ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ, НГУ)
Факультет информационных технологий
Кафедра параллельных вычислений
09.03.01 Информатика и вычислительная техника. Программная инженерия и компьютерные науки

РАЗРАБОТКА И РЕАЛИЗАЦИЯ АЛГОРИТМА ДИНАМИЧЕСКОЙ АДАПТАЦИИ КОЛИЧЕСТВА РАБОЧИХ ПОТОКОВ В СИСТЕМЕ LUNA

Выполнил: Валитов Андрей Александрович, студент группы 19205 ФИТ НГУ
Научный руководитель: Власенко Андрей Юрьевич, к.т.н., доцент каф. ПВ ФИТ НГУ
Соруководитель: Перепёлкин Владислав Александрович, ст. преп. каф. ПВ ФИТ НГУ
Рецензент: Чесноков Вадим Владимирович, рук. отд. мобильных технологий, ООО “ТруИнжиниринг”

23.06.2023

Новосибирск 2023

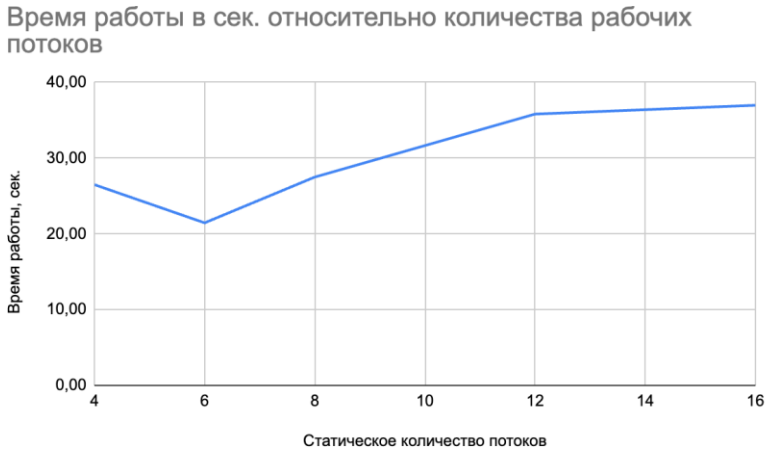
Предметная область

Система автоматического конструирования параллельных программ - система, которая строит параллельную программу автоматически по её описанию на относительно высоком уровне абстракции.

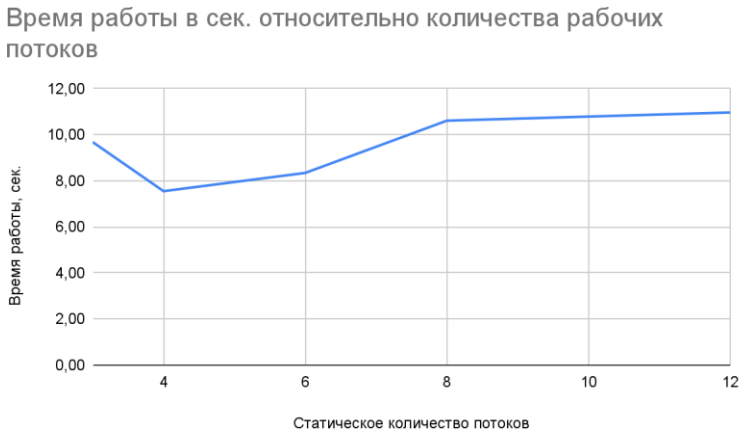
Эффективность таких систем, в частности, зависит от эффективности исполнительной системы и ее внутренних алгоритмов. К ним относится организация рабочих потоков, которую можно разделить на:

- **Статический выбор количества потоков** - задание количества рабочих потоков на этапе подготовки к исполнению программы.
- **Динамический выбор количества потоков** - задание количества рабочих потоков во время исполнения программы на основе анализа хода исполнения.

