



**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

НАУЧНЫЙ ДОКЛАД

**по результатам подготовленной
научно-квалификационной работы (диссертации)**

ФИО Зиганурова Лилия Фаилевна

Направление подготовки 09.06.01 Информатика и вычислительная техника

**Профиль (направленность) программы Математическое моделирование,
численные методы и комплексы программ**

Аспирантская школа по компьютерным наукам

Аспирант _____ /Зиганурова Л.Ф. /
подпись

Научный руководитель _____ /Щур Л.Н. /
подпись

Директор Аспирантской школы по компьютерным наукам _____ /Объедков С.А. /
подпись

Москва, 2019

Содержание

Введение.	3
Цель и задачи исследования.	5
Степень разработанности темы исследования.	6
Научная новизна.	7
Основные результаты исследования и положения, выносимые на защиту.	9
Краткое содержание работы.	9
Апробация результатов исследования.	29
Литература	32

Введение.

За последние два десятилетия был совершен гигантский рывок в масштабах параллельных вычислений, выполняющихся на суперкомпьютерах (Таблица 1). В настоящий момент количество вычислительных ядер в суперкомпьютерах доходит до десятков миллионов, без учета ядер на графических ускорителях. Помимо этого, ядра центрального процессора могут иметь многопоточность.

Год	Суперкомпьютер из TOP-500	Количество ядер	Производительность на LINPACK
1998	ASCI Red	9К	1.3 TFlop/s
2000	ASCI White	8К	4.9 TFlop/s
2004	BlueGene/L	32К	70 TFlop/s
2008	BlueGene/L	213К	487 TFlop/s
2010	Jaguar (Cray XT5-HE)	224К	1.7 PFlop/s
2012	Sequoia (BlueGene/Q)	1.6 М	18 PFlops
2014	Tianhe-2A	3.1М	34 PFlops
2016	Sunway TianhuLight	10.6М	125 PFlop/s
2018	Summit (IBM)	2.3М	144 PFlop/s

Таблица 1: Изменение количества ядер на суперкомпьютерах в период с 1998 по 2018 г. [1]

Моделирование сложных систем на суперкомпьютерах – это один из инструментов, способных привести к новым научным открытиям [2]. Для эффективного использования доступного на сегодняшний день параллелизма в целях решения одной задачи необходима разработка новых параллельных алгоритмов, компиляторов, библиотек и других программных инструментов. Один из методов, позволяющих запускать одну задачу на множестве вычислительных узлов (процессоров/ядер), это метод параллельного моделирования дискретных событий (PDES, Parallel Discrete Event Simulation) [9]. Этот метод используется как в фундаментальной науке, так

и в прикладных исследованиях в сфере экономики, медицины, социологии и т.д. Примерами применения PDES могут служить моделирование кинетическим методом Монте Карло [3], 3D моделирование процесса спекания [4], моделирование интернета вещей [5], моделирование нейроморфных систем [6] и др.

В методе PDES происходит декомпозиция одной большой задачи на подзадачи, каждая из которых выполняется своим логическим процессом. Изменение состояния системы происходит в определенные дискретные моменты времени, называемые *дискретными событиями*. Обычно эти времена распределены по экспоненциальному закону. Логический процесс имеет свое *локальное виртуальное время* [7] и имеет локальную очередь событий, которые он обрабатывает в порядке возрастания их временных меток. Количество обрабатываемых событий при моделировании методом PDES на современных суперкомпьютерах доходит до пятисот миллиардов в секунду [8]. Логические процессы взаимодействуют между собой при помощи сообщений и могут генерировать события для других логических процессов. Для корректности моделирования необходимо, чтобы события обрабатывались строго в порядке возрастания их временных меток. Для обеспечения этого условия существуют специальные механизмы синхронизации, которые можно классифицировать по трем группам: консервативные, оптимистические и алгоритмы Freeze-and-Shift [9, 10].

Объектом исследования диссертационной работы является набор локальных времен всех логических процессов (Рисунок 1) – так называемый профиль локальных виртуальных времен (ЛВВ). Профиль ЛВВ эволюционирует в ходе моделирования и в каждый фиксированный момент модельного времени может быть описан двумя характеристиками – средним значением ЛВВ и средней шириной (среднеквадратичным отклонением). Профиль ЛВВ растет по-разному, в зависимости от алгоритма синхронизации и топологии взаимодействия ЛП. В диссертационном исследовании мы строим модели эволюции профиля ЛВВ в консервативном и оптимистическом алгоритмах PDES с различным типом коммуникаций между логическими процессами и изучаем поведение среднего значения ЛВВ и средней ширины в зависимости от интересующих параметров. Изучая поведение профиля ЛВВ можно сделать выводы об эффективности и масштабируемости алго-

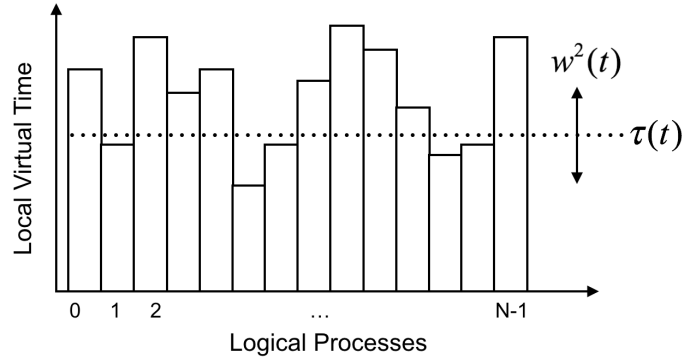


Рисунок 1: Профиль локальных времен в фиксированный момент модельного времени t . Каждый ЛП имеет свое значение локального времени. $\tau(t)$ – это среднее значение локального времени по всем логическим процессам, а $w^2(t)$ – среднеквадратичная ширина профиля.

ритмов. Скорость роста профиля ЛВВ соответствует утилизации процессорного времени, а ширина профиля ЛВВ отражает степень рассинхронизации между параллельными процессами.

Цель и задачи исследования.

Целью диссертационного исследования является изучение моделей эволюции локальных времен процессов в различных алгоритмах синхронизации при параллельном дискретно-событийном моделировании.

Задачами исследования являются:

1. Изучить современную научную литературу в области PDES.
2. Изучить современную научную литературу по свойствам и применению сетей малого мира.
3. Исследовать модель эволюции локальных времен процессов в консервативном алгоритме PDES в размерности 1+1 на регулярной топологии и топологии малого мира.

4. Исследовать модель эволюции локальных времен процессов в оптимистическом алгоритме PDES в размерности $1+1$ на регулярной топологии и топологии малого мира.
5. Установить соответствие между результатам моделирования и результатами экспериментальных данных, полученных с использованием симулятора ROSS [11].

Степень разработанности темы исследования.

Модель эволюции профиля локальных времен процессов для консервативного алгоритма предложена в [12]. Авторы рассматривают одномерную систему с периодическими граничными условиями, где взаимодействие между ЛП происходит локально между ближайшими соседями. Каждому ЛП соответствует один процессорный элемент (узел/процессор/ядро/нить). Показано, что профиль локальных времен имеет аналогию с неравновесным ростом поверхности в физике. Локальное случайное увеличение времени в модели консервативного алгоритма PDES соответствует отложению случайного количества материала в локальных минимумах растущей поверхности (как, например, при молекулярной эпитаксии).

В случае, когда логические процессы взаимодействуют локально, рост профиля ЛВВ в консервативном алгоритме можно описать уравнением Кардара-Паризи-Жанга (KPZ) [13]. Авторы утверждают, что эффективность алгоритма соответствует плотности локальных минимумов профиля ЛВВ. Моделирование эволюции профиля ЛВВ в консервативном алгоритме и грубое приближение показало, что алгоритм является асимптотически масштабируемым, поскольку при увеличении числа ЛП скорость роста профиля остается ненулевой, а ширина профиля расходится. Ненулевая скорость роста профиля ЛВВ означает, что консервативный алгоритм не имеет мертвых состояний, при этом растущая ширина профиля говорит о плохой масштабируемости исследуемого алгоритма. Таким образом, аналогия между профилем локальных времен в PDES и ростом поверхности позволила отнести

модель роста профиля ЛВВ в консервативном алгоритме с локальными взаимодействиями к классу универсальности KPZ и сделать выводы о свойствах алгоритма в пределе бесконечного количества логических процессов.

Помимо этого, на основе такой аналогии позднее все алгоритмы синхронизации PDES были классифицированы на три класса – консервативные, оптимистические и алгоритмы FaS, соответствующие периодическим, открытым и фиксированным граничным условиями уравнения KPZ [10].

