента рождаемости, динамика медианного для выборки значения которого показана на рис. 2.3.7.1, где виден скачкообразный рост среднего СКР не только в 2007 г. — очевидно, вследствие запуска Федеральной программы материнского капитала, но и в 2012 г., когда стартовало большинство региональных программ.

Таким образом, для апробации работы новой конструкции демографической модели России были выбраны следующие регионы с проектами, соответствующими региональным программами материнского капитала (в табл. 2.3.7.1 представлены только те условия программ, которые использовались в эксперименте).

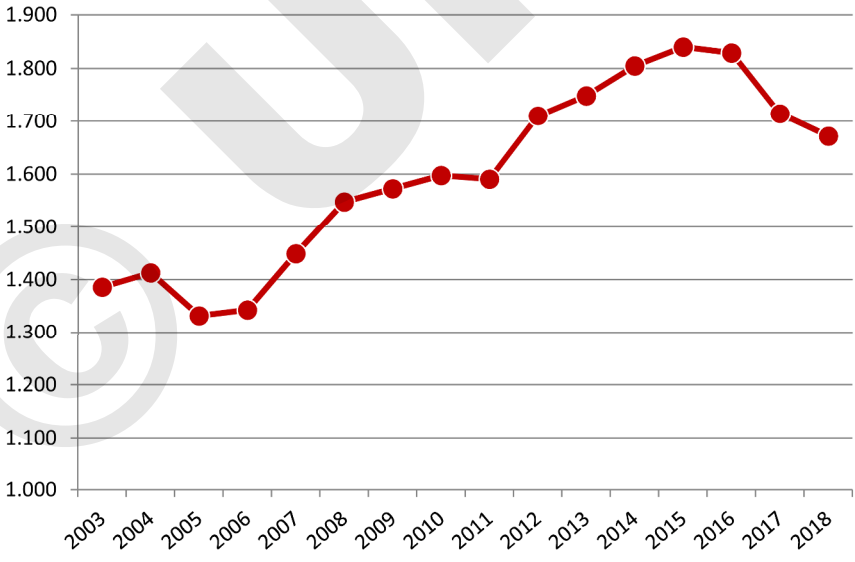


Рис. 2.3.7.1. Динамика медианного по выборке регионов значения суммарного коэффициента рождаемости, данные Росстата

Таблица 2.3.7.1

Параметры региональных программ материнского (семейного) капитала, участвовавших в эксперименте

№ п/п	Регионы	Сумма, тыс. руб.	Период действия, годы	Условия предоставления
1	г. Санкт-Петербург	100 (с ежегодной индексацией)	2012–2026	Рождение 3-го ребенка
2	Краснодарский край	100 (с ежегодной индексацией)	2011–2021	Рождение 3-го ребенка
3	Ставропольский край	100	2011–2015	Рождение 3-го ребенка
4	Ростовская область	100	2012–2018	• Рождение 3-го ребенка; • Среднедушевой доход семьи ниже регионального прожиточного минимума
5	Тюменская область	40	2012–2021	Рождение 3-го ребенка
6	Ханты-Мансийский АО — Югра	116	2012–2021	Рождение 3-го ребенка
7	Ямало-Ненецкий АО	350 (с ежегодной индексацией)	2011–2021	Рождение 3-го ребенка
8	Новосибирская область	100	2012–2021	Рождение 3-го ребенка
9	Республика Саха (Якутия)	100 (с ежегодной индексацией)	2011–2021	Рождение 3-го ребенка

© URSS

Таблица 2.3.7.2

Прирост числа рождений, связанный с реализацией региональных проектов материнского капитала (тыс. человек) и его значимость (доля от населения региона в текущем году, %)

№ п/п	Регионы	Годы								
		2011	**2012**	**2013**	**2014**	**2015**	**2016**	**2017**	**2018**	**Итого**
1	г. Санкт-Петербург		7,7 (0,2)	4,1 (0,1)	5,8 (0,1)	6,9 (0,2)	6,4 (0,2)	6,1 (0,1)	5,7 (0,1)	**42,7**
2	Краснодарский край	17,1 (0,4)	15,7 (0,3)	15,3 (0,3)	12,1 (0,3)	12,0 (0,2)	11,6 (0,2)	10,8 (0,2)	9,0 (0,2)	**103,6**
3	Ставропольский край	18,6 (0,7)	13,5 (0,5)	10,6 (0,4)	6,6 (0,3)	6,3 (0,2)				**55,7**
4	Ростовская область		13,2 (0,3)	10,7 (0,3)	10,0 (0,2)	6,9 (0,2)	6,0 (0,1)	7,4 (0,2)	5,8 (0,1)	**59,9**
5	Тюменская область		5,1 (0,2)	4,3 (0,1)	4,8 (0,1)	6,0 (0,2)	4,9 (0,1)	5,2 (0,2)	1,8 (0,1)	**32,3**
6	Ханты-Мансийский АО — Югра		7,9 (0,5)	5,5 (0,4)	4,2 (0,3)	3,2 (0,2)	4,1 (0,3)	4,1 (0,3)	2,2 (0,1)	**31,2**
7	Ямало-Ненецкий АО	8,5 (1,6)	3,9 (0,7)	5,0 (0,9)	2,4 (0,5)	2,9 (0,5)	2,1 (0,4)	1,3 (0,2)	1,6 (0,3)	**27,8**
8	Новосибирская область		11,2 (0,4)	5,7 (0,2)	4,8 (0,2)	4,6 (0,2)	5,0 (0,2)	4,9 (0,2)	2,7 (0,1)	**39,0**
9	Республика Саха (Якутия)	5,3 (0,5)	4,5 (0,5)	2,7 (0,3)	2,3 (0,2)	2,5 (0,2)	1,6 (0,2)	1,9 (0,2)	1,6 (0,2)	**22,5**
	Всего	**49,5 (0,2)**	**82,8 (0,3)**	**63,8 (0,3)**	**53,0 (0,2)**	**51,5 (0,2)**	**41,7 (0,2)**	**41,8 (0,2)**	**30,4 (0,1)**	**414,5**

Эксперимент проводился по схеме «без проекта и с проектом», то есть все многочисленные параметры модели (такие как год начала и период имитации; параметры смертности и рождаемости и индексы их изменения; параметры Федеральной программы материнского капитала и т. д.), так же, как и процедуры учета миграционных потоков, оставались неизменными и менялся только файл исходной информации по региональным проектам. В базовом варианте региональных проектов не было, а во втором варианте вводились девять представленных выше проектов, после чего для этих регионов сравнивались полученные варианты результатов моделирования. Результаты эксперимента показаны в табл. 2.3.7.2 и 2.3.7.3.

Данные табл. 2.3.7.2 свидетельствуют о том эффекте, который оказало добавление в модель блока, имитирующего реакцию агентов на введение региональных программ материнского капитала. И хотя относительная значимость увеличения числа рождений на каждом шаге имитации в указанных регионах незначительна, но в целом за весь рассматриваемый период абсолютный прирост рождений по регионам выборки составил свыше четырехсот тысяч человек. Рост числа рождений закономерно привел к росту численности населения и, соответственно, к снижению отклонения результатов моделирования от фактических значений этого показателя (см. табл. 2.3.7.3).

Следует отметить те изменения в программном комплексе, связанные с реализацией нового блока модели, которые могли существенно повлиять на эффективность работы суперкомпьютерного варианта модели: а) добавление нового класса агентов-семей и создание соответствующей популяции; б) усложнение процедур установки стартового состояния системы; в) добавление на каждом шаге имитации новых стадий (и соответствующих точек синхронизации, то есть возможных задержек в ходе работы модели); г) существенное увеличение количества связей агентов и, соответственно, существенный рост трафика сообщений между агентами. Все эти изменения в программном комплексе привели к изменению технических характеристик работы суперкомпьютерных вариантов модели.

Таблица 2.3.7.3

Сравнение отклонения результатов моделирования численности населения с учетом миграции от фактических значений для двух вариантов экспериментов: 1 — без региональных проектов и 2 — с реализацией региональных программ материнского капитала, %

№ п/п	Регионы	2012		2013		2014		2015		2016		2017		2018	
		1	2	1	2	1	2	1	2	1	2	1	2	1	2
1	г. Санкт-Петербург	−3,2	−3,2	−3,7	−3,6	−4,2	−4,0	−4,7	−4,5	−5,3	−4,9	−5,8	−5,4	−6,3	−5,7
2	Краснодарский край	−2,1	−1,7	−2,2	−1,5	−2,4	−1,5	−2,6	−1,5	−2,7	−1,5	−2,9	−1,4	−2,9	−1,3
3	Ставропольский край	−2,3	−1,7	−2,4	−1,3	−2,5	−1,1	−2,7	−1,1	−2,9	−1,0	−3,1	−1,3	−3,2	−1,4
4	Ростовская область	−1,3	−1,3	−1,4	−1,4	−1,5	−1,2	−1,6	−1,1	−1,7	−1,0	−1,8	−0,9	−1,7	−0,7
5	Тюменская область	−3,1	−2,9	−3,4	−3,1	−3,7	−3,3	−4,0	−3,5	−4,5	−3,8	−4,8	−4,0	−4,9	−3,9
6	Ханты-Мансийский АО — Югра	−3,9	−3,9	−4,5	−4,2	−5,1	−4,4	−5,7	−4,8	−6,1	−5,0	−6,5	−5,2	−6,8	−5,3
7	Ямало-Ненецкий АО	−4,4	−3,0	−4,9	−2,8	−5,6	−2,7	−6,3	−2,8	−7,1	−3,0	−7,6	−3,2	−7,8	−3,1
8	Новосибирская область	−1,5	−1,5	−1,7	−1,4	−1,8	−1,3	−2,0	−1,3	−2,2	−1,4	−2,4	−1,4	−2,4	−1,1
9	Республика Саха (Якутия)	−0,8	−0,5	−1,0	−0,3	−1,3	−0,2	−1,7	−0,3	−2,0	−0,4	−2,1	−0,3	−2,1	0,0
	Максимальное отклонение	**−4,4**	**−3,9**	**−4,9**	**−4,2**	**−5,6**	**−4,4**	**−6,3**	**−4,8**	**−7,1**	**−5,0**	**−7,6**	**−5,4**	**−7,8**	**−5,7**
	Минимальное отклонение	**−0,8**	**−0,5**	**−1,0**	**−0,3**	**−1,3**	**−0,2**	**−1,6**	**−0,3**	**−1,7**	**−0,4**	**−1,8**	**−0,3**	**−1,7**	**0,0**
	Число регионов с отклонением ≤ 2,0%	**3**	**5**	**3**	**5**	**3**	**5**	**3**	**5**	**2**	**5**	**1**	**5**	**1**	**5**

Так, при сопоставимой суммарной численности агентов значительно (более чем в два раза) возросло среднее время выполнения имитации на шаге, и также вдвое возросло среднее число сообщений, отправляемых на шаге агентами-людьми. Тем не менее общее время имитации продолжает оставаться вполне приемлемым. Например, общее время работы агентной модели на одном процессоре при прогнозировании на 17 лет в первом варианте (без семей) составило при численности агентов 8 млн около 7,7 минут, а во втором варианте (4 млн агентов-людей с добавлением образованных ими агентов-семей) — 16,9 минут.

В новом варианте несколько снизилась и масштабируемость (ускорение работы модели, достигаемое при увеличении числа процессоров суперкомпьютера), хотя и этот важный показатель эффективности параллельной работы модели остается на высоком уровне (ускорение почти в 30 раз для 64 процессоров), что показано на рис. 2.3.7.2.

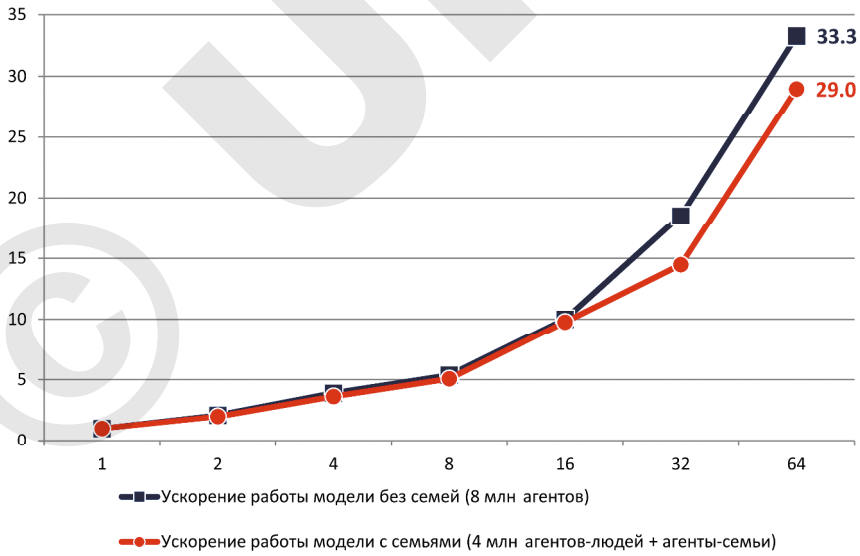


Рис. 2.3.7.2. Сравнение ускорения работы двух вариантов модели, достигаемого при использовании различного числа процессоров суперкомпьютера, раз

* * *

Проведенные эксперименты позволяют сделать следующие выводы.

1. Использованная при разработке демографической агент-ориентированной модели России система проектирования «МЁБИУС» позволяет разрабатывать сложные модели, включающие агентов разных типов, поддерживающих социальные, в том числе иерархические, связи. Система «МЁБИУС» при этом обеспечивает их автоматическую декомпозицию на множестве вычислительных процессоров для эффективного распараллеливания программного кода на суперкомпьютерах.
2. Система «МЁБИУС» обеспечивает синхронизацию реакции отдельных элементов агентной модели благодаря механизму разделения ее работы на стадии, а последующее агрегирование их характеристик и состояний по иерархии позволяет оценить влияние планируемых мер на результирующие показатели, в качестве которых выбираются целевые для экспериментатора.
3. Добавление в модель такой конструкции как «Проекты» позволяет приблизить характер проведения экспериментов к привычным приемам планирования, применяемым практиками, что увеличивает привлекательность модели как инструмента подготовки управленческих решений.

Первоочередными направлениями дальнейшего развития представленного программного комплекса нам видятся: а) введение в рассмотрение других видов проектов — ежемесячных денежных выплат, программ дополнительного образования для детей и др., которые, как показывают проведенные социологические исследования [58], значимы для принятия молодыми семьями решения о рождении ребенка; и б) разработка механизмов имитации миграции агентов как их осознанного выбора с учетом показателей качества жизни в различных регионах и их значимости для агентов. Ключевым субъектом и в том, и в другом случае должна быть семья.

2.4. Применение цифровых моделей для оценки стратегий социального развития

В рамках программы Президиума РАН «Механизмы обеспечения отказоустойчивости современных высокопроизводительных и высоконадежных вычислений» были разработаны и апробированы несколько агент-ориентированных моделей, построенных, в том числе, с использованием геоинформационных и суперкомпьютерных технологий.

1. Демографические агентные модели нескольких регионов (Москвы, Санкт-Петербурга, Вологодской области и др.) и страны в целом. Эти модели позволяют прогнозировать изменения возрастно-половой структуры населения соответствующих субъектов с учетом его пространственного размещения [4].
2. Отраслевая модель, реализованная на примере ракетно-космической промышленности и предназначенная для оценки мер государственной политики в промышленной сфере, направленных на реализацию крупных проектов и связанных с корректировкой соответствующих финансовых затрат [59]. Также модель может использоваться для оценки экономических ограничений и санкций, направленных против рассматриваемой отрасли.
3. Социо-эколого-экономическая агентная модель региона, действующими акторами которой является агенты двух видов: люди и предприятия. Первый тип агентов определяет демографическую ситуацию, а также часть представителей этого вида связана с агентами-предприятиями, являясь их работниками, создающими продукт соответствующей отрасли. Второй тип агентов, помимо производства продукции, осуществляет выбросы (как в атмосферу, так и в водные объекты), тем самым оказывая негативное влияние на экологию региона и, как следствие, на показатели агентов первого типа, связанных со здоровьем и работоспособностью. В процессе проведения компьютерных экспериментов пользователь может менять параметры механизмов регулирования выбросов (имитация экологических требований со стороны регулирующих органов), а агенты модели, оперируя доступными для них действиями, могут сокращать объемы своих выбросов. Инст-

© URSS

румент позволяет получать средне- и долгосрочные прогнозы изменения уровня загрязнений с учетом пространственного размещения производства и стационарных источников выбросов [43].

4. Агент-ориентированная региональная модель «Губернатор», являющаяся своего рода отработанным фреймворком для построения региональных моделей, позволяет имитировать социально-экономические системы соответствующих территорий и предоставляет возможность управлять бюджетным процессом моделируемого субъекта России [60].
5. Комплексная агент-ориентированная модель, построенная с использованием геоинформационных технологий и имитирующая социально-экономические системы нескольких стран Европейского союза. В модели рассматриваются агенты различного уровня (люди, отрасли, страны), что позволяет оценивать возникающие в процессе компьютерных симуляций эффекты во многих сферах социально-экономических систем соответствующих государств. Агенты нижнего уровня определяются широким набор параметров (помимо половозрастных характеристик учитывается уровень образования, здоровья, квалификации, жизненные приоритеты и др.), что, в свою очередь, дает возможность оценивать демографические изменения в Европейском союзе с учетом важных факторов, таких как внутренние установки людей, влияющие на их стратегию репродуктивного поведения, а также миграционный приток извне. Данная модель разработана в рамках исследования, осуществляемого совместно с Международным институтом прикладного системного анализа (International Institute for Applied System Analysis) [61].
6. Агентная модель, имитирующая организации различного вида, отличающиеся по набору характеристик, одной из которых является тип корпоративной культуры. Модель позволяет рассчитать последствия слияний рассматриваемых организаций с позиции уровня удовлетворенности их сотрудников и возможных действий по [62].
7. Модель поведения человека (включая девиантное/противоправное) в зависимости от структуры его личности и условий внешней

среды, учитывающая различные жизненные цели человека, их значимость, а также критерии оценки достижения поставленных целей. Позволяет получать оценки изменения уровня недовольства населения и отдельных его групп с учетом их пространственного размещения. Эта модель разрабатывалась в рамках Программы фундаментальных научных исследований Президиума РАН «Фундаментальные исследования по проблеме экономической безопасности», реализуемой совместно с Институтом законодательства и сравнительного правоведения при Правительстве РФ и Федеральной службой по финансовому мониторингу [63].

8. Агент-ориентированная модель, воспроизводящая процесс трудовой миграции из Китая в Россию и рассматривающая большой набор программных классов (агенты-люди, предприятия, отрасли и регионы), широкой перечень их параметров и, самое главное, — каналы распространения информации между агентами нижнего уровня, использующими родственные связи и сеть знакомых для получения сведений относительно условий проживания и работы в России. В модели учтены как объективные факторы миграции — заработная плата, наличие рабочих мест и т. д., так и побуждающие к миграции субъективные установки, влияющие на конечное решение и основывающиеся на личностных характеристиках агентов [42].
9. Модель для прогнозирования эпидемиологической динамики, в которой в явном виде учитываются: а) особенности протекания конкретного инфекционного заболевания, вызвавшего эпидемию; б) неоднородность населения региона с точки зрения его восприимчивости к инфекции; и в) социальные связи отдельных людей, существенным образом влияющие на частоту контактов и как следствие — на вероятность передачи инфекции. Особенностью конструкции данной модели, отличающей ее от других известных моделей эпидемий, является использование механизма формирования семей, что делает имитацию контактов на уровне отдельного агента максимально приближенной к реальности. Модель апробирована на примере эпидемии COVID-19 в г. Москве, но используемые в модели социально-демографическая

структура населения и эпидемиологические характеристики конкретной инфекции являются параметрами модели, что делает ее достаточно универсальной и позволяет производить ее настройку на особенности других инфекций и других регионов [64].

Описанные в монографии модели являются лишь маленькой частью огромного количества научно-практических наработок в сфере агент-ориентированного моделирования, но и они наглядно показали, что в мире все чаще возникают проблемы, которые трудно решить, используя только традиционные математические методы. Так, процессы, в которых важно пространственное размещение взаимодействующих индивидов, а также дифференциация последних по широкому набору параметров естественным образом обуславливает применение агентных моделей.

Вместе с тем, по нашему опыту отметим, что не слишком сложная модель этого класса включает около 10–20 функций, определяющих поведение каждого индивида и около 10 базовых параметров. Соответственно, крупные модели, включающие в себя миллионы агентов и большое количество единиц модельного времени, в принципе не могут быть решены аналитическим способом. Их решением является усредненный результат по множеству прогонов, что само по себе, согласно утверждению Стивена Вольфрама, создает приращение знаний.

Одна из целей монографии — подведение промежуточных итогов в развитии искусственных обществ и придание дополнительного импульса к дальнейшему развитию принципиально нового инструментария, который позволит решать сложнейшие задачи современности. Причем это не обязательно должны быть агентные модели, а, возможно, нечто совершенно другое, поскольку столь динамичное развитие информационных технологий и их применение (в том числе приведенные примеры) дает основания предполагать появление новых возможностей обработки больших объемов данных. Каждый год различные издания опрашивают специалистов ИКТ-сферы по поводу прогнозов возникновения сильного искусственного интеллекта, который может привести как к стабильному безкризисному развитию человеческого общества, так и к технологической сингулярности с непредсказуемыми последствиями.

Литература к главе 2

1. *Макаров В. Л.*, *Ву Ц., Ву З., Хабриев Б. Р.*, *Бахтизин А. Р.* Современные инструменты оценки последствий мировых торговых войн // Вестник Российской академии наук. 2019. Т. 89. № 7. С. 745–754.
2. *Макаров В. Л.*, *Ву Ц., Ву З., Хабриев Б. Р.*, *Бахтизин А. Р.* Мировые торговые войны: сценарные расчет последствий // Вестник Российской академии наук. 2020. Т. 90. № 2. С. 169–179, DOI: 10.31857/S0869587320020097.
3. *Бахтизин А. Р.* Агент-ориентированные модели экономики. М.: Экономика, 2008.
4. *Макаров В. Л.*, *Бахтизин А. Р.* Социальное моделирование — новый компьютерный прорыв (агент-ориентированные модели). М.: Экономика, 2013. 295 с.
5. *Gardner M.* The fantastic combinations of John Conway's new solitaire game «life» // Scientific American. 1970. October. № 4.
6. *Schelling T. C.* Dynamic Models of Segregation // Journal of Mathematical Sociology. 1971. 1 (2). P. 143–186.
7. *Epstein J. M.*, *Axtell R.* Growing artificial societies: Social science from the bottom up. Brookings Institution Press; The MIT Press, 1996
8. *Epstein J. M.* Generative Social Science: Studies in Agent-Based Computational Modeling. Princeton University Press, 2006. 356 p.
9. *Epstein J. M.* Agent_Zero: Toward Neurocognitive Foundations for Generative Social Science. Princeton University Press, 2013. 249 p.
10. *Bisset K. R., Chen Jiangzhuo, Feng Xizhou*, *Anil Kumar V. S. Marathe M. V.* EpiFast: A Fast Algorithm for Large Scale Realistic Epidemic Simulations on Distributed Memory Systems / Department of Computer Science and Virginia Bioinformatics Institute Virginia Tech. Blacksburg, USA, 2009.
11. *Cockrell R. C., Christley S., Chang E., An G.* Towards Anatomic Scale Agent-Based Modeling with a Massively Parallel Spatially Explicit General-Purpose Model of Enteric Tissue (SEGMEnT_HPC) // PLoS ONE. 2015. 10 (3): e0122192. doi:10.1371/journal.pone.0122192.
12. *Shoham Y.* Agent Oriented Programming: Technical Report. Computer Science Department, Stanford University, 1990.
13. Microsoft Corporation: Axum Programmer's Guide. 2009.
14. *Gustafsson N.* Axum language overview (v0.8). 2009. Jun. 16.
15. *Zia K.*, *Riener A.*, *Farrahi K.*, *Ferscha A.* An Agent-Based Parallel Geo-Simulation of Urban Mobility during City-scale Evacuation. Simulation. 2013. DOI: 10.1177/0037549713485468.
16. *Auld J.*, *Hope M.*, *Ley H.*, *Sokolov V.*, *Xua B.*, *Zhang K.* POLARIS: Agent-based modeling framework development and implementation for integrated

travel demand and network and operations simulations // Transportation Research Part C: Emerging Technologies. 2016. V. 64. March. P. 101–116.
17. *Lubaś R.*, *Wąs J.*, *Porzycki J.* Cellular Automata as the basis of effective and realistic agent-based models of crowd behavior // The Journal of Supercomputing. 2016. V. 72. 6. P. 2170–2196. URL: https://doi.org/10.1007/s11227–016–1718–7
18. *Cordasco G.*, *Scarano V.*, *Spagnuolo C.* Distributed MASON: A scalable distributed multi-agent simulation environment // Simulation Modelling Practice and Theory. 2018. V. 89. P. 15–34. ISSN 1569–190X. URL: https://doi.org/10.1016/j.simpat.2018.09.002
19. *Borges F.*, *Gutierrez-Milla A.*, *Luque E.*, *Suppi R.* Care HPS: A high performance simulation tool for parallel and distributed agent-based modeling // Future Generation Computer Systems. 2017. V. 68. P. 59–73. ISSN 0167–739X. URL: https://doi.org/10.1016/j.future.2016.08.015
20. *Glimcher P. W., Fehr E.* (eds.). Neuroeconomics: Decision Making and the Brain: Second Edition, 2013. P. 1–577.
21. *Deissenberg C., van der Hoog S., Dawid H.* EURACE: A Massively Parallel Agent-Based Model of the European Economy // Applied Mathematics and Computation. 2008. 204. P. 541–552.
22. *Raberto M.*, *Cincotti S.*, *Teglio A.* Economic Policy and the Financial Crisis. Routledge Frontiers of Political Economy. Ch. 9. Taylor & Francis, 2014.
23. *Teglio A., Mazzocchetti A., Ponta L., Raberto M., Cincotti S.* Budgetary rigour with stimulus in lean times: Policy advices from an agent-based model. Working Papers 2015/04, Economics Department, Universitat Jaume I, Castellón (Spain). https://ideas.repec.org/p/jau/wpaper/2015–07.html.
24. *Luhmann N.* Die Wirtschaft der Gesellschaft. Suhrkamp. 1988.
25. *Fleischmann A.* A Model for a Simple Luhmann Economy // Journal of Artificial Societies and Social Simulation. 2005. V. 8. № 2. URL: https://www.jasss.org/8/2/4.html
26. *LeBaron B.* Agent-based computational finance // Handbook of computational economics. 2006. 2. P. 1187–1233.
27. *LeBaron B.* Microconsistency in Simple Empirical Agent-Based Financial Models // Computational Economics. 2019. 1–19. DOI: 10.1007/s10614–019–09917–8.
28. *Axtell R. L.* 120 Million Agents Self-Organize into 6 Million Firms: A Model of the U.S. Private Sector // Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems (AAMAS'16). Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2016. P. 806–816.
29. *Axtell R. L.* Zipf distribution of U.S. Firm Sizes // Science. 2001. 293 (5536). P. 1818–1820.

30. *Axtell R. L.* Endogenous Dynamics of Multi-Agent Firms // SSRN Electronic Journal. 2015. DOI: 10.2139/ssrn.2827059.
31. *Yakovenko V. M.*, *Rosser J. B.* Colloquium: Statistical mechanics of money, wealth, and income // Rev. Mod. Phys. 2009. 81. 1703. Published 2 December 2009.
32. *Peckham R.* Economies of contagion: financial crisis and pandemic // Economy and Society. 2013. 42: 2. P. 226–248. DOI: 10.1080/03085147.2012.718626.
33. *Tesfatsion L.* Agent-Based Computational Economics: Modelling Economies as Complex Adaptive Systems. 2002. URL: http://www.econ.iastate.edu/tesfatsi
34. *Kirman A. P.* Whom or what does the representative individual represent? // Journal of economic perspectives. 1992. 6 (2). P. 117–136.
35. *Hommes C.*, *LeBaron B.* (eds.). Computational Economics: Heterogeneous Agent Modeling. Elsevier, 2018.
36. *Farmer J.*, *Foley D.* The economy needs agent-based modelling // Nature. 2009. 460. P. 685–686. URL: https://doi.org/10.1038/460685a
37. *Rand W.*, *Rust R. T.* Agent-based modeling in marketing: Guidelines for rigor // International Journal of Research in Marketing. 2011. V. 28. 3. September. P. 181–193. DOI: 10.1016/j.ijresmar.2011.04.002
38. *Halaška M.*, *Šperka R.* Is there a Need for Agent-based Modelling and Simulation in Business Process Management? // Organizacija. 2018. 51 (4). P. 255–269. URL: https://doi.org/10.2478/orga-2018–0019
39. *Бахтизин А. Р.*, *Ильин Н. И.*, *Макаров В. Л.*, *Сушко Е. Д.*, *Хабриев Б. Р.* Программно-аналитический комплекс «МЁБИУС» — инструмент планирования, мониторинга и прогнозирования социально-экономической системы России // Искусственные общества. 2020. Т. 15. Вып. 4. URL: https://artsoc.jes.su/s207751800012303–2–1. DOI: 10.18254/S207751800012303–2.
40. *Бахтизин А. Р.*, *Макаров В. Л.*, *Максаков А. А.*, *Сушко Е. Д.* Демографическая агент-ориентированная модель России и оценка ее применимости для решения практических управленческих задач // Искусственные общества. 2021. Т. 16. Вып. 2 URL: https://artsoc.jes.su/s207751800015357–1–1. DOI: 10.18254/S207751800015357–1.
41. *Макаров В. Л.*, *Бахтизин А. Р.*, *Бекларян Г. Л.*, *Акопов А. С.* Разработка программной платформы для крупномасштабного агент-ориентированного моделирования сложных социальных систем // Программная инженерия. 2019. № 4. С. 167–177. DOI: 10.17587/prin.10.167–177.
42. *Макаров В. Л.*, *Бахтизин А. Р.*, *Сушко Е. Д.*, *Агеева А. Ф.* Агент-ориентированный подход при моделировании трудовой миграции из Китая в Россию // Экономика региона. 2017. Т. 13. Вып. 5. С. 331–341.
43. *Макаров В. Л.*, *Бахтизин А. Р.*, *Сушко Е. Д.* Агент-ориентированная модель как инструмент регулирования экологии региона // Журнал

Новой экономической ассоциации. 2020. № 1 (45). С. 151–171. DOI: 10.31737/2221–2264–2020–45–1–6.

44. *Макаров В. Л.*, *Бахтизин А. Р.*, *Хабриев Б. Р.* Оценка эффективности механизмов укрепления государственного суверенитета России // Финансы: теория и практика / Finance: Theory and Practice. 2018. 22 (5): 6–26. URL: https://doi.org/10.26794/2587–5671–2018–22–5–6–26
45. *Diaz B. A.* Agent-Based Models on Social Interaction and Demographic Behaviour (Ph.D. Thesis). Wien: Technische Universität, 2010. 85.
46. *Stillwell J., Clarke M.* (eds.). Population Dynamics and Projection Methods // Understanding Population Trends and Processes. 2011. V. 4. 222. DOI: 10.1007/978–90–481–8930–4.
47. *Billari F. C., Prskawetz A.* (eds.). Agent-Based Computational Demography: Using Simulation to Improve Our Understanding of Demographic Behaviour. Heidelberg: Springer, 2003. 210.
48. *Wu B. M., Birkin M. H.* Agent-Based Extensions to a Spatial Microsimulation Model of Demographic Change // Heppenstall A. J. et al. (eds.). Agent-Based Models of Geographical Systems. 2012. P. 347–360. DOI: 10.1007/978–90–481–8927–4–16.
49. *Вишневский А. Г.* Воспроизводство населения и общество: История, современность, взгляд в будущее. М.: Финансы и статистика, 1982. 287 с.
50. Родители и дети, мужчины и женщины в семье и обществе / Под науч. ред. Т. М. Малевой, О. В. Синявской; Независимый институт социальной политики. М.: НИСП, 2007. 640 с.
51. *Шубат О. М.*, *Багирова А. П.* Прогнозирование вторых рождений у российских женщин: социолого-статистический подход // Проблемы прогнозирования. 2014. № 3. С. 131–140.
52. *Тарасов В. Б.* От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. М.: URSS, 2002. 352 с.
53. *Бахметова Г. Ш.* Методы демографического прогнозирования. М.: Финансы и статистика, 1982. 159 с.
54. *Шахотько Л. П.*, *Терещенко С. М.* Компьютерное решение задачи построения демографических прогнозов // Вопросы статистики. 1999. № 10. С. 57–65.
55. *Макаров В. Л.*, *Бахтизин А. Р.*, *Васенин В. А.*, *Роганов В. А.*, *Трифонов И. А.* Средства суперкомпьютерных систем для работы с агент-ориентированными моделями // Программная инженерия. 2011. № 3.
56. *Karypis G.*, *Kumar V.* METIS-unstructured graph partitioning and sparse matrix ordering system, version 2.0. 1995. URL: http://dm.kaist.ac.kr/kse625/resources/metis.pdf

57. The Global Terrorism Index 2016 // Institute for Economics and Peace. URL: http://economicsandpeace.org/wp-content/uploads/2016/11/Global-Terrorism-Index-2016.2.pdf
58. *Шабунова А. А.*, *Ростовская Т. К.* О необходимости разработки модели оптимальных условий для формирования и реализации демографических установок // Экономические и социальные перемены: факты, тенденции, прогноз. 2020. Т. 13. № 4. С. 38–57. DOI: 10.15838/esc.2020.4.70.2.
59. *Макаров В. Л.*, *Бахтизин А. Р.*, *Сушко Е. Д.* Управление деятельностью отрасли в агент-ориентированной модели на примере госкорпорации «Роскосмос» // Теория и практика институциональных преобразований в России / Сборник научных трудов под ред. Б. А. Ерзнкяна. Вып. 35. М.: ЦЭМИ РАН, 2016. С. 20–29.
60. *Макаров В. Л.*, *Бахтизин А. Р.*, *Сушко Е. Д.* Компьютерное моделирование взаимодействия между муниципалитетами, регионами, органами государственного управления // Проблемы управления. 2013. № 6. С. 31–40.
61. *Макаров В. Л.*, *Бахтизин А. Р.*, *Бекларян Г. Л.*, *Акопов А. С.*, *Ровенская Е. А.*, *Стрелковский Н. В.* Укрупненная агент-ориентированная имитационная модель миграционных потоков стран Европейского Союза // Экономика и математические методы 2019. Т. 55. № 1. С. 3–15. URL: https://emm.jes.su/S042473880004044–7–1. DOI: 10.31857/S042473880004044–7.
62. *Денисова С. В.*, *Бахтизин А. Р.* Моделирование процесса слияний организаций с помощью агент-ориентированной модели // Проблемный анализ и государственно-управленческое проектирование. 2011. № 2.
63. *Макаров В. Л.*, *Бахтизин А. Р.*, *Сушко Е. Д.*, *Бурилина М. А.* Разработка математической модели для количественной оценки последствий незаконных финансовых операций // Управленческое консультирование. 2016. № 3. С. 85–96.
64. *Макаров В. Л.*, *Бахтизин А. Р.*, *Сушко Е. Д.*, *Агеева А. Ф.* Моделирование эпидемии COVID-19 — преимущества агент-ориентированного подхода // Экономические и социальные перемены: факты, тенденции, прогноз. 2020. Т. 13. № 4. С. 58–73. DOI: 10.15838/esc.2020.4.70.3.

Глава 3
Методическое обеспечение исследования качества жизни в цифровой экономике

3.1. Качество жизни как сфера применения стандартизации, метрологии и управления качеством в условиях цифровизации

Цифровизация, — в таких ее проявлениях, как всеобщая компьютеризация, сетевые и облачные технологии, большие данные и многое другое, — неузнаваемо изменила модель потребления, его технологическую основу. Проектирование и теоретическое изучение стандартных технологий составляют основу для создания математических описаний перспективных технологий, предопределяют развитие соответствующей группы технологий.

Эволюция технологий, связанная с типизацией процессов и автоматизацией проектирования, сопровождается появлением особых требований к материально-техническому, кадровому обеспечению сложных технологических процессов. Последующая реализация технологий требует детализации их описания. При этом конкретизируются не только последовательности действий и операций, но и числовые характеристики процессов, возникает необходимость в метрологическом обеспечении. В результате разрабатывается технологическая документация, содержащая информацию о характеристиках и точности измерительных систем, сроках поверки измерительного оборудования и т. д. Теоретически стало возможным создание технологий, которые могли бы работать без вмешательства человека неограниченное количество времени.

В таких условиях особого внимания заслуживает применение научного подхода, сформированного на основе экономики качества,

неоднократно и убедительно доказавшего свою надежность применительно к решению сложных социально-экономических задач [1].

Экономика качества задействует обширный комплекс научных дисциплин применительно к объекту и уровню исследования (рис. 3.1.1).

Рис. 3.1.1. Применение наук в исследовании качества

Для изучения качества продукции (1-й уровень) применяются фундаментальные науки, прежде всего такие, как философия, математика, физика, химия, метрология. Оценка качества процессов (2-й уровень), качества работы предприятия, всех его подразделений и служб (3-й уровень), качества среды обитания (4-й уровень) и, наконец, качества жизни (5-й уровень) требует применения всего арсенала научных дисциплин [2, 3].

Основу концепции экономики качества с момента ее возникновения составляют метрология, стандартизация и управление

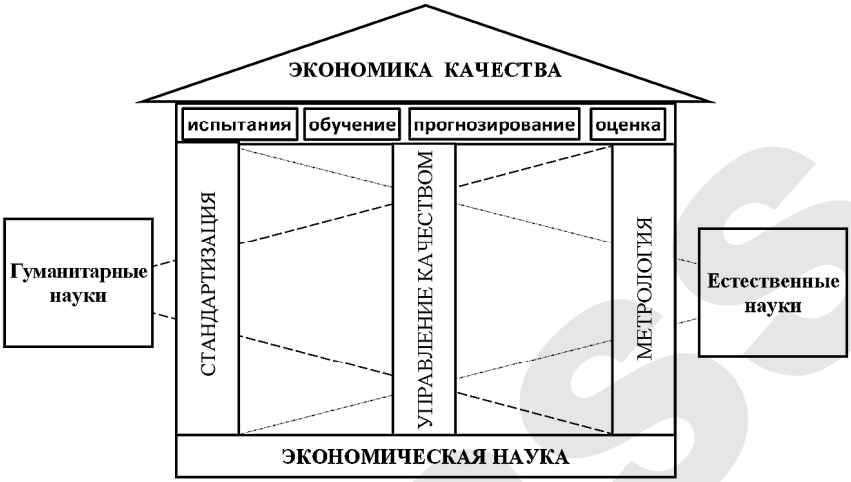


Рис. 3.1.2. Экономика качества в совокупности базовых элементов

качеством. Система взглядов, образованная экономикой качества, выявляет все новые грани качества, которые открываются сквозь призму метрологии (как науки об измерениях и стандартизации, основанной на объединенных достижениях науки, техники и практического опыта и составляющей основу долговременного развития) и управления качеством, сфокусированного на выполнении требований качества и его улучшении [4]. И особую роль в этом играет экономическая наука (рис. 3.1.2).

Наука о качестве сочетает и синтезирует знания, приемы и методы многих направлений экономической науки. Сохраняя свою самостоятельность, наука о качестве представляет собой часть экономической науки, изучающей взаимосвязь качественных характеристик объектов или явлений с экономическими показателями.

Непосредственной целью экономики качества как науки являются описание, объяснение и предсказание закономерностей воздействия качества на процессы и явления общественной жизни. Экономика качества дает возможность по-новому взглянуть на многие явления и закономерности общественных и производственных отношений с позиций роли качества, его влияния на весь

© URSS

ход социально-экономического развития. Совместное использование методов экономики качества и математического моделирования способствует формированию системного подхода к управлению развитием цифровой экономики.

Экономика качества, во взаимодействии метрологии, стандартизации и управления качеством, играет важнейшую организующую роль в цифровой экономике (рис. 3.1.3).

Стандартизация направлена на достижение оптимальной степени упорядочения в определенной области посредством установления положений для всеобщего и многократного использования в отношении реально существующих или потенциальных задач [5]. В последние 15 лет основной целью стандартизации стало обеспечение повышения качества жизни как единого комплекса повышения качества продукции, услуг, окружающей среды, других составляющих качества жизни.

Рис. 3.1.3. Система обеспечения качества в цифровой экономике

© URSS

После Второй мировой войны порядка 13 % роста экономики СССР, США и стран Европы были достигнуты благодаря стандартам, которые явились средством распространения технологий, практики управления бизнесом и приобретения других знаний, рассматриваемых в качестве элементов системы инноваций. Данные современных международных исследований по выявлению степени влияния стандартизации на экономику показали, что в странах — членах ВТО влияние стандартизации на рост ВВП составляет более 27 %, на рост производительности труда — 30 %. На микроуровне преимущества от стандартизации оцениваются как вклад в валовую прибыль компании на уровне 0,15–5 % годовых доходов от продаж.

Стандартизация обеспечивает правильность, сопоставимость результатов измерений, распространяет эти качества на экономику в целом, опираясь на метрологию. Метрология предоставляет единые методы измерения и оценки показателей цифровой экономики. Связь метрологии со стандартизацией обусловлена стандартизацией единиц измерений, системы государственных эталонов, средств измерений и методов поверок, создании стандартных образцов свойств и состава веществ. Любой предмет, явление или процесс нуждается в измерении и оценке, а уже затем стандартизируется.

В нашей стране ежедневно производятся миллионы измерений, свыше 2 млн человек занимаются измерениями в связи со своей профессиональной деятельностью. Согласно имеющимся данным, доля затрат на измерения в России составляет 5–10 % затрат общественного труда, а в отраслях промышленности, производящих сложную технику, она достигает 20–30 % [6]. Экономический эффект от метрологической деятельности в Российской Федерации и развитых странах мира составляет от 2 до 3,5 % ВВП.

По количеству позиций, включенных в базу данных Международного бюро мер и весов и характеризующих признанные мировым метрологическим сообществом измерительные и калибровочные возможности (СМС), Российская Федерация в 2020 г. вышла на первое место в мире, опередив Китай, США, Германию и другие страны [7]. Лидерство России в мировой метрологии, обеспеченное

достижениями отечественных ученых в сфере науки и техники, открывает хорошие стартовые возможности для ее цифровизации.

Развитие цифровой метрологии происходит одновременно в четырех сферах: законодательной, организационной, методической и технической. Это и создание качественной правовой базы, и техническое переоснащение метрологических организаций и подразделений на предприятиях, а также совершенствование подготовки кадров, соответствующих новому уровню требований к метрологическому обеспечению цифровой экономики.

Перед специалистами государственных научных метрологических институтов поставлена задача в ближайшие годы разработать программу цифровой трансформации российской метрологии по направлениям:

- цифровая трансформация метрологических услуг, в том числе создание инфраструктуры для цифровых сертификатов калибровки, создание «метрологического облака»;
- метрология в анализе больших данных — разработка методов анализа и машинного обучения для Big Data;
- метрология коммуникационных систем нового поколения, в том числе для сетей 5G;
- метрология интеллектуальных средств измерений — самонастройка и самокалибровка;
- метрология для моделирования и виртуальных приборов [8].

Институциональную основу российского механизма цифровизации метрологической деятельности составляют структуры, в рамках которых развитие цифровой метрологии находится в компетенции Минпромторга, а непосредственно за выполнение работ отвечает Росстандарт. Для координации этой деятельности в структуре Росстандарта было создано Управление цифрового развития технического регулирования.

В Метрологической академии, объединившей ведущих ученых и специалистов в области обеспечения единства измерений, действуют девять координационных научных советов, и практически каждый из них сегодня рассматривает вопросы цифровизации

метрологической деятельности по своим направлениям. Советы по законодательной метрологии, по фундаментальным проблемам метрологии, по аналитическому исследованию потребности в развитии метрологии, обсуждают и предлагают решения, касающиеся общих проблем состояния и перспектив цифровой метрологии. Научные советы по метрологии в физикохимии и в медицине в первую очередь занимаются проблемами совершенствования приборной базы и повышения точности и эффективности измерений в этих отраслях, что прямо связано с цифровизацией.

Важнейшая роль развития цифровой метрологии в процессе цифровизации экономики была особо подчеркнута в приветственном обращении директора Международного бюро мер и весов (далее — МБМВ) Мартина Милтона и директора Международного бюро законодательной метрологии (далее — МОЗМ) Энтони Доннеллана по случаю Всемирного дня метрологии 20 мая 2022 г. В обращении отмечено, что «цифровая трансформация в области метрологии может принести большую пользу. Например, ускорится ввод измерительных продуктов и услуг на рынок, и меньше времени будет занимать процесс утверждения, что, в свою очередь, окажет благоприятное воздействие на инновационное развитие, темпы продвижения продукции и обеспечение устойчивости» [9].

Со своей стороны МБМВ и МОЗМ очень многое делают для создания и применения цифровых измерительных платформ в мировой и национальных экономиках, для распространения передового опыта в сфере цифровизации и внедрения цифровой основы для всех пользователей данных по измерениям.

На новом, цифровом этапе развития промышленности, известном как «Индустрия 4.0», во главу угла поставлен переход от встроенных к киберфизическим производственным системам. В отличие от встроенных систем, где центральные блоки управления встраиваются в объекты, которыми эти системы управляют, в киберфизических производственных системах происходит объединение виртуального и физического мира посредством Интернета (сетей) и данных — так «умные» объекты взаимодействуют друг с другом.

В «Индустрии 4.0» объединение всей сферы деятельности современного предприятия в единую информационную среду позволяет оптимизировать используемую информацию, интегрировать бизнес-процессы в единую систему, получить доступ ко всем данным, существующим на предприятии, для анализа с целью оптимизации управленческих процессов (рис. 3.1.4).

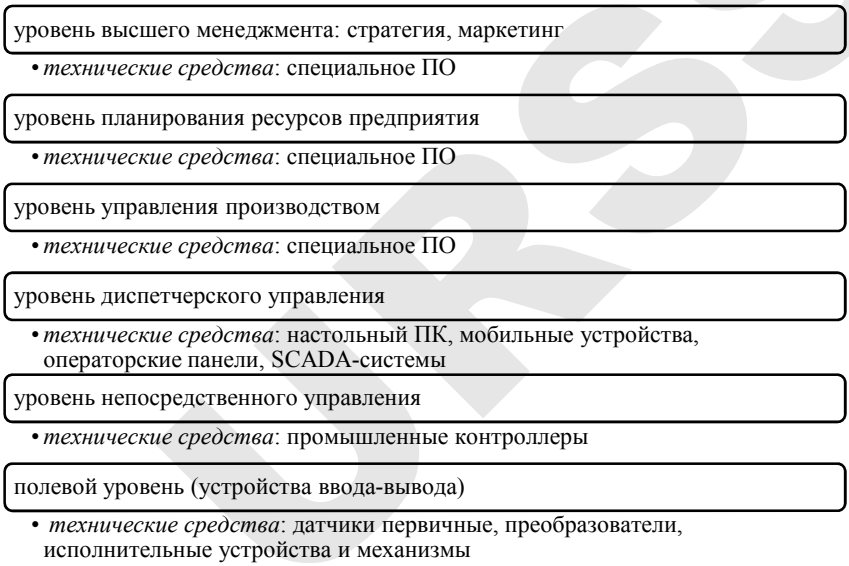


Рис. 3.1.4. Пирамида автоматизации

На самой вершине пирамиды автоматизации, характеризующей организационную структуру предприятия, располагается уровень высшего менеджмента, на самом нижнем уровне находятся технические измерения, результаты которых поступают в существующие АСУ через устройства ввода-вывода, вручную. Автоматизация и роботизирование процесса измерений, создание виртуальных лабораторий позволит уменьшить как погрешности измерения, так и добиться исключения человеческого фактора при работе с отчетной документацией. Несомненно, удобна и система дистан-

ционной калибровки и поверки средств измерения, выдачи электронных сертификатов, которые могли бы автоматически вноситься в базу данных предприятия.

Хранение таких объемов информации, в свою очередь, требует внедрения облачных технологий, организации и систематизации данных. Обработка больших объемов информации невозможна без инструментов Big Data, предсказание и ведение аналитики — без методов машинного обучения. Внедрение таких технологий, несомненно, требует метрологической оценки точности и корректности работы.

Проведение дистанционных измерений, калибровки и поверки возможно реализовать, например, методами машинного зрения. Дальнейшая передача данных об измерении беспроводным путем тоже требует особого внимания. Пока не разработано законченной и целостной теории оценки погрешности при проведении подобных процедур.

В научном метрологическом сообществе появляются проекты создания национальных и международных метрологических облаков, виртуальных лабораторий, цифровых двойников, дистанционных процедур прохождения поверки и калибровки средств измерений, выдачи электронных сертификатов. Тенденция однозначно направлена в сторону средств измерений с распределенными, частично виртуализированными компонентами и применения облачных сервисов (Cloud Services).

Грамотная организация хранения данных и обращения к ним, обработка и анализ становятся новой задачей современной законодательной метрологии. Успешному цифровому развитию законодательной метрологии призвано способствовать внедрение цифровой инфраструктуры. Так, внедрение «метрологического облака» (Metrology Cloud) обеспечивает: присоединение существующих инфраструктур данных, дифференцированный доступ всех партнеров к «метрологическому облаку» как к базе данных и надежному пункту доступа, новые концепции для координации, концентрации, упрощения, гармонизации и обеспечения качества метрологических услуг.

Разработка Metrology Cloud направлена на внедрение пользующейся доверием базовой платформы для цифровой инфраструктуры качества путем присоединения существующих информационных инфраструктур и баз данных и дифференцированного доступа всех партнеров для цифрового укрепления законодательной метрологии.

В Российской Федерации цифровая трансформация экономики является одной из приоритетных государственных задач, для реализации которой разработаны соответствующие национальные проекты. Их осуществление направлено на создание инфраструктуры, в успешной деятельности которой значительная роль принадлежит развитию цифровой метрологии.

Разработка цифровой инфраструктуры состоит из двух параллельных потоков. Первый поток — это разработка законодательства, обеспечивающего признание цифровой платформы, а второй поток — это развитие самой цифровой платформы. Очень сложная задача для руководителя проекта — координация этих двух взаимосвязанных потоков, ведь для работы с цифровой платформой необходимо подготовить изменения в законодательстве, связанные с переходом на цифровую платформу.

Законодательство включает в себя национальный закон о метрологии и ряд связанных с ним законодательных и технических актов. Техническая платформа включает программный продукт и серверные мощности.

Техническая составляющая цифровой, в значительной мере — метрологической инфраструктуры в России, как и за рубежом, — цифровая платформа. Данная платформа объединяет информацию и оперирует данными об эталонах, средствах измерений, справочными материалами и другими объектами законодательной метрологии, предоставляя доступ к связанным сервисам и документам. Так, доступные для измерительного прибора данные должны автоматически связываться с данными всех инструментов, используемых для его калибровки. Для этого необходимые данные калибровочных инструментов также должны присутствовать на цифровой платформе.

Чтобы реализовать идею цифровой метрологической платформы, необходимо правильно поставить и решить три задачи:

- спроектировать архитектуру цифровой платформы, определить инфраструктуру (необходимо спроектировать архитектуру цифрового решения, реализовать структуру облачного хранилища);
- разработать мотивационные мероприятия для участников или пользователей системы законодательной метрологии для ускорения внедрения цифровой платформы в максимально короткий период времени;
- обеспечить юридическую значимость данных, хранящихся на цифровой платформе.

В сфере законодательной метрологии для реализации работы в новой российской цифровой инфраструктуре также поставлен ряд задач, среди которых:

- поверка законодательно регулируемых средств измерений (СИ);
- создание цепочки прослеживаемости от СИ (или других признанных стандартов, если они им соответствуют) до конечного пользователя;
- определение процедур утверждения типа;
- аттестация стандартных образцов

и т. д.

Законодательная метрология в России большое внимание уделяет процедуре поверки средств измерений. Результатом, как правило, является не только заключение о пригодности или непригодности средства измерений, но также информация о характеристиках средства измерения. По разным оценкам в Российской Федерации проводится порядка 50–60 млн поверок в год. Задача хранения и обработки такого большого количества данных очень сложная.

Через информацию о поверке средств измерений Росстандарт получает данные обо всех используемых средствах измерений. В связи с этим очень важно оценить необходимую мощность создаваемой технической платформы. Платформа должна быть способна обрабатывать около 500 поверок от 2000 проверяющих органов в

минуту. Это означает, что при максимальной производительности платформа должна быть способна обрабатывать несколько тысяч поверок в минуту. Еще одна важная техническая проблема заключается в том, что данные на платформе требуют цифровых подписей. Информация должна быть защищена от несанкционированных изменений, в том числе хакерских атак.

Чтобы обеспечить более мягкий переход к цифровой инфраструктуре, на первом этапе возможно комбинирование деятельности: работа как с использованием цифровой инфраструктуры, так и по старинке, с применением бумажных оригиналов сертификатов и других документов. Тем более что в соответствии с Федеральным законом «Об обеспечении единства измерений» [10] с 24 сентября 2020 г. при регистрации результатов поверки индивидуальных приборов учета и других средств измерений приоритет принадлежит электронной записи в Федеральном информационном фонде по обеспечению единства измерений, а бумажная форма может быть выдана по желанию потребителя, но она носит только справочный характер.

В настоящий момент версия цифровой метрологической платформы частично реализована путем создания федерального информационного фонда по обеспечению единства измерений ФГИС «Аршин» [11].

Техническая платформа содержит как защищенные паролем платформы, так и открытые порталы. Открытый портал используется в свободном доступе для получения информации в области законодательной метрологии в России (включая все применимые международные документы, например рекомендации МОЗМ на русском и английском языках). Можно проверить информацию о прохождении процедуры поверки средств измерения по их серийному номеру и типу на ФГИС «Аршин». Всего техническая платформа насчитывает 19 модулей. Для крупных организаций, выполняющих множество метрологических задач, реализован механизм пакетной передачи данных. Этого можно добиться как с помощью файлов определенного формата, так и с помощью специального сервиса API технической платформы.

Важным вопросом является наличие в системе информации о метрологических работах, выполненных до 24 сентября 2020 г. При подготовке законопроекта было решено не требовать от участников цифровой инфраструктуры размещать всю информацию о ранее выполненных услугах. Отчеты и документации по ним принимаются в «старом» бумажном формате. Действительно, ввод всех этих данных в цифровую платформу стал бы бременем для предприятий и кажется неразумным. С другой стороны, Росстандарт потребовал от своих организаций вводить всю информацию, начиная с 2010 г. Это объясняет, почему в год проводится около 50 млн поверок, а в целом платформа уже содержит более 300 млн наборов поверочных данных.

Цифровизация метрологии имеет большое значение для всех пользователей средств измерений, которыми являются не только компании и органы власти, но и частные лица. Это очень важный момент, потому что ключевой задачей внедрения информационных технологий в России является в том числе защита граждан. Теперь с помощью мобильного приложения любой желающий может проверить наличие поверочной документации весов в супермаркете, ТРК на АЗС или на счетчик воды в своем доме или квартире. Таким образом, этот проект также имел существенное социальное значение.

Данные, размещаемые на цифровой платформе, очень важны для метрологического надзора. Измерительные приборы, период поверки которых истек и для которых не было зарегистрировано никаких новых поверок, можно легко получить с платформы. Целевые меры реагирования могут применяться к организациям, владеющим этими инструментами. Для аккредитационного надзора также важны данные об организациях, выполняющих большое количество метрологических задач. Они могут сосредоточиться на таких случаях и заниматься прослеживаемостью. Для проверяющих органов появилась возможность работать с результатами в режиме реального времени, что сразу снимает целый ряд проблем, связанных с неточной или даже фальсифицированной информацией.

Еще одно важное преимущество заключается в том, что из-за ошибок, связанных с человеческим фактором, иногда приходится аннулировать результаты метрологических решений, что может привести к юридическим последствиям. Для властей это влечет за собой отзыв их сертификатов и дополнительные расходы на исправление ошибок. Теперь система либо не допускает таких ошибок, либо (в худшем случае) позволяет их обнаруживать раньше. В этой связи чрезвычайно важным является вопрос совершенствования подготовки кадров для метрологической деятельности.

Например, на кафедре метрологического обеспечения и промышленной безопасности Санкт-Петербургского государственного университета аэрокосмического приборостроения (ГУАП) проводится углубленное изучение технических дисциплин будущими специалистами-метрологами, что позволяет им после окончания университета в полной мере ориентироваться в работе на сложнейших предприятиях аэрокосмического комплекса России, обеспечивать точность и качество измерений при создании новой техники и оборудования.

В последнее время на кафедре особое внимание уделяется изучению вопросов цифровой метрологии, без которой невозможно представить развитие современной авиационной и космической техники. Для лучшего обеспечения учебного процесса специалистами кафедры и других подразделений ГУАП в 2021 г. было издано учебное пособие «Цифровая метрология», а в 2022 г. подготовлен учебник «Основы цифровой метрологии», в котором обобщены и представлены последние данные о методах обработки и оценивания точности измерений, о влиянии цифровизации на метрологическую деятельность [12].

На состоявшейся 2 августа 2022 г. встрече премьер-министра РФ Михаила Мишустина и руководителя Росстандарта Антона Шалаева было отмечено, что цифровизация стала ключевым направлением работы в системе обеспечения единства измерений. Уже переведены в цифровой формат такие услуги, как утверждение типа средств измерений или типа стандартных образцов и свидетель-

ства о соответствии средств измерений метрологическим характеристикам, согласование отзывных программ в области автомобильных колесных транспортных средств и другие. Разрабатывается более 30 стандартов для реализации новых технологий, в том числе для применения технологий искусственного интеллекта в клинической медицине, в транспортной области, робототехнике, биометрии и биомониторинге [13].

При всей значимости цифровой метрологии не менее важным является цифровое развитие стандартизации.

Роль стандартизации как инструмента экономики качества хорошо заметна в Федеральном законе «О стандартизации в Российской Федерации» [14]: «деятельность по разработке (ведению), утверждению, изменению (актуализации), отмене, опубликованию и применению документов по стандартизации и иной деятельности, направленной на достижение упорядоченности в отношении объектов стандартизации».

Цели и задачи стандартизации определены следующим образом.

1) содействие социально-экономическому развитию Российской Федерации;
2) содействие интеграции Российской Федерации в мировую экономику и международные системы стандартизации в качестве равноправного партнера;
3) улучшение качества жизни населения страны;
4) обеспечение обороны страны и безопасности государства;
5) техническое перевооружение промышленности;
6) повышение качества продукции, выполнения работ, оказания услуг и повышение конкурентоспособности продукции российского производства.

Представляется очевидным, что отдельные аспекты безопасности людей, технической и информационной совместимости, экономии ресурсов, обеспечения стабильности деятельности лежат в сфере стандартизации. Ее развитие оказывает большое влияние на технический прогресс, устранение технических барьеров, становление очередного технологического уклада.

Стандарт на всем протяжении существования и развития стандартизации — это норма, которой необходимо следовать. В условиях цифровой экономики, когда так велика неопределенность содержания цифровых активов и технологий управления ими, стандарт обретает новое, во многом определяющее значение для экономики.

Важнейшим средством управления в экономике является стандартизация промышленности, с которой зачастую и начинается государственная промышленная политика. На институциональном уровне принципы, цели и задачи, механизмы и основные положения стандартизации закреплены в Федеральном законе «О стандартизации в Российской Федерации» и соответствующих подзаконных актах.

Среди документов, разработанных совместно Минпромторгом России и Росстандартом с активным привлечением экспертного сообщества, — Федеральный закон «О внесении изменений в Федеральный закон «О стандартизации в Российской Федерации» от 30 декабря 2020 г. № 523-ФЗ. На этом этапе законодательно закреплена новая задача, в рамках которой необходимо применение инструментов стандартизации в целях разработки и внедрения передовых технологий, в том числе цифровых технологий, для достижения и поддержания технологического лидерства Российской Федерации в высокотехнологичных (инновационных) секторах экономики.

Технологический аспект поставленной задачи состоит в автоматизации работ по стандартизации и обеспечении их прозрачности. Для этого предусмотрено использование информационной системы (которая является государственным информационным ресурсом) на всех этапах разработки документов по стандартизации — от их планирования до редактирования.

Главное в этой деятельности — нормативное закрепление цифровых технологий в национальной системе стандартизации. Разработка стандартов еще с ноября 2017 г. полностью ведется через интернет-портал Федеральной государственной информационной системы «Береста». За это время нам удалось в полной мере оценить влияние цифровых решений на процесс стандартизации.

Цифровизация позволила существенно ускорить разработку стандартов и сделать ее более прозрачной за счет привлечения всех заинтересованных сторон.

В промышленности и научной сфере очень важным является введение в нормативно-правовое регулирование и применение новых видов документов по стандартизации: техническая спецификация и технический отчет, в которых будет представлена возможность устанавливать характеристики и методы испытаний в отношении инновационной продукции. Это внесет свой вклад в том числе в реализацию утвержденного распоряжением Правительства РФ от 6 ноября 2021 г. № 3142-р «Стратегического направления в области цифровой трансформации обрабатывающих отраслей промышленности» [15]. При этом предусмотрено внедрение таких технологий, как искусственный интеллект, новые производственные технологии, робототехника и сенсорика, новые коммуникационные интернет-технологии, интернет вещей, технологии виртуальной и дополненной реальности.

Проекты «Стратегического направления в области цифровой трансформации обрабатывающих отраслей промышленности» — «Умное производство», «Цифровой инжиниринг», «Новая модель занятости», «Продукция будущего» — трудно представить без цифровизации стандартизации. Стране нужно единое информационное пространство стандартизации, обеспеченное единой платформой; единая программа развития промышленности, которая позволит всем отраслям и всем предприятиям общаться на одном языке и работать по единым стандартам. Основой этой платформы должны стать именно единые государственные стандарты качества.

В числе мер по совершенствованию механизмов стандартизации для обеспечения соответствия существующей системы технического регулирования целям развития цифровой экономики, можно перечислить [16]:

- работу в партнерстве с отраслевыми, корпоративными, международными и иностранными системами стандартизации;
- ускоренное принятие национальных стандартов на основе отраслевых (корпоративных) и международных документов;

- обеспечение возможности применения документов по стандартизации на английском языке, обеспечивающих максимально комфортное развитие современных технологий;
- формирование библиотеки национальных стандартов по приоритетным направлениям в машиночитаемом формате.

В связи с ускоренными темпами развития цифровой экономики и ее растущими потребностями, за сравнительно небольшой промежуток времени предстоит выполнить колоссальный объем работы. Например, только перевод информационных стандартов в XML-формат охватывает весь фонд стандартов, а это порядка 32 000 документов. Принципы принятия решений, способы их реализации, контроль исполнения, планирование возможных улучшений — все это входит в сферу управления качеством. Система обеспечения качества в цифровой экономике и опирается на взаимосвязь метрологии, стандартизации и управления качеством, и поддерживается их непрерывным развитием.

