ОТЧЕТ О ПРОДЕЛАННОЙ РАБОТЕ С ИСПОЛЬЗОВАНИЕМ ОБОРУДОВАНИЯ ИВЦ НГУ

1.Аннотация

В области системного параллельного программирования актуальной проблемой является автоматическое конструирование эффективной параллельной программы. Под эффективностью понимается время выполнения программы, расход памяти и т.п. Автоматическое конструирование эффективной программы является алгоритмически труднорешаемой задачей в общем случае, но решается в частных случаях. Вектором развития языков и систем автоматического конструирования параллельных программ является накопление частных решений и эвристик. Поэтому необходимо исследование и создание эффективных параллельных программ, реализующих конкретные задачи. Под реализацией задачи понимается разработка программы, решающей данную задачу. Опыт, накопленный при их создании, позволит создать новые методы и алгоритмы автоматического построения эффективных параллельных программ.

2. Тема работы

Эффективная реализация краевой задачи фильтрации двухфазной жидкости в системе LuNA

3. Состав коллектива

Студент: Нуштаев Юрий Юрьевич, Факультет информационных технологий, кафедра параллельных вычислений, 4-й курс, группа 19207

Научный руководитель: Маркова Валентина Петровна, к.т.н., доцент. параллельных вычислений ФИТ НГУ, с.н.с. ИВМиМГ СО РАН

Соруководитель: Перепёлкин Владислав Александрович, ст. преп. каф. параллельных вычислений ФИТ НГУ, н.с. ИВМиМГ СО РАН

4. Научное содержание работы

4.1 Постановка задачи

Цель работы: разработать эффективную LuNA программу задачи фильтрации двухфазной жидкости в трёхмерной области при наличии скважин и проанализировать её нефункциональные свойства.

Достижение поставленной цели требует решения следующих задач:

1) разработать эффективную программу, реализующую краевую задачу фильтрации двухфазной жидкости в системе LuNA;

2) провести сравнение нефункциональных свойств программы LuNA и программы MPI, а именно оценить эффективность;

4.2 Современное состояние проблемы

Автоматизация конструирования параллельных программ позволяет снижать сложность разработки, отладки и модификации параллельных программ

Важно обогащать систему конструирования различными частными системными алгоритмами и эвристиками.

Полезно вручную разрабатывать программы для различных классов задач, чтобы обобщать полученные результаты и включать в систему автоматического конструирования программ эвристики полученные при разработки.

Анализ средств параллельного программирования показал, что нефункциональные свойства программ достигаются только на определённом множестве задач, это множество задач может отличаться от системы к системе.

4.3 Подробное описание работы, включая используемые алгоритмы.

Поведение — это множество реализаций. В 4.3.1 описано поведение фрагментированной программы в 2 этапа, каждый этап уточняет поведение фрагментированной программы.

4.3.1 Описание фрагментированной программы

Опишем множество ФД.

Пусть FG — количество ФД, на которое разбиты вектора, тогда опишем множество ФД. Большинство названий ФД соответствует названиям структур данных из анализа 2.1. Множество G представляет собой множество индексов по пространственной координате, индекс — номер итерации в цикле. Множество F — множество индексов от 0..FG-1, индекс — номер фрагмента вектора. |F|=FG.

Рассмотрим множество всех ФД D = CF ∪ MF ∪ IF, где CF — фрагменты векторов, MF — фрагменты матриц, IF — фрагменты с промежуточными значениями.

Множество фрагментов векторов CF = V1 ∪ V2 ∪ pRes ∪ R ∪ pU ∪ pVb ∪ P ∪ new_pRes, где:

1. $V1 = \{ V1_{i,j} | \forall i \in G, \forall j \in F \}$
2. $V2 = \{ V2_{i,j} | \forall i \in G, \forall j \in F \}$
3. $pRes = \{ pRes_{i,j} | \forall i \in G, \forall j \in F \}$
4. $R = \{ R_{i,j} | \forall i \in G, \forall j \in F \}$
5. $pU = \{ pU_{i,j} | \forall i \in G, \forall j \in F \}$
6. $pVb = \{ pVb_{j} | \forall j \in F \}$
7. $P = \{ P_{i,j} | \forall i \in G, \forall j \in F \}$
8. $new\_pRes = \{ new\_pRes_{i,j} | \forall i \in G, \forall j \in F \}$

Множество фрагментов матриц MF = m_pB ∪ m_pD ∪ m_pBt, где:

1. m_pB
2. m_pD
3. m_pBt

Множество ФД с промежуточными значениями IF = qr ∪ ps ∪ a ∪ minus_a ∪ mult_v_a ∪ k1 ∪ k2 ∪ r ∪ Rnorm ∪ Iters ∪ m_K, где:

1. $qr = \{ qr_i | \forall i \in G \}$
2. $ps = \{ ps_i | \forall i \in G \}$
3. $a = \{ a_i | \forall i \in G \}$
4. $minus\_a = \{ minus\_a_i | \forall i \in G \}$
5. $mult\_v\_a = \{ mult\_v\_a_i | \forall i \in G \}$

6. k1
7. $k2 = \{ k2_i | \forall i \in G\}$
8. $r = \{ r_i | \forall i \in G\}$
9. $Rnorm = \{ Rnorm_i | \forall i \in G\}$
10. Iters
11. m_K

Рассмотрим множество ФВ.

Разобьём данное множество ФВ на три подмножества: в первое подмножество войдут ФВ, соответствующие векторным операциям; во второе — соответствующие матричным операциям; в третье — соответствующие остальным операциям.

ФВ представлены в виде триплетов <in, mod, out>, где in — множество входных ФД, mod —программный модуль, out — множество выходных ФД.

Векторные ФВ:

1. $<\{R_{i+1}\}, Norm_i, \{Rnorm_i\}>$
2. $<\{a_i, P_{i+1}, pU_i, \}, AddKV\_r_i, \{pU_{i+1}\}>$
3. $<\{minus\_a_i, new\_pRes_i, R_i, \}, AddKV\_r_i, \{R_{i+1}\}>$
4. $<\{mult\_v\_a_i, pVb\}, AddKV\_pRes_i, \{pRes_i\}>$
5. $<\{R_i, R_i\}, vector\_mult_i, \{qr_i\}>$
6. $<\{R_i\}, vector\_set\_init_i, \{P_i\}>$
7. $<\{k1, R_i, k2_i, P_i\}, set\_lin_i, \{P_{i+1}\}>$
8. $<\{pVb, P_{i+1}\}, vector\_mult_i, \{tmp\_a_i\}>$
9. $<\{new\_pRes_i, P_{i+1}\}, vector\_mult_i, \{ps_i\}>$

Матричные ФВ:

1. $<\{P_{i+1}, m\_pB\}, mult\_v\_add_i, \{V1_i\}>$
2. $<\{V1_i, m\_pD\}, progonka_i, \{V2_i\}>$
3. $<\{pRes_i, m\_pBt, V2_i\}, t\_mult\_v\_add_i, \{new\_pRes_i\}>$

Остальные ФВ:

1. $<\{qr_i, qr_{i-1}\}, set\_div_i, \{k2_i\}>$
2. $<\{tmp\_a_i, m\_K\}, set\_mult_i, \{mult\_v\_a_i\}>$

3. $<\{qr_i, ps_{i-1}\}, set\_div_i, \{a_i\}>$

4. $<\{a_i\}, set\_minus_i, \{minus\_a_i\}>$

4.3.2 Описание поведения

В данном разделе будет определено поведение путём определения функций отображения, отношения эквивалентности ФД и т.п. Описание поведения будет проведено в два этапа.

Этап 1:

На данном этапе определим функции отображения на узлы для ФД и ФВ, удаление ФД. Все узлы пронумерованы неотрицательными целыми числами по порядку, ввиду этого отображение происходит на это множество номеров.

Опишем функцию отображения векторных ФД на узлы мультикомпьютера. Пусть NP — количество узлов мультикомпьютера, FG — количество ФД, j — индекс ФД. Тогда F(j) = {j * NP / FG} — множество (номер) узлов, на котором находится ФД с индексом j.

Необходимо, чтобы ФВ с индексом j, обрабатывающий ФД с индексом j, выполнялся на узле, где уже есть все необходимые данные, следовательно, функция отображения ФД на узлы мультикомпьютера и функция отображения ФВ на узлы мультикомпьютера имеют одинаковую зависимость от j.

Так как все узлы пронумерованы, то достаточно отображать на целое число, тогда можно задать отображение на множество узлов. Для матричных ФД и ФД с промежуточными значениями опишем функцию как F(j) = {0} — то есть все ФД отображаются на основной (нулевой узел).

Так как ФВ, на выход которым подаётся матричные ФД или ФД с промежуточными значениями, существуют не только на основном узле, то отображение матричных ФД и ФД с промежуточными значениями на основной узел — это ещё не оптимальное решение. Оптимальное решение данной проблемы будет описано на этапе 2.

Для определения поведения в полной мере необходимо задать их удаление. В основном это относится к ФД, подвергающимся изменениям, например, к векторным ФД и ФД с промежуточным значениями. Зададим

следующие поведение для таких ФД: у нас есть информация, что определённый ФД будет потреблен N число раз, тогда будем считать, что после N потреблений данный ФД будет удалён из памяти.

Этап 2:

На данном этапе уточним поведение, задав репликацию некоторых ФД на множество узлов и отношение эквивалентности ФД.

Для матричных ФД и ФД с промежуточными значениями функция отображения F(I) = {0}; для работы ФВ необходимо, чтобы эти ФД были доступны на каждом узле. Используем другую функцию отображения. Проведём репликацию для каждого ФД R {0} → {0..NP-1}, данная функция производит репликацию для каждого матричного ФД и ФД с промежуточными значениями на всё множество доступных узлов.

Важно переиспользовать память, а именно чтобы все ФД из D, имеющие пространственную координату i — номер итерации цикла, отображались на один и тот же буфер в памяти. Для этого заведём отношение эквивалентности на множестве ФД. Два ФД эквивалентны, если они имеют один и тот же буфер в памяти. Например, фрагмент $R_{1,j} \equiv R_{2,j}$, то есть эти два ФД эквивалентны.

4.4 Полученные результаты

Тестирование итоговой реализации, описанной в разделе 4.3. Было проведено тестирование на двух кластерах: НГУ и МВС. Причём на МВС тестирование проводилась на двух коммуникационных подсистемах: Ethernet и InfiniBand.

Для тестирования было выбрано оптимальное количество потоков для каждого кластера и каждой реализации. Выбор оптимального количества потоков осуществлялся с помощью запуска программы с изменением количества потоков от 1 до количества потоков, при котором результаты по времени становились хуже по отношению к предыдущему запуску программы. Было взято количество потоков, при котором программа отрабатывала за наименьшее время. Результаты представлены в таблице 1.

Таблица 1 — Оптимальное количество потоков на разном количестве узлов

Количество потоков					
	Кластер НГУ		Кластер МВС		
Кол—во узлов	MPI	LuNA	MPI	LuNA	LuNA IB
1	8	12	8	18	21
2	4	15	8	18	21
4	4	15	8	18	21
6	4	15	—	—	—
8	—	—	8	18	21
12	—	—	8	18	21
16	—	—	8	18	21

4.4.1 Тестирование на кластере НГУ

Результат представлен на рисунке 1.

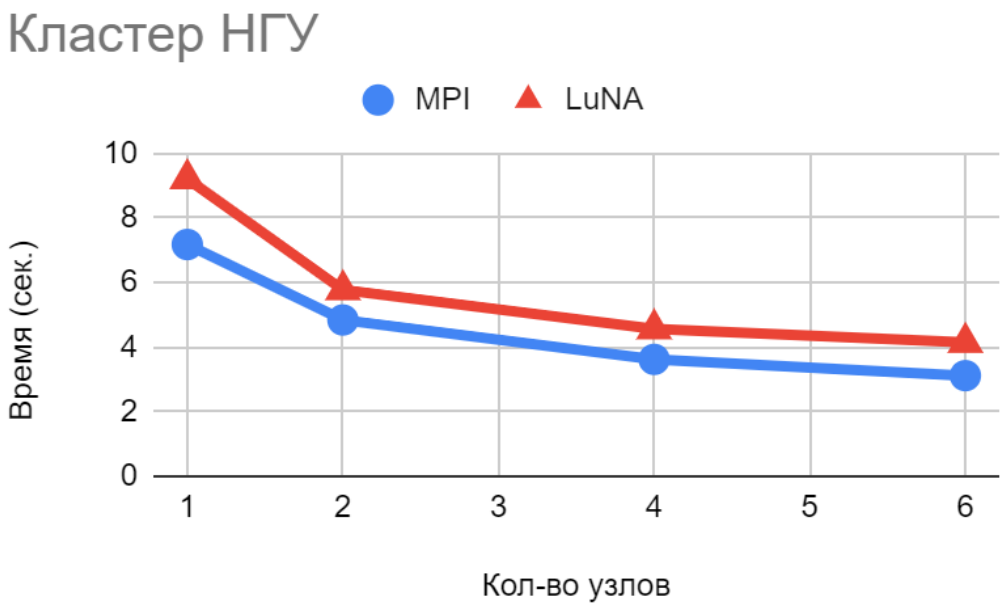

Рисунок 1 — Тестирование на кластере НГУ

На графике представлены результаты тестирования реализации алгоритма фильтрации в системе LuNA. По вертикали время откладывается в секундах, по горизонтали — количество узлов. По графику можно сделать вывод, что отставание реализации LuNA можно оценить в 30% от MPI реализации, что оценивается как удовлетворительный результат.

5. Эффект от использования кластера в достижении целей работы.

Использование кластера является необходимым. Система LuNA является системой предназначенной для мультикомпьютеров в том числе и типа “СуперЭВМ”. Программы решающие задачи такого рода, обычно выполняются на кластерах.

6. Перечень публикаций, содержащих результаты работы

1. Нуштаев Ю. Ю. Эффективная реализация краевой задачи фильтрации двухфазной жидкости в системе LuNA Материалы 61-й Междунар. науч. студ. конф. 17-26 апреля 2023 г. / Новосиб. гос. ун-т. — Новосибирск: ИПЦ НГУ, 2023

2. Кудрявцев А.А., Малышкин В.Э., Нуштаев Ю.Ю., Перепёлкин В.А., Спирин В.А. Эффективная фрагментированная реализация краевой задачи фильтрации двухфазной жидкости // журнал "Проблемы информатики", 2023, №2