Модель [12] учитывала только локальные взаимодействия логических процессов. В реальности между ЛП могут возникать не только локальные, но и удаленные связи. В работах [14, 15] изучается влияние малого количества случайных дальних взаимодействия на синхронизацию процессов в консервативном алгоритме PDES. Моделирование роста профиля ЛВВ проводилось на топологии, близкой по свойствам к топологии малого мира, но не являющейся ею по определению. Добавление малого числа случайных дальних взаимодействий между ЛП существенно снижает ширину профиля ЛВВ, при этом сохраняя скорость роста положительной. Консервативный алгоритм на такой топологии становится полностью масштабируемым, поскольку при увеличении количества логических процессов ширина остается постоянной (а не растет, как на регулярной топологии), а скорость снижается незначительно.

Еще одна область исследования эффективности метода PDES посвящена анализу производительности реальных систем параллельного дискретно-событийного моделирования на различных симуляторах (ROSS [11], PDEVs [18], μsik [19] и др.). Анализ эффективности обычно проводится в терминах количества обработанных событий в единицу времени [20–22]. Обычно такие исследования являются аппаратно и модельно-зависимыми и их результаты не носят универсальный характер.

Научная новизна.

Научная новизна исследования заключается в применении такого известного способа изучения метода PDES, как моделирование эволюции профи-

ля ЛВВ [10, 12], к новым алгоритмам синхронизации с новой топологией взаимодействия логических процессов. Помимо этого, в диссертационном исследовании впервые проводится сравнение полученных в ходе моделирования данных с результатами запуска реальных моделей на симуляторе ROSS.

Более подробно научная новизна исследования раскрывается в следующих пунктах.

1. В ходе диссертационного исследования была построена и изучена модель роста профиля ЛВВ консервативного алгоритма PDES на топологии малого мира. Отличие от [14] заключается в топологии взаимодействия логических процессов. В [14] каждый ЛП связан ровно с тремя ЛП, два из которых – ближайшие соседи, а один выбран случайным образом из оставшихся. Дальние связи “включаются” с вероятностью p . Такая модель по свойствам напоминает топологию малого мира, но не является ею по определению. В нашей работе мы объединяем ЛП в регулярную топологию, а затем с вероятностью p добавляем дальние взаимодействия (либо переписываем ближние связи случайным образом). Такой способ построения сети со свойствами малого мира является классическим [17] и лучше отражает свойства реальных моделируемых систем.
2. Впервые предложена модель эволюции ЛВВ для оптимистического алгоритма PDES. Модель изучена на регулярной топологии и топологии малого мира. В результате исследования показано, что модель оптимистического алгоритма PDES может быть отнесена к классу направленной перколяции (Directed Percolation) [16].
3. По нашей просьбе разработчики фреймворка ROSS добавили в программу функцию вывода локальных времен процессов. Ранее никто не изучал поведение локальных времен в моделях PDES, обычно измерение производительности параллельного дискретно-событийного моделирование происходило в количестве обработанных событий за единицу времени.

Основные результаты исследования и положения, выносимые на защиту.

1. Построена и изучена модель роста профиля ЛВВ для консервативного алгоритма PDES на различных топологиях малого мира.
2. Выявлена зависимость скорости роста и ширины профиля ЛВВ от концентрации дальних связей p между логическими процессами.
3. Предложена и изучена модель роста профиля ЛВВ для оптимистического алгоритма PDES на различных топологиях малого мира.
4. Показано, что модель роста профиля ЛВВ для оптимистического алгоритма на регулярной топологии может быть отнесена к классу направленной перколяции.
5. Проведено сравнение с результатами экспериментальных данных на симуляторе ROSS.

Краткое содержание работы.

В **первой главе** описаны основные понятия и концепции метода параллельного дискретно-событийного моделирования, а также дана классификация алгоритмов синхронизации. Во **второй главе** описана связь между ростом профиля локальных времен в алгоритмах PDES и моделями роста поверхности в физике. В **третьей главе** приведен обзор литературы по сетям малого мира и описаны используемые в работе топологии. В **главе 4** описан процесс и результаты моделирования профиля ЛВВ в консервативном алгоритме, а в **главе 5** – в оптимистическом. Сравнение результатов

моделирования с данными, полученными в ходе case-studies на симуляторе ROSS приведено в **главе 6**.

Топология малого мира. Впервые определение сети малого мира было дано в работе Ваттса и Строгатца [17]. Сети малого мира характеризуются малым значением среднего кратчайшего пути и высоким значением коэффициента кластеризации. Такие свойства наблюдаются во многих технологических, биологических, социальных и информационных сетях.

Есть несколько способов, которыми можно организовать топологию со свойствами малого мира. Первый способ – сначала построить регулярную топологию, где каждый узел соединен с $2k$ ближайшими узлами (Рисунок 2а), а затем с вероятностью p “отцепить” ребро от ближайшего узла и соединить его с другим случайным узлом решетки (Рисунок 2б). Вторым способом заключается в добавлении поверх регулярного графа с вероятностью p новых ребер между случайными узлами (Рисунок 2в,г).

Мы провели исследование на трех различных топологиях, построенных по правилам, описанным выше. Эти три сети отличаются изначальным количеством соседей в регулярной топологии, а также способ конструирования из нее сети малого мира (с переписыванием или добавлением связей). Для простоты будем использовать сокращенные названия “*A (add) or R (rewrite) – параметр k*”. Все используемые топологии изображены на рисунке 2. Во всех используемых топологиях малого мира средний кратчайший путь растет логарифмически с размером графа, однако эти топологии отличаются значением коэффициента кластеризации. В модели сети $A-k1$, $C(p) \approx 0$. Строго говоря, эта сеть не является сетью малого мира, поскольку не удовлетворяет одному из критериев. В моделях сети $A-k2$ и $R-k2$ коэффициент кластеризации близок к $1/2$, что говорит о принадлежности этих топологий к классу сетей малого мира.

Описание модели для консервативного алгоритма. Консервативный алгоритм полностью избегает ошибки причинности, т.е. такие ситуации, при которых логический процесс получает сообщение “из прошлого” – с временной меткой меньшей, чем его текущее локальное время [9,23]. Обработка событий логическим процессом будет безопасна, если его локальное

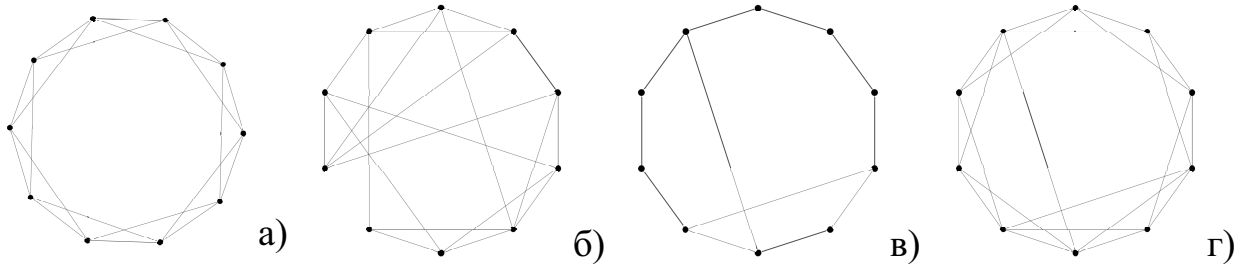


Рисунок 2: а) Одномерная топология, в которой каждый узел соединен с $2k$ соседями, с периодическими граничными условиями. Параметр $k = 2$.

б) Модель $R-k2$, т.е. модель Ваттса и Страттса, в которой малая доля связей переписывается случайным образом.) в) Модель $A-k1$, где каждый узел соединен с $2k$ соседними, и с небольшой долей случайных дальних связей, добавленных поверх регулярной топологии. г) Модель $A-k2$, отличается от $A-k1$ только начальным количеством соседей.

время меньше или равно локальному времени тех логических процессов, от которых он зависит. Если это условие не выполняется, ЛП остается заблокированным и ждет, пока соседние ЛП не “догонят” его во времени.

Коммуникационный граф взаимодействия ЛП статичен и задается матрицей смежности D такой, что если ЛП $_i$ зависит от ЛП $_j$, то $D(i, j) = 1$, в противном случае, если ЛП $_i$ и ЛП $_j$ независимы друг от друга, то $D(i, j) = 0$.

Обозначим буквой N количество логических процессов. На одном шаге моделирования t каждый логический процесс будет иметь свое локальное виртуальное время $\tau_i(t)$. Профилем локальных времен мы называем набор $\{\tau_1(t), \tau_2(t), \dots, \tau_N(t)\}$.

Мы начинаем моделирование эволюции ЛВВ с плоского профиля: $\tau_i(t = 0) = 0$, $i = 1, 2, \dots, N$. Увеличиваем локальное время только у тех процессов, чье время меньше или равно локальному времени соседей. Такой процесс называется *активным*. Если его время больше, чем у его соседей, то на данном шаге моделирования его локальное время не изменяется. Времена наступления событий в системе распределены экспоненциально. Правило обновления профиля ЛВВ на каждом шаге моделирования t выглядит следующим образом:

$$\tau_i(t + 1) = \begin{cases} \tau_i(t) + \eta_i & \text{если } \tau_i(t) \leq \{\tau_j(t)\}_{D(i,j)=1}, \\ \tau_i(t) & \text{в противном случае,} \end{cases} \quad (1)$$

где η_i – это случайная величина, распределенная по закону Пуассона с единичным математическим ожиданием, $\{\tau_j(t)\}_{D(i,j)=1}$ – это локальные времена логических процессов, которые соединены с ЛП_{*i*} локальной либо дальней связью, $i = 1..N$.

После каждого обновления профиля рассчитывается среднее значение локального времени по всем логическим процессам $\tau(t)$, средняя скорость профиля $v(t)$ и средняя квадратичную ширину профиля $w^2(t)$.

Среднее локальное виртуальное время иногда называют средней высотой профиля. Оно представляет собой среднее арифметическое по локальным временам всех логических процессов:

$$\tau(t) = \left\langle \frac{1}{N} \sum_{i=1}^N \tau_i(t) \right\rangle. \quad (2)$$

Треугольные скобки $\langle \cdot \rangle$ в этом уравнении и далее означают усреднение по независимым реализациям профиля на шаге моделирования t .

Средняя скорость профиля показывает загруженность процессорных элементов, или эффективность работы алгоритма. Для консервативного алгоритма средняя скорость рассчитывается по формуле:

$$v(t) = \left\langle \frac{N(t)}{N} \right\rangle, \quad (3)$$

где $N(t)$ – это количество активных ЛП, т.е. тех ЛП, которые изменили свое время на данном шаге моделирования.

Средняя квадратичная ширина профиля отражает степень синхронизации процессорных элементов и рассчитывается по формуле

$$w^2(t) = \left\langle \frac{1}{N} \sum_{i=1}^N [\tau_i(t) - \tau(t)]^2 \right\rangle. \quad (4)$$

Чем меньше разброс локальных времен, тем лучше ЛП синхронизированы. Большая дисперсия локального времени процессов указывает на то, что часть ЛП отстает, а часть – слишком торопится. Это увеличивает время ожидания для тех ЛП, которые находятся впереди по времени.

Мы изучаем изменение средней скорости профиля локальных времен v и средней квадратичной ширины профиля локальных времен w^2 , как функций шага моделирования t , количества логических процессов N и концентрации дальних связей между логическими процессами p .

Основные результаты моделирования профиля ЛВВ в консервативном алгоритме. Мы исследовали влияние топологии малого мира на свойства консервативного алгоритма синхронизации PDES, используя модель эволюции профиля ЛВВ. Модели были исследованы на нескольких реализациях сети малого мира, которые отличаются некоторыми свойствами и процедурой внедрения дальних связей.

В модели консервативного алгоритма *на регулярной топологии* средняя ширина профиля растет со временем логарифмически:

$$w^2(t) \sim t^{2\beta}, \quad (5)$$

а затем выходит на насыщение (рис. 3а). Величина насыщения (максимальная ширина профиля ЛВВ) зависит от количества логических процессов линейно (рис. 3б):

$$w_{\infty}^2(N) \sim N^{2\alpha} \quad (6)$$

Значения показателей α и β , полученные путем аппроксимации данных, равны 0.49(1) и 0.326(5), соответственно. Значения этих показателей близки к критическим индексам класса KPZ.

Средняя скорость роста профиля ЛВВ в пределе бесконечного размера системы v_0 равна 0.246410(7) [10]. Из этого следует, что при использовании консервативной синхронизации с локальными взаимодействиями только четверть логических процессов активна в один момент времени. Степень рассинхронизации между логическими процессами увеличивается пропорционально их количеству ($w^2 \sim N$).

На топологии малого мира получены следующие основные результаты.

Средняя скорость роста профиля. Средняя скорость $\langle v \rangle$ уменьшается с концентрацией дальних связей p и ее значение меньше для сетей $A-k2$ и $R-k2$, поскольку в этих сетях каждый ЛП зависит минимум не от двух, а от четырех ближайших ЛП. Наличие дополнительных зависимостей с большим

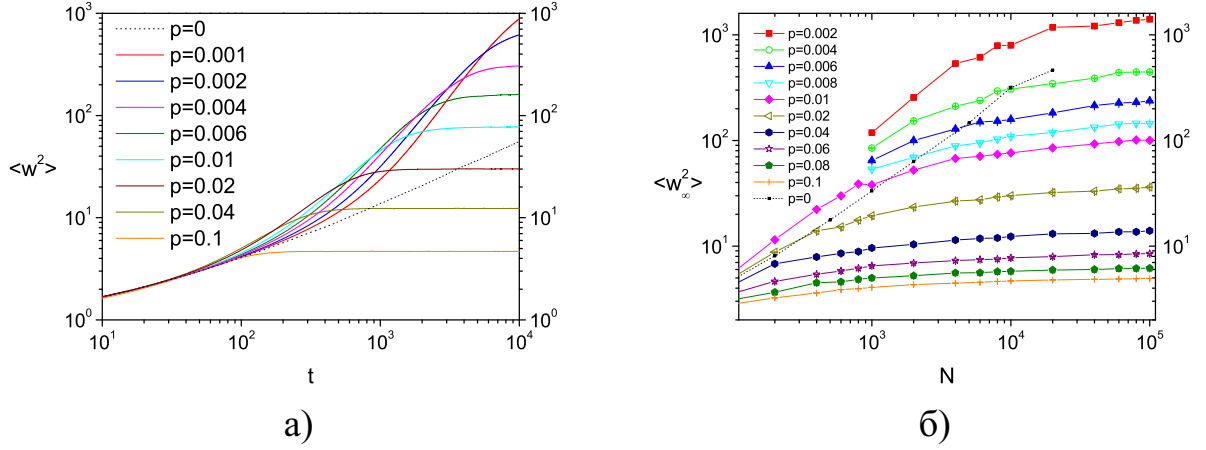


Рисунок 3: а) Средняя квадратичная ширина w^2 как функция времени для системы из $N = 10^4$ логических процессов и для различных значений p . Усреднение взято по 1500 независимым реализациям. Черной прерывистой линией обозначена кривая для регулярной топологии $p = 0$. б) Зависимость средней величины насыщения w_∞^2 от количества логических процессов N при разных p [24].

числом ближайших соседей существенно уменьшает среднюю скорость $\langle v \rangle$. Так, для $p = 0$, средняя скорость $\langle v_0 \rangle = 0.14674(7)$. Тем не менее, при увеличении параметра p на небольшое значение, скорость остается позитивной, что означает, что консервативный алгоритм синхронизации на сетях малого мира поддерживает ненулевую утилизацию процессорного времени. Например, $\langle v \rangle = 0.221370(7)$ в сети $A-k1$ для значения $p = 0.01$ и $\langle v_0 \rangle = 0.246410(7)$ для $p = 0$.

Мы обнаружили степенную зависимость скорости профиля от концентрации дальних связей p : $\langle v \rangle = \langle v_0 \rangle - \text{const } p^B$ со значением $B = 0.306(4)$ для сети $A-k1$. Показатель B для сетей $A-k2$, $A-k4$ и $A-k8$ равен примерно $0.44(1)$, а для сети $R-k2$ $B = 0.450(2)$. Таким образом, чем большее координационное число графа, тем медленнее скорость роста профиля, при этом значение коэффициента кластеризации влияет на поведение профиля ЛВВ.

Средняя ширина профиля. Как было сказано ранее, модель эволюции профиля локальных времен на регулярной топологии (только с ближними взаимодействиями) принадлежит классу универсальности KPZ, в котором критические показатели α и β имеют значения $1/2$ и $1/3$, соответственно. Добавление даже малого числа дальних связей между ЛП существенно изменяет поведение профиля ЛВВ. Критический показатель роста β зависит логарифмически от концентрации дальних связей, $\beta \sim -\ln p$, а критиче-

	Регулярная топология $p = 0$	Топология малого мира $p > 0$
$w^2(t)$	$w^2(t) \sim t^{2\beta}, t < t_x$ $\beta = 0.326(5)$	$w^2 \sim t^{2\beta}$ $\beta \sim \ln(p)$
$w_\infty^2(N)$	$w_\infty^2(N) \sim N^{2\alpha}$ $\alpha = 0.49(1)$	$w_\infty^2(N) \sim \text{const}$ $\alpha = 0$
v	$v_0 = 0.246410(7)$	$v = v_0 - \Delta v$ $\Delta v \sim p^{0.306(4)}$

Таблица 2: Сводная таблица поведения скорости v и квадратичной ширины w^2 профиля локальных времен в модели консервативного алгоритма.

ский показатель шероховатости α равен нулю. Нулевой показатель шероховатости означает, что ширина профиля остается постоянной при росте размера системы.

Был обнаружен коллапс данных для ширины профиля как функции концентрации дальних связей p для двух реализаций сети малого мира с ненулевым коэффициентом кластеризации. Отсутствие коллапса с данными топологии $A-k1$ может быть объяснено тем, что в сети $A-k1$ коэффициент кластеризации равен нулю (следовательно, топология $A-k1$ не является сетью малого мира).

На языке процесса моделирования, полученные результаты могут быть интерпретированы следующим образом. Во-первых, дополнительные дальние коммуникации в топологии взаимодействия ЛП означают большее количество проверок зависимостей и, как следствие, большее количество пересылок сообщений, что уменьшает среднюю утилизацию процессорного времени. Тем не менее, утилизация остается положительной, что показывает, что консервативный алгоритм лишен мертвых состояний. Во-вторых, десинхронизация между ЛП остается конечной и уменьшается с количеством дальних коммуникаций. Таким образом, консервативный алгоритм на сетях малого мира становится *полностью масштабируемым*, означая, что в пределе большого числа ЛП: 1) прогресс моделирования остается позитивным; 2) десинхронизация времен логических процессов становится конечной.

Такой подробный анализ модели консервативного алгоритма синхронизации PDES на сетях малого мира позволит связать параметры нашей мо-

дели с параметрами моделирования конкретных систем. Это исследование может быть полезным при оптимизации моделирования методом PDES.

Описание модели для оптимистического алгоритма. В оптимистическом алгоритме вместо блокирующих конструкций (как в консервативных алгоритмах) используется механизм обнаружения и исправления ошибок [25]. ЛП параллельно обрабатывают события в течение некоторого установленного временного окна, а затем происходит проверка причинности вычислений. Если ЛП получил сообщение с меткой меньшей, чем его ЛВВ, то состояние этого ЛП “откатывается” до того времени, когда получение этого события было бы безопасным. Если при этом ЛП отправлял сообщения другим ЛП, то для отмены этих преждевременно отправленных сообщений рассылаются так называемые антисообщения. Антисообщения отличаются от обычных сообщений только наличием отрицательного знака. При попадании в одну очередь двух одинаковых сообщений с разными знаками оба сообщения уничтожаются.

Поскольку в оптимистическом алгоритме локальное время логических процессов может как увеличиваться, так и уменьшаться, мы разбиваем один шаг моделирования на две части: продвижение вперед и откат назад. Во время продвижения вперед каждый ЛП_{*i*} увеличивает свое ЛВВ на случайную величину η_i , распределенную по закону Пуассона со средним значением, равным единице:

$$\tau_i(t + 1) = \tau_i(t) + \eta_i, \quad i = 1, 2, \dots, N. \quad (7)$$

Откат происходит в том случае, если нарушена причинность вычислений. Будем считать, что количество откатов во время одного шага моделирования – это пуассоновская случайная величина со средним b . Во время каждого отката случайным образом выбирается один ЛП_{*j*}. Считаем, что причинность может быть нарушена равновероятно любым ЛП, от которого зависит j -тый ЛП. Выбираем равновероятно ЛП_{*r*} из числа тех, для которых $D(j, r) = 1$. Далее τ_j сравнивается со временем τ_r , и в случае, если τ_j больше ЛВВ ЛП_{*r*}, τ_j уменьшается до τ_r [26]

$$\tau_j = \begin{cases} \tau_r, & \text{если } \tau_j < \tau_r \\ \tau_j, & \text{в противном случае.} \end{cases} \quad (8)$$

После каждого шага t , т. е. после каждого обновления профиля, мы рассчитываем среднее значение локального времени по всем логическим процессам $\tau(t)$, среднюю скорость профиля $v(t)$ и среднюю квадратичную ширину профиля $w^2(t)$. Среднее время и средняя ширина профиля рассчитываются так же, как в модели консервативного алгоритма (формулы 2, 4). Средняя скорость для оптимистического алгоритма также отображает эффективность обработки событий, но рассчитывается по другой формуле:

$$v(t) = \langle \tau(t) - \tau(t-1) \rangle, \quad (9)$$

где $\tau(t)$ – это среднее по логическим процессам время на шаге t , а $\tau(t-1)$ – среднее по логическим процессам время в на предыдущем шаге. В случае, когда в модели оптимистического алгоритма нет откатов, то есть параметр $b = 0$, средняя скорость будет равна единице.

Удобно ввести параметр $q = 1/(1+b)$, который может быть интерпретирован как интенсивность роста поверхности (growth rate).

Средняя скорость роста профиля ЛВВ $v(t)$ – это среднее приращение высоты профиля за один шаг моделирования (уравнение 9). Средняя скорость не зависит от времени, но зависит от параметра q , интенсивности роста. С ростом b при некотором значении $q = q_c$ скорость профиля снижается до нуля, т.е. слишком большое количество откатов не позволяет профилю ЛВВ продвигаться вперед. Система оказывается в *абсорбирующем состоянии* (absorbing state). Мы можем ожидать согласно гипотезе Грассбергера [33], что модель будет принадлежать к классу универсальности направленной перколяции.

Для нахождения зависимости $v(q)$ при каждом значении параметра q производится усреднение функции $v(t)$ по шагам моделирования, при этом в усреднении не учитываются значения для шага времени, меньшего типичного интервала релаксации скорости. График зависимости средней скорости v от параметра q представлен на рисунке 4а. Предполагая, что

$v \sim \xi_{\parallel}^{-1} \sim (q - q_c)^{\nu_{\parallel}}$ [27,28], мы аппроксимировали график функцией вида

$$v(q) = v_0(q - q_c)^{\nu_{\parallel}}. \quad (10)$$

Полученные значения параметров фитирования: $v_0 = 1.26(2)$, $q_c = 0.136(1)$, $\nu_{\parallel} = 1.78(2)$.

На рисунке 4б показана зависимость средней скорости v от $(q - q_c)$ в дважды логарифмической шкале. В целом результаты численного моделирования хорошо попадают на кривую $(q - q_c)^{1.78(2)}$, но вблизи критической точки данные отклоняются. Это можно объяснить недостаточной точностью вычислений и эффектом конечного размера системы. Для того, чтобы проверить влияние конечного размера, мы провели моделирование систем из 250, 500, 1000 и 5000 логических процессов. Чем больше моделируемая система, тем ближе полученные значения к предсказанной кривой, однако даже для достаточно больших систем ($N = 5000$) заметно влияние конечности размера системы.

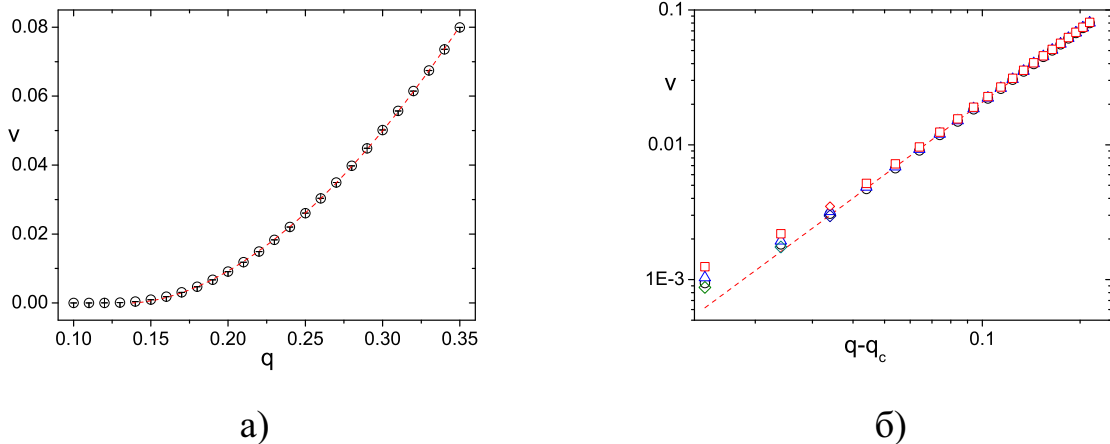


Рисунок 4: а) Средняя скорость профиля v как функция параметра q для системы из $N = 1000$ логических процессов на регулярной топологии $p = 0$. б) Средняя скорость профиля v как функция параметра $(q - q_c)$ в системах разного размера на регулярной топологии $p = 0$. Зеленые ромбы для $N = 5000$; круглые черные точки для $N = 1000$; треугольные синие точки для $N = 500$; квадратные красные точки для $N = 250$.

Таким образом, скорость роста профиля ЛВВ в модели оптимистического алгоритма на регулярной топологии имеет фазовый переход между активной фазой (когда скорость роста профиля ненулевая) и состоянием пиннинга (когда профиль не растет) в точке $q_c = 0.136(1)$. Найденная нами

критическая экспонента $\nu_{\parallel} = 1.78(2)$ близка к экспоненте в классе направленной перколяции, $\nu_{\parallel}^{DP} \approx 1.73$. К классу направленной перколяции также относятся контактные процессы [31,32], модели распространения эпидемий без иммунизации [33], каталитические реакции [34–36], модели случайного блуждания (branching-annihilating random walks) [37, 38], модели роста поверхностей [39,40], и другие [27].