Управление качеством создает гарантии получения прибыли, снижения издержек, сокращения сроков изготовления и реализации продукции, обеспечивает стабильность и надежность любой социально-экономической деятельности, повышает ее эффективность. Опыт многолетних научных исследований убедительно доказывает, что с помощью методов управления качеством можно решить любую проблему на любом уровне управления, независимо от общественного строя, формы собственности, характера и размера производства, численности персонала.

Вклад российских ученых, членов Академии наук, в развитие управления качеством особенно значителен. С введением научных основ обеспечения качества стали применяться элементы стимулирования работников, разрабатываться математические методы оценки качества. Разработка основ системного подхода к качеству началась в середине прошлого столетия, в 1970-х годах были созданы комплексные системы управления качеством продукции. Важным итогом этой работы стала разработка научных принципов всеобщего управления качеством. Спустя несколько лет было поло-

жено начало использованию систем менеджмента качества на основе международных стандартов. Наиболее распространенными и востребованными сегодня являются системы менеджмента качества, созданные в соответствии с требованиями стандартов ISO (International Organization for Standardization) серии 9000 [17]. Более миллиона компаний и организаций в 170 странах мира сертифицированы на соответствие стандарту ISO 9001 [18].

Принятая на рубеже XXI в. новая версия международных стандартов ISO 9000 сконцентрировала управление качеством на интересах потребителя. Стандарты на системы менеджмента качества распространяются и на отрасли экономики. В условиях цифровой экономики это обеспечит эффективное решение задач управления на всем жизненном цикле изделия — от замысла до утилизации последнего экземпляра. В основе современного подхода к управлению качеством лежит риск-ориентированное мышление. Оно предполагает оценку вероятности возникновения дефектов и степень их влияния на продукцию и потребителя.

Возрастание уровня риска в условиях цифровой трансформации смещает акцент в управлении качеством на менеджмент рисков. Все более широкое применение находят стандарты ГОСТ Р 51897–2011 «Менеджмент риска. Термины и определения» [19], ГОСТ Р ИСО 31000–2010 «Менеджмент риска. Принципы и руководство» [20]. Это делает управление более гибким, позволяет целенаправленно достигать поставленных целей.

Основываясь на допущении о трансформации цифровой экономики в цифровую социо-эколого-экономическую систему, можно с уверенностью утверждать, что еще одно направление — построение систем экологического менеджмента в соответствии с требованиями стандартов ISO серии 14 000 (ISO 14000) [21], не только не потеряет своей актуальности, но и найдет новое, более широкое применение. Это стандарты Национальный стандарт РФ ГОСТ Р ИСО 14001–2016 «Системы экологического менеджмента. Требования и руководство по применению» [22], ГОСТ Р ИСО 14004–2007 «Системы экологического менеджмента. Общее руководство по принципам, системам и

методам обеспечения функционирования» [23], Стандарт ГОСТ Р ИСО 14001–2016 полностью совместим со стандартами ГОСТ Р ИСО 9001–2015 [24] (система менеджмента качества) и стандартом ГОСТ 12.0.230–2007 «Система стандартов безопасности труда. Система управления охраной труда. Общие требования» [25].

Среди преимуществ, которые образуются в результате внедрения стандартов, можно отметить не только повышение эффективности использования ресурсов и рост результативности деятельности. Процесс создания и спецификации цифровых активов организаций значительно облегчается, например, в условиях интегрированной системы менеджмента, удовлетворяющей требованиям не только стандартов на систему менеджмента качества, но и экологического менеджмента. Большую пользу здесь может принести ГОСТ Р ИСО 19011–2012 [26], предполагающий совместное проведение аудитов СМК и окружающей среды.

Особого внимания в условиях цифровой трансформации заслуживает социальное развитие и в этом смысле чрезвычайно полезен ГОСТ Р ИСО 26000–2012 «Руководство по социальной ответственности» [27]. Это и подотчетность, и прозрачность, и этичное поведение, и уважение интересов заинтересованных сторон, и соблюдение верховенства закона, соблюдение международных норм поведения и соблюдение прав человека. Необходимость использования упомянутого стандарта особенно актуальна при работе с разными видами цифровых активов.

Дальнейшая востребованность инструментов экономики качества определена не только их положительным вкладом в управление системами, но и их организующей ролью для цифровой экономики. Появление новых направлений ее развития уже обусловило возникновение потребностей бизнеса, общества и населения в новых стандартах. Примером таких стандартов являются стандарты в области устойчивого развития. На международном уровне это стандарт ISO/TR 37121:2017 «Устойчивое развитие сообществ. Перечень действующих директивных рекомендаций и принципов по устойчивому развитию городов и их устойчивости к негативным

внешним воздействиям» [28]. В настоящее время он содержит компиляцию (описание) 105 систем индикаторов для мониторинга (самооценки) городов, применяемых во всем мире [29].

Достижения в сфере российской стандартизации, востребованные на международных площадках, значительно усилят позиции России в области высоких технологий, упрочат ее конкурентоспособность, особенно в сегодняшних непростых условиях. Государственная программа «Развитие цифровой экономики в России» [30] предусматривает создание единых систем идентификации, информационной безопасности, единых стандартов, которые расширят возможности решения проблем инновационной экономики, обусловленных высоким уровнем цифровой трансформации.

Применение метрологии и стандартизации способствует универсализации величин, понятий и процессов, а внедрение методов управления качеством повышает эффективность работы с информацией [31]. Использование инструментов экономики качества при проведении моделирования цифровой экономики целесообразно практически на каждом этапе и имеет смысл, особенно при решении сложных практических задач с помощью суперкомпьютеров.

Экономика качества, таким образом, предоставляет комплексный подход для решения сложных практических задач цифровой экономики и позволяет оптимизировать протекающие в ней сквозные цифровые процессы, вывести их на следующий, более высокий уровень развития. Следует отметить, что объектом управления может выступать как цифровая экономика в целом, так и регион, территория, а также и отдельные составляющие цифровой экономики, например цифровые активы.

Рассматривая качество как комплексную систему методов и инструментов, применяемую во всей системе экономических отношений, экономика качества позволяет найти оптимальные решения любых социально-экономических проблем на всех иерархических уровнях управления [32]. Для этих целей применяется многоуровневая система управления качеством как совокупность организационной структуры, методов работы, процессов и ресурсов,

необходимых для осуществления воздействия на качество посредством мер оперативного характера на трех взаимосвязанных и взаимовлияющих друг на друга уровнях:

- микроуровень — предприятие, корпорация;
- мезоуровень — город, отрасль, регион;
- макроуровень — страна в целом.

Многоуровневая система управления качеством нацелена на последовательный рост благосостояния населения, эффективное воспроизводство и модернизацию производства, повышение инвестиционной привлекательности региона и дальнейшее развитие рыночных механизмов хозяйствования, укрепление конкурентоспособности, повышение качества жизни населения [33]. Данная система включает в себя составляющие, позволяющие учитывать потребности населения региона и оценивать степень их удовлетворения; отслеживать и своевременно реагировать на изменения в отношениях между государством, бизнесом и обществом на фоне цифровизации протекающих процессов. Можно с уверенностью утверждать, что в условиях инновационной экономики основу разработки принципиальных подходов к уточнению роли инструментов экономики качества в оценке качества жизни населения региона составляет система управления качеством.

«Входом» системы на макро-уровне являются интересы государства, аккумулирующие выработанные в согласии с обществом важнейшие направления социально-экономического развития, в которых решающую роль играют вопросы качества: экономические, социальные, военные, международные, информационные, экологические. «Выходом» системы являются результаты реализации национальных интересов, а именно: обеспечение конкурентоспособности, безопасности, обороноспособности, повышение престижа страны в мире, укрепление позиций на внутреннем и внешнем информационном рынке, улучшение экологической ситуации. В этом ряду удовлетворение потребностей общества в целом, как и отдельных индивидов, следует рассматривать как полноценную защиту прав и интересов граждан со стороны государства.

Цикл функционирования системы управления качеством на любом уровне начинается и заканчивается мониторингом ситуации, в котором важную роль играет оценка показателей качества обществом. Результаты мониторинга и анализа общественного мнения относительно уровня достижения требуемых показателей являются важнейшей базой для принятия стратегических управленческих решений. Реализация национальной стратегии в области качества призвана быть основой обеспечения устойчивого развития, ориентации созидательной деятельности работников на постоянное повышение качества товаров и услуг, а также снижение отрицательного воздействия на окружающую среду. Современный уровень развития информационных технологий позволяет осуществлять мониторинг этих факторов в реальном времени.

Цифровизация экономики и сферы потребления в нынешних условиях все в большей степени выступает решающим фактором во взаимодействии всех участников борьбы за качество товаров и услуг. Она не только ускоряет и удешевляет процессы производства и сбыта продукции, но и меняет характер взаимоотношений производителей и потребителей, позволяя сделать его более оперативным и прозрачным. Значительная роль в этом принадлежит моделям управления качеством на уровне предприятий. Они включают развитие различных систем менеджмента, в том числе базирующихся на стандартах ИСО, моделей Европейской и национальных премий по качеству, проведение регулярных национальных и региональных конкурсов по оценке качества продукции и услуг.

Для развития системного подхода на уровне предприятий — микроуровне — с применением всех инструментов экономики качества была разработана методика анализа возможностей совершенствования деятельности предприятий на основе применения методов самооценки. Критерии методики, по которым проводится оценка, универсальны, их можно применять к любому предприятию. Ее использование дает организации возможность более точно определить свое место на рынке, существенно повысить эффек-

тивность стратегического планирования и улучшить качество продукции, опираясь на оценки конечного потребителя.

В цифровой экономике региона методология экономики качества создает надежную базу для разработки важнейших целевых стратегических ориентиров развития и обеспечивает получение достоверных оценок результативности стратегических решений,

Рис. 3.1.5. Показатели качества жизни в системе целей «Стратегии 2035»

принимаемых на всех уровнях власти. На региональном уровне находит широкое применение модель системы управления качеством для уровня федерального округа, предложенная для использования в процессе создания Комплексной научно-технической программы Северо-Западного федерального округа РФ на период до 2030 г. Еще одно направление практического применения многоуровневой системы управления качеством — долгосрочная стратегия развития Санкт-Петербурга.

Главной целью «Стратегии 2035» является повышение качества жизни людей (рис. 3.1.5). В числе приоритетов: город инноваций, комфортный город и открытый город.

Достижение цели предопределяет развитие человеческого капитала, повышение качества городской среды, обеспечение устойчивого экономического роста, обеспечение эффективности управления и развитие гражданского общества. «Стратегия 2035» особо выделяет качество жизни как ведущий элемент системы целей социально-экономической политики Санкт-Петербурга. Это наиболее емкий, интегральный показатель, характеризующий одновременно уровень развития экономики, состояние городской среды, включая экологию, качество и доступность услуг организаций здравоохранения, образования, культуры и социального обслуживания населения.

Научный подход к разработке стратегии, привлечение ученых, представителей бизнес-сообщества, государственной власти позволил верно определить направления развития Санкт-Петербурга. В числе приоритетов — развитие «экономики знаний», то есть науки, образования, здравоохранения, ИТ-технологий, биотехнологий. Сегодня «экономика знаний» формирует около 25 % ВРП Санкт-Петербурга. Для сравнения, в США > 40 % ВВП, в Европе — 30 % [34, с. 590].

Предстоящие изменения не только не уменьшат роль человека, но, наоборот, в новом цифровом обществе на первый план выйдет именно человеческий фактор. Только человек способен создавать те или иные ИТ-технологии для решения определенных задач. Поэтому задача развития цифровой экономики тесно связана с развитием данных отраслей. Именно они являются решающим фактором

при повышении качества условий жизнедеятельности человека, а значит, непосредственно воздействуют на развитие человеческого капитала — развитие его интеллекта, здоровья, знаний и умений [35].

При разработке стратегии были учтены показатели, отражающие ориентиры развития ведущих стран и городов мира. Несомненно, времени на создание такого масштабного документа было мало, тем более что необходимая для этого нормативная и методическая база практически отсутствовала. Однако Комитетом по экономической политике и стратегическому планированию, Экономическим советом при губернаторе Санкт-Петербурга, другими комитетами правительства и общественными организациями города была проделана большая работа. Особо следует обратить внимание на интернет-сайт стратегии, на котором все заинтересованные стороны, каждый житель Петербурга мог высказать свои предложения и замечания. Создание сайта и свободный доступ любому жителю города обеспечил максимальную прозрачность процесса разработки стратегии и реальные условия учета интересов всех заинтересованных сторон. Активное участие петербуржцев в обсуждении проекта убедительно подтвердило заинтересованность горожан в переменах и развитии [34, с. 561].

В Стратегии сформированы три сценария социально-экономического развития Санкт-Петербурга в долгосрочной перспективе: базовый сценарий, сценарий «базовый плюс» и целевой сценарий. Базовый сценарий предполагает в долгосрочном периоде переход от стагнации к умеренному восстановлению темпов роста экономики Санкт-Петербурга. Сценарий «базовый плюс» предполагает более благоприятные условия развития внешней среды, чем в базовом сценарии. В экономике Санкт-Петербурга ожидается ускорение темпов экономического роста и активизация инвестиционной и инновационной деятельности в отраслях экономики Санкт-Петербурга, в том числе за счет трансфера технологий и реализации политики импортозамещения. С учетом того, что рост экономики будет приводить к дополнительной потребности в трудовых ресурсах, предприятия и организации будут вынуждены увеличи-

вать эффективность деятельности за счет автоматизации производственных и вспомогательных процессов, что приведет к увеличению количества высокопроизводительных рабочих мест и увеличению доли высокотехнологичных предприятий в ВРП.

В соответствии с целевым сценарием ожидается выход российской экономики на траекторию устойчивого роста при одновременном обеспечении макроэкономической сбалансированности и постепенного перехода на несырьевую модель развития. В Санкт-Петербурге с учетом условий развития внешней среды целевой сценарий будет характеризоваться наиболее успешным использованием его конкурентных преимуществ и инструментов ускорения социально-экономического роста путем реализации Правительством Санкт-Петербурга мер поддержки и стимулирования экономики, разработанных для развития приоритетных отраслей, технологий и территорий.

В качестве примера приведем несколько целевых значений индикаторов устойчивого развития Санкт-Петербурга (значения, которых они должны достигнуть к 2035 г.).

- Ожидаемая продолжительность жизни при рождении — 80 лет.
- Отношение обеспеченности населения местами в общеобразовательных организациях к потребности — 100 %.
- Удельный вес численности выпускников профессиональных образовательных организаций очной формы обучения, трудоустроившихся в течение первого года после окончания обучения, в общей численности выпускников профессиональных образовательных организаций очной формы обучения — 70 %.
- Доля населения, имеющего денежные доходы ниже величины прожиточного минимума, в общей численности населения Санкт-Петербурга — 2,6 %.
- Коэффициент Джини — 0,370 единиц.
- Индекс физического объема ВРП (ежегодно в среднем за период) —105 %.
- Доля продукции высокотехнологичных и наукоемких отраслей экономики в ВРП — 45 %.

- Доля внутренних затрат на исследования и разработки в ВРП — 7 %.
- Увеличение производительности труда относительно уровня 2017 г. — в 1,8 раз.
- Доля инвестиций в основной капитал в ВРП — 25 %.
- Темп роста объемов несырьевого неэнергетического экспорта товаров (всего; ежегодно в среднем за период) — 105,98 %.
- Увеличение выработки на одного занятого в промышленности (к уровню 2017 г., без учета данных по видам деятельности «Добыча полезных ископаемых» и «Производство кокса и нефтепродуктов») — в 2,9 раза.
- Доля инновационной продукции в общем объеме продукции в обрабатывающих производствах (без учета данных по виду деятельности «Производство кокса и нефтепродуктов») — 30 %.
- Удельный вес численности высококвалифицированных работников в общей численности квалифицированных работников — 41 % [36].

Следует полагать, что у Санкт-Петербурга есть шансы на успех в новой цифровой экономике. Эксперты отмечают, что в городе есть как специалисты мирового уровня в области разработки программного обеспечения, так и существенные технологические разработки. Кроме того, существует возможность подготовки специалистов в данной отрасли. В результате цифровизация должна стать одним из «локомотивов» развития города. Вторым же «локомотивом» станут как раз отрасли, которые обеспечивают развитие человеческого капитала, то есть составляющие «экономики знаний».

В городе уже предпринимаются шаги по созданию предпосылок для перехода к «умному городу». В их числе можно назвать работы по созданию аппаратно-программного комплекса «Безопасный город» в Кронштадте. Такая система способствует преодолению множества проблем. Например, в сфере энергоэффективности и ресурсосбережения.

Разработка стратегических планов регионального развития и повышение качества жизни населения, особенно в условиях эконо-

мических кризисов, могут и должны опираться на инструменты экономики качества — метрологию, стандартизацию и управление качеством. В определении стратегических ориентиров для достижения лучшего качества жизни населения и как приоритетные показатели, в Стратегию экономического и социального развития Санкт-Петербурга на период до 2030 г., а затем и в «Стратегию 2035», впервые в России включены основные показатели экономики качества.

3.2. Разработка информационно-методического обеспечения оценки качества жизни

Как показал опыт проведения научных исследований качества, такие сложные задачи, как разработка стратегии, оценка благосостояния населения в регионах, требуют применения новейших информационных технологий. По мнению К. Шваба, основателя Всемирного экономического форума, теперь совокупности «компьютеров, программного обеспечения и сетей… настолько сложны и интегрированы, что способны уже трансформировать общества и глобальную экономику» [37]. Согласно исследованиям ЦЭМИ РАН, для решения задач, ключевым элементом которых является моделирование поведения искусственного общества, требуется создание агент-ориентированных моделей с числом агентов до 10^9 и использование суперкомпьютерных технологий для проведения симуляций [38].

Влияние качества жизни непосредственно на состояние экономики в целом и экономическое поведение людей подчеркивает важность проблемы разработки адекватной методологии измерения качества жизни населения.

В Институте проблем региональной экономики РАН на концептуальной основе экономики качества совместно с ЦЭМИ РАН выполнено исследование по Программе Президиума РАН № 27 «Механизмы обеспечения отказоустойчивости современных высокопроизводительных и высоконадежных вычислений». Результаты

работы, проведенной ЦЭМИ РАН с применением суперкомпьютерных технологий, позволили ИПРЭ РАН адекватно оценить колоссальный объем необходимых вычислений и прийти к решению о необходимости формирования базы данных «Качество жизни (База данных для построения модели качества жизни)» с закрепленным правом интеллектуальной собственности, дополняющей методическую разработку оценки и измерения качества жизни населения.

Проведенные в ИПРЭ РАН исследования проблематики измерения и оценки качества жизни дают основание определить предметную область следующим образом. *Качество жизни* — это оценочная категория, которая обобщенно характеризует параметры всех составляющих жизни человека: его потенциала, жизнедеятельности и условий жизнедеятельности по отношению к стандарту или эталону, который выработан и институционализирован обществом и/или существует в индивидуальном сознании человека [39].

Согласно методике ИПРЭ РАН оценка качества жизни предполагает проведение работы в следующих направлениях.

1. Формирование системы целевых показателей для оценки и измерения качества жизни в целях регионального управления.
2. Выделение принципов применения системы показателей оценки и измерения качества жизни в интересах регионального управления.
3. Обоснование теоретической базы моделирования обобщенного показателя качества жизни.
4. Разработка механизма построения обобщенного показателя качества жизни.
5. Моделирование оценки факторов качества жизни для регионов Российской Федерации.
6. Разработка рекомендаций по применению теоретико-методологической базы экономики качества и оценки качества жизни в целях роста эффективности регионального управления.

При построении системы целевых показателей качества жизни подлежат учету показатели, отражающие человека и его потребности, характеризующие условия жизнедеятельности человека и среду,

способствующую его всестороннему развитию, здоровой и полноценной жизни. В связи с этим сформирована система целевых показателей, в том числе в сфере образования, для оценки и измерения качества жизни в целях регионального управления.

Система целевых показателей качества жизни содержит показатели разного уровня сложности. Самые простые — единичные показатели — измеряют одну из характеристик качества жизни, а более сложные — групповые и комплексные показатели — являются результатом обработки измерений нескольких характеристик. Как результат совокупности комплексных показателей формируется обобщенный показатель, интегрирующий все учтенные ранее характеристики качества жизни.

В общем случае система целевых показателей для измерения качества жизни основана на использовании информации (более 100 показателей), представленной в базе данных российской государственной статистики, отчетах региональных служб, других статистических данных, полученных из официальных источников информации. Данная система содержит набор показателей, предложенных нами ранее для применения в многоуровневой системе управления качеством [40].

Вектор качества жизни формируется как результат взаимодействия некоторых определяющих его направление и значение векторов. При количественном измерении качества жизни происходит векторное сложение частных факторов, что делает интегральный показатель одномерным упрощением этих факторов.

Выработанный в ИПРЭ РАН подход к теоретико-методологическому обоснованию уровня и качества жизни в контексте инновационной экономики строится на следующих принципах:

1) охват показателями векторов, определяющих вектор качества жизни;
2) построение рейтинга качества жизни населения региона;
3) введение полученных результатов в систему регионального управления посредством многоуровневой системы управления качеством.

Состояние (значение) поля качества жизни, образованного случайной функцией нескольких аргументов, описывается в виде иерархической совокупности функциональных зависимостей (рис. 3.2.1).

Показатели групп родственных характеристик качества жизни составляют основу для формирования соответствующих интегральных показателей. В своей основе они содержат более частные показатели следующего уровня, характеризующие отдельные аспекты качества жизни.

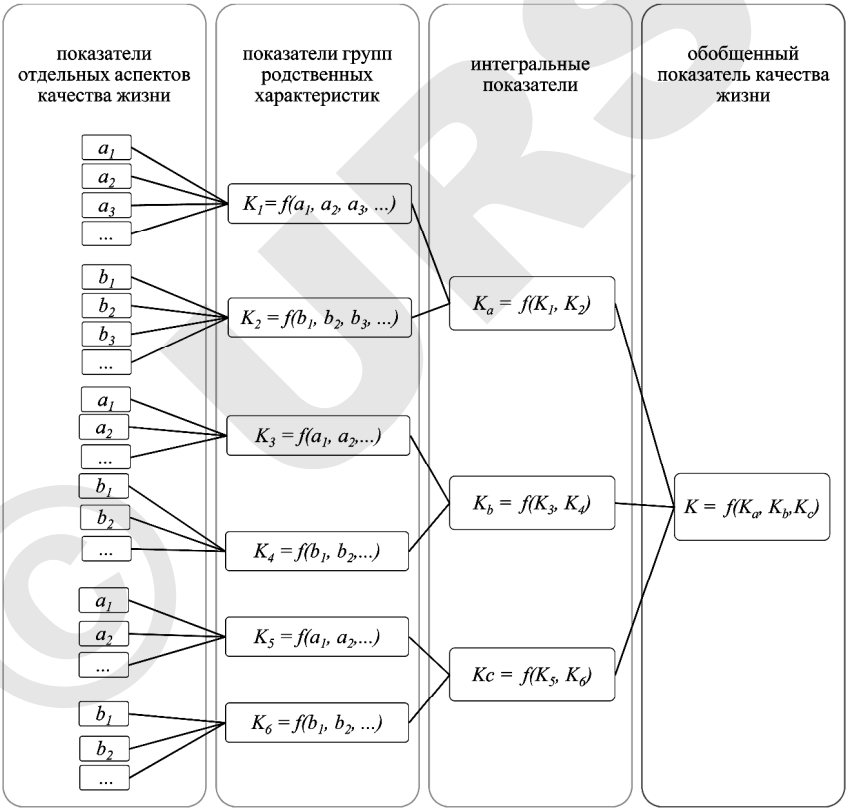


Рис. 3.2.1. Иерархическое представление функциональной зависимости в системе показателей качества жизни

Основу выделения принципов применения системы показателей оценки и измерения качества жизни в интересах регионального управления составило применение методов экономики качества, благодаря которому мы не просто оцениваем качество жизни, а получаем возможность его непрерывного улучшения. Предложенный подход ориентирован на построение такого механизма количественной оценки качества жизни населения региона, который дает возможность в рамках одного блока модели измерить и стандартизировать показатель качества жизни, а в рамках второго блока — эффективно использовать полученные значения показателя качества жизни для целей стратегического управления в рамках многоуровневой системы управления качеством, используя системы обратной связи.

Логика подхода схематично представлена на рис. 3.2.2.

Центральный элемент схемы — модель многоуровневой системы управления качеством, которая представляет собой совокупность организационной структуры, методов работы, процессов и ресурсов, необходимых для осуществления воздействия на качество

Рис. 3.2.2. Структурная схема моделирования и применения количественной оценки качества жизни населения

посредством мер оперативного характера на трех взаимосвязанных и взаимовлияющих друг на друга уровнях: предприятия, региона, страны в целом.

Получение количественных оценок качества жизни на основе применения эконометрических методов на концептуальной основе экономики качества имеет свою очередность (рис. 3.2.3).

Рассматриваемая методология позволила впервые разработать модель механизма количественной оценки качества жизни с применением элементов экономики качества: метрологии, стандартизации и управления качеством, обеспечивающего сочетание требований к необходимому качеству жизни и возможностей выбора оптимальных способов удовлетворения этих требований. В этом состоит и привлекательность методики, и ее отличие от других инструментов, которые мы проанализировали в процессе исследования.

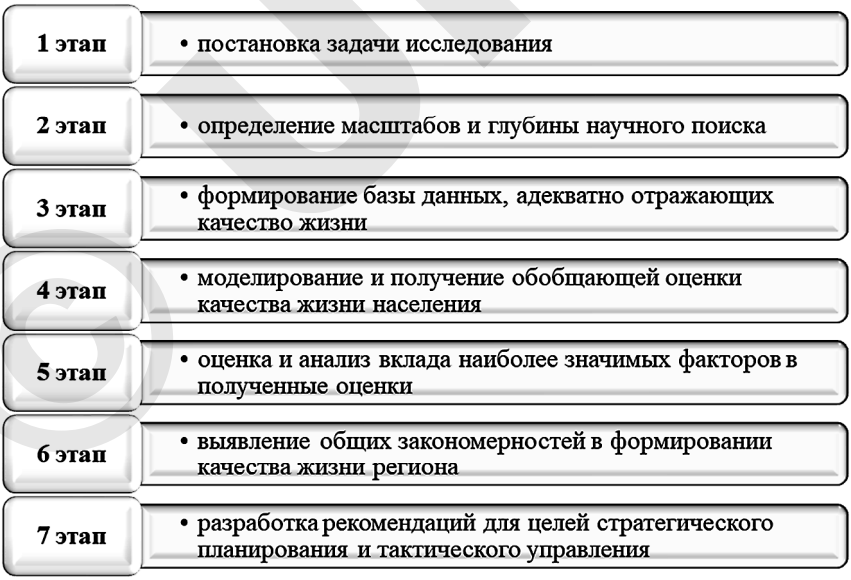


Рис. 3.2.3. Этапы методического обоснования оценки качества

В настоящее время известно более 15 отечественных и зарубежных методик количественной оценки качества жизни населения, основу оценок в которых составляют характеристики и показатели населения, среди которых: здоровье, уровень культуры и образованности, материальное благополучие, а также развитость региональной инфраструктуры, политическая стабильность и безопасность, комфортность климатических условий и состояние окружающей среды и многие другие. Сравнительный анализ наиболее известных методик оценки качества жизни визуализирован на рис. 3.2.4.

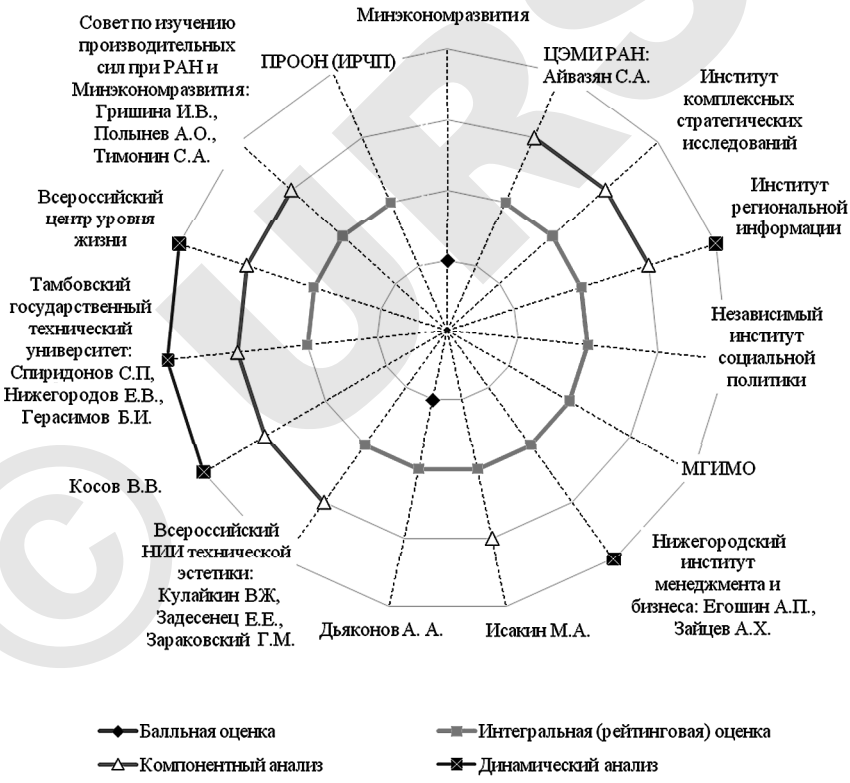


Рис. 3.2.4. Разные виды оценки и анализа в методиках оценки качества жизни населения

Моделирование и получение обобщающей оценки качества жизни на базе модели многоуровневой системы управления качеством делает востребованной количественную оценку качества жизни населения при выборе стратегических ориентиров, методов текущего управления, а также при формировании их критериев и шкал оценки качества жизни. В результате открываются новые перспективы для роста эффективности регионального управления в условиях цифровизации. Так, оценка качества жизни по методике, предложенной ИПРЭ РАН, наряду с совокупностью показателей, характеризующих здоровье, среду жизнедеятельности, безопасность, институциональные особенности региона, предполагает учет удельного веса инновационных товаров, работ, услуг в общем объеме отгруженных товаров, работ, услуг в регионе. Это отражает оригинальный взгляд авторов на проблемы современный экономики и отличает данную методику от большинства других методик.

Исходную базу моделирования составляет пространственная социально-экономическая система, которая рассматривается как совокупность субъектов РФ и представлена в заданный период времени — месяц, год и т. п. — в системе различных пространственных координат. В качестве основного «типологического инструмента» исследования рассматривается нелинейный типологический анализ, реализованный в функциональной форме на основе принципов многомерного шкалирования и в рамках «цифровой парадигмы».

Важнейшей составляющей методики ИПРЭ РАН является база данных «Качество жизни (База данных для построения модели качества жизни)» с закрепленным правом интеллектуальной собственности (рис. 3.2.5) [41].

База данных «Качество жизни (База данных для построения модели качества жизни)» представляет собой собрание справочных таблиц, объединенных в группы логическими связями. Таблицы БД содержат полный список первичных статистических показателей, использованных в построении модели качества жизни, первичные индикаторы для построения моделей методом главных компонент,

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации базы данных

№ 2021622426

БД Качество жизни (База данных для построения модели качества жизни)

Правообладатель: ***Федеральное государственное бюджетное учреждение науки Институт проблем региональной экономики Российской академии наук (RU)***

Авторы: ***Окрепилов Владимир Валентинович (RU), Шматко Алексей Дмитриевич (RU), Гагулина Наталья Львовна (RU)***

Заявка № **2021622344**
Дата поступления **28 октября 2021 г.**
Дата государственной регистрации
в Реестре баз данных ***09 ноября 2021 г.***

Руководитель Федеральной службы по интеллектуальной собственности

Г.П. Ивлиев

Рис. 3.2.5. Свидетельство о государственной регистрации базы данных «Качество жизни (База данных для построения модели качества жизни)»

результаты факторной корреляционной модели для стандартизированных индикаторов, результаты факторной ковариационной модели.

Основная цель создания базы данных состоит в развитии методов и инструментов измерения показателей качества жизни для анализа и моделирования качества жизни.

База данных «Качество жизни», обеспечивающая методику ИПРЭ РАН, предназначена для проведения пилотного моделирования показателей качества жизни и создания надежной основы для принятия стратегических решений, максимального использования ресурсных и производственных возможностей региона в интересах роста благосостояния населения и повышения качества жизни. Полученные результаты могут служить информационно-аналитической основой для формирования систем государственного управления регионами, учитывающих типологические особенности объектов управления. База данных может быть применена при выборе стратегических ориентиров управления, а также при формировании критериев и шкал оценки качества жизни для стандартизации системы государственного управлении регионами.

Основные технологические принципы комплексного исследования пространственных экономических систем регионального уровня сформулированы в ходе совместного исследования, проведенного в ИПРЭ РАН Центром региональных проблем экономики качества и Лабораторией математических методов анализа данных в 2020–2022 гг.

Значительные усилия были предприняты авторами по формированию системы первичных данных. Основное требование, поставленное к исходной информации, — адекватность целям и задачам проводимого исследования, а также статистическая представительность соответствующих данных. При формировании первичных данных использовались официальная статистическая методология и следующие информационные источники.

- Единая межведомственная информационно-статистическая система (ЕМИСС). Часть показателей ЕМИСС дублируется,

так как в разное время в системе имело место изменение методик подсчета показателей (один и тот же показатель может быть в нескольких вариантах — до 2013 г., с 2013–2016 гг., с 2017 г. и т. д.). Причем кодов у показателей в этой системе нет, поиск данных занимает много времени. По состоянию на 01.10.2021 — это 65 ведомств, 7944 показателя, собранных по 548 формам федерального статистического наблюдения.

- Витрина статистических данных — это относительно новый формат платформы для работы с данными Росстата (июль 2019), он дает больше возможностей для быстрого поиска и обработки данных, можно представлять в виде графиков и таблиц интересующие показатели, а также выгружать результат в различных форматах. Находится в состоянии наполнения данными из ЕМИСС. Каждый показатель имеет уникальный паспорт. В частности, следует отметить возможность выбора формы представления многомерных данных в Excel-формате. Данные в витрине постоянно пополняются, предсказать их появление затруднительно, поэтому процесс формирования БД требует мониторинга состояния данных в витрине.
- Публикации Росстата, сформированные в виде сборников: «Регионы России», «2019 — Инвестиции в России», «Россия в цифрах», «Рабочая сила, занятость и безработица в России (по результатам выборочных обследований рабочей силы, 2020)», «Итоги сплошного наблюдения за деятельностью субъектов малого и среднего предпринимательства за 2015 г., 29.09.2018». При этом отметим, что не все показатели, которые существуют в базах данных Росстата, печатаются в сборниках.

Кроме информации из указанных здесь источников, в проведенных исследованиях использованы также результаты обследований, выполненные в рамках официальной статистической методологии — выборочные обследования, результаты экспертных опросов и т. п.

И, наконец, обратим особое внимание на вопросы обеспечения проводимых исследований официальной статистической инфор-

мацией. В частности, необходимые данные отсутствуют для регионов, в которых безработица носит институциональный (структурный) характер, требующий профессионально-квалификационную дифференциацию. В этом случае необходимо формирование специального регионального информационного ресурса (с соблюдением требований ОСМ), для регионов, определение которых осуществляется в рамках специальной тематической типологии. В работах [42–45] представлен пример реализации такого подхода для Санкт-Петербурга.

На выбор состава показателей, выступающих в виде первичных данных и заложенных в основу моделирования качества жизни, повлияла система целевых показателей многоуровневой системы управления качеством. После сравнительного анализа сборников статистики разных лет показатели подобраны нами таким образом, чтобы максимально приблизить область исследования качества жизни к данной системе целевых показателей.

При проведении комплексной верификации сформированной системы первичных данных 2009–2022 гг. осуществлялись:

- стандартный «технический» контроль заполненных данных за 2009–2020 гг.;
- заполнение пропущенных значений в данных 2009–2020 гг.; при этом использовались аппараты статистических индексов и линейных (по параметрам) регрессионных моделей (всего — стандартный набор из 11 моделей);
- построение оценочных (прогнозных) значений для данных 2021–2022 гг.

В результате количество пропущенных значений в верифицированной системе первичных данных снизилось с 37,2 % (из которых 14,3 % — прогнозные) до 1,4 %.

Информационная база исследования охватывает данные Росстата для 82 субъектов РФ за 2009–2020 гг., включенные в систему первичных показателей контекстного блока моделирования «Качество жизни». Всего в первичную базу исследования вошло 24 494 валидных и 14 538 пропущенных значений. Последние составили 37,2 % от об-

щего количества значений (39 032), из которых 14,3 % относились к прогнозным оценкам на 2021 и 2022 гг. После проведения процедуры верификации первичных данных объем пропущенных значений снизился до 1,4 %.

Перед применением к данным различных алгоритмов анализа их структуры всегда возникает необходимость выбора способов предобработки, которая в задаче визуализации данных заключается в оцифровке, нормировке и выборе метрики. В дальнейшем предполагается, что значения предобработанных (первичных) данных для каждой переменной центрированы:

$$\frac{1}{N}\sum_{i=1}^{N} x_{i,l} = 0 \quad \forall \; l = 1, \ldots, m; \tag{3.2.1}$$

где $x_{i,l}$ — значение l-го показателя $x_i \epsilon X$.

Далее рассматриваются следующие типы первичных индикаторов: масштабные (S-индикаторы), удельные (U-индикаторы) и стандартизированные (Z-оценки[1]).

Масштабные индикаторы характеризуют вклад показателя по определенному субъекту РФ в соответствующий общероссийский показатель для конкретного года статистического измерения. Они определяются для значений статистического показателя, характеризующих состояния различных субъектов РФ, но в один и тот же интервал времени, а сумма значений показателя по всем субъектам РФ, образующим базу сравнения, равна 100 % [46]. Данные индикаторы измеряются в процентах и позволяют сравнивать значения показателей разной природы.

Удельные индикаторы (индикаторы группы «U») являются результатом всевозможных классификаций и измеряются в процентах.

Стандартизированные индикаторы получаются преобразованием из исходных с помощью стандартизации, заключающейся в центрировании средним значением и нормированием величиной среднеквадратического отклонения рассматриваемого показателя от своего среднего значения. Шкалы всех стандартизированных индикаторов в качестве «нуля» имеют свои средние значения, а

[1] В лексике IBM SPSS Statistics — Z-оценки (Z-scores).

в качестве масштаба («единицы») используется «единица» разброса значений — среднее квадратическое отклонение, что позволяет сравнивать значения различных индикаторов как попарно, так и в составе группы, в том числе независимо от их физической природы и т. п.

Многокритериальный (векторный) характер задачи оценки качества жизни с применением сложной системы индикаторов обусловил необходимость применения факторного анализа.

Факторный анализ является тем инструментом, который обеспечивает сведение оригинального многомерного вектора к вектору меньшей размерности при помощи стандартизованной математической методики, не зависящей от субъективного человеческого выбора. Переменные нового вектора состояния называются факторами, которые являются линейными комбинациями исходных переменных. Факторы или факторные переменные независимы по отношению друг другу. Задача факторного анализа состоит в формировании небольшого числа факторных компонент, соответствующих интегральным показателям качества жизни.

Согласно сформулированной нами гипотезе научного исследования, обобщенный показатель качества жизни является результирующим его основных интегральных составляющих. В соответствии с данной гипотезой и должна решаться задача факторного анализа. Этапы проведения факторного анализа методом главных компонент сведены на рис. 3.2.6. Выявленные при проведении факторного анализа компоненты представляют собой сложные линейные комбинации исходных показателей и имеют большое значение для проведения дальнейшего сравнительного анализа.

В числе методик, которые могут быть использованы для проведения сравнительного анализа полученных результатов, — известная отечественная методика, автором которой является С. А. Айвазян (ЦЭМИ РАН) [47]. Данная методика имеет фундаментальный характер, а интегральный показатель качества жизни, согласно методике, определяется качеством населения, благосостоянием населения, качеством социальной сферы, экологии и природно-климатическими условиями.

Рис. 3.2.6. Процесс проведения факторного анализа

© URSS

3.3. Исследование качества жизни в российских регионах

Опираясь на основные методические положения оценки качества жизни населения, предложенные в ИПРЭ РАН и обеспеченные базой данных «Качество жизни (База данных для построения модели качества жизни)» с закрепленным правом интеллектуальной собственности, нами проведено моделирование оценки качества жизни для регионов Российской Федерации. Приведем результаты совместных усилий Центра региональных проблем экономики качества и Лаборатории математических методов анализа данных ИПРЭ РАН по пилотному моделированию, реализованному с применением технологий классического метода главных компонент (далее — МГК) [48].

Полный список первичных статистических показателей, отобранных для проведения расчетов и использованных в исследовании, содержится в табл. 3.3.1. В качестве первичной базы данных (далее — БД) в проведенных исследованиях использовались данные Росстата для 82 субъектов РФ за 2009–2020 гг. Дополнительно в первичную БД были добавлены расчетные (оценочные и прогнозные) данные со значениями рассматриваемых показателей за 2021 и 2022 гг.

Таблица 3.3.1

Показатели первичной БД для построения модели качества жизни

Код	Наименование показателя	Единица измерения
g001	Общие коэффициенты рождаемости. Число родившихся на 1000 человек населения	чел.
g002	Общие коэффициенты смертности. Число умерших на 1000 человек населения	чел.
g003	Коэффициенты младенческой смертности. Число детей, умерших в возрасте до 1 года, на 1000 родившихся живыми	чел.
g004	Мощность амбулаторно-поликлинических организаций тысяч посещений в смену	тыс. посещений

Окончание таблицы 3.3.1

Код	Наименование показателя	Единица измерения
g005	Численность врачей всех специальностей на 10 000 человек населения	чел.
g006	Плотность населения, g006_1/g006_2 · 1000	чел./км^2
g007	Использование свежей воды	млн м^3
g008	Среднедушевые денежные доходы (в месяц)	руб.
g009	Потребительские расходы в среднем на душу населения (в месяц)	руб.
g010	Удельный вес расходов домашних хозяйств на оплату жилищно-коммунальных услуг в процентах от общей суммы потребительских расходов	%
g011	Численность учителей в организациях, осуществляющих образовательную деятельность по образовательным программам начального, основного среднего общего образования	тыс. чел.
g012	Численность обучающихся в общеобразовательных организациях	тыс. чел.
g013	Обеспеченность учителями в общеобразовательных школах, g011/g012	тыс. чел./ тыс. чел.
g014	Общая площадь жилых помещений, приходящаяся в среднем на одного жителя	м^2
g015	Объем коммунальных услуг на душу населения	руб.
g016	Объем транспортных услуг на душу населения	руб.
g017	Объем бытовых услуг на душу населения	руб.
g020	Квалифицированные безработные — всего	тыс. чел.
g021	Удельный вес квалифицированных безработных в общем количестве безработных	%
g022	Численность зрителей театров на 1000 человек населения	чел.
g023	Число посещений музеев на 1000 человек населения	—
g024	Число спортивных сооружений (спортивные залы)	—
g025	Детские оздоровительные лагеря. Численность детей, отдохнувших в ДОЛ за лето	тыс. чел.
g026	Коллективные средства размещения. Численность размещенных лиц	тыс. чел.

Код	Наименование показателя	Единица измерения
g027	Выбросы загрязняющих веществ, отходящих от стационарных источников, в атмосферный воздух	тыс. т
g028	Сброс загрязненных сточных вод в поверхностные водные объекты	млн $м^3$
g029	Объем оборотной и последовательно используемой воды	млн $м^3$
g030	Численность работников государственных органов и органов местного самоуправления	чел.
g031	Численность работников территориальных органов федеральных органов исполнительной власти	чел.
g032	Число зарегистрированных преступлений на 100 000 человек населения	—
g033	ВРП в текущих основных ценах — всего	млн руб.
g034	Удельный вес инновационных товаров, работ, услуг в общем объеме отгруженных товаров, работ, услуг в процентах от общего объема отгруженных товаров, выполненных работ, услуг	%

Всего сформированная первичная БД содержала 24 494 валидных и 14 538 пропущенных значений. Последние составляли 37,2 % от общего количества значений (39 032), из которых 14,3 % относились к прогнозным оценкам на 2021 и 2022 гг. Применение комплексной системы верификации данных, разработанной в Лаборатории математических методов анализа данных ИПРЭ РАН, позволило снизить объем пропущенных значений до 1,4 %.

Отметим, что первичными объектами статистического наблюдения в исследовании являлись субъекты РФ (далее — СРФ). За единичный (элементарный) объект указанной БД принято «состояние» СРФ, представленное данными рассматриваемых показателей за конкретный год. Последнее подчеркивает «панельный» характер используемой статистической информации.

Для получения оценок качества жизни, которые обладают всеми необходимыми характеристиками для применения в рамках многоуровневой системы управления качеством, к множеству первичных показателей, представленных в табл. 3.3.1, применены

принципы системы и комплексности. Согласно первому — в рамках общих метрологических схем рассматривались наиболее существенные форматы категоризации понятий предметной области «Качество жизни» (далее — КЖ). Реализация принципа комплексности сводилась к наличию в формируемой выборке показателей представителей всех рассматриваемых выделенных групп — категорий исследуемой предметной области. При этом первичные показатели имели критериальную форму представления, что дает возможность использовать их в качестве информационной базы для разработки целевых интегральных показателей в задачах анализа и прогнозного оценивания динамики регионального развития СРФ с позиций КЖ.

Далее, вся совокупность первичных показателей была разделена по уровню значимости на «масштабные» и «удельные» индикаторы (табл. 3.3.2). Масштабные индикаторы обладают большей значимостью для оценки КЖ в масштабе страны, а удельные — на региональном уровне.

Таблица 3.3.2

Первичные индикаторы для построения модели КЖ

№ п/п	Код	Наименование индикатора
Масштабные индикаторы		
1	**s_gП04**	% СРФ в: Мощность амбулаторно-поликлинических организаций
2	**s_gП07**	% СРФ в: Использование свежей воды
3	**s_gП11**	% СРФ в: Численность учителей в организациях, осуществляющих общеобразовательную деятельность
4	**s_gП12**	% СРФ в: Численность обучающихся в общеобразовательных организациях
5	**s_gП20**	% СРФ в: Квалифицированные безработные — всего
6	**s_gП24**	% СРФ в: Число спортивных сооружений (спортивные залы)
7	**s_gП25**	% СРФ в: Численность детей, отдохнувших в детских оздоровительных лагерях за лето
8	**s_gП26**	% СРФ в: Численность размещенных лиц в коллективных средствах размещения

№ п/п	Код	Наименование индикатора
9	**s_gП27**	% СРФ в: Выбросы загрязняющих веществ в атмосферный воздух, отходящих от стационарных источников
10	**s_gП28**	% СРФ в: Сброс загрязненных сточных вод в поверхностные водные объекты
11	**s_gП29**	% СРФ в: Объем оборотной и последовательно используемой воды
12	**s_gП30**	% СРФ в: Численность работников госорганов и органов местного самоуправления
13	**s_gП31**	% СРФ в: Численность работников территориальных органов федеральных органов исполнительной власти
14	**s_gП33**	% СРФ в: ВРП в текущих основных ценах — всего
Удельные (структурные) индикаторы		
15	**u_gП01**	Число родившихся на 1000 человек населения
16	**u_gП02**	Число умерших на 1000 человек населения
17	**u_gП03**	Число детей, умерших в возрасте до 1 года, на 1000 родившихся живыми
18	**u_gП05**	Численность врачей всех специальностей на 10 000 человек населения
19	**u_gП06**	Плотность населения (чел.)
20	**u_gП08**	Среднедушевые денежные доходы (в месяц)
21	**u_gП09**	Потребительские расходы в среднем на душу населения (в месяц)
22	**u_gП10**	Удельный вес расходов домашних хозяйств на оплату жилищно-коммунальных услуг (% от общей суммы потребительских расходов)
23	**u_gП13**	Обеспеченность учителями в общеобразовательных школах
24	**u_gП14**	Общая площадь жилых помещений, приходящаяся в среднем на одного жителя
25	**u_gП15**	Объем коммунальных услуг на душу населения
26	**u_gП16**	Объем транспортных услуг на душу населения
27	**u_gП17**	Объем бытовых услуг на душу населения
28	**u_gП21**	Удельный вес квалифицированных безработных в общем количестве безработных
29	**u_gП22**	Численность зрителей театров на 1000 человек населения

Окончание таблицы 3.3.2

№ п/п	Код	Наименование индикатора
30	**u_gП23**	Число посещений музеев на 1000 человек населения
31	**u_gП30**	Кол-во работников госорганов и органов местного самоуправления на 1 федерального чиновника
32	**u_gП32**	Число зарегистрированных преступлений на 100 000 человек населения
33	**u_gП34**	Удельный вес инновационных товаров, работ, услуг в общем объеме отгруженных товаров, работ, услуг

В дальнейшем при построении моделей используется система Z-значений (Z-score), построенная для системы первичных индикаторов (как масштабных, так и удельных). Указанная система показателей может быть представлена в виде трех состояний.

1. Как самостоятельная система всей совокупности показателей, характеризующих категорию «Качество жизни населения РФ».

2. Как выборка из общей системы показателей, нашедших применение в многоуровневой системе управления качеством:

- *группа 1:* демографическая обстановка;
- *группа 2:* качество жилья;
- *группа 3:* качество окружающей среды;
- *группа 4:* качество услуг здравоохранения;
- *группа 5:* развитие физической культуры и спорта;
- *группа 6:* качество образовательных услуг;
- *группа 7:* качество услуг культурно-досуговой сферы;
- *группа 8:* экономическое развитие территории;
- *группа 9:* уровень доходов населения;
- *группа 10:* кадры государственных органов и органов местного самоуправления;
- *группа 11:* преступность.

3. Как выборка из общей системы показателей, характеризующих подходы научной школы академика Армении С. А. Айвазяна (далее — подход ЦЭМИ РАН):

- *группа 1*: качество населения;
- *группа 2*: благосостояние (уровень жизни населения);
- *группа 3*: качество социальной сферы;
- *группа 4*: качество экологической ниши;
- *группа 5*: природно-климатические условия;
- *группа 6*: экономический потенциал жизненной среды (среды жизнеобитания), экономическое развитие территорий.

Ниже приведены соответствующие данные, использованные при реализации двух указанных подходов. В частности, введены специальные нормировки первичных показателей — индикаторов и проведен сравнительный анализ их метрологической емкости (мощности). В качестве последней рассматривается % объясненной дисперсии для выбранной системы представления в рамках построенных на их основе факторных моделей, реализующих классический метод главных компонент в ковариационном формате. В табл. 3.3.3 и 3.3.4 представлены результаты отнесения первичных индикаторов к указанным группировкам показателей качества жизни.

Таблица 3.3.3

Распределение показателей первичной БД в категоризации многоуровневой системы управления качеством

№ группы показателей КЖ в соответствии с группировкой (1)	**Количество показателей в группе**	**№ показателя в первичной БД (таблице 3.3.2)**
1	4	15–17, 19
2	1	24
3	4	2, 9–11
4	2	1, 18
5	1	6
6	3	3, 4, 23
7	4	7, 8, 29, 30
8	4	5, 14, 28, 33
9	6	20–22, 25
10	3	12, 13, 31
11	1	32

Таблица 3.3.4

Распределение показателей первичной БД
в категоризации методики ЦЭМИ РАН

№ группы показателей КЖ в соответствии с группировкой (2)	Количество показателей в группе	№ показателя в первичной БД (табл. 3.3.2)
1	4	15–17, 19
2	9	1, 18, 20–22, 24–27
3	9	3, 4, 6–8, 23, 29, 30, 32
4	4	3, 9–11
5	совпадающих показателей в данной группе нет	
6	7	5, 12–14, 28, 31, 33

Далее, при проведении сравнительного анализа представлений указанных масштабных и удельных индикаторов КЖ использовались следующие схемы стандартизации как технологии предобработки данных:

- построение для рассматриваемой системы индикаторов совокупности соответствующих Z-оценок;
- разработка систем W-оценок, как результата соответствующих нормировок системы Z-индикаторов.

В последнем случае реализуется метод «взвешивания» Z-индикаторов в рамках рассматриваемой группировки показателей предметной области объектов исследования, удовлетворяющий следующему принципу «равномерности»:

- суммарный вес показателей, отнесенных к одной группе, одинаков для любой группы рассматриваемой категоризации, например, равен 1;
- в рамках одной группы все показатели имеют один и тот же вес; например, $1/ng$, где ng — количество показателей — индикаторов, отнесенных к рассматриваемой группе.

© URSS

В результате реализации указанных принципов для системы Z-индикаторов для приведенных выше группировок, используя данные табл. 3.3.3 и 3.3.4, получаем две системы W-индикаторов — соответственно W1 и W2.

Ниже приведены результаты анализа указанных систем индикаторов классическим МГК — в корреляционном формате для Z-индикаторов и в ковариационном — для W-индикаторов. Для построенных корреляционных и ковариационных моделей ниже приводятся данные для четырех главных компонент (соответствующие выделенные суммы квадратов и нагрузок извлечения). Сводные данные представлены в табл. 3.3.5.

Таблица 3.3.5

Общие модели классического МГК для систем индикаторов «Качество жизни населения РФ»

Компонента (фактор)	**Начальные собственные значения**[1]		
	Значение	**% дисперсии**	**Кумулятивный %**
Система Z-индикаторов			
1	10,69	32,41	**32,41**
2	4,69	14,21	**46,62**
3	2,88	8,74	**55,36**
4	1,95	5,92	**61,27**
от 05 до 33	от 1,58 до 0,01		38,73
Система W1-индикаторов			
1	1,87	35,10	**35,10**
2	1,35	25,36	**60,46**
3	0,78	14,61	**75,07**
4	0,34	6,28	**81,36**
от 05 до 33	от 0,15 до 0,01		18,64

[1] Здесь и далее в тексте и в таблицах выделены суммы квадратов нагрузок извлечения.

© URSS

Окончание таблицы 3.3.5

Компонента (фактор)	Начальные собственные значения		
	Значение	% дисперсии	Кумулятивный %
Система масштабных W1-индикаторов			
1	1,78	**81,22**	**81,22**
2	0,14	6,40	87,62
3	0,07	3,10	90,72
4	0,06	2,61	93,33
Система удельных W1-индикаторов			
1	1,360	**43,335**	43,335
2	0,784	**24,983**	68,319
3	0,340	**10,822**	**79,141**
4	0,139	4,438	83,579
5	0,098	3,130	86,709
6	0,081	2,568	89,277
7–19	от 0,06 до 0,00		10,723
Система W2-индикаторов			
1	0,25	28,84	**28,84**
2	0,15	17,74	**46,59**
3	0,10	11,58	**58,16**
4	0,06	7,10	**65,27**
от 05 до 33	от 5,40 до 0,01		34,73

Система Z-индикаторов (корреляционная модель). Всего было отобрано четыре компоненты с общей «объясняющей силой» **61,3 %**. Причем первый фактор имеет самую высокую «объясняющую силу» (32,4 %), второй — 14,26 %, третий и четвертый примерно равнозначны по «мощности» второму (соответственно, 8,7 % и 5,9 %).

Система W1-индикаторов (ковариационная модель). Всего было отобрано четыре компоненты с общей «объясняющей силой» **81,3 %**. Причем первый фактор имеет самую высокую «объясняющую силу» (35,1 %), второй — 20,9 %, третий и четвертый примерно равнозначны по «мощности» второму (соответственно, 14,6 % и 6,3 %).

Система W2-индикаторов (ковариационная модель). Всего было отобрано четыре компоненты с общей «объясняющей силой» **65,3 %**. Причем первый фактор имеет самую высокую «объясняющую силу» (28,8 %), второй — 17,7 %, третий и четвертый примерно равнозначны по «мощности» второму (соответственно, 11,6 % и 7,1 %).

Проанализируем представленные ковариационные модели МГК, основываясь на понятии **метрологической емкости** (мощности) группировки показателей КЖ относительно заданного множества показателей качества жизни. Величину метрологической емкости будем определять как информационную значимость (долю объясненной дисперсии) для ковариационной модели МГК при заданном количестве извлеченных компонент.

В результате приходим к выводу, что по уровню метрологической емкости модель, основанная на показателях, составляющих выборку согласно подходу ЦЭМИ РАН (модель «КЖ-ЦЭМИ»), уступает модели, построенной на основе выборки показателей, принятой согласно методике оценки качества жизни ИПРЭ РАН. В первом случае мы имеем значения: **65,3 %** для четырех извлеченных компонент и **46,6 %** — для двухфакторной модели, а во втором случае: **81,4 %** и **60,5 %**, соответственно. Поэтому при проведении последующего анализа представляется оправданным применение метода главных компонент лишь для системы **W1-индикаторов** (масштабных и удельных).

В связи с эти рассмотрим матрицу компонент для построенной факторной модели согласно Методике ИПРЭ РАН. В табл. 3.3.6 приведены лишь данные, для которых соответствующие факторные веса (коэффициенты корреляции) превосходят установленные пороговые значения 0,4.

Таблица 3.3.6

Матрица компонент для факторной модели согласно методики ИПРЭ РАН

Код индикатора	Наименование индикатора	Компонента			
		1	2	3	4
W1s_gП24	W1-знач.: % СРФ в: Число спортивных сооружений (спортивные залы)	**0,97**			
W1s_gП12	W1-знач.: % СРФ в: Численность обучающихся в общеобразовательных организациях	**0,95**			
W1s_gП11	W1-знач.: % СРФ в: Численность учителей в организациях, осуществляющих общеобразовательную деятельность	**0,95**			
W1s_gП31	W1-знач.: % СРФ в: Численность работников территориальных органов федеральных органов исп. власти	**0,94**			
W1s_gП30	W1-знач.: Кол-во работников госорганов и органов местного самоуправления на 1 федерального чиновника	**0,93**			
W1s_gП04	W1-знач.: % СРФ в: Мощность амбулаторно-поликлинических организаций	**0,93**			
W1s_gП33	W1-знач.: % СРФ в: ВРП в текущих основных ценах — всего	**0,80**			
W1s_gП26	W1-знач.: % СРФ в: Численность размещенных лиц в коллективных средствах размещения	**0,79**			
W1s_gП28	W1-знач.: % СРФ в: Сброс загрязненных сточных вод в поверхностные водные объекты	**0,78**			
W1s_gП25	W1-знач.: % СРФ в: Численность детей, отдохнувших в детских оздоровительных лагерях за лето	**0,65**			–0,41
W1s_gП29	W1-знач.: % СРФ в: Объем оборотной и последовательно используемой воды	0,55			
W1u_gП06	W1-знач.: Плотность населения (чел.)	0,55			0,48

Код индикатора	Наименование индикатора	Компонента			
		1	2	3	4
W1u_gП09	W1-знач.: Потребительские расходы в среднем на душу населения (в месяц)	0,51			0,46
W1s_gП07	W1-знач.: % СРФ в: Использование свежей воды	0,46			
W1u_gП14	W1-знач.: Общая площадь жилых помещений, приходящаяся в среднем на одного жителя		**–0,86**	0,50	
W1u_gП32	W1-знач.: Число зарегистрированных преступлений на 100 000 человек населения		**0,69**	0,67	
W1u_gП01	W1-знач.: Число родившихся на 1000 человек населения		**0,69**		
W1u_gП03	W1-знач.: Число детей, умерших в возрасте до 1 года, на 1000 родившихся живыми		0,48		
W1u_gП02	W1-знач.: Число умерших на 1000 человек населения		–0,42	0,53	
W1u_gП15	W1-знач.: Объем коммунальных услуг на душу населения			0,42	
W1u_gП05	W1-знач.: Численность врачей всех специальностей на 10 000 человек населения				**0,82**
W1u_gП22	W1-знач.: Численность зрителей театров на 1000 человек населения	0,40			0,60
W1u_gП08	W1-знач.: Среднедушевые денежные доходы (в месяц)	0,47			0,50
W1u_gП16	W1-знач.: Объем транспортных услуг на душу населения	0,40			0,46
W1u_gП23	W1-знач.: Число посещений музеев на 1000 человек населения				0,42
W1u_gП30	W1-знач.: Кол-во работников госорганов и органов местного самоуправления на 1 федерального чиновника				–0,4

GURSS

Как показывают полученные результаты, наиболее существенными являются две первые компоненты (их суммарная информативность –65 %). Причем в первой компоненте доминирует 10 масштабных индикаторов (с изменением факторных весов от 0,97 до 0,65). Во второй компоненте преобладает 3 удельных индикатора с факторными весами разной направленности, абсолютные величины которых находятся в интервале от 0,86 до 0,69. Разные знаки факторных весов здесь интерпретируется как наличие определенных балансовых соотношений для соответствующих индикаторов качества жизни. Еще 12 типологизирующих индикаторов (2 масштабных и 10 удельных) находятся в зоне относительно высокой информативности.

Таким образом, в построенных моделях четко проявляются два заложенных в систему первичных индикаторов начала: масштабность (вклад в общесистемную эффективность) и удельная значимость (локальная эффективность). Поэтому представляется вполне целесообразным моделировать эти качества раздельно. Например, в форме реализации методом главных компонент отдельно для масштабных и удельных индикаторов с последующим совместным анализом двух построенных моделей (как правило, с решением возникающих задач «балансировки»).

Проанализируем данные построенных факторных ковариационных моделей для масштабных и удельных индикаторов качества жизни, приведенных нами ранее в табл. 3.3.2. Информация о величине объясненной дисперсии для первых четырех компонент (факторов) моделей содержится в табл. 3.3.7.

В системе масштабных W1-индикаторов всего было отобрано четыре компоненты с общей «объясняющей силой» **93,3 %**. Первый фактор обладает самой высокой «объясняющей силой» (**81,2 %**), что, по существу, означает одномерность модели.

В системе удельных W1-индикаторов отобрано шесть компонент с общей «объясняющей силой» **89,3 %.** Первый фактор имеет самую высокую «объясняющую силу» (**43,3 %**). Построенная модель,

Таблица 3.3.7

Ковариационные модели, полученные методом главных компонент

Компонента (фактор)	Начальные собственные значения		
	Значение	% дисперсии	Кумулятивный %
ФМ «Масш». Система масштабных W1-индикаторов			
1	**1,78**	**81,22**	**81,22**
2	0,14	6,40	87,62
3	0,07	3,10	90,72
4	0,06	2,61	**93,33**
ФМ «Уд». Система удельных W1-индикаторов			
1	**1,360**	**43,335**	43,335
2	**0,784**	**24,983**	68,319
3	**0,340**	**10,822**	**79,141**
4	0,139	4,438	83,579
5	0,098	3,130	86,709
6	0,081	2,568	89,277
7–19	от 0,06 до 0,00		10,723

по существу, является трехмерной. Сгладить информационные различия трех выделенных компонент можно было бы за счет вращения, но мы для этого в дальнейшем прибегнем к использованию аппарата R-шкалирования и применим гибридные модели МГК, основанные на синтезе классического МГК и функциональных моделей многомерного метрического шкалирования [49].

В табл. 3.3.8 и 3.3.9 приведены матрицы компонент — значений коэффициентов корреляции типологизируемых индикаторов и построенных компонент для соответствующих ковариационных моделей, полученных методом главных компонент.

Таблица 3.3.8

Матрица компонент для факторной модели «КЖ-2021_Масш»

Код индикатора	Наименование W1-значений индикаторов	Компонента	
		1	2
W1s_gП024	% СРФ в: Число спортивных сооружений (спортивные залы)	**0,985**	
W1s_gП012	% СРФ в: Численность обучающихся в общеобразовательных организациях	**0,962**	
W1s_gП011	% СРФ в: Численность учителей в организациях, осуществляющих общеобразовательную деятельность	**0,949**	
W1s_gП030	% СРФ в: Численность работников территориальных органов федеральных органов исполнительной власти	**0,946**	
W1s_gП031	% СРФ в: Кол-во работников госорганов и органов местного самоуправления на 1 федерального чиновника	**0,945**	
W1s_gП004	% СРФ в: Мощность амбулаторно-поликлинических организаций	**0,929**	
W1s_gП033	% СРФ в: ВРП в текущих основных ценах — всего	**0,794**	0,401
W1s_gП028	% СРФ в: Сброс загрязненных сточных вод в поверхностные водные объекты	**0,784**	
W1s_gП026	% СРФ в: Численность размещенных лиц в коллективных средствах размещения	**0,778**	0,522
W1s_gП025	% СРФ в: Численность детей, отдохнувших в детских оздоровительных лагерях за лето	**0,692**	–0,560
W1s_gП029	% СРФ в: Объем оборотной и последовательно используемой воды	**0,585**	–0,456
W1s_gП027	% СРФ в: Квалифицированные безработные — всего	0,420	**–0,539**
W1s_gП020	% СРФ в: Квалифицированные безработные — всего		
W1s_gП007	% СРФ в: Использование свежей воды	0,461	

Таблица 3.3.9

Матрица компонент для факторной модели «КЖ-2021_Уд»

Код индикатора	Наименование W1-значений индикаторов	Компонента			
		1	2	3	4
W1u_gП014	Общая площадь жилых помещений, приходящаяся в среднем на одного жителя	**0,848**	**0,522**		
W1u_gП032	Число зарегистрированных преступлений на 100 000 человек населения	**–0,736**	**0,675**		
W1u_gП001	Число родившихся на 1000 человек населения	**–0,696**			
W1u_gП003	Число детей, умерших в возрасте до 1 года, на 1000 родившихся живыми	**–0,505**			
W1u_gП015	Объем коммунальных услуг на душу населения	0,407			
W1u_gП017	Объем бытовых услуг на душу населения				
W1u_gП034	Удельный вес инновационных товаров, работ, услуг в общем объеме отгруженных товаров, работ, услуг				
W1u_gП002	Число умерших на 1000 человек населения		**0,557**		
W1u_gП010	Удельный вес расходов домашних хозяйств на оплату жилищно-коммунальных услуг (% от общей суммы потребительских расходов)				
W1u_gП005	Численность врачей всех специальностей на 10 000 человек населения			**0,892**	
W1u_gП022	Численность зрителей театров на 1000 человек населения			**0,709**	

© URSS

Окончание таблицы 3.3.9

Код индикатора	Наименование W1-значений индикаторов	Компонента			
		1	**2**	**3**	4
W1u_gП008	Среднедушевые денежные доходы (в месяц)			**0,643**	
W1u_gП009	Потребительские расходы в среднем на душу населения (в месяц)			**0,622**	
W1u_gП016	Объем транспортных услуг на душу населения			**0,579**	
W1u_gП006	Плотность населения (чел.)			**0,549**	
W1u_gП023	Число посещений музеев на 1000 человек населения				
W1u_gП013	Обеспеченность учителями в общеобразовательных школах, g011/g012				**–0,821**
W1u_gП021	Удельный вес квалифицированных безработных в общем количестве безработных				**0,523**
W1u_gП030	Кол-во работников госорганов и органов местного самоуправления на 1 федерального чиновника			–0,444	0,496

Далее вводим систему интегральных оценок уровня масштабности и уровня структурной значимости состояния *x* субъектов РФ.

Интегральные оценки уровня масштабности:

$$Ind\mathrm{Ma}(x) = \frac{1}{\sigma\mathrm{Ma}} \sum_{j=1,2,3,4} \sigma\mathrm{Ma}_j F\mathrm{Ma}_j\,(x);\ \ \forall\, x\ \in X, \qquad (3.3.1)$$

где $F\mathrm{Ma}_j$ — значение j-й компоненты (фактора) модели ***«КЖ-2021_Масш»***, которые для любого состояния субъекта вычисляются с помощью матрицы коэффициентов оценок компонент (индиви-

дуальных оценок факторов) построенной модели; σМа$_j$ — % объясненной дисперсии фактором *FMa$_j$*. Суммарно:

$$\sigma\text{Ма} = \sum_{o=1,2,3,4} \sigma\text{Ма}_j.$$

Интегральные оценки уровня структурной значимости:

$$Ind\text{Уд}(x) = \frac{1}{\sigma\text{Уд}} \sum_{j=1,2,3,4} \sigma\text{Уд}_j F\text{Уд}_j(x); \ \forall x \in X, \qquad (3.3.2)$$

где *FУд$_j$* — значение *j*-й компоненты (фактора) модели ***«КЖ-2021_Уд»***, которые для любого состояния ОСН вычисляются с помощью матрицы коэффициентов оценок компонент (индивидуальных оценок факторов); σУд$_j$ — % объясненной дисперсии фактором *F*Уд$_j$, формирующий сумму:

$$\sigma\text{Уд} = \sum_{o=1,2,3,4} \sigma\text{Уд}_j.$$

Далее, вычислим стандартизированные представления (Z-оценки) индикаторов $Ind\text{Ма}(x)$ и $Ind\text{Уд}(x)$ и проведем квантификацию $ZInd\text{Ма}(x)$, $ZInd\text{Уд}(x)$, согласно условию:

$$\text{Ур}ZInd = \begin{cases} 1 \text{ (экстремально низкий), если } ZInd < -3; \\ 2 \text{ (низкий уровень), если } -3 \le ZInd < -0{,}5; \\ 3 \text{ (средний уровень), если } -0{,}5 \le ZInd < 0{,}5; \\ 4 \text{ (высокий уровень), если } 0{,}5 \le ZInd < 3; \\ 5 \text{ (экстремально высокий), если } 3 \le ZInd. \end{cases} \qquad (3.3.3)$$

Распределения полученных интегральных индикаторов уровня масштабности и структурной значимости УрМа и УрУД представлены в табл. 3.3.10 и 3.3.11. Результаты, полученные в разрезе федеральных округов, сведены в табл. 3.3.12–3.3.15.

Всего в табл. 3.3.10 представлено 846 элементарных объектов БД — состояний СРФ. Заметим, что состояние СРФ участвует в процессе применения МГК, в нем отсутствуют пропущенные значения (все значения валидны).

© URSS

Таблица 3.3.10

Представление сопряженности интегральных индикаторов масштабности и структурной значимости — количество состояний СРФ

Уровни масштабности (УрМа)	Уровни значений интегрального индикатора структурной значимости (УрУД)					
	Экстремально низкий	Низкий	Средний	Высокий	Экстремально высокий	Итого
Экстремально низкий		1				1
Низкий		40	109	65		214
Средний	8	125	229	132		494
Высокий		21	74	24	5	124
Экстремально высокий			4	9		13
Итого	8	187	416	230	5	846

Как следует из табл. 3.3.10:

1) состояния СРФ с экстремально низкими значениями интегральных индикаторов КЖ, как масштабных, так и удельных, практически отсутствуют (соответственно, 1 и 8 позиций);
2) экстремально высокие уровни масштабности имеет лишь Москва (13 позиций). Более подробную информацию можно получить при рассмотрении результатов картирования для построенной модели. При этом локальная эффективность КЖ (интегральный индикатор структурной значимости) достигает лишь уровней «средний» и высокий — соответственно 4 и 9 позиций;
3) основная масса состояний СРФ распределена в «квадрате» ячеек с уровнями «низкий — средний — высокий» для обоих интегральных индикаторов КЖ (табл. 3.3.11).

К доминантным уровням относятся состояния СРФ, которые попадают в ячейки табл. 3.3.11 с наполнением выше 50 %. Более подробная детализация может быть получена в рамках разработанной системы картирования результатов МГК.

© URSS

Таблица 3.3.11

Доминантные уровни интегральных индикаторов масштабности и структурной значимости

Уровни масштабности	Интегральный индикатор структурной значимости			
	Низкий	Средний	Высокий	Итого
	% в уровне структурной значимости			
Низкий	21,4	26,2	28,3	
Средний	**66,8**	**55,0**	**57,4**	
Высокий	11,2	17,8	10,4	
Итого	99,5 %	99,0 %	96,1 %	
	% в уровне масштабности			
Низкий	18,7	**50,9**	30,4	100 %
Средний	25,3	46,4	26,7	98,4 %
Высокий	16,9	**59,7**	19,4	96,0 %

Таблица 3.3.12

Распределения индикатора интегрального уровня структурной значимости для федеральных округов РФ на основании ковариационной модели МГК, построенной для удельных W1-индикаторов состояний субъектов РФ

Наименование ФО	Уровень структурной значимости для ФО					
	Экстремально низкий	Низкий	Средний	Высокий	Экстремально высокий	Итого
Центральный ФО		8	96	96		200
Северо-Западный ФО		3	60	46	5	114
Южный ФО		31	35	8		74
Северо-Кавказский ФО	4	47	3	1		55
Приволжский ФО		16	99	37		152
Уральский ФО		18	32	1		51
Сибирский ФО	3	48	54	11		116
Дальневосточный ФО	1	16	37	30		84
Итого	8	187	416	230	5	846

© URSS

Из табл. 3.3.12 следует уточнение данных табл. 3.3.10.

1. Практически отсутствуют состояния СРФ с экстремально низким уровнем интегрального индикатора структурной значимости. Всего 8 позиций: в Северо-Кавказском, Сибирском и Дальневосточном ФО России (соответственно, в количестве 4, 3 и 1).
2. Экстремально высокие уровни структурной значимости имеет лишь Северо-Западный ФО (Санкт-Петербург) — 5 позиций. Более подробная информация — в результатах картирования для построенной модели.
3. Основная масса состояний СРФ распределена в интервале значений уровней «низкий — средний — высокий».

Таблица 3.3.13

Доминантные уровни интегральных индикаторов структурной значимости для ФО России

Наименование ФО	**Интегральный индикатор структурной значимости**			
	Низкий	**Средний**	**Высокий**	**Итого**
	% в уровне структурной значимости			
Центральный ФО	4,3	**23,1**	**41,7**	—
Северо-Западный ФО	1,6	14,4	**20**	—
Южный ФО	16,6	8,4	3,5	—
Северо-Кавказский ФО	25,1	0,7	0,4	—
Приволжский ФО	8,6	**23,8**	16,1	—
Уральский ФО	9,6	7,7	0,4	—
Сибирский ФО	**25,7**	13	4,8	—
Дальневосточный ФО	8,6	8,9	13	—
Итого	100	100	100	—
	% в федеральном округе			
Центральный ФО	4	**48**	48	100
Северо-Западный ФО	2,6	52,6	40,4	95,6
Южный ФО	41,9	**47,3**	10,8	100
Северо-Кавказский ФО	85,5	**5,5**	**1,8**	92,7
Приволжский ФО	10,5	65,1	24,3	100
Уральский ФО	35,3	**62,7**	2	100
Сибирский ФО	41,4	**46,6**	**9,5**	97,4
Дальневосточный ФО	19	44	35,7	98,8

В рассматриваемой ситуации к ФО с доминантным уровнем структурной значимости отнесем ФО, которые попадают в ячейки табл. 3.3.13 с наполнением выше 20 %. При этом Центральный ФО — вследствие экстремально высокого уровня структурной значимости Москвы. Более подробная детализация также может быть получена в рамках разработанной системы картирования результатов МГК. В итоге доминантные ФО, в которых доля состояний СРФ с высоким уровнем структурной значимости превосходит 20 %: Центральный, Северо-Кавказский, Сибирский, Северо-Западный, Приволжский и Дальневосточный ФО.

Таблица 3.3.14

Распределения индикатора интегрального уровня масштабности для Федеральных округов РФ на основании ковариационной модели МГК, построенной для масштабных W1-индикаторы состояний субъектов РФ

Наименование ФО	Интегральный индикатор уровня масштабности для ФО					
	Экстремально низкий	Низкий	Средний	Высокий	Экстремально высокий	Итого
Центральный ФО		60	107	20	13	200
Северо-Западный ФО			76	38		114
Южный ФО		6	56	12		74
Северо-Кавказский ФО			43	12		55
Приволжский ФО	1	91	57	3		152
Уральский ФО		24	13	14		51
Сибирский ФО		21	70	25		116
Дальневосточный ФО		12	72			84
Итого	1	214	494	124	13	846

Из табл. 3.3.14 следует:

1. Практически отсутствуют состояния СРФ с экстремально низкими значениями интегральных масштабных индикаторов КЖ (всего 1 позиция).
2. Экстремально высокие уровни масштабности имеет лишь Центральный ФО (за счет Москвы — 13 позиций).

3. Как и ранее, для интегрального индикатора масштабности основная масса состояний ФО распределена в «квадрате» ячеек с уровнями «низкий — средний — высокий».

Таблица 3.3.15

Доминантные уровни интегральных индикаторов масштабности для ФО России

Наименование ФО	Интегральный индикатор масштабности			
	Низкий	Средний	Высокий	Итого
	% в уровне масштабности			
Центральный ФО	**28,0**	**21,7**	16,1	
Северо-Западный ФО		15,4	**30,6**	
Южный ФО	2,8	11,3	9,7	
Северо-Кавказский ФО		8,7	9,7	
Приволжский ФО	**42,5**	11,5	2,4	
Уральский ФО	11,2	2,6	11,3	
Сибирский ФО	9,8	14,2	**20,2**	
Дальневосточный ФО	5,6	14,6		
Итого	100 %	100 %	100 %	
	% в Федеральном округе			
Центральный ФО	**30,0**	**53,5**	10,0	93,5 %
Северо-Западный ФО		**66,7**	**33,3**	100 %
Южный ФО	8,1	**75,7**	16,2	100 %
Северо-Кавказский ФО		**78,2**	**21,8**	100 %
Приволжский ФО	**59,9**	**37,5**	2,0	100 %
Уральский ФО	**47,1**	**25,5**	27,5	100 %
Сибирский ФО	18,1	**60,3**	**21,6**	100 %
Дальневосточный ФО	14,3	**85,7**		100 %

Доминантность с высоким уровнем масштабности в РФ характерна для двух ФО: Северо-Западного и Сибирского, а доминантность с низким уровнем наблюдается в Центральном и Приволжском ФО.

В региональном масштабе все ФО являются доминантными на среднем уровне. При этом лишь три ФО являются доминантными на высоком уровне: Северо-Западный, Северо-Кавказский и Сибирский ФО, При этом высокий уровень в первом случае определяется Санкт-

Петербургом. Низкий же уровень доминантности в рассматриваемом смысле представлен Приволжским, Уральским и Центральным ФО.

Примеры визуализации результатов построенных моделей приведены ниже (рис. 3.3.1, 3.3.2).

В заключение отметим, что представленная система моделирования оценки качества жизни для регионов Российской Федерации в настоящее время получила развитие как система функциональных локально-линейных моделей типологизации регионов, представленных небольшим количеством по сравнению с исходными показателями — первичными индикаторами. При этом, как правило, количество интегральных индикаторов составляет не более двух. Один из них аккумулирует масштабные первичные индикаторы, а другой — удельные (структурные). В случае, когда интегральных индикаторов оказывается более двух, применяются специальные инструменты их декомпозиции и сравнительного анализа.

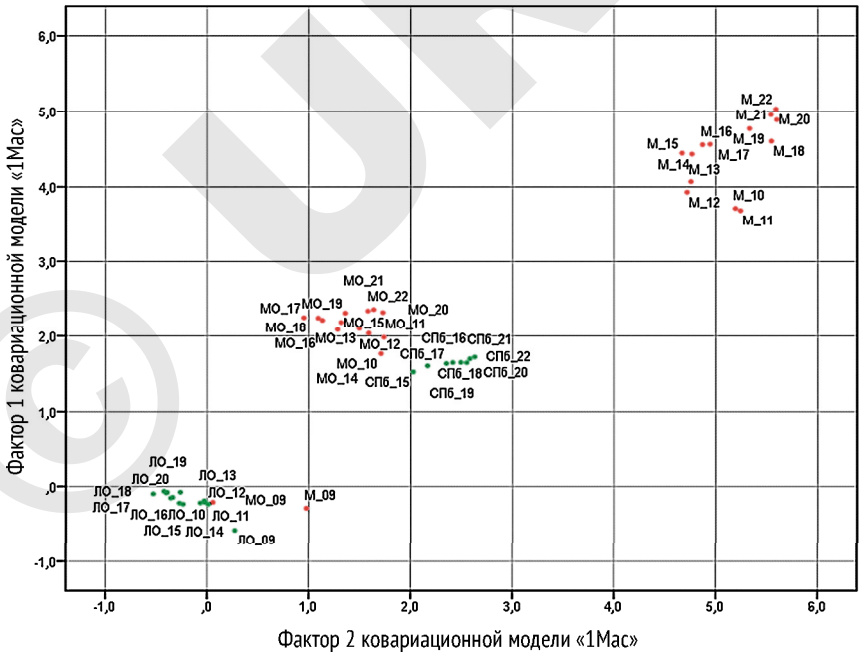


Рис. 3.3.1. Москва, Санкт-Петербург, Московская и Ленинградская области на диаграмме рассеивания (модель для W1-индикаторов)

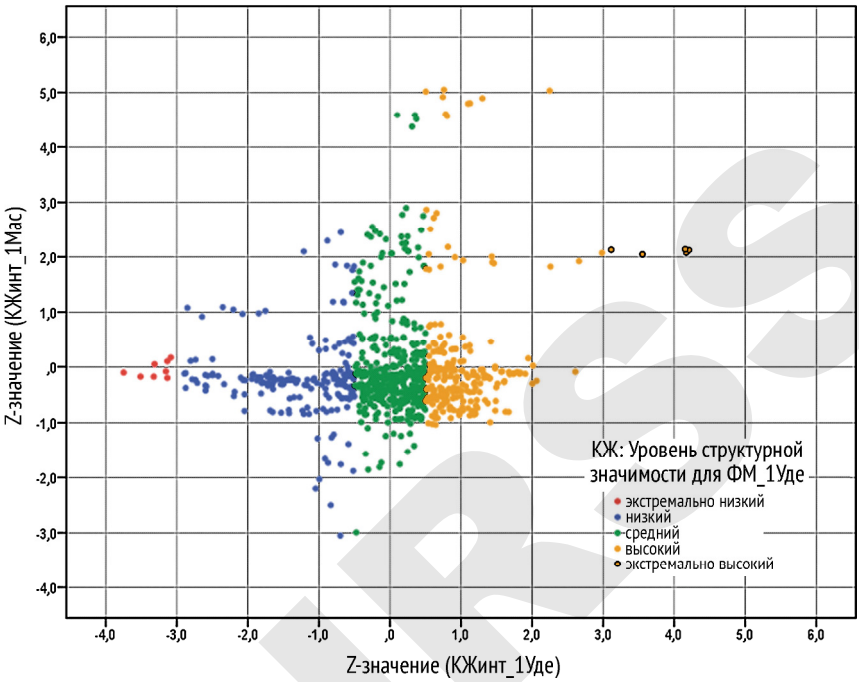


Рис. 3.3.2. Индикация уровня структурной значимости (модели для W1-индикаторов)

В частности, гибридный МГК, являющийся синтезом классического МГК и функциональных моделей ММШ [50–53].

Для построенной совокупности интегральных индикаторов формируется система типов состояний регионов. Как правило, количество типов относительно невелико, а временна́я траектория перемещения состояний каждого региона в построенном типологическом пространстве характеризуется (кроме координат в типологическом пространстве) принадлежностью региона к конкретному типу и интегральными динамическими характеристиками масштабности и структурной значимости.

Теоретическая значимость проведенного научного исследования состоит в совместном применении идей экономики качества и методологии разработки больших агент-ориентированных моде-

лей. Полученные результаты представляют большой интерес в связи с их перспективностью для дальнейшего применения в целях регионального управления, поскольку создают основу для пространственно-временного представления изменчивости качества жизни в пределах конкретно взятого региона.

Учитывая значимость экономики качества для развития социально-экономических систем, особый научный интерес вызывает поиск возможностей расширения методической базы аппарата количественной оценки качества жизни за счет внедрения показателей экономики качества в действующий перечень показателей, которые использованы в процессе моделирования. Введение метрологии, стандартизации и управления качеством в процесс оценки и управления качеством жизни позволит привести в соответствие современные требования к качеству жизни с целями и условиями инновационного развития в регионах.

Панельная формула, включающая потенциально значимые переменные в наборе регрессоров, имеет следующую форму:

$$y = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{i=1}^{l} b_i z_i + \sum_{i=1}^{m} d_i D_i + \varepsilon, \qquad (3.3.4)$$

где y — обобщающий показатель качества жизни в регионе;
a_0 — константа;
a_i — коэффициент при экономической переменной x_i ($i = 1, 2, \ldots, n$);
b_i — коэффициенты при дополнительных переменных z_i ($i = 1, 2, \ldots, l$);
d_i — коэффициент при фиктивной переменной ($i = 1, 2, \ldots, m$);
D_i — фиктивная (дамми) переменная, отражающая групповой эффект (например, региональный и т. д.);
ε — случайная составляющая (стохастическая ошибка).

Коэффициенты при независимых переменных показывают, на сколько пунктов изменится темп прироста при изменении соответствующей переменной на единицу. Их можно определить методом наименьших квадратов при условии, что стохастическая ошибка удовлетворяет стандартным ограничениям. Фиктивная (дамми) переменная может принимать значения 0 или 1, то есть, по сути, позволяет составить две разные модели, учитывающие какие-либо различия. Напри-

мер, наличие в регионе утвержденной стратегии развития означает равенство дамми-переменной 1, отсутствие стратегии — нулю.

Развитие методологического аппарата и инструментов измерения качества жизни в отечественной и зарубежной науке и практике свидетельствует о преобразовании качества жизни из стихийного в управляемый процесс. С использованием принципов системного подхода нами показано [34], что в многоуровневой системе управления качеством необходимое качество жизни в регионе обеспечивается в процессе выполнения цикла PDCA (Plan — Do — Check — Action), которым предусмотрено:

- определение целевых показателей развития (*Plan* — планирование);
- реализация целевых научно-технических и социально-экономических программ по приоритетным направлениям исследований и разработок, региональных программ (*Do* — сделать, производство);
- мониторинг и анализ целевых показателей (*Check* — проверка, контроль);
- корректирующие действия (*Action* — действие).

Показатели качества жизни находятся под управлением на каждом этапе цикла PDCA: и на этапе определения целевых показателей развития экономики региона, и на этапе реализации целевых программ и разработок, и в связи с мониторингом и анализом целевых показателей, и при выполнении корректирующих действий.

По существу нами построен второй блок модели механизма количественной оценки качества жизни (рис. 3.2.2), который дает возможность эффективно использовать полученные в рамках первого блока модели значения показателя качества жизни для целей стратегического управления в рамках многоуровневой системы управления качеством, используя системы обратной связи. Благодаря применению теоретико-методологической основы экономики качества показатели качества жизни в рамках системы региональной экономики находятся под управлением.

Социально-экономическое пространство Российской Федерации изначально является дифференцированным и имеет сложную орга-

низационную структуру, что оказывает непосредственное влияние на качество жизни. Неравномерное региональное развитие российских регионов определяет потребность в формировании эффективной государственной политики по устранению региональных различий в качестве жизни населения [54]. Способность органов власти эффективно использовать особенности подведомственных им территорий выступает одним из условий решения большинства региональных проблем, в том числе и проблем качества жизни.

Чтобы достижения цифровой экономики стали движущей силой пространственного развития, очень важно определять направления региональной политики, положительно влияющие на качество жизни населения и устойчивое развитие в долгосрочной перспективе. Совершенствование региональной политики как теории и практики территориального управления требует углубления ее концептуальных основ.

Экономика качества располагает необходимой концептуальной и теоретико-методологической базой, предоставляет инструментарий для эффективного поиска решения задач повышения качества жизни населения региона [55]. В условиях цифровой экономики направление процесса стандартизированного управления качеством жизни в русло инновационного развития наилучшим образом способствует разработке, реализации и поддержке стратегии экономического и социального развития региона.

Процесс стандартизированного управления качеством жизни рассматривается как стандартизованный и упорядоченный набор фаз, технических и управленческих решений, промежуточных результатов и контрольных точек. В перспективе это позволит по-новому подойти к возможностям улучшения механизмов реализации стратегии социально-экономического развития Санкт-Петербурга до 2035 г. («Стратегии 2035») и существенно повысит конкурентоспособность Северо-Запада.

Новый теоретико-методологический подход, основанный на совместном применении экономики качества и применении методологии разработки больших агент-ориентированных моделей, способен внести значительный вклад в решение непростой задачи реализации новых возможностей цифровой экономики. Благодаря эффективному применению экономики качества приращение на-

учного знания о современных высокопроизводительных и высоконадежных вычислениях создает условия для долговременного устойчивого развития цифровой экономики, обеспечивая качество роста и качество жизни населения в масштабе региона, национальной экономики и на международном уровне.

Повышение качества жизни представляет собой сложнейшую социальную, экономическую и управленческую задачу, поиск научного решения которой существенно облегчается при совместном применении экономики качества и методологии разработки больших агент-ориентированных моделей.

Литература к главе 3

1. *Евтушенко С. Н.* Цифровой суверенитет и технологическая рента России. Доклад на Национальном форуме информационной безопасности. Москва, 2–3 февраля 2017 г. С. 5.
2. *Боробова М. А.*, *Парфишина Г. М.*, *Гафарова В. А.* Роль экономики качества в повышении эффективности социально-экономического развития // Теория и практика современной науки. 2015. № 4. URL: https://readera.org/140266578 (дата обращения: 06.08.2022).
3. *Окрепилов В. В.* Экономика качества — важнейшее направление развития экономической науки // Проблемы прогнозирования. 2022. № 5. С. 78–90.
4. Устойчивое развитие: вызовы и возможности: сборник научных статей / Под ред. канд. экон. наук Е. В. Викторовой. СПб.: Изд-во СПбГЭУ, 2020. 333 с. URL: https://unecon.ru/sites/default/!les/collection_of_events_proceedings_2.pdf (дата обращения: 25.08.2022).
5. *Окрепилов В. В.* Эволюция качества. СПб.: Наука, 2008. 744 с.
6. *Окрепилов В. В.*, *Крутиков В. Н.*, *Элькин Г. И.* Экономическая составляющая в обеспечении единства измерений. Измерительная техника. 2014. № 2. С. 3–8.
7. Метрологической академии — 30 лет: сборник материалов / Под науч. ред. академика РАН В. В. Окрепилова. СПб., 2022. 128 с.
8. Цифровая трансформация метрологии — пять задач Росстандарта. URL: https://www.rst.gov.ru/portal/gost/home/presscenter/news (дата обращения: 05.08.2022).
9. Метрология в эпоху цифровизации. Приветственное обращение директора МБМВ Мартина Милтона и директора МБЗМ Энтони Доннеллана во Всемирный день метрологии 20 мая 2022 г. // Измерительная техника. 2022. № 5. С. 3.

10. Федеральный закон от 27.12.2019 г. № 496-ФЗ «О внесении изменений в Федеральный закон «Об обеспечении единства измерений». URL: http://www.kremlin.ru/acts/bank/45004 (дата обращения: 20.10.2022).
11. Федеральный информационный фонд по обеспечению единства измерений ФГИС «Аршин». Официальный сайт. URL: https://fgis.gost.ru/fundmetrology/registry
12. Цифровая метрология / Ю. А. Антохина, В. В. Окрепилов, Е. А. Фролова, Н. Ю. Ефремов, А. С. Степашкина; под ред. д-ра экон. наук, проф., академика РАН В. В. Окрепилова. СПб.: ГУАП, 2021. 181 с.
13. Встреча Михаила Мишустина с руководителем Федерального агентства по техническому регулированию и метрологии Антоном Шалаевым. 2 августа 2022. URL: http://government.ru/news/46158/ (дата обращения: 03.08.2022).
14. Федеральный закон от 29 июня 2015 г. № 162-ФЗ «О стандартизации в Российской Федерации» (с изменениями и дополнениями). URL: https://base.garant.ru/71108018/ (дата обращения: 15.08.2022).
15. Распоряжение Правительства РФ от 6 ноября 2021 г. № 3142-р «Об утверждении стратегического направления в области цифровой трансформации обрабатывающих отраслей промышленности». URL: https://www.garant.ru/products/ipo/prime/doc/402914382/ (дата обращения: 13.08.2022).
16. Вызовы цифровой экономики: условия, ключевые институты, инфраструктура: сборник статей I Всероссийской научно-практической конференции (г. Брянск, 21–22 марта 2018 г.). Брянск: Брянск. гос. инженерно-технол. ун-т., 2018. 288 с.
17. *Окрепилов В. В.* Цифровизация управления и обеспечение качества жизни // Труды научно-практической конференции с международным участием «Цифровая экономика и Индустрия 4.0: новые вызовы» / Под ред. А. В. Бабкина. 2018. С. 50–63.
18. ИСО — ISO 9001 — Системы менеджмента качества. URL: https://www.iso.org/ru/iso-9001-quality-management.html (дата обращения: 20.10.2022).
19. ГОСТ Р 51897–2011. Менеджмент риска. Термины и определения. URL: https://internet-law.ru/gosts/gost/51596/ (дата обращения: 20.10.2022).
20. ГОСТ Р ИСО 31000–2010. Менеджмент риска. Принципы и руководство. URL: https://internet-law.ru/gosts/gost/51461/ (дата обращения: 20.10.2022).
21. ISO — ISO 14000 family — Environmental management. URL: https://www.iso.org/iso-14001-environmental-management.html (дата обращения: 20.10.2022).
22. Национальный стандарт РФ ГОСТ Р ИСО 14001–2016 «Системы экологического менеджмента. Требования и руководство по применению» (утв. приказом Федерального агентства по техническому регулирова-

нию и метрологии от 29.04.2016 № 285-ст). URL: https://base.garant.ru/71544070/ (дата обращения: 20.10.2022).
23. ГОСТ Р ИСО 14004–2007. Системы экологического менеджмента. Общее руководство по принципам, системам и методам обеспечения функционирования. URL: https://internet-law.ru/gosts/gost/47949/ (дата обращения: 20.10.2022).
24. ГОСТ Р ИСО 9001–2015. Системы менеджмента качества. Требования. URL: https://internet-law.ru/gosts/gost/60764 (дата обращения: 20.10.2022).
25. ГОСТ 12.0.230–2007. Межгосударственный стандарт. Система стандартов безопасности труда. Системы управления охраной труда. Общие требования (введен в действие Приказом Ростехрегулирования от 10.07.2007 № 169-ст) (ред. от 31.10.2013). URL: https://www.consultant.ru/document/cons_doc_LAW_135558/ (дата обращения: 20.10.2022).
26. ГОСТ Р ИСО 19011–2012. Руководящие указания по аудиту систем менеджмента. URL: https://internet-law.ru/gosts/gost/52229/ (дата обращения: 20.10.2022).
27. ГОСТ Р ИСО 26000–2012. Руководство по социальной ответственности. URL: https://internet-law.ru/gosts/gost/53986/ (дата обращения: 20.10.2022).
28. ISO — ISO/TR 37121:2017 — Sustainable development in communities — Inventory of existing guidelines and approaches on sustainable development and resilience in cities. URL: https://www.iso.org/standard/63790.html (дата обращения: 20.10.2022).
29. *Окрепилов В. В.*, *Иванова Г. Н.*, *Чудиновских И. В.* Цифровая экономика: проблемы и перспективы // Экономика Северо-Запада: проблемы и перспективы развития. 2018. № 3–4 (56–57).
30. Развитие цифровой экономики в России. Программа до 2035 г. // Система информационно-аналитических ресурсов по инновационной и технологической тематике. URL: http://innclub.info/wp-content/uploads/2017/05/strategy.pdf (дата обращения: 02.10.2022).
31. *Окрепилов В. В.*, *Гагулина Н. Л.* Цифровая трансформация экономики: новое качество // Управление качеством в интересах устойчивого развития: труды научно-практической конференции. 14–15 марта 2019 г. СПб.: Политех-пресс, 2019. С. 131–139.
32. *Окрепилов В. В.*, *Гридасов А. Г.* Национальная система управления качеством как основа модернизации экономики в интересах защиты прав потребителей // Национальная концепция качества: государственная и общественная защита прав потребителей: сборник тезисов докладов международной научно-практической конференции / Под ред. д-ра экон. наук, проф. Е. А. Горбашко. СПб.: Культ-информ-пресс, 2019. С. 275–279.

33. *Окрепилов В. В.* Перспективы создания многоуровневой системы управления качеством // Стандарты и качество. 2009. № 1. С. 58–65.
34. *Окрепилов В. В.* Развитие науки о качестве. Избранные труды. СПб.: Политех-пресс, 2019. 1004 с.
35. *Окрепилов В. В.*, *Чудиновских И. В.* Процесс цифровизации Санкт-Петербурга как социально-экономической системы. Цифровизация экономических систем: теория и практика: монография / Под ред. д-ра экон. наук, проф. А. В. Бабкина. СПб.: Политех-пресс, 2020. 722 с. С. 251–273.
36. «Стратегия социально-экономического развития Санкт-Петербурга на период до 2035 г.». Приложение к Закону Санкт-Петербурга от 19 декабря 2018 г. № 771–164 «О Стратегии социально-экономического развития Санкт-Петербурга на период до 2035 г.».
37. *Schwab K.* The Fourth Industrial Revolution, Geneva: World Economic Forum. — URL: https://www.weforum.org/about/the-fourth-industrial-revolution-by-klaus-schwab (дата обращения: 22–09–2021).
38. *Макаров В. Л.*, *Бахтизин А. Р.*, *Сушко Е. Д.*, *Сушко Г. Б.* Разработка агент-ориентированной демографической модели России и ее суперкомпьютерная реализация // Вычислительные методы и программирование. 2018. Т. 19. № 4. С. 368–378.
39. *Окрепилов В. В.*, *Гагулина Н. Л.* Развитие оценки качества жизни населения региона // Журнал экономической теории. 2019. Т. 16. № 3. С. 318–330.
40. *Окрепилов В. В.* Многоуровневая система управления качеством как движущий фактор пространственного инновационного развития // Экономика и управление. 2012. № 11 (85). С. 14–19.
41. БД Качество жизни (База данных для построения модели качества жизни). Свидетельство о государственной регистрации базы данных № 2021622426. Российская Федерация / В. В. Окрепилов, А. Д. Шматко, Н. Л. Гагулина; заявитель и правообладатель — Федеральное государственное бюджетное учреждение науки «Институт проблем региональной экономики Российской академии наук». № 20221622426; заявка № 2021622344 от 28.10.2021; опубл. 09.11.2021. 1 с.
42. *Перекрест В. Т.*, *Перекрест И. В.* Региональные особенности государственного регулирования сферы занятости России в контексте результатов экономико-математического моделирования системы региональных рынков труда // Экономика Северо-Запада: проблемы и перспективы развития. 2021. № 1. С. 50–60. ISSN 2411–4588.
43. *Кузнецов С. В.*, *Курзенев В. А.*, *Перекрест В. Т.*, *Перекрест И. В.* Потребности региональной экономики в профессиональных кадрах: вопросы цифровизации исчисления и особенности региональной безработицы //

Экономика Северо-Запада: проблемы и перспективы развития, 2020. № 1. ISSN 2411–4588.

44. *Перекрест В. Т.*, *Курзенев В. А.*, *Перекрест И. В.* Особенности формирования структуры баланса трудовых ресурсов на рынке труда // Управленческое консультирование. 2015. № 5. ISSN 1726–1139. С. 72–80.
45. *Курзенев В. А.*, *Перекрест В. Т.*, *Перекрест И. В.*, *Чернейко Д. С.* Технология обеспечения профессиональными кадрами развивающихся экономических систем // Управленческое консультирование. 2018. № 2. ISSN 1726–1139. С. 55–64.
46. *Перекрест В. Т.*, *Шакирова Н. И.* Эконометрическое сопоставление инновационных систем субъектов РФ методами многомерного шкалирования // Управленческое консультирование. Актуальные проблемы государственного и муниципального управления. 2011. № 3. С.146–159.
47. *Айвазян С. А.* Анализ качества и образа жизни населения (эконометрический подход). ЦЭМИ РАН. М: Наука, 2012. 432 с.
48. *Зиновьев А. Ю.* Визуализация многомерных данных. Красноярск: Изд-во КГТУ, 2000. 180 с.
49. *Перекрест В. Т.* Нелинейный типологический анализ социально-экономической информации: математические модели и вычислительные методы. Л.: Наука, 1983. 176 с.
50. *Айвазян С. А.*, *Бухштабер В. М.*, *Енюков И. С.*, *Мешалкин Л. Д.* Прикладная статистика. Классификация и снижение размерности. М.: Финансы и статистика, 1989. 607 с.
51. *Jolliffe I. T.* Principal Component Analysis. Series: Springer Series in Statistics. 2nd ed. NY: Springer, 2002. XXIX. 487 p. 28 illus.
52. *Айвазян С. А.*, *Мхитарян В. С.* Теория вероятностей и прикладная статистика. Т. 1. М.: Юнити, 2001. 656 с.
53. *Дубров А. М.*, *Мхитарян В. С.*, *Трошин Л. И.* Многомерные статистические методы. М.: Финансы и статистика, 2000. 350 с.
54. Междисциплинарное исследование процессов трансформации социально-экономического пространства и территориального развития регионов России: монография / Под науч. ред. д-ра экон. наук, проф., академика РАН В. В. Окрепилова, д-ра экон. наук, проф. С. В. Кузнецова. СПб.: ГУАП, 2021. 469 с.
55. Национальная концепция качества: подготовка управленческих кадров: сборник тезисов докладов национальной научно-практической конференции с международным участием / Под ред. д-ра экон. наук, проф. Е. А. Горбашко. СПб.: Изд-во СПбГЭУ, 2020. 429 с.

Заключение

Эволюция экономических систем — постоянный и непрерывный, сложно прогнозируемый процесс, в ходе которого происходят изменения как отдельных элементов в экономике, так и трансформация ее основных сфер, что находит отражение в общих изменениях структуры экономической системы. Начавшаяся в последние десятилетия цифровая трансформация порождает новый технологический уклад, появляются не только целые новые отрасли экономики, но и изменяются законы функционирования и развития социально-экономической системы, ее сущность и содержание.

Инновационный характер современного экономического развития предопределил опережающий рост влияния информационных технологий на все ключевые области экономики и обусловил формирование долговременных технологических трендов. Сегодня очевидно, что цифровые активы представляют собой новый экономико-правовой феномен и составляют важнейшую часть экономики, фундаментом которой все в большей степени становится информация.

Наряду с позитивными тенденциями в социально-экономическом развитии в условиях повышенных темпов цифровизации повышается и уровень неопределенности в выборе наиболее эффективных трендов, возрастает риск принятия ошибочных решений на всех уровнях управления — от предприятия до государственной власти.

Исследование проблематики современной экономики, обусловленной цифровой трансформацией, равно как и самой трансформации, показало, что оптимальный путь решения проблемы состоит в совместном применении методов экономики качества и математического моделирования, которое способствует формиро-

ванию системного подхода к управлению развитием в условиях цифровой трансформации.

Авторы монографии пришли к такому выводу на основе проведенных в течение нескольких лет совместных исследований. Были проанализированы свойства и характеристики цифровой экономики, ее основа, а также тенденции цифровой трансформации и ее технологические тренды. Создана основа для совместных фундаментальных исследований сложных практических задач цифровой экономики на базе методологии разработки больших агент-ориентированных моделей, воссоздающих в искусственной среде структуру и особенности функционирования реальных социально-экономических систем, а также реализации подобных моделей в суперкомпьютерной среде. Проанализированы возможности применения результатов зарубежного опыта разработки агентных моделей, реализованных с использованием суперкомпьютерных технологий, а также результатов вычислительных экспериментов в решении специфических проблем цифровой экономики, обусловленных множественными взаимодействиями экономических агентов.

Проведенные исследования показали, что тенденции цифровой трансформации, а также технологические тренды в современной экономике обусловили такой уровень развития информационных технологий, при котором существенно изменились институциональные условия заключения контрактов, а следовательно — и создания цифровых активов.

Глубокая включенность экономики качества в сквозные процессы цифровой экономики объясняет ее востребованность при построении новых системных механизмов для решения задач цифровой экономики. Применение метрологии, стандартизации, управления качеством при проведении моделирования цифровой экономики целесообразно практически на каждом этапе и повышает уровень прогнозирования при решении сложных практических задач с помощью суперкомпьютеров.

Для моделирования поведения искусственного общества в целом требуется создание агент-ориентированных моделей с числом

агентов до 10^9 и использование суперкомпьютерных технологий для проведения симуляций.

Новый теоретико-методологический подход, основанный на совместном применении экономики качества и применении методологии разработки больших агент-ориентированных моделей, способен внести значительный вклад в решение непростой задачи реализации новых возможностей цифровой экономики. Благодаря эффективному применению экономики качества приращение научного знания о современных высокопроизводительных и высоконадежных вычислениях создает условия для долговременного устойчивого развития цифровой экономики, обеспечивая качество роста и качество жизни населения в масштабе региона, национальной экономики и на международном уровне.

В отечественной практике впервые сформулированы наиболее актуальные на международном уровне темы в сфере информационно-коммуникационных технологий и высокопроизводительных вычислений, выявлены долгосрочные и фундаментальные в своем развитии флагманские проекты в данной отрасли знаний. Предложен новый теоретико-методологический подход, основанный на применении методов стандартизации, метрологии и управления качеством, который применим для дальнейшего исследования сформулированных наиболее актуальных тем и флагманских проектов цифровой экономики.

© ВИНИТИ

Макаров Валерий Леонидович,
Окрепилов Владимир Валентинович,
Бахтизин Альберт Рауфович

Научные решения сложных экономических и социальных задач с помощью суперкомпьютеров. — М.: ЛЕНАНД, 2023.

Редактор: А. Г. Гридасов

Формат 60×90/16. Печ. л. 26

Подписано в печать: 09.03.2023

© URSS