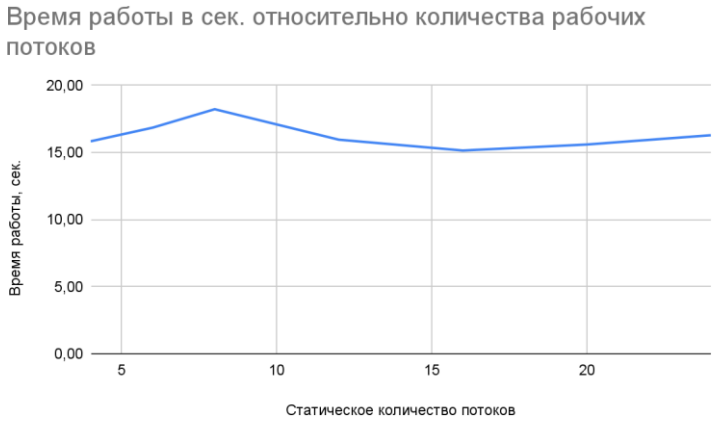
Блочное умножение матриц



Решение одномерного эллиптического уравнения методом IADE



Трёхмерное моделирование самогравитирующего пылевого облака методом частиц-в-ячейках



Актуальность

С точки зрения оптимизации исполнения программ исполнение с оптимальным или близким к нему количеством рабочих потоков может привести к уменьшению общего времени работы программы. Зачастую именно данная характеристика является определяющей для конечного пользователя. Характер загрузки программы в процессе исполнения меняется динамически, поэтому актуальна разработка алгоритма, который позволит динамически находить оптимальное количество потоков.

Цель и задачи работы

Цель работы – разработка и реализация алгоритма динамической адаптации количества рабочих потоков в системе LuNA.

Для достижения цели были поставлены следующие **задачи**:

1. Проанализировать особенности проблемы динамической адаптации количества рабочих потоков в системе LuNA;
2. Сформировать требования к разрабатываемому алгоритму;
3. Разработать алгоритм, удовлетворяющий требованиям из п. 2;
4. Реализовать разработанный алгоритм и встроить его в систему LuNA;
5. Провести экспериментальное исследование времени выполнения практически значимых вычислительных задач при активированном алгоритме.

Научная новизна и практическая значимость

Научная новизна работы состоит в том, что был разработан и реализован алгоритм, который позволил динамически адаптировать количество рабочих потоков.

Практическая ценность работы заключается в том, что с помощью реализованного алгоритма было достигнуто ускорение работы практически значимых LuNA программ по сравнению с их выполнением со статическим количеством потоков.

Обзор родственных работ

Системы автоматического конструирования параллельных программ и средства параллельного программирования	Описание системы в соответствии с критериями обзора
Charm++	Статическое количество рабочих потоков, равное числу ядер на узле. Рабочие потоки привязываются к ядрам. Есть балансировщик нагрузки на рабочие потоки.
PaRSEC	Статическое количество рабочих потоков, задаваемое во время компиляции. Рабочие потоки привязываются к ядрам.
Legion	Рабочие потоки создаются в соответствии с описанием отображения программы на вычислительные ресурсы узла.
OpenMP	Выбор количества потоков и правило отображения их на ядра определяются реализацией компилятора.
LuNA	Статическое количество потоков, задаваемое по умолчанию системой, либо пользователем через опцию при запуске программы. Нет механизма контроля отображения потоков на ядра узла.

LuNA и ее исполнительная система

LuNA – система на основе задач. Задачи формируются **фрагментами вычислений** (ФВ) – независимыми единицами программы, содержащие описание входных/выходных фрагментов данных и **фрагментов кода** (ФК).

Задачи попадают в общую очередь задач пула потоков исполнительной системы. Их исполняют рабочие потоки. Результат выполнения задачи – изменение состояния исполнительной системы.

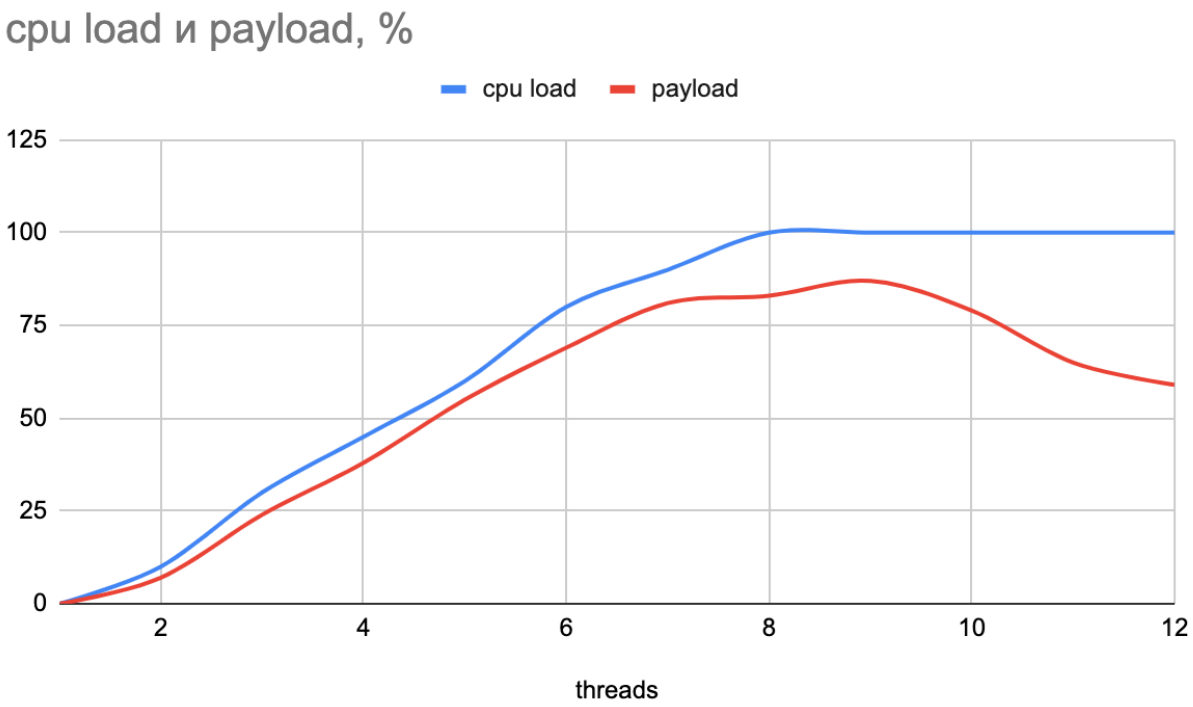
Система LuNA позволяет также внедрить снятие различных нефункциональных характеристик с выполненных фрагментов кода и использовать их далее в других модулях.

Постановка задачи

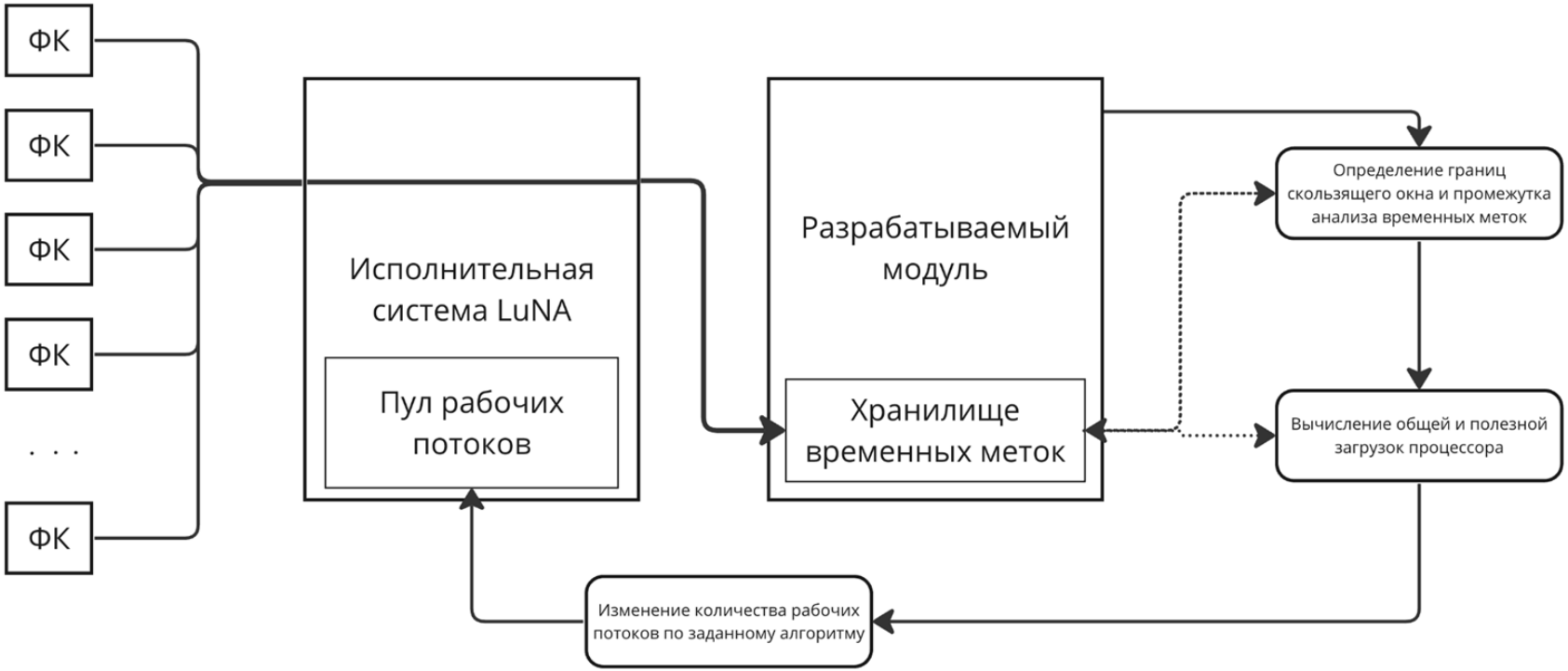
Оптимальное количество потоков на отрезках времени установить в общем случае сложно. Однако, можно рассматривать функцию приближенного оптимального количества потоков g(T) и использовать прогностический подход для поиска нового значения на следующем отрезке времени, используя информацию с уже пройденных отрезков.

Таким образом, задача адаптации количества рабочих потоков в реальном времени сводится к вычислению функции g(T) на основе прогностического подхода на каждом отрезке разбиения.

Разработка. Идея алгоритма



Разработка. Общая схема алгоритма



Разработка. Общая и полезная загрузка

Общая загрузка процессора с N ядрами за период времени T вычисляется по формуле (1), где P_i – загрузка i-го ядра узла.

$$cpuload_{\mathrm{T}} = \frac{\sum_{i=1}^{N} P_i}{\mathrm{T} \times N} \quad (1)$$

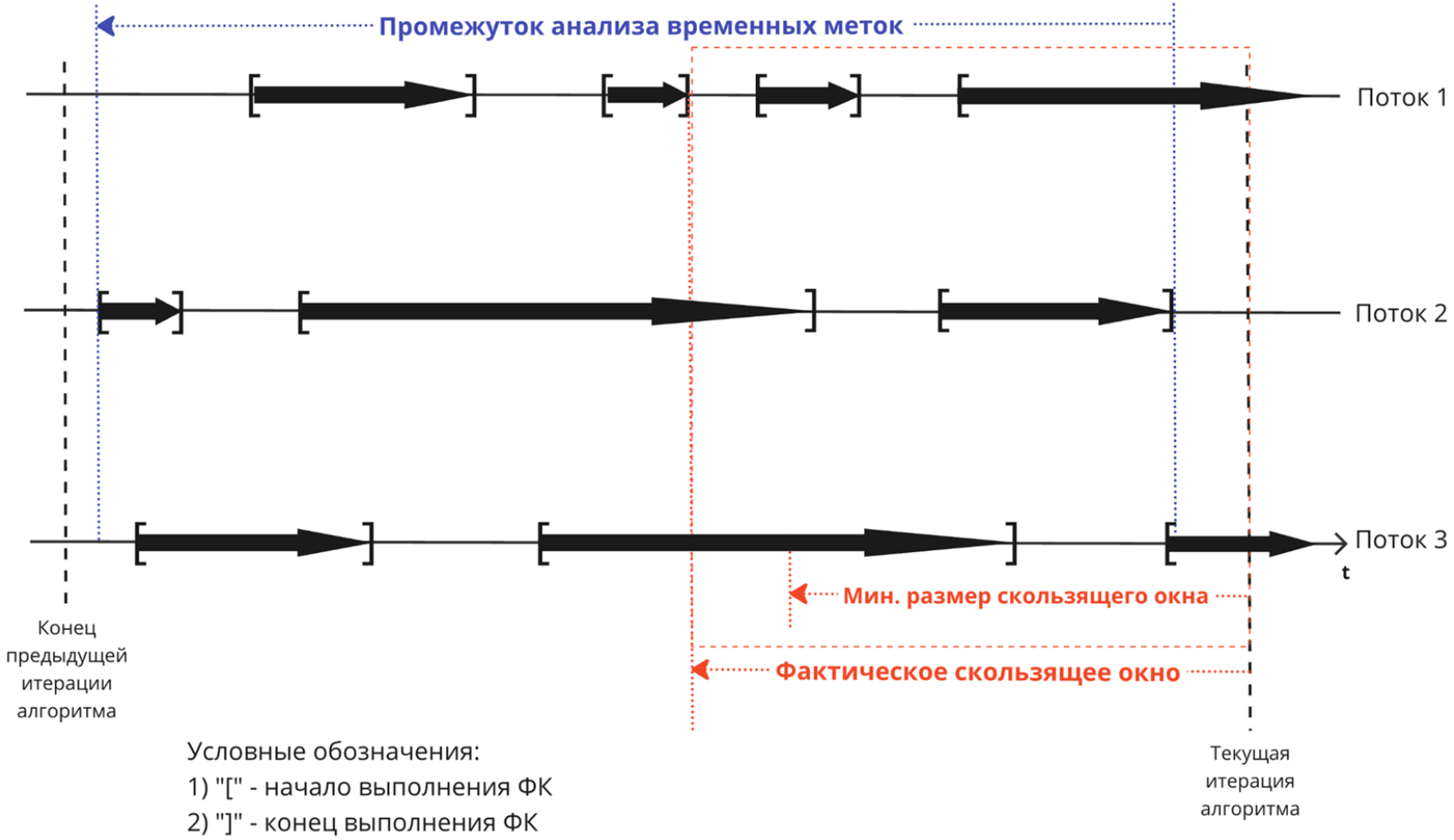
Полезная загрузка i-го ядра за период времени T вычисляется по формуле (2), где $C_{i,j}$ – время, затраченное на исполнение фрагментов кода j-м рабочим потоком i-го ядра узла.

$$payload_{\mathrm{T},\,i} = \frac{\sum_{j} C_{i,j}}{\mathrm{T}} \quad (2)$$

Полезная загрузка процессора с N ядрами за период времени T вычисляется по формуле (3).

$$payload_{\mathrm{T}} = \frac{\sum_{i=1}^{N} payload_{\mathrm{T},\,i}}{N} \quad (3)$$

Разработка. Определение окна обновления



Реализация. Особенности реализации

- Применена техника кодогенерации;
- Решены проблемы синхронизации доступа к общим ресурсам;
- Алгоритм оптимизирован по памяти;
- Внедрена возможность логирования работы модуля.

```
time, cpuload, payload, threads, jobs, trend, status
1680872471.82, 58.248338, 45.537552, 5, 20396, 2, Growth
1680872479.82, 66.586555, 53.930929, 7, 18831, 3, Growth
1680872487.82, 77.609286, 65.004397, 10, 17394, 4, Growth
1680872495.82, 85.766096, 74.905049, 14, 16137, 5, Growth
1680872503.82, 91.142514, 82.068153, 19, 14983, 6, Growth
1680872511.82, 88.295997, 80.854370, 18, 13980, -2, Single fall
1680872519.82, 82.043321, 75.362502, 19, 13057, 2, Single growth
1680872527.82, 78.157566, 71.484704, 18, 11955, -2, Single fall
1680872535.82, 75.571546, 68.618821, 19, 10937, 2, Single growth
1680872543.82, 75.771230, 69.044296, 19, 9933, 2, Single growth
1680872551.82, 75.759276, 69.026246, 19, 8880, 2, Single growth
1680872559.82, 74.443303, 67.633185, 18, 0, -2, Single fall
```

Пример работы алгоритма из логов

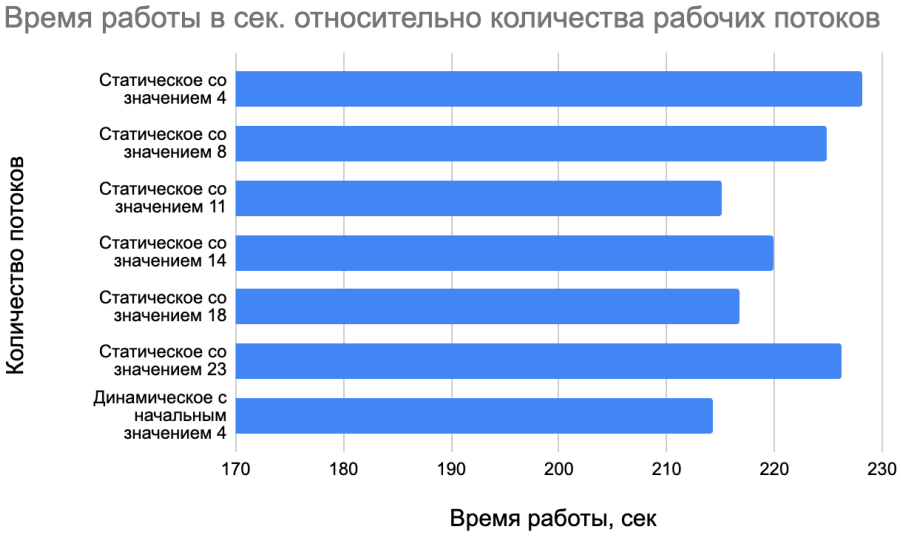
Тестирование алгоритма

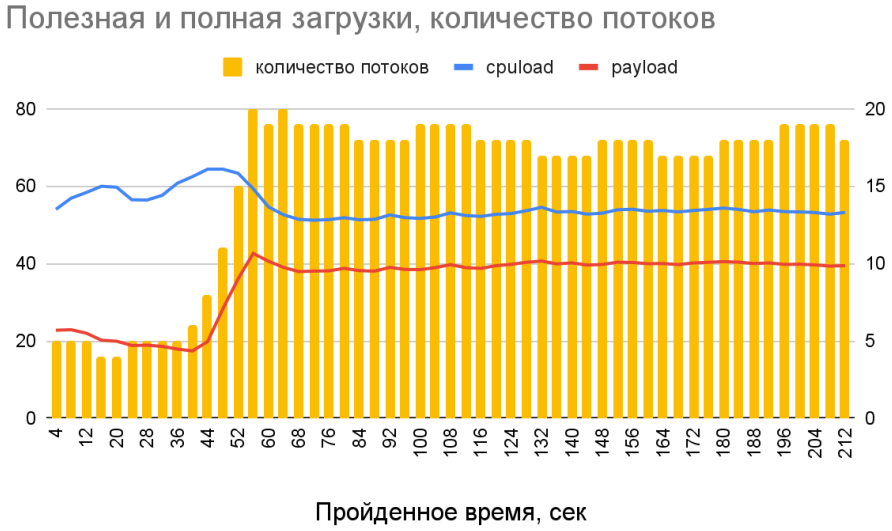
Тестирование алгоритма динамической адаптации количества рабочих потоков проводилось на кластере ИВЦ НГУ (запрашивалось 2 узла по 12 ядер на каждом) и на одиночном узле (8 ядер).

Для проверки эффективности работы алгоритма были проведены тесты на трех вычислительных LuNA программах, представляющих практически значимый интерес.

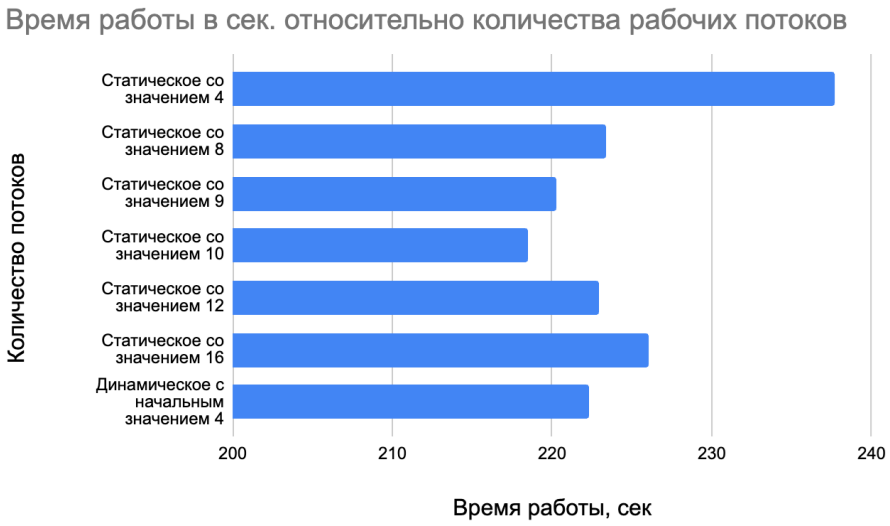
Цель тестирования – проанализировать влияние работы алгоритма на общее время выполнения LuNA программы.

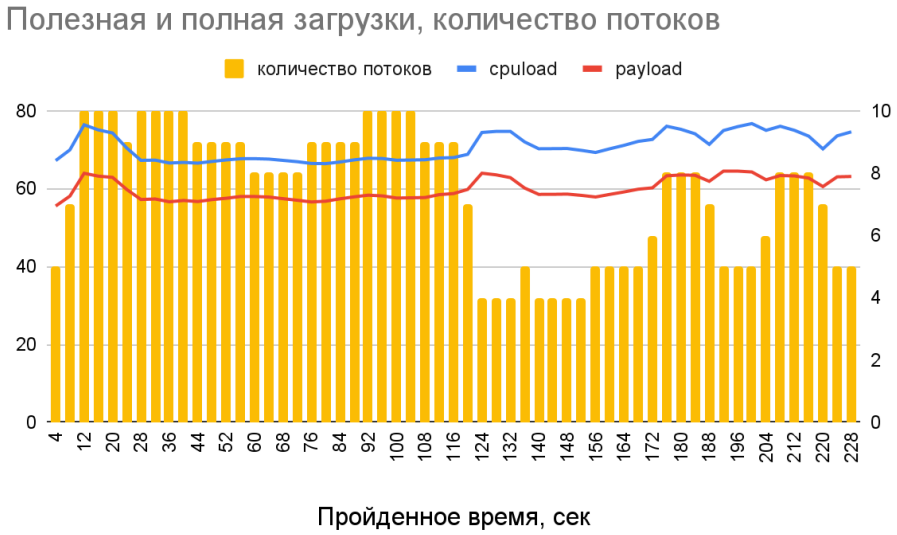
Тестирование. Решатель модельного трехмерного уравнения теплопроводности в единичном кубе



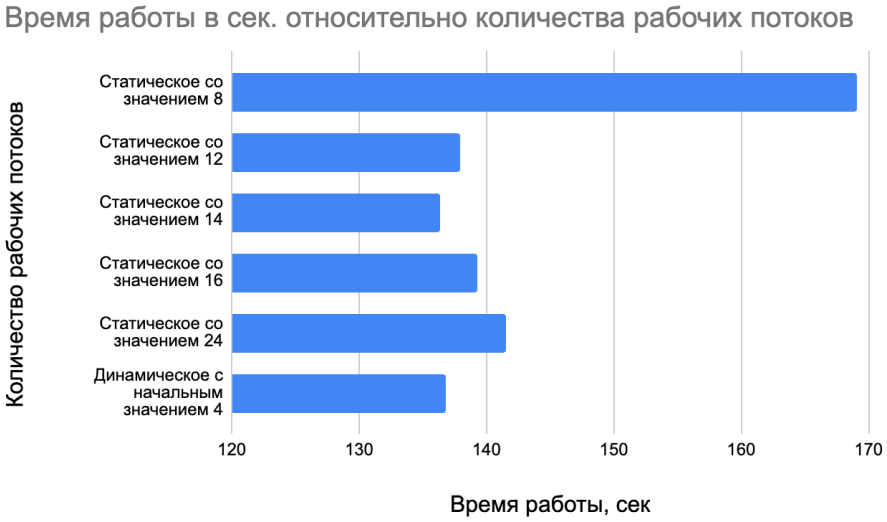


Тестирование. Решение уравнения Пуассона итерационным методом



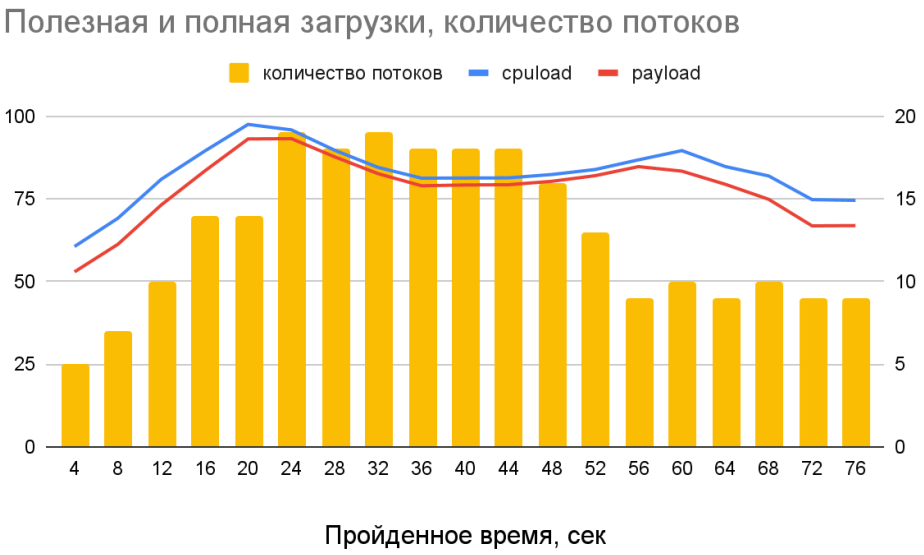


Тестирование. Блочное умножение матриц (кластер ИВЦ НГУ)



Количество рабочих потоков	Время работы, сек	Ускорение
Статическое со значением 8	169,07	0,816
Статическое со значением 12	137,94	1,000
Статическое со значением 14	136,4	1,011
Статическое со значением 16	139,28	0,990
Статическое со значением 24	141,55	0,974
Динамическое с начальным значением 4	136,79	1,008

Тестирование. Блочное умножение матриц



Апробация работы

Данная работа была опубликована на 61-й Международной научно-студенческой конференции, г. Новосибирск, 2023 г.

Публикация по теме:

1. Валитов, А. А. Разработка и реализация алгоритма динамической адаптации количества рабочих потоков в системе LuNA / А. А. Валитов // Информационные технологии : Материалы 61-й Междунар. науч. студ. конф. 17–26 апреля 2023 г. / Новосиб. гос. ун-т. — Новосибирск : ИПЦ НГУ, 2023 (в печати).

Заключение

Был разработан и реализован алгоритм динамической адаптации количества рабочих потоков на основе введенных характеристик полезной и общей загрузок процессора. С помощью алгоритма удалось ускорить LuNA программы, решающие практически значимые вычислительные задачи.

Защищаемые положения:

1. разработан алгоритм динамической адаптации количества рабочих потоков;
2. разработанный алгоритм был реализован и встроен в систему фрагментированного программирования LuNA в качестве модуля исполнительной системы LuNA;
3. проведено экспериментальное исследование работоспособности и эффективности реализованного алгоритма.

Дальнейшие планы

1. Усовершенствование алгоритмов определения количества рабочих потоков. В частности, использование схемы предиктор-корректор;
2. Расширение модуля динамической адаптации количества потоков с целью учета всех видов потоков исполнения (сетевых, возможных новых системных) в системе LuNA;
3. Интеграция алгоритма динамической адаптации количества потоков с алгоритмами балансировки нагрузки с целью возможности адаптации не только в рамках одного узла, но и всего мультикомпьютера.

Спасибо за внимание!

Характеристика задачи блочного умножения матриц

Для блочного умножения матриц характерно большое количество однотипных операций умножения и равномерное изменение количества задачи в очереди. Исполнительная система генерирует для каждых подматриц свои задачи в очереди, далее по мере возможностей рабочие потоки выполняют их.

Для задач с равномерным изменением количества задач в очереди ожидается, что алгоритм динамической адаптации количества рабочих потоков сможет относительно близко приблизиться к оптимальному количеству потоков, но при этом не даст большого прироста по сравнению со статическим количеством потоком, равному количеству ядер узла. Для таких задач динамичность изменений оказывается менее эффективна в силу отсутствия большой необходимости в адаптации количества потоков.

Характеристика итерационной задачи решения уравнения Пуассона

Для данной задачи свойственно малое количество задач в очереди почти на всем протяжении времени работы. Сначала формируется большое количество задач, связанных с инициализацией данных, которые выполняются в течение нескольких секунд. После большая часть вычислений выполняется в рамках нескольких задач. Очередь задач довольно часто остается пустой, поскольку рабочие потоки успевают выполнять задачи быстрее, чем они формируются в очереди.

Для задач с малыми размерами очередей ожидается, что алгоритм динамической адаптации количества рабочих потоков подстроится под частоту появления новых задач в очереди.

Характеристика решателя модельного трехмерного уравнения теплопроводности в единичном кубе

Для решателя модельного трехмерного уравнения теплопроводности в единичном кубе характерно неравномерное изменение размера очереди задач в ходе работы программы. Агенты порождают новые задачи порциями, поэтому в разные моменты времени размер очереди бывает то небольшим (порядка 10-100 задач), то относительно большим (1000-5000 задач). Ожидается, что именно для таких задач динамическая адаптация количества рабочих потоков может показать себя с лучшей стороны – при достаточно высокой частоте обновления (2-6 секунд) количества потоков алгоритм позволяет подстроиться под текущие изменения, быстро находя оптимальные показатели количества потоков в данный промежуток.