МОДЕЛЬ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ ДЛЯ ОЦЕНКИ МАСШТАБИРУЕМОСТИ ИТЕРАЦИОННЫХ АЛГОРИТМОВ НА КЛАСТЕРНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМАХ

05.13.11 - математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей

Диссертация на соискание ученой степени кандидата физико-математических наук

Ежова Надежда Александровна

Научный руководитель:
СОКОЛИНСКИЙ Леонид Борисович,
доктор физ.-мат. наук, профессор

Исследование выполнено при финансовой поддержке Правительства РФ в соответствии с Постановлением №211 от 16.03.2013 г. (соглашение № 02.A03.21.0011) и Министерства образования и науки РФ (государственное задание 2.7905.2017/8.9).

Цель диссертационной работы

Разработать и исследовать новую модель параллельных вычислений для итерационных алгоритмов применительно к многопроцессорным системам с распределенной памятью, позволяющую предсказывать границу масштабируемости алгоритма на ранних стадиях его проектирования, и построить на ее основе параллельный каркас для кластерных вычислительных систем экзафлопсного уровня производительности

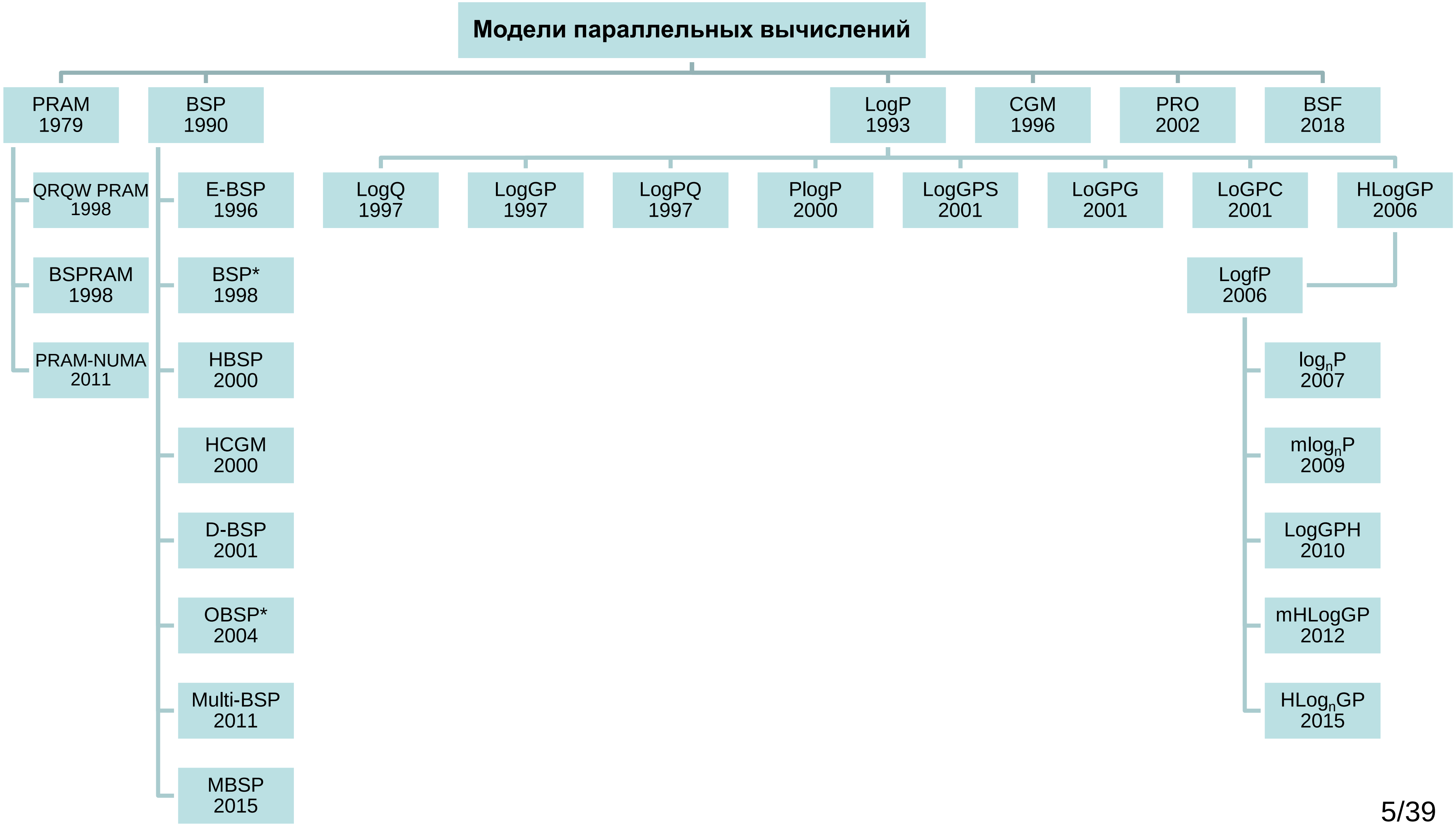
Основные задачи

1. Разработать модель параллельных вычислений
2. Создать на языке C++ параллельный каркас, основанный на разработанной модели
3. Выполнить проектирование и реализацию визуального конструктора программ на языке C++ в соответствии с разработанными моделью и параллельным каркасом
4. Провести вычислительные эксперименты для верификации предложенной модели и разработанного программного обеспечения

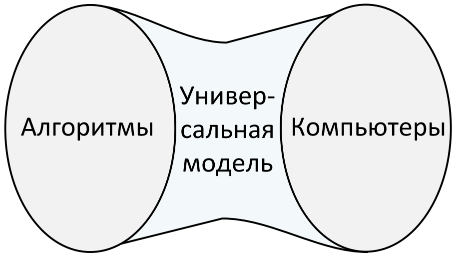
Модель параллельных вычислений

- Модель параллельных вычислений – это фреймворк (система правил и ограничений) для описания и анализа параллельных алгоритмов и программ
- Применение
  - Прогноз времени выполнения
  - Оценка масштабируемость алгоритма

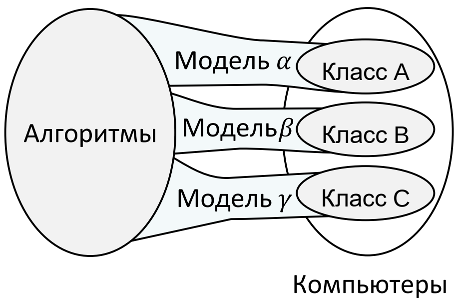
Дерево моделей параллельных вычислений



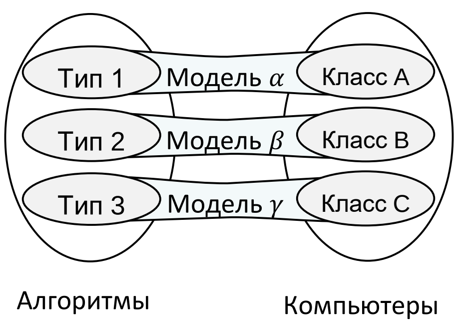
Модель, соединяющая алгоритмы и компьютеры



\+ Универсальность
\- Невозможно создать ***адекватную*** универсальную модель для современных компьютеров



\+ Адекватность
± Ограниченная универсальность
\- Сложность практического применения



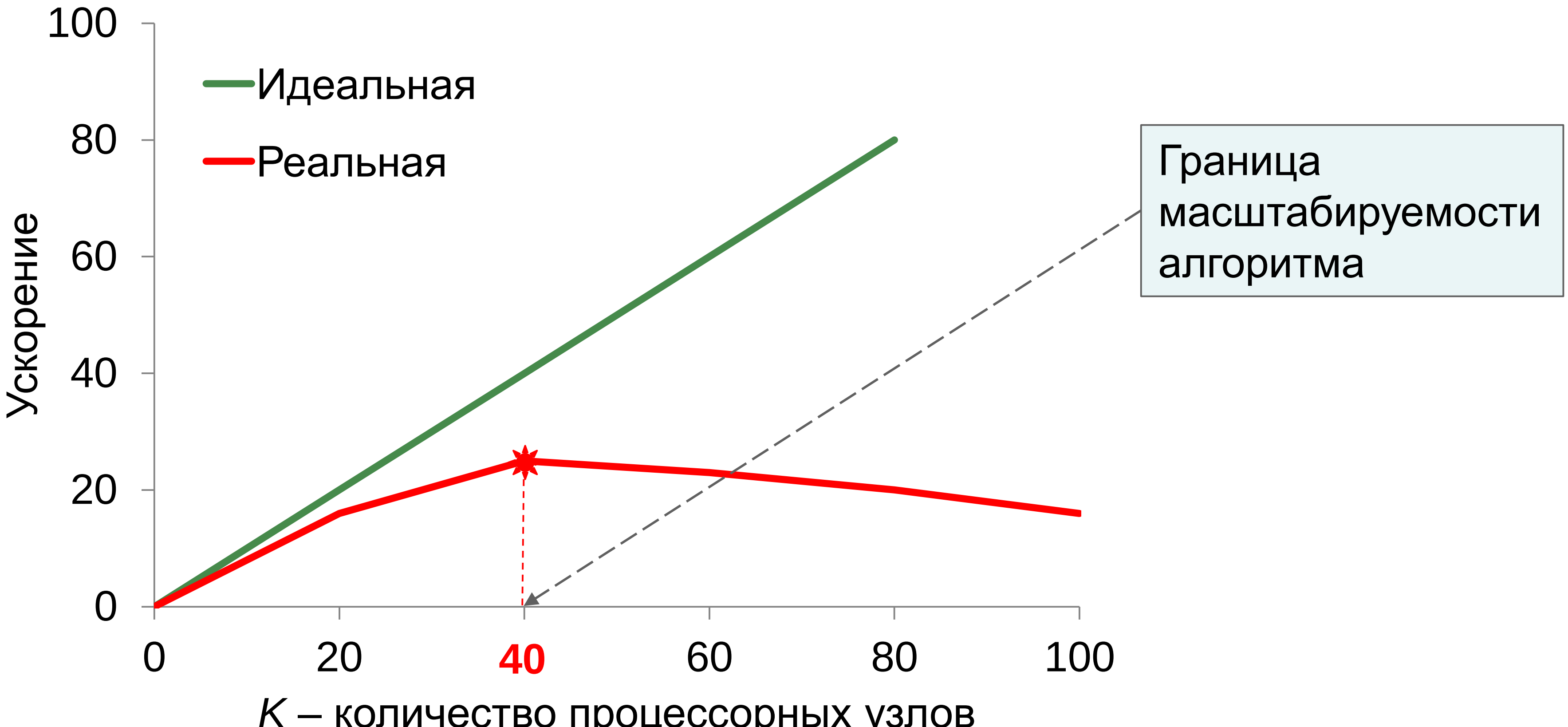
\+ Адекватность
\+ Простота использования
\- Универсальность

Модель параллельных вычислений BSF (Bulk-Synchronous Farm)

- Область применения:
  - Многопроцессорные системы с распределенной памятью
  - Параллельные итерационные алгоритмы с высокой вычислительной сложностью
- Позволяет предсказать:
  - ускорение параллельного алгоритма
  - **границу масштабируемости параллельного алгоритма** (уникальное качество модели BSF)

Кривая ускорения – основной индикатор масштабитруемости

$$a(K) = \frac{t_1}{t_K}$$

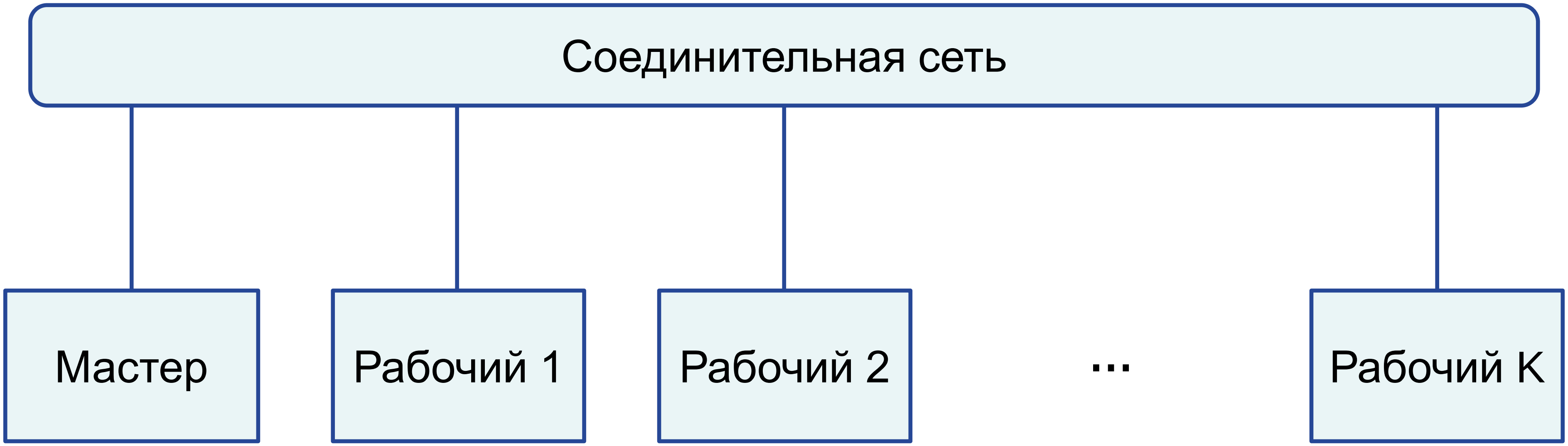


Фундамент модели BSF

Модель BSF
(Bulk-Synchronous Farm)

- Схема программирования SPMD (Single Program – Multiple Data)
- Парадигма «Мастер-рабочие» (Farm Skeleton)
- Модель параллельных вычислений BSP (Bulk-Synchronous Parallelism)
- BMF (Bird-Meertens formalism)

BSF-компьютер



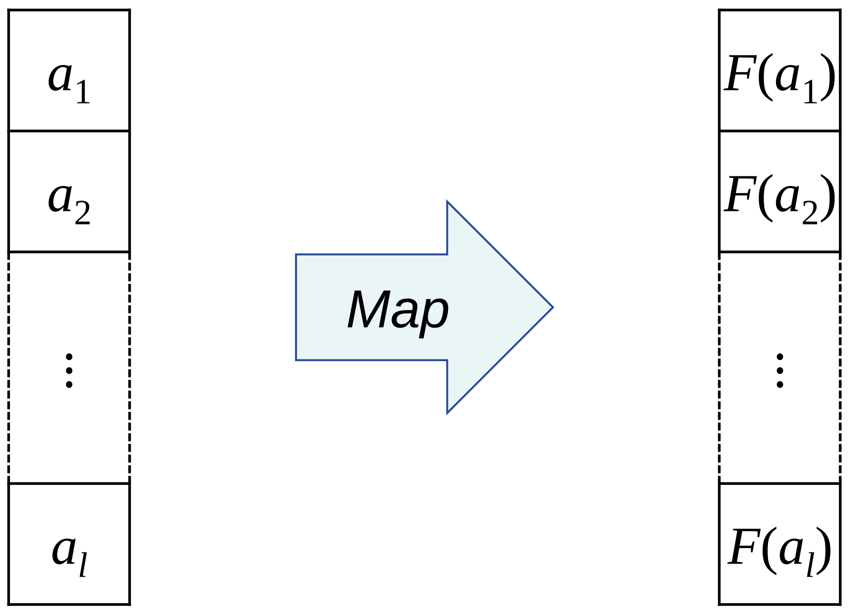
Процессорные узлы

Представление алгоритма в виде операций над списками

- Функции высшего порядка:
  - Map
  - Reduce

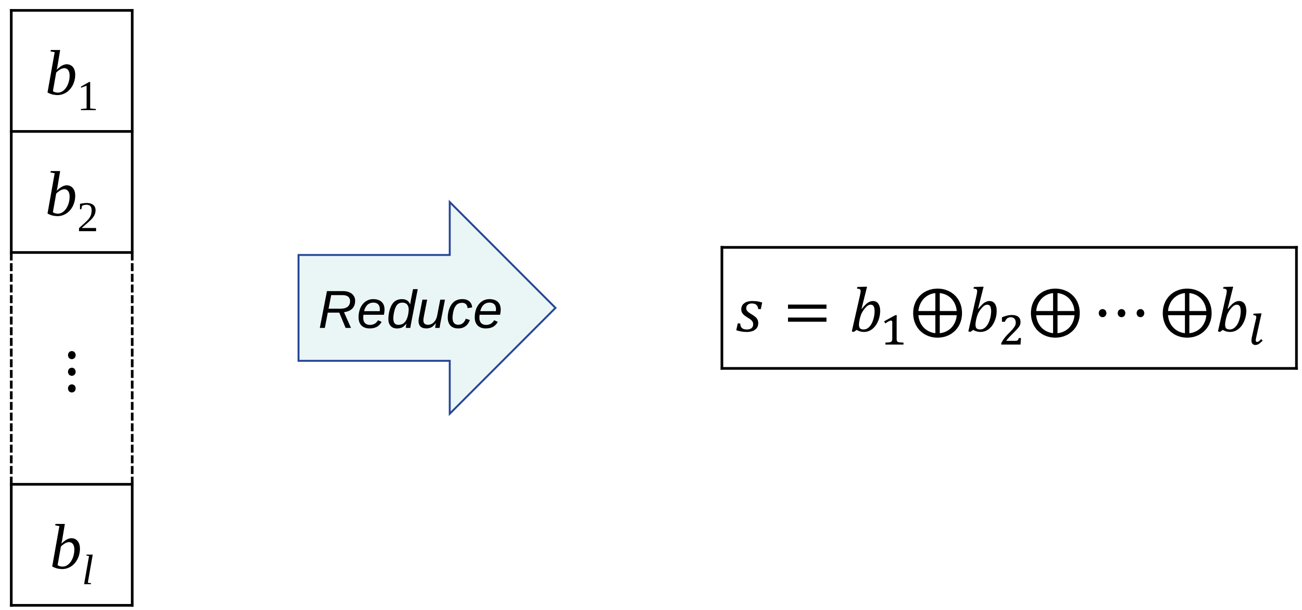
(Формализм Бёрда-Миртенса)

Функция высшего порядка *Map*



$$Map(F, [a_1, \ldots, a_l]) = [F(a_1), \ldots, F(a_l)]$$

Функция высшего порядка *Reduce*



$$Reduce(\oplus, [b_1, \dots, b_l]) = b_1 \oplus \cdots \oplus b_l$$

Шаблон последовательного алгоритма в модели BSF

1. $i \coloneqq 0; Input(A, x_0)$
2. $B \coloneqq Map\left(F_{x_i}, A\right)$
3. $s \coloneqq Reduce(\oplus, B)$
4. $x_{i+1} \coloneqq Compute(x_i, s); i \coloneqq i + 1$
5. **if** $StopCond(x_{i-1}, x_i)$ **go to** 7
6. **go to** 2
7. $Output(x_i)$; **stop**

i	- номер итерации
$A \in [\mathcal{A}]$	- список исходных элементов данных
x_0	- начальное приближение
$F_x: \mathcal{A} \to \mathcal{B}$	- параметризованная функция
$B \in [\mathcal{B}]$	- список результирующих элементов
$\oplus$	- ассоциативная операция
x_i	- *i*-тое приближение

Шаблон параллельного алгоритма в модели BSF

Шаг	Мастер	Рабочий j (j=1,…,K)
1.	$i \coloneqq 0; Input(x_0)$	$Input\left(A^{[j]}\right)$
2.	$SendToWorkers(x_i)$	$RecvFromMaster(x_i)$
		$B^{[j]} \coloneqq Map\left(F_{x_i}, A^{[j]}\right)$
3.		$s^{(j)} \coloneqq Reduce\left(\oplus, B^{[j]}\right)$
	$RecvFromWorkers([s^{(1)}, \dots, s^{(K)}])$	$SendToMaster(s^{(j)})$
	$s \coloneqq Reduce\left(\oplus, [s^{(1)}, \dots, s^{(K)}]\right)$	
4.	$x_{i+1} \coloneqq Compute(x_i, s);\ i \coloneqq i + 1$	
5.	**if** $StopCond(x_{i-1}, x_i)$ **go to** 7	
6.	**go to** 2	**go to** 2
7.	$Output(x_i)$; **stop**	

Стоимостная метрика модели BSF

K	– количество рабочих
l	– длина обрабатываемого списка
L	– латентность (время посылки сообщения длиной в 1 байт)
t_s	– время передачи сообщения от мастера рабочему (без учета латентности)
t_r	– время передачи сообщения от рабочего мастеру (без учета латентности)
t_{Map}	– время выполнения функции Map для всего списка исходных данных
t_a	– время выполнения операции $\oplus$
t_p	– время на обработку результатов итерации

Ускорение в модели BSF

Время решения задачи
1 мастером и
1 рабочим

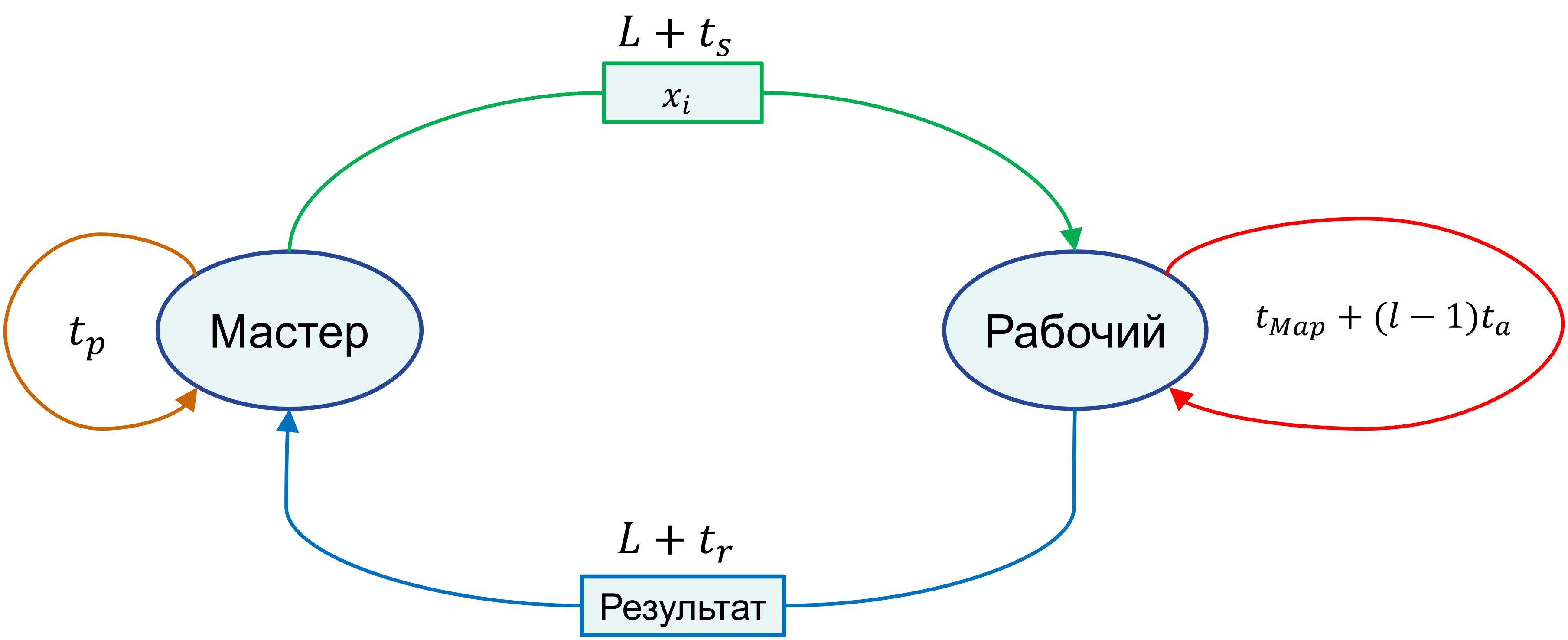
$$a(K) = \frac{T_1}{T_K}$$

Количество рабочих

Время решения задачи
1 мастером и
K рабочими

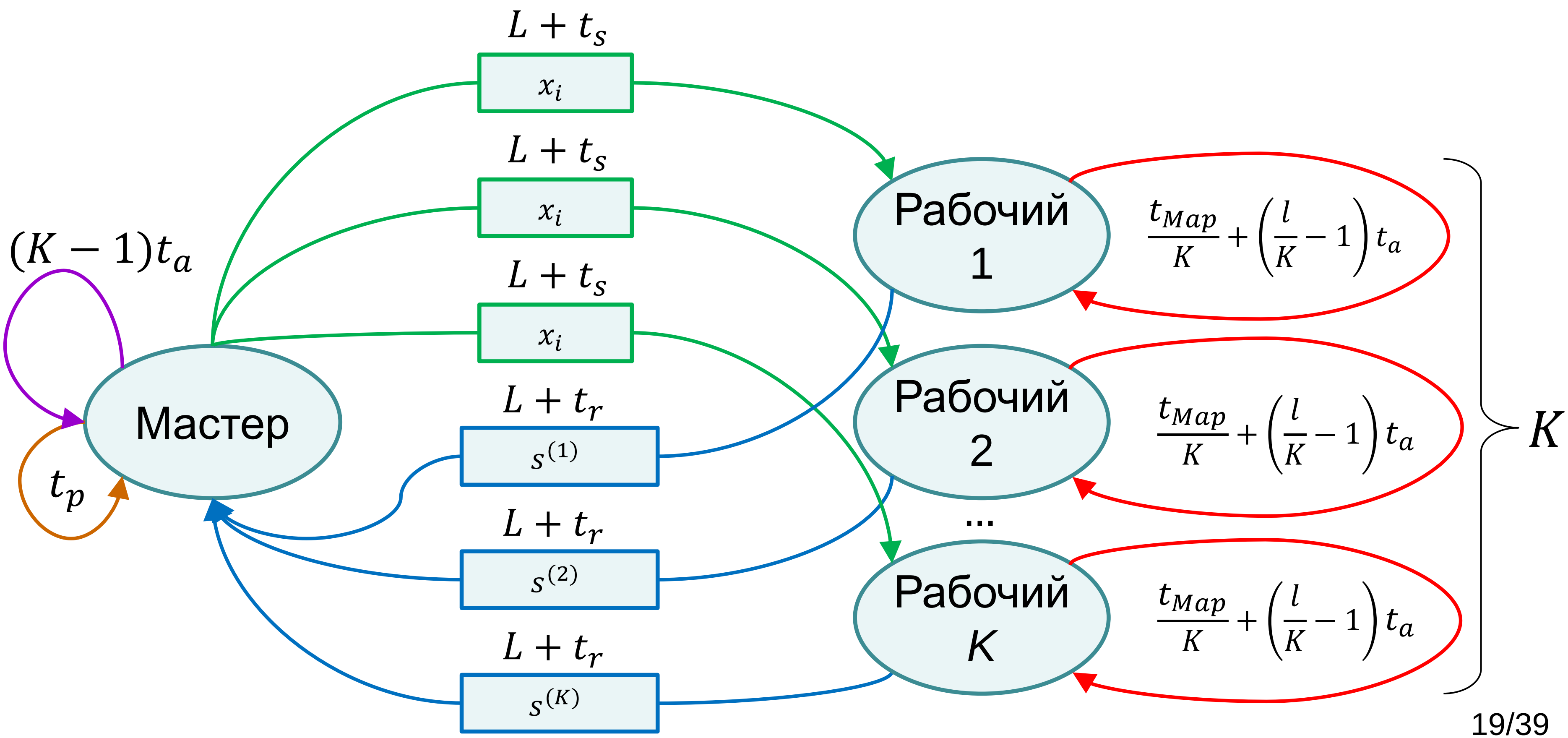
Оценка времени T_1

$$T_1 = L + t_s + t_{Map} + (l-1)t_a + L + t_r + t_p$$



Оценка времени T_K

$$T_K = K(L + t_s) + \frac{t_{Map}}{K} + \left(\frac{l}{K} - 1\right) t_a + K(L + t_r) + (K-1)t_a + t_p$$



Ускорение для модели BSF

$$a_{BSF}(K) = \frac{2L + t_s + t_r + t_p + t_{Map} + lt_a}{K(2L + t_s + t_r + t_a) + \frac{(t_{Map} + lt_a)}{K} - t_a + t_p}$$

Теорема о границе масштабируемости BSF-алгоритма

Время выполнения операции *Map*

Время выполнения операции Reduce

Граница масштабируемости BSF-алгоритма

Время выполнения ⊕

$$K_{Max} = \sqrt{\frac{t_{Map} + l t_a}{2L + t_s + t_r + t_a}}$$

Латентность

Время посылки сообщения одному рабочему

Время получения результата от одного рабочего

Параллельный BSF-каркас

- C++
- MPI + OpenMP
- Исходные коды
  https://github.com/nadezhda-ezhova/BSF-MR

BSF-Studio

- Веб-приложение для быстрой разработки BSF-программ на языке C++
- Базируется на BSF-каркасе

Верификация модели BSF

- Имитационное моделирование
- Алгоритм Jacobi
- Гравитационная задача

Имитационное моделирование

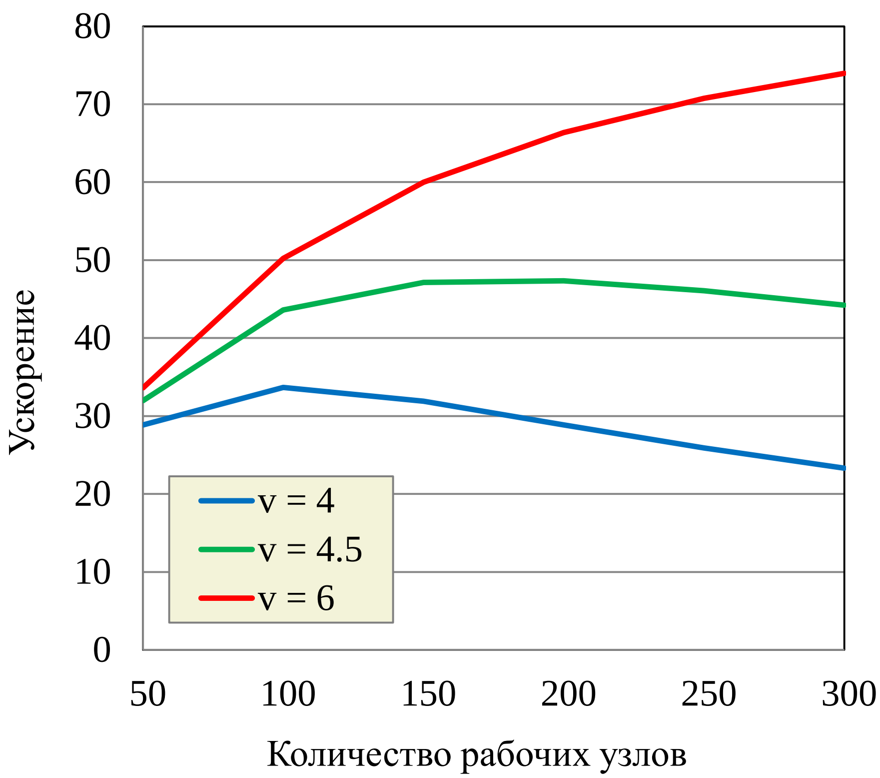
Время выполнения задания рабочими узлами

$$v = \lg\left(\frac{t_w}{t_s}\right)$$

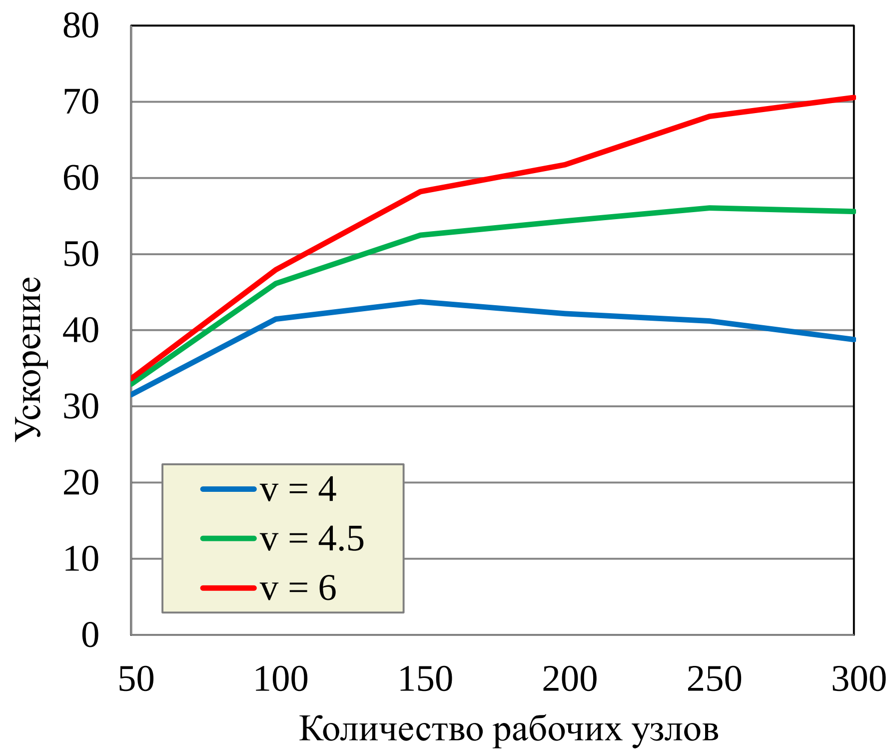
Время передачи сообщения от мастера рабочему

- дополнительный параметр *v*
- связывает t_w и t_s
- показывает адекватность модели при разных соотношениях

Результаты исследования ускорения при имитационном моделировании



Аналитическая модель



Имитационная модель

Алгоритм Jacobi для приближенного решения СЛАУ

$$Ax = b$$

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix} \quad x = (x_1, \dots, x_n) \quad b = (b_1, \dots, b_n)$$

$$C = \begin{pmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{n1} & \cdots & c_{nn} \end{pmatrix} \quad c_{ij} = \begin{cases} -\dfrac{a_{ij}}{a_{ii}}, \forall j \neq i \\ 0, \forall j = i \end{cases}$$

$$d = (d_1, \dots, d_n) \qquad d_i = b_i / a_{ii}$$

$$x^{(k+1)} = Cx^{(k)} + d$$

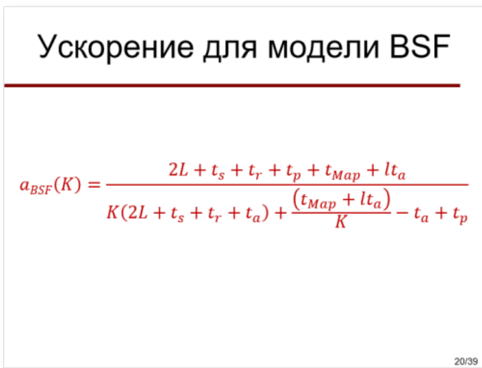
Алгоритм Jacobi-BSF

$F_x(j) = \left(c_{1j}x_j, \dots, c_{nj}x_j\right)$, j – номер столбца матрицы C

1. $Input(C, d); k \coloneqq 0; x^{(0)} \coloneqq d$
2. $[g^1, \dots, g^n] \coloneqq Map\left(F_{x^{(k)}}, [1, \dots, n]\right)$
3. $g \coloneqq Reduce(+, [g^1, \dots, g^n])$
4. $x^{(k+1)} \coloneqq g + x^{(k)}$; $k \coloneqq k + 1$
5. **if** $\left\|x^{(k)} - x^{(k-1)}\right\|^2 < \varepsilon$ **go to** 7
6. **go to** 2
7. $Output\left(x^{(k)}\right)$; **stop**

BSF-оценки для алгоритма Jacobi-BSF

τ_{op} - время выполнения одной операции с плавающей точкой
τ_{tr} - время пересылки вещественного числа (без учета латентности)
n - количество уравнений
L - латентность



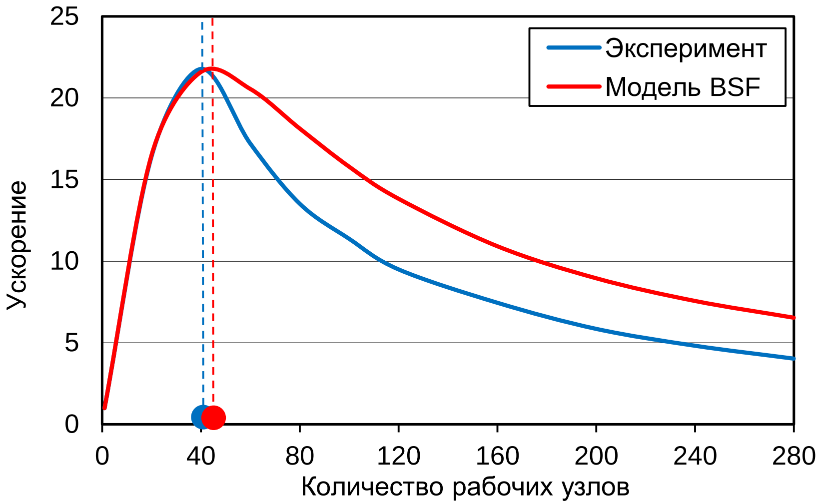
$$\Rightarrow a_{Jacobi}(K) = \frac{2(L + \tau_{tr} n) + \tau_{op} n(3n - K + 5)}{K^2\left(2(L + \tau_{op} n) + 3\tau_{op} n(n + K)\right)}$$

$$\Rightarrow K_{Max} = \sqrt{\frac{\tau_{op} n(3n - K)}{2(L + \tau_{tr} n) + \tau_{op} n}}$$

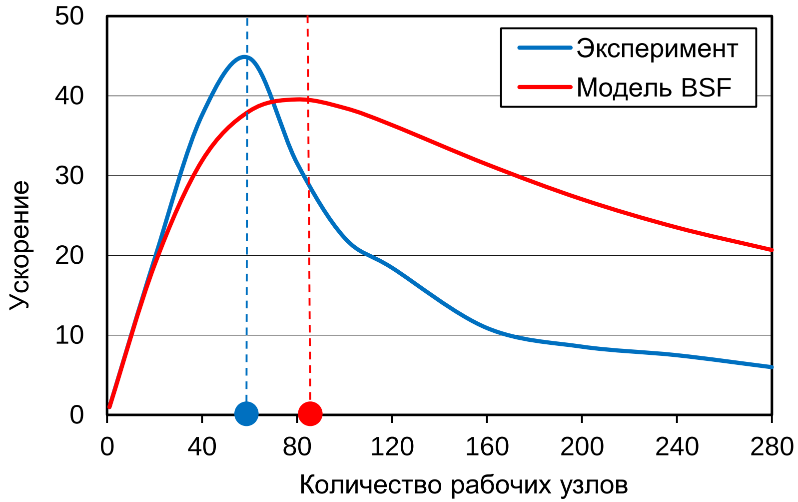
$$\boldsymbol{K_{Jacobi}^{Max} = O\left(\sqrt{n}\right)}$$

Ускорение алгоритма Jacobi-BSF: теория и практика

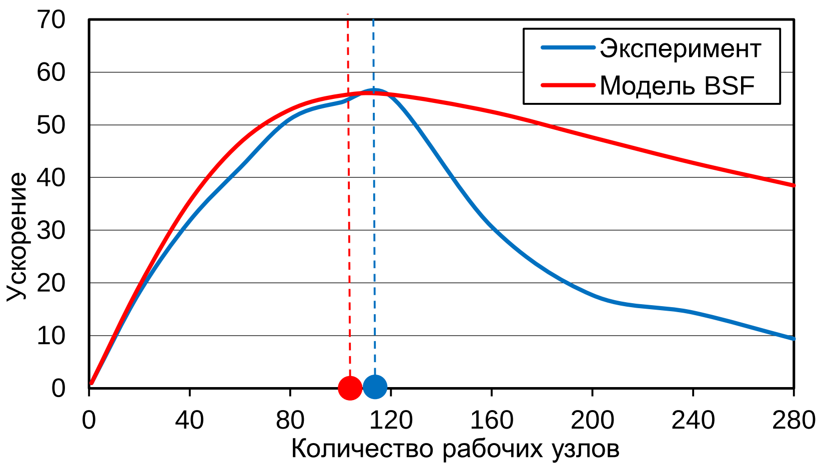
$n = 1\,500$



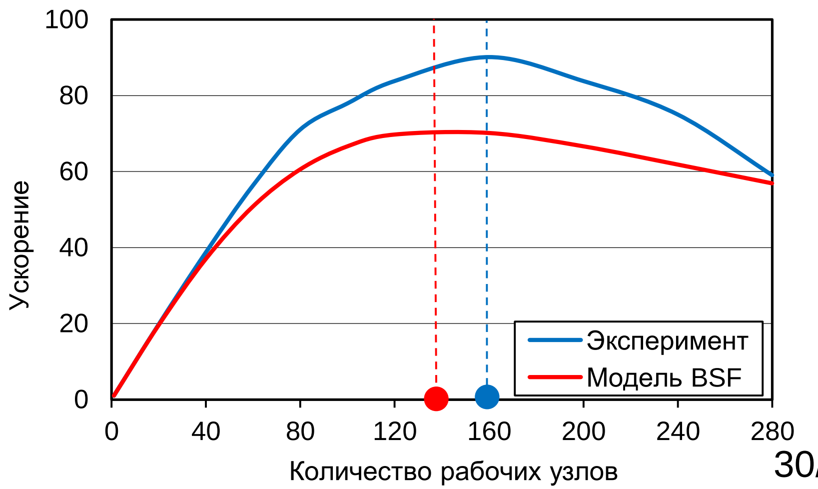
$n = 5\,000$



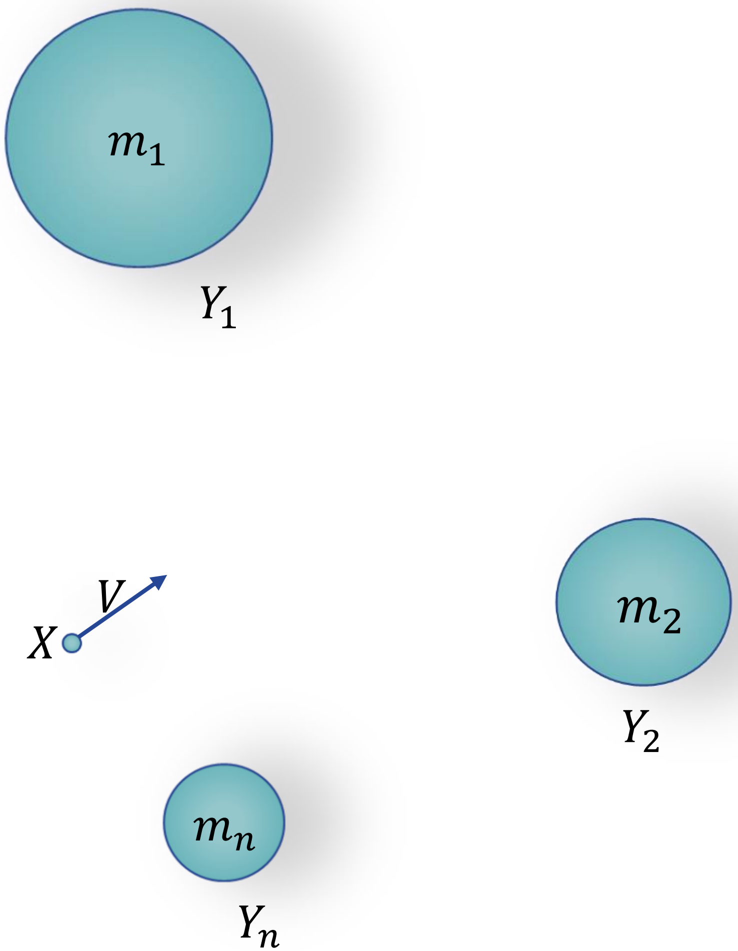
$n = 10\,000$



$n = 16\,000$

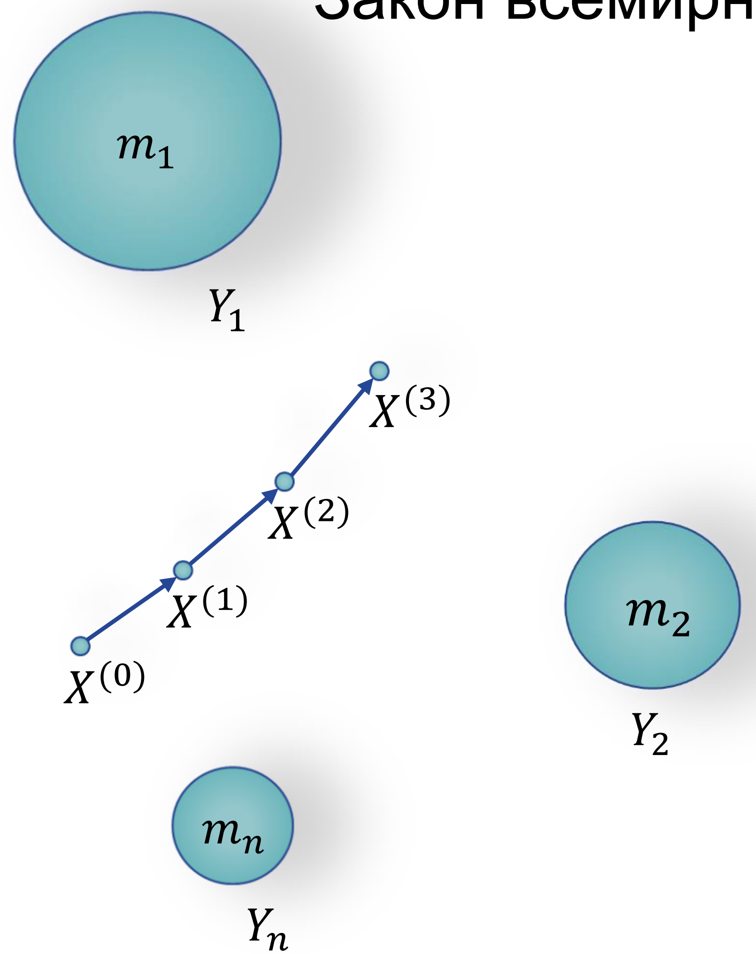


Гравитационная задача



Материальная точка X с малой массой m_x движется с начальной скоростью V среди n неподвижных тел $Y_1, Y_2, \ldots, Y_n$ с большими массами $m_1, m_2, \ldots, m_n$.

Необходимо рассчитать траекторию точки X с шагом Δt



Закон всемирного тяготения: $F_i = G \frac{m_i m_x}{\|Y_i - X\|^3}(Y_i - X)$

Второй закон Ньютона: $A_i = \frac{F_i}{m_x}$

$$\alpha_X(Y_i) = G \frac{m_i}{\|Y_i - X\|^3}(Y_i - X)$$

$$A_i = \alpha_X(Y_i)$$

$$A = \sum_{i=1}^{n} A_i$$

$$V^{(k+1)} = V^{(k)} + A \cdot \Delta t$$

$$X^{(k+1)} = X^{(k)} + V^{(k+1)} \cdot \Delta t$$

Алгоритм решения гравитационной задачи по шаблону BSF

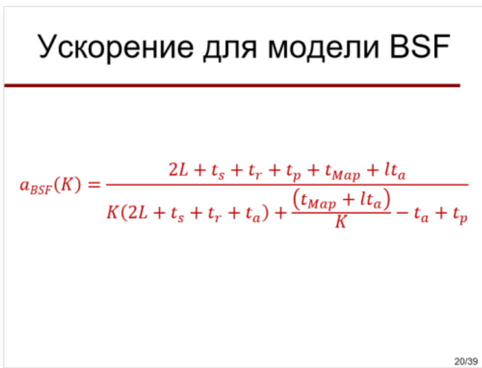
(T – конечный момент времени, Δt - шаг по времени)

1. $t \coloneqq t_0; X^{(0)} \coloneqq 0; V^{(0)} \coloneqq 0;\ Input([(Y_1, m_1), \dots, (Y_n, m_n)])$
2. $[A_1, \dots, A_n] \coloneqq Map(\alpha_X, [(Y_1, m_1), \dots, (Y_n, m_n)])$
3. $A \coloneqq Reduce(+, [A_1, \dots, A_n])$
4. $V^{(t+\Delta t)} \coloneqq V^{(t)} + A \cdot \Delta t$

   $X^{(t+\Delta t)} \coloneqq X^{(t)} + V^{(t+\Delta t)} \Delta t$
5. Если $t \geq T$, перейти на шаг 7
6. $t \coloneqq t + \Delta t$; перейти на шаг 2
7. Стоп

BSF-оценки для алгоритма Gravitation-BSF

τ_{op} - время выполнения одной операции с плавающей точкой
τ_{tr} - время пересылки вещественного числа (без учета латентности)
n - количество больших тел
L - латентность



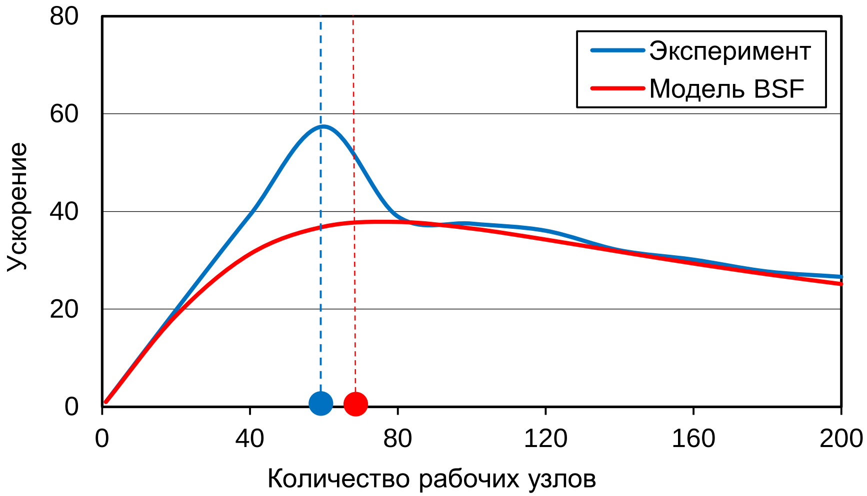
$$\Rightarrow a_{BSF}(K) = \frac{2L + 6\tau_{tr} + 14\tau_{op} + 20n\tau_{op} + 3l\tau_{op}}{K\left(2L + 96 + 14\tau_{op}\right) + \frac{20n\tau_{op} + 3l\tau_{op}}{K} + 11\tau_{op}}$$

$$\Rightarrow K_{Max} = \sqrt{\frac{20n\tau_{op} + 3l\tau_{op}}{2L + 6\tau_{tr} + 14\tau_{op}}}$$

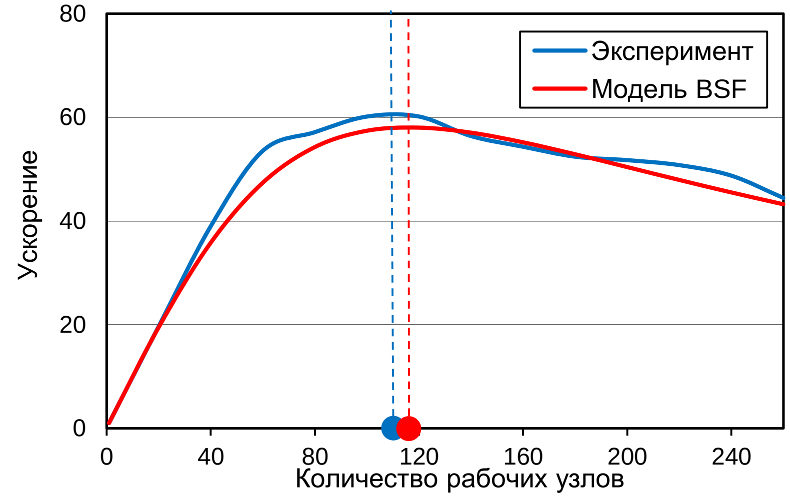
$$\boldsymbol{K_{Gravitation}^{Max} = O\left(\sqrt{n}\right)}$$

Ускорение гравитационного алгоритма: теория и практика

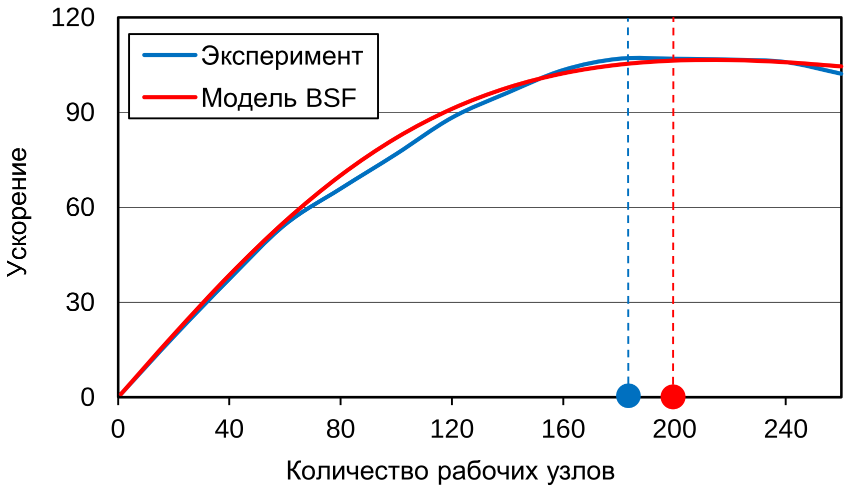
$n = 450$



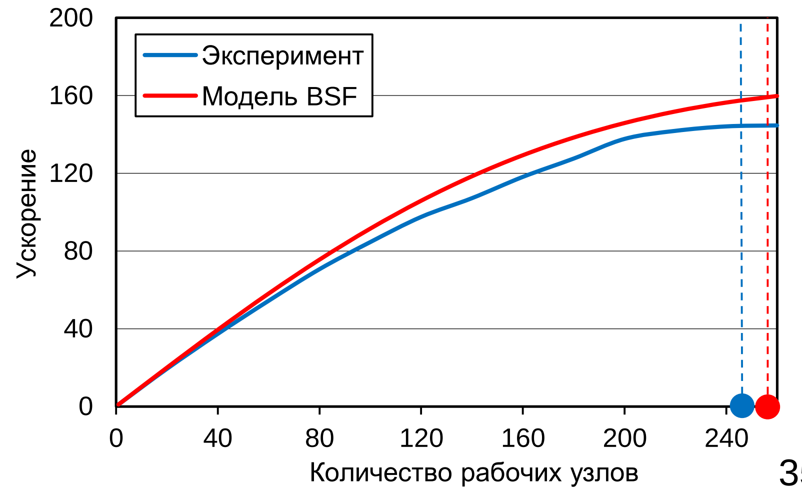
$n = 600$



$n = 900$



$n = 1200$



Публикации

Статьи в журналах из перечня ВАК

1. Ежова, Н.А. Исследование масштабируемости итерационных алгоритмов при суперкомпьютерном моделировании физических процессов / Н.А. Ежова, Л.Б. Соколинский // Вычислительные методы и программирование: новые вычислительные технологии. –2018. –Т. 19, № 4. –С. 416–430.
2. Ежова, Н.А. Модель параллельных вычислений для многопроцессорных систем с распределенной памятью / Н.А. Ежова, Л.Б. Соколинский // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. -2018. –Т. 7, № 2. –С. 32-49.
3. Ежова, Н.А. Модель параллельных вычислений BSF-MR / Н.А. Ежова, Л.Б. Соколинский // Системы управления и информационные технологии. –2019. № 3(77). –С. 15–21.
4. Ежова, Н.А. Обзор моделей параллельных вычислений / Н.А. Ежова, Л.Б. Соколинский // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. –2019. –Т. 8, № 3. –С. 58-91.
5. Ежова, Н.А. Программная поддержка модели BSF / Н.А. Ежова, Л.Б. Соколинский // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. –2019. –Т. 8, № 4.

Публикации

Статьи в изданиях, индексируемых в SCOPUS

6. Ezhova, N.A. Scalability Evaluation of Iterative Algorithms Used for Supercomputer Simulation of Physical processes / N.A. Ezhova, L.B. Sokolinsky // Proceedings - 2018 Global Smart Industry Conference, GloSIC 2018. –IEEE, 2018. –10 p. DOI: 10.1109/GloSIC.2018.8570131.
7. Ezhova, N. Verification of BSF Parallel Computational Model / N. Ezhova // Proceedings of the 3rd Ural Workshop on Parallel, Distributed, and Cloud Computing for Young Scientists. CEUR Workshop Proceedings. -2017. -Vol. 1990. P. 30–39. URL: http://ceur-ws.org/Vol-1990/paper-04.pdf.

Основные результаты, выносимые на защиту

1. Разработана модель параллельных вычислений BSF
2. Разработан компилируемый BSF-каркас на языке C++ с использованием библиотек параллельного программирования MPI и OpenMP
3. Выполнены проектирование и реализация визуального конструктора программ BSF-Studio
4. С использованием BSF-каркаса созданы BSF-программы для известных численных итерационных алгоритмов

Спасибо за внимание!

Вопросы?