Поведение скорости роста профиля для модели оптимистического алгоритма на сетях малого мира изменяется несущественно. Поскольку дополнительные зависимости между ЛП провоцируют больше откатов, то в среднем скорость профиля снижается, положение критической точка q_c и критический показатель ν_{\parallel} при этом смещаются в сторону больших значений (Таблица 3).

p	v_0	q_c	ν
0	1.26(2)	0.136(1)	1.78(2)
0.001	1.27(4)	0.140(2)	1.80(3)
0.01	1.32(4)	0.158(2)	1.82(3)
0.1	1.64(7)	0.213(3)	1.96(5)

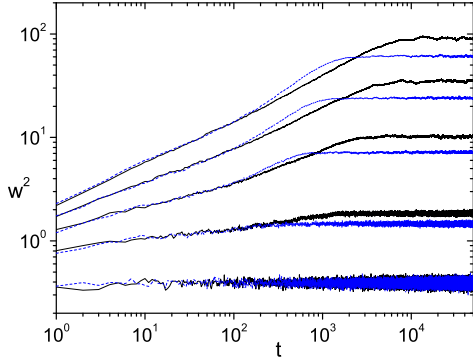
Таблица 3: Значения параметров функции (10) для регулярной топологии и топологий малого мира с различными значениями параметра p .

В моделях роста ширина характеризуется t_{\times} , типичным значением шага моделирования для выхода на насыщение. Известно, что $t_{\times} \sim N^z$, где z - динамическая экспонента. В нашей модели в период $t < t_{\times}$ ширина растет степенным образом

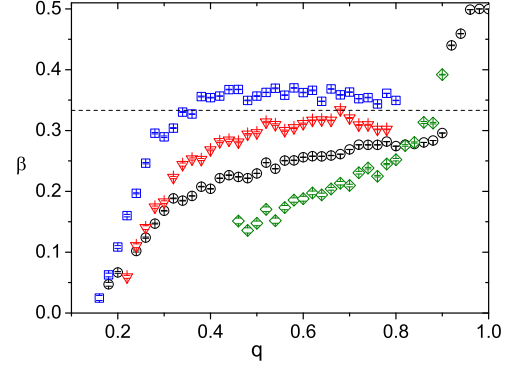
$$w^2(t) \sim t^{2\beta}.$$

График изменения ширины профиля ЛВВ со временем для разных топологий представлен на рисунке 5а. Чем меньше параметр q , тем меньше ширина профиля ЛВВ. Поскольку откаты выравнивают времена логических процессов, увеличение их количества, т.е. уменьшение параметра q , делают профиль ЛВВ более гладким. Добавление небольшого числа дальних связей также улучшает синхронизацию – ширина профиля при одинаковом параметре q на сетях малого мира ниже, чем на регулярной топологии.

Мы также исследовали, как изменяется показатель роста β в зависимости от параметра q , доли дальних связей p и размера системы N .



а)



б)

Рисунок 5: а) Зависимость средней ширины профиля ЛВВ от времени для разных значений параметра q снизу вверх: $[0.1, 0.2, 0.3, 0.4, 0.5]$. Черной линией обозначена ширина для регулярной топологии $p = 0$, а синей прерывистой – для топологии малого мира с параметром $p = 0.01$. Результаты усреднены по 1000 независимым реализациям. б) Зависимость критического индекса β от параметра q на различных топологиях для системы из $N = 10^3$ логических процессов. Черными кружками обозначены результаты на регулярной топологии $p = 0$, синими квадратами – результаты для топологии малого мира с $p = 0.001$, красными треугольниками – для $p = 0.01$ и зелеными ромбами – для $p = 0.1$. Черной пунктирной линией обозначено значение критического индекса для класса KPZ $\beta = 1/3$.

В критической точке $q \approx 0.136$ в системе из $N = 10^5$ логических процессов на регулярной топологии ($p = 0$) ширина профиля в области $t < t_\times$ растет по степенному закону (Рисунок 5а)

$$w^2(t < t_\times) = 0.618(2) + 0.00043(5)t^{0.553(9)},$$

т.е. показатель $\beta \approx 0.277$. В таблице 4 приведено значение показателя β в критической точке q_c для регулярной топологии ($p = 0$) в зависимости от количества логических процессов. По таблице видно, что β увеличивается с количеством логических процессов. Нас интересует показатель β в термодинамическом пределе бесконечного размера системы, $N \rightarrow \infty$.

На рисунке 5б изображен график зависимости критического индекса роста β от параметра q для различных топологий взаимодействия логических процессов. Поведение критического индекса можно разделить на три режима: при малых, умеренных и больших значениях параметра q . При малых q

N	t_{\times}	A	B	β
10^3	< 100	0.42(2)	0.17(2)	0.016(9)
10^4	≈ 20000	0.600(6)	0.006(2)	0.13(3)
10^5	> 50000	0.618(2)	0.00043(5)	0.277(9)

Таблица 4: Результаты аппроксимации ширины в области $t < t_{\times}$ функцией $w^2(t) = A + Bt^{2\beta}$ для систем разного размера N в критической точке q_c .

профиль локальных времен практически не растет, поэтому и ширина профиля w^2 не растет и остается близкой к нулю. Для умеренных $q \in [0.4; 0.8]$, или в активной фазе, где профиль растет с ненулевой скоростью v и имеется ненулевое количество откатов, во всех топологиях показатель β либо выходит на константу, близкую к $1/3$, либо стремится к этому значению. Для точного определения показателей β к графику 5б необходимы поправки на конечный размер системы. При $q > 0.8$ наблюдается режим, при котором профиль растет практически свободно, без откатов ($q = 1$ соответствует чистому свободному росту). В этом случае ширина профиля растет быстрее всего, показатель β стремится к значению $1/2$.

По-видимому, в активной фазе, при q значительно больше q_c , но несколько меньше 1, показатель β совпадает с показателем из класса KPZ, как в консервативном алгоритме. При q вблизи единицы, система растет по закону случайного роста. Таким образом, модель оптимистического алгоритма в одномерном случае ($p = 0$) демонстрирует несколько режимов поведения в зависимости от значения q – она имеет две критические точки ($q = q_c$ и $q = 1$) и поведение в классе KPZ для промежуточных значений q .

Таким образом, модель эволюции профиля ЛВВ для оптимистического алгоритма по формулировке похожа на модель роста SOS (Solid-on-Solid) [30]. При этом, однако, у нее есть три режима – режим в классе направленной перколяции, режим случайного роста и промежуточный режим, близкий к классу универсальности KPZ. Такое поведение может быть также сопоставлено с поведением поверхности роста в quenched-KPZ.

С точки зрения проведения параллельных вычислений, введение связей типа малого мира улучшает синхронизацию процессорных элементов при небольшом замедлении вычислений. Если консервативный алгоритм свободен от блокировок (deadlock), то в оптимистическом алгоритме при

увеличении частоты нарушения причинности, возможен переход шероховатости в абсорбирующее состояние с нулевой общей производительностью. При умеренных значениях частоты нарушения причинности, общая производительность положительна и десинхронизация выходит на насыщение с ростом числа параллельных процессов.

Проведение моделирование на симуляторе ROSS. Система ROSS (Rensselaer's optimistic simulation systems) [11] – это модульная, написанная на языке C, система параллельного дискретно-событийного моделирования, использующая алгоритм синхронизации Time Warp. ROSS обладает модульной структурой, имеет умеренные требования к памяти за счет использования механизма обратных вычислений.

Симулятор ROSS зарекомендовал себя как надежный программный продукт, ставший стандартом среди ученых, изучающих метод PDES и использующих его для решения прикладных задач [20–22, 42, 43].

Описание и настройка модели PCS. PCS (personal communication service) – это модель беспроводной (сотовой) сети [44–46]. На всей области покрытия сети находятся сотовые вышки. Когда в системе возникают звонки на мобильные устройства, они обрабатываются определенной вышкой. Если абонент во время разговора перемещается между разными зонами (покрываемые разными вышками), то звонок переходит в обслуживание другой вышки при условии, что там есть свободный канал. Если все каналы заняты, звонок обрывается.

Логические процессы – это сотовые вышки. Они расположены на квадратной решетке. В начале программы логические процессы равномерно распределяются по КП и по ПЭ. Внутри ЛП есть цикл – EventHandler (планировщик событий). В нем есть следующие основные процедуры: инициализация ЛП, обработка сообщений, обработка откатов (используются обратные вычисления), завершение работы ЛП, а также сбор статистики. В системе есть 4 основных вида событий: NextCall – это событие, которое генерируется и обрабатывается на одном и том же ЛП (т.е. случай, когда абонент не покидает зону покрытия одной вышки), CompletionCall – завершение звонка, MoveCallIn – “вхождение” звонка из соседней соты, MoveCallOut –

“уход” звонка из зоны покрытия вышки. В ROSS используется экспоненциальное распределение между событиями.

При моделировании можно настраивает следующие параметры модели: количество ЛП, количество КП, начальное количество событий на одном ЛП, среднее время между звонками, не покидающими зоны одной соты, среднее время между звонками, переходящими из зоны покрытия одной вышки в зону другой, средняя продолжительность звонка.

Соотношение моделей OSW и PCS. В модели роста профиля ЛВВ при оптимистическом алгоритме синхронизации мы исследуем поведение локальных времен при изменении двух **независимых** параметров: процента дальних связей p и интенсивности роста q (growth rate). Для краткости нашу модель роста профиля ЛВВ при оптимистическом алгоритме будем называть модель OSW (Optimistic Small World).

В модели PCS напрямую изменять параметры p и q нельзя. Однако, изменяя параметр MOVE_CALL_MEAN, отвечающего за среднее время между звонками, переходящими из одной зоны покрытия сети в другую, можно влиять на количество удаленных событий (remote events). Процент удаленных событий мы обозначаем буквой p' . Влияние параметра MOVE_CALL_MEAN на p' легко объяснить. Удаленные события в системе PCS возникают только в том случае, когда абонент перешел из одной зоны в другую, все остальные звонки обрабатываются ЛП локально. А количество таких “мобильных” абонентов обратно пропорционально параметру MOVE_CALL_MEAN (MCM).

Важно подчеркнуть, что процент удаленных событий p' численно не равен проценту дальних взаимодействий p в модели OSW. В нашей модели не предусмотрен режим, когда ЛП вообще не взаимодействуют друг с другом и все обрабатываемые события являются исключительно локальными. Само понятие локальности в модели OSW подразумевает не отсутствие взаимодействий между ЛП (как это определено в симуляторе ROSS), а взаимодействие лишь соседних ЛП между собой. Мы предполагаем, что ЛП организованы в регулярную топологию и каждый ЛП общается сообщениями как минимум с двумя соседними ЛП. В этом случае $p = 0$. Когда $p > 0$, возникают **дополнительные** взаимозависимости между ЛП, находя-

щимися друг от друга на расстоянии больше 1 (считая по количеству узлов между ЛП). В модели PCS нулевое количество удаленных событий $p' = 0$ означает, что все события возникают и обрабатываются локально на одном ЛП, и все ЛП независимы друг от друга. При $p' > 0$ возникают пересылки данных, но все они происходят между **ближайшими** (соседними) ЛП. Иначе говоря, можно вывести такое соотношение: $p' = 1$ эквивалентно $p = 0$. Тем не менее, смысл у обоих параметров p и p' одинаковый: они отвечают за дополнительные взаимодействия между логическими процессами.

Интенсивность роста в модели роста ЛВВ при оптимистической синхронизации определена формулой:

$$q = \frac{1}{1 + b} \quad (11)$$

где b - это средняя “продолжительность” откатов, учитывая, что “продолжительность” продвижения вперед равна единице. Аналогом параметра q в модели PCS может служить следующее соотношение:

$$q' = 1 - \frac{\text{кол-во откатов}}{\text{кол-во обработанных событий}} \quad (12)$$

Обозначим этот параметр q' . Мы обнаружили, что на самом деле этот параметр **зависит** от количества удаленных событий:

$$q' = q'(p'), \quad (13)$$

поэтому изменять p' и q' по-отдельности невозможно. Для того, чтобы сравнить результаты с двух моделей (нашей модели OSW и модели PCS), было проведено моделирование поведения профиля ЛВВ при оптимистической синхронизации для пары параметров p и q , равных полученным в ходе моделирования системы сотовой связи PCS значениям p' и q' при определенном настраиваемом параметре MOVE_CALL_MEAN (MCM).

Результаты моделирования. Было проведено моделирование на 64 ядрах компьютерного кластера R2D26 [47]. С таким числом процессорных элементов удобнее распределять ЛП и КР по квадратной решетке. Количество ЛП и КР совпадает и равно 256. Обычно один КР обрабатывает события

с нескольких ЛП, однако, поскольку сбор статистики (информации о локальных временах) ведется на базе КР (выводится лишь одно минимальное локальное среди всех логических процессов на этом КР), установка нескольких ЛП на одном КР привело бы к потере информации о локальных временах каждого логического процесса. Поскольку в ROSS не предусмотрена консервативная реализация модели PCS, мы провели только анализ оптимистической синхронизации.

Было изучено, как изменяется скорость и ширина профиля ЛВВ в модели PCS. Данные, получаемые непосредственно с симулятора ROSS, представляют собой таблицу, состоящую из полей: номер PE, номер КР, текущее GVT (глобальное виртуальное время), текущее LVT. В этом примере имеется один процессорный элемент и 4 логических процесса, по одному на каждый КР. Данные собираются в момент расчета GVT. Далее данные преобразуются в таблицу вида: временной шаг t , среднее ЛВВ, средняя ширина профиля ЛВВ. За временной шаг t в модели PCS мы берем момент, когда происходит вычисление GVT, то есть некоторое фиксированное количество событий, определяемые в ROSS параметрами $batch \times gvt\ interval$. Были взяты значения по умолчанию равные 16×16 . Для нахождения среднего LVT, необходимо найти все значения локального времени LVT при фиксированном GVT и усреднить. Ширина профиля ЛВВ есть средне-квадратичное отклонение локальных времен от среднего значения LVT.

Среднее ЛВВ растет линейно с t , и для того, чтобы найти скорость роста профиля ЛВВ, нужно найти наклон прямой LVT(t). Ширина профиля ЛВВ в модели PCS практически сразу выходит на насыщение и не зависит от t . Поэтому для нахождения $\langle w_{\infty}^2 \rangle$ усредняются все значения $w^2(t)$ по времени. Помимо средней скорости и средней ширины профиля ЛВВ, анализируются такие показатели, как Event rate – это количество обработанных событий в секунду, $Efficiency = 1.0 - \frac{E_{RB}}{E_{Net}}$, где E_{RB} общее число откатов, E_{Net} – общее число обработанных сообщений при последовательном моделировании [42], и время выполнения программы ExTime. Полученные данные приведены в таблице 5.

Результаты показали, что качественно поведение ЛВВ в сравниваемых моделях очень похоже и в целом наши предсказания верны. При большем количестве удаленных событий скорость роста профиля, Event rate и эффек-

тивность снижаются, а время выполнения программы увеличивается, при этом ширина профиля уменьшается. На рисунке 6 можно сравнить поведение скорости роста профиля ЛВВ в модели оптимистического алгоритма, скорости роста профиля ЛВВ, Event rate и обратного времени в модели PCS в зависимости от “частоты откатов” (параметра q или q'). Средняя скорость в модели OSW действительно отражает утилизацию процессорного времени, и по поведению она ближе к показателю Event Rate, чем к скорости роста профиля в модели PCS.

Мы аппроксимировали график скорости роста профиля ЛВВ в модели OSW и график event rate в модели PCS вблизи нуля ($q = 0$) функцией $v = v_0(q - q_c)^\nu + \text{const}$ и получили следующие значения параметров:

$$\text{event_rate} \sim (q' - 0.100(2))^{1.54(3)} \quad (14)$$

$$v \sim (q - 0.11(1))^{3,0(2)} \quad (15)$$

Критическая точка q_c , при которой моделирование существенно замедляется (т.е. где скорость роста профиля становится равной нулю) совпадает для обоих случаев и равна приблизительно $q_c \approx 0.1$, а показатели степени отличаются примерно в два раза ($\nu \approx 1.5$ для модели PCS и $\nu \approx 3$ для модели OSW).

На рисунке ?? показана зависимость скорости роста профиля и показателя Event rate от доли дальних взаимодействий p и p' . Качественно все три графика очень похожи, несмотря на различие параметров p и p' , описанное выше.

Таким образом, можно сказать, что скорость роста профиля в модели OSW действительно отражает такой показатель реального моделирования, как event rate (основной параметр, по которому оценивают эффективность моделирования во время case-studies [20–22]). Поскольку точка остановки моделирования в системе ROSS определяется количеством обработанных событий, время моделирования обратно пропорционально параметру event rate

$$\text{Execution Time} \sim \frac{1}{\text{Event Rate}},$$

соответственно, чем выше скорость роста профиля, тем быстрее завершается моделирование. Небольшие отличия в поведении скорости в модели

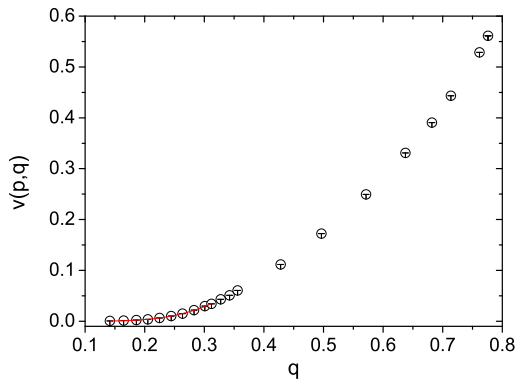
MCM	remote,%	q'	v	w^2	EvRate	ExTime
300	18.21	0.24	1675.29(2)	16225	2315642.3	103.2613
450	13.68	0.30	2351.31(3)	26004	3471274.8	60.4699
600	10.78	0.36	2988.14(6)	35628	4787282.1	40.7795
800	8.28	0.43	3805.5(1)	47197	6775426.8	27.1874
1000	6.66	0.50	4586.6(1)	56834	8933732.9	19.8764
1250	5.31	0.57	5432.3(1)	67373	11633740.6	14.8056
1500	4.39	0.64	6191.5(1)	72562	14386931.4	11.7267
1750	3.73	0.68	6730.7(2)	82408	16507380.8	10.0664
2000	3.23	0.71	7125.8(3)	93951	18337217.9	8.9581
3000	2.09	0.76	7807.3(8)	198909	21639091.6	7.3862
4000	1.54	0.7759	8057.8(5)	361910	23043327.3	6.8399
4500	1.35	0.7761	8100.1(3)	492374	23222975.3	6.7551

Таблица 5: Зависимость различных показателей от среднего времени между “перемещающимися” звонками MoveCallMean (MCM), значение NextCallMean = 360.

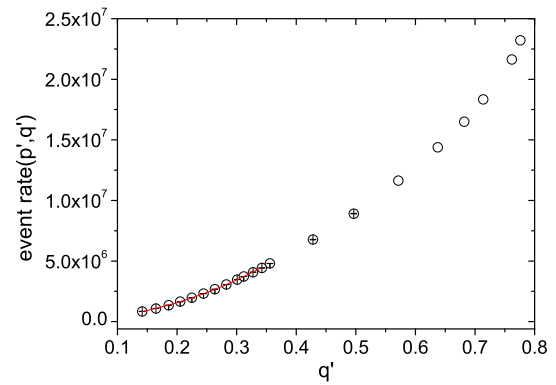
OSW и PCS можно объяснить тем, что в системе PCS могут возникать дополнительные корреляции, которые мы не учитываем в нашей модели OSW.

Что касается поведения средней ширины профиля ЛВВ, то качественно результаты также похожи (Рисунок 7), хотя в модели PCS наблюдается более резкое изменение ширины профиля в зависимости от q' и p' . Ширина профиля отражает степень рассинхронизации между процессами. Видно, что в модели PCS ширина дольше остается умеренной, даже при достаточно больших q' .

Таким образом, параметры p и q в реальном моделировании изменяются вследствие изменения других параметров, связанных с физической системой (например, среднее время между перемещающимися звонками), а также они зависят друг от друга. Скорость роста профиля ЛВВ в модели OSW действительно отражает эффективность моделирования, а именно, Event rate. Критическая точка q_c для OSW и PCS совпадает, а показатель степени ν в два раза выше для модели OSW, чем для модели PCS. Ширина профиля ЛВВ в моделях OSW и PCS также имеют схожее поведение.

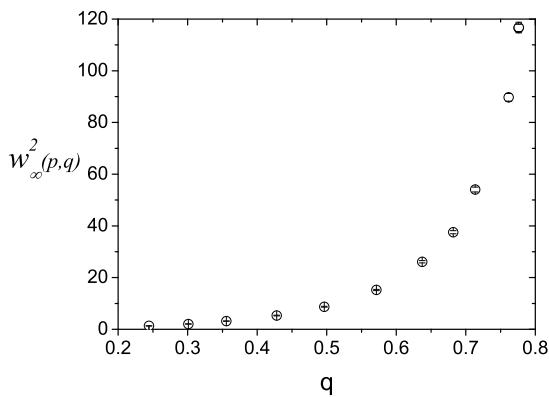


а) Скорость роста профиля в нашей модели OSW как функция q .

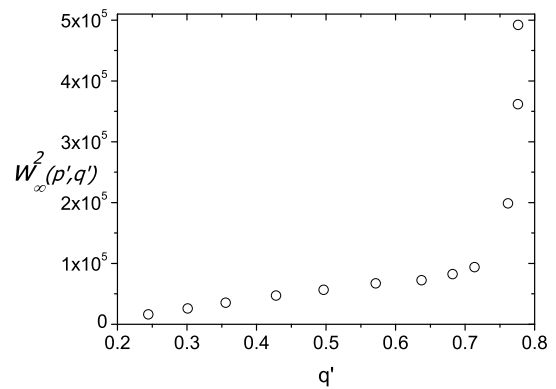


б) Event rate как функция q' .

Рисунок 6: Сравнение поведения скорости роста профиля ЛВВ в модели оптимистического алгоритма, скорости роста профиля ЛВВ, Event rate и обратного времени в модели PCS в зависимости от “частоты откатов”.



а) Поведение средней ширины профиля в нашей модели OSW в зависимости от параметра q .



б) Поведение средней ширины профиля в модели PCS в зависимости от параметра q' .

Рисунок 7: Сравнение поведения средней ширины профиля ЛВВ.

Апробация результатов исследования.

Основные результаты диссертации докладывались и обсуждались на конференциях:

1. International Conference on Computer Simulation in Physics and beyond, September 6-10, 2015, Moscow, Russia, постерный доклад “Virtual Time Profile Modeling in Parallel Discrete Event Simulation”, <http://csp2015.ac.ru/index.html>
2. International Conference “Supercomputer Simulations in Science and Engineering”, September 6-10, 2016, Moscow, Russia, устный доклад, <http://ssse2016.ac.ru/>
3. Landau Days 2017. June 26-29 Chernogolovka, Russia, постерный доклад “Synchronisation in parallel event simulation algorithm on the small-world network”, <http://landaudays.itp.ac.ru/ld2017/>
4. XXIX IUPAP Conference in Computational Physics (CCP2017), Paris, July 9th - July 13th, 2017, устный доклад: “Synchronisation of Conservative Parallel Discrete Event Simulations in Small World Network”, <https://ccp2017.sciencesconf.org>
5. The Sixth China-Russia Conference on Numerical Algebra with Applications (CRCNAА 2017), 28–30 августа 2017, Moscow, Russia, устный доклад “Analysis of Local Virtual Time Profile in Parallel Discrete Event Simulation”, <https://miem.hse.ru/crcnaa/>
6. International Conference on Computer Simulation in Physics and beyond, October 9-12, 2017, Moscow, Russia, устный доклад, <http://csp2017.ac.ru/registrartion.html>

7. VIII International Conference “Distributed Computing and Grid-technologies in Science and Education” (GRID), 10-14 сентября, Дубна, Россия, устный доклад “Properties of The Parallel Discrete Event Simulation Algorithms on Small-World Communication Networks”, <http://indico.jinr.ru/event/447/>
8. International Conference on Computer Simulation in Physics and beyond September 24-27, 2018, Moscow, Russia, устный доклад “The analysis of optimistic parallel discrete event simulation algorithm on small-world networks”, <http://csp2018.ac.ru>
9. The 5th International Conference on Matrix Methods in Mathematics and applications, August 19-23, 2019 Moscow, Russia, устный доклад, <http://deeptensor.ml/mmma2019/>

Публикации по теме диссертации.

1. Ziganurova L., Novotny M. A., Shchur L. N. Model for the evolution of the time profile in optimistic parallel discrete event simulations //Journal of Physics: Conference Series, 681(1), 012047 (2016).
2. Shchur L., Ziganurova L. Simulation of virtual time profile in conservative parallel discrete event simulation algorithm for small-world network // Lobachevskii Journal of Mathematics, 38(5), 967-970 (2017).
3. Ziganurova L., Shchur L. Properties of the Conservative Parallel Discrete Event Simulation Algorithm. In: Malyskin V. (eds) Parallel Computing Technologies. PaCT 2017. Lecture Notes in Computer Science, vol 10421. Springer, Cham (2017)
4. Хомутов Е. В., Зиганурова Л. Ф. Анализ виртуального времени в оптимистическом алгоритме параллельного моделирования дискретных событий // В кн.: Новые информационные технологии в автоматизированных системах: материалы двадцать первого научно-практического семинара. М.: Институт прикладной математики им. М.В. Келдыша РАН, 2018. С. 294-299. (2018)

5. Ziganurova, L., Shchur, L. N. Synchronization of conservative parallel discrete event simulations on a small-world network // Physical Review E, 98(2), 022218 (2018)
6. Properties of The Parallel Discrete Event Simulation Algorithms on Small-World Communication Networks in Proceedings of the VIII International Conference “Distributed Computing and Grid-technologies in Science and Education” (2018)
7. Щур Л.Н., Зиганурова Л.Ф. "Синхронизация процессов при параллельном моделировании дискретных событий Журнал экспериментальной и теоретической физики, том 156 (4) (2019)

Литература

1. Top 500 of supercomputers. Электронный ресурс: <https://www.top500.org/>
2. Workshop on Research Challenges in Modeling and Simulation for Engineering Complex Systems (2016)
3. Nilmeier J. P., Marian J. A rigorous sequential update strategy for parallel kinetic Monte Carlo simulation //Computer Physics Communications. – 2014. – Т. 185. – №. 10. – С. 2479-2486.
4. Garcia Cardona C., Tikare V., Plimpton S. J. Parallel simulation of 3D sintering //International Journal of Computational Materials Science and Surface Engineering. – 2010. – Т. 4. – №. 1. – С. 37-54.
5. D'Angelo G., Ferretti S., Ghini V. Simulation of the Internet of Things // 2016 International Conference on High Performance Computing & Simulation (HPCS). – IEEE, 2016. – С. 1-8.
6. Plagge M. et al. Nemo: A massively parallel discrete-event simulation model for neuromorphic architectures //ACM Transactions on Modeling and Computer Simulation (TOMACS). – 2018. – Т. 28. – №. 4. – С. 30.
7. Jefferson, D. R. (1985). Virtual time. ACM Transactions on Programming Languages and Systems (TOPLAS), 7(3), 404-425.
8. Barnes Jr P. D. et al. Warp speed: executing time warp on 1,966,080 cores //Proceedings of the 1st ACM SIGSIM Conference on Principles of Advanced Discrete Simulation. – ACM, 2013. – С. 327-336.
9. Fujimoto R. M. Parallel discrete event simulation //Communications of the ACM. – 1990. – Т. 33. – №. 10. – С. 30-53.

10. Shchur L. N., Novotny M. A. Evolution of time horizons in parallel and grid simulations //Physical Review E. – 2004. – T. 70. – №. 2. – C. 026703.
11. Carothers C. D., Bauer D., Pearce S., ROSS: A high-performance, low-memory, modular Time Warp system //Journal of Parallel and Distributed Computing. – 2002. – T. 62. – №. 11. – C. 1648-1669.
12. Korniss G. et al. From massively parallel algorithms and fluctuating time horizons to nonequilibrium surface growth //Physical review letters. – 2000. – T. 84. – №. 6. – C. 1351.
13. Kardar M., Parisi G., Zhang Y. C. Dynamic scaling of growing interfaces //Physical Review Letters. – 1986. – T. 56. – №. 9. – C. 889.
14. Korniss G. et al. Suppressing roughness of virtual times in parallel discrete-event simulations //Science. – 2003. – T. 299. – №. 5607. – C. 677-679.
15. Guclu H. et al. Synchronization landscapes in small-world-connected computer networks //Physical Review E. – 2006. – T. 73. – №. 6. – C. 066115.
16. Hinrichsen H. Non-equilibrium phase transitions with long-range interactions //Journal of Statistical Mechanics: Theory and Experiment. – 2007. – T. 2007. – №. 07. – C. P07006.
17. Watts D. J., Strogatz S. H. Collective dynamics of ‘small-world’ networks //Nature. – 1998. – T. 393. – №. 6684. – C. 440.
18. Cardoen B. et al. A PDEVs simulator supporting multiple synchronization protocols: implementation and performance analysis //Simulation. – 2018. – T. 94. – №. 4. – C. 281-300.
19. Perumalla K. S. /spl mu/sik-a micro-kernel for parallel/distributed simulation systems //Workshop on Principles of Advanced and Distributed Simulation (PADS’05). – IEEE, 2005. – C. 59-68.
20. Wilsey P. A. Some properties of events executed in discrete-event simulation models //Proceedings of the 2016 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation. – ACM, 2016. – C. 165-176.

21. Williams B. et al. Performance characterization of parallel discrete event simulation on knights landing processor //Proceedings of the 2017 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation. – ACM, 2017. – C. 121-132.
22. Eker A. et al. Performance implications of global virtual time algorithms on a knights landing processor //Proceedings of the 22nd International Symposium on Distributed Simulation and Real Time Applications. – IEEE Press, 2018. – C. 87-96.
23. Vee V. Y., Hsu W. J. Parallel discrete event simulation: A survey //Technical Report, Centre for Advanced Information Systems. – 1999.
24. Ziganurova L., Shchur L. N. Synchronization of conservative parallel discrete event simulations on a small-world network //Physical Review E. – 2018. – T. 98. – №. 2. – C. 022218.
25. Jefferson D., Fujimoto R. A Brief History of Time Warp //Advances in Modeling and Simulation. – Springer, Cham, 2017. – C. 97-134.
26. Ziganurova L., Novotny M. A., Shchur L. N. Model for the evolution of the time profile in optimistic parallel discrete event simulations //Journal of Physics: Conference Series. – IOP Publishing, 2016. – T. 681. – №. 1. – C. 012047.
27. Hinrichsen H. Non-equilibrium critical phenomena and phase transitions into absorbing states //Advances in physics. – 2000. – T. 49. – №. 7. – C. 815-958.
28. Reis F. D. A. Depinning transitions in interface growth models //Brazilian journal of physics. – 2003. – T. 33. – №. 3. – C. 501-513.
29. Grassberger P. On phase transitions in Schlögl's second model //Zeitschrift für Physik B Condensed Matter. – 1982. – T. 47. – №. 4. – C. 365-374.
30. Odor G. Universality classes in nonequilibrium lattice systems //Reviews of modern physics. – 2004. – T. 76. – №. 3. – C. 663.
31. Liggett T. M. Interacting particle systems. – Springer Science & Business Media, 2012. – T. 276.

32. Dickman R., Burschka M. A. Nonequilibrium critical poisoning in a single-species model //Physics Letters A. – 1988. – T. 127. – №. 3. – C. 132-137.
33. Grassberger P. On the critical behavior of the general epidemic process and dynamical percolation //Mathematical Biosciences. – 1983. – T. 63. – №. 2. – C. 157-172.
34. Ziff R. M., Gulari E., Barshad Y. Kinetic phase transitions in an irreversible surface-reaction model //Physical Review Letters. – 1986. – T. 56. – №. 24. – C. 2553.
35. Schlogl F. Chemical reaction models for non-equilibrium phase transitions //Zeitschrift fur physik. – 1972. – T. 253. – №. 2. – C. 147-161.
36. Albano E. V. Spreading analysis and finite-size scaling study of the critical behavior of a forest fire model with immune trees //Physica A: Statistical Mechanics and its Applications. – 1995. – T. 216. – №. 3. – C. 213-226.
37. Takayasu H., Tretyakov A. Y. Extinction, survival, and dynamical phase transition of branching annihilating random walk //Physical review letters. – 1992. – T. 68. – №. 20. – C. 3060.
38. Hoyuelos M., Albano E. V., Martín H. O. A multilayer contact process //Journal of Physics A: Mathematical and General. – 1997. – T. 30. – №. 2. – C. 431.
39. Kertesz J., Wolf D. E. Anomalous roughening in growth processes //Physical review letters. – 1989. – T. 62. – №. 22. – C. 2571.
40. Alon U. et al. Roughening transition in a one-dimensional growth process //Physical review letters. – 1996. – T. 76. – №. 15. – C. 2746.
41. Fujimoto R. M., Panesar K. S. Buffer management in shared-memory Time Warp systems //Acm sigsim simulation digest. – 1995. – T. 25. – №. 1. – C. 149-156.
42. Bauer Jr D. W., Carothers C. D., Holder A. Scalable time warp on blue gene supercomputers //Proceedings of the 2009 ACM/IEEE/SCS 23rd Workshop

on Principles of Advanced and Distributed Simulation. – IEEE Computer Society, 2009. – С. 35-44.

43. Jagtap D., Abu-Ghazaleh N., Ponomarev D. Optimization of parallel discrete event simulator for multi-core systems //2012 IEEE 26th International Parallel and Distributed Processing Symposium. – IEEE, 2012. – С. 520-531.
44. Carothers C. D. et al. Distributed simulation of large-scale PCS networks //Proceedings of International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems. – IEEE, 1994. – С. 2-6.
45. Carothers C. D., Fujimoto R. M., Lin Y. B. A case study in simulating PCS networks using Time Warp //ACM SIGSIM Simulation Digest. – 1995. – Т. 25. – №. 1. – С. 87-94.
46. Carothers, Bauer, Pearce ROSS: Renselaer’s Optimistic Simulation System. User’s Guide, 2002
47. Вычислительный узел R2D26. Электронный ресурс:
<http://rscf158.comphys.ru/r2d26.pdf>