МЕТОД РАСПРЕДЕЛЕННОГО РЕШЕНИЯ ВЫЧИСЛИТЕЛЬНО СЛОЖНЫХ ОПТИМИЗАЦИОННЫХ ЗАДАЧ С ИСПОЛЬЗОВАНИЕМ ГЕТЕРОГЕННОГО КОЛЛЕКТИВА ВЫЧИСЛИТЕЛЬНЫХ УСТРОЙСТВ

Клименко А.Б.,
Институт информационных наук и технологий безопасности
Российского государственного гуманитарного университета, Москва, Россия
Anna_klimenko@mail.ru
Баринов А.А.
Институт проблем управления им. В.А. Трапезникова РАН, Москва, Россия
barseniyy@yandex.ru

Аннотация. В данной статье представлен метод распределенного решения вычислительно сложных оптимизационных задач с использованием гетерогенного коллектива вычислительных устройств. Предлагается использование принципов параллельных независимых запусков метаэвристических алгоритмов в совокупности с подбором объема блока вычислений в соответствии с критериями времени и размера блока.

Ключевые слова: оптимизация, распределенные вычисления, метаэвристики, управление вычислениями, оптимизация времени решения задач.

Введение

В настоящее время одним из перспективных направлений распределенных вычислений является организация решения вычислительно сложных задач, в том числе, задач оптимизации, в коллективах вычислителей краевого слоя сети. В качестве основных тому причин можно назвать необходимость локализации обработки данных в непосредственной близости от их источников, необходимость снижения времени решения многих оптимизационных задач (например, задач машинного обучения на краю сети), а также необходимость разгрузки сетевой инфраструктуры. В некоторых работах также указывают на требования к безопасности обработки данных, как на причину перехода к вычислениям на краю сети.

В связи с этим ведутся интенсивные исследования возможностей решения различных задач оптимизации на краевых устройствах, включая также краевые сервера.

Предметные области, в рамках которых решаются таким образом задачи оптимизации, достаточно разнообразны, но в рамках данного исследования можно выделить два следующих класса:

- решение задач, относящихся непосредственно к организации функционирования края сети и туманного слоя, управление ресурсами сети [1–3];
- решения прикладных/пользовательских задач на устройствах [4–6].

Проведенный анализ публикаций, представленных в открытой печати, демонстрирует достаточно глубокую проработку решения задач первого класса, однако, без детальной проработки возможностей распределенного их решения. В целом, исследования акцентируют снимание на формировании интегральных целевых функций, на выборе критериев оптимизации и на формировании параметров метаэвристических алгоритмов для решения частных случаев оптимизационных задач со многими ограничениями.

Второй класс – решение пользовательских оптимизационных задач на краевых устройствах – в опубликованных результатах исследований включает вопросы сокращения времени решения задач путем распараллеливания оптимизационных алгоритмов, например, в [7] представлены методы организации распределенного решения задачи на основе островной модели, а в [8] представлен метод распределенного решения задачи машинного обучения в гетерогенной среде.

Последняя работа [8] представляет особенный интерес, поскольку в ней внимание акцентируется на минимизации времени обучения нейронной сети в условиях распределения фрагментов задачи обучения по гетерогенному сообществу краевых вычислительных устройств. В данном исследовании с целью минимизации времени обучения сформирована задача составления расписаний машинного обучения для каждого устройства при соблюдении ограничений на скорость передачи данных. Управляемым параметром является выделяемый ресурс канала передачи данных, при фиксированных трудоемкостях фрагментов машинного обучения, которые определяются структурой нейронной сети и, таким образом, задача сводится к формированию распределения выполняемого кода для каждого раунда обучения. Предложенный метод предполагает распределенное решение задачи машинного

обучения в режиме «master-slave» где распределению подлежит каждый раунд решения задачи обучения нейросети, что, в свою очередь, предполагает частый обмен данными между устройствами.

Целью данного исследования является разработка метода распределенного решения сложных оптимизационных задач с использованием гетерогенного коллектива вычислительных устройств, ограниченных в плане вычислительной мощности. Отличием от аналогов является использование параллельных независимых запусков метаэвристик, что снижает коммуникационные издержки и позволяет пренебречь параметрами каналов передачи данных, а также двухкритериальная постановка задачи распределения вычислений задачи оптимизации, критериями которой являются время выполнения расчетов (makespan) и вычислительная сложность блоков расчетов. Кроме того, для решения непосредственно задачи распределения решения задачи оптимизации предлагается использование метода сокращения ресурсопотребления вычислительных ресурсов, более подробно описанного в [9].

1. Формализация задачи распределения параллельных независимых запусков метаэвристик для решения вычислительно сложных задач оптимизации

Будем полагать, что пользовательская задача оптимизации решается одним из метаэвристических методов, т.е. является итерационной и стохастической процедурой. Также будем оценивать вычислительную сложность решения задачи оптимизации количеством вызовов целевой функций, полагая целевую функцию неизменной на протяжении вычислений, при этом n вызовов целевой функции формируют блок, трудоемкость которого и прикрепление к устройству должны быть определены в процессе формирования и распределения блоков по устройствам.

Реализация распределенного решения пользовательской оптимизационной задачи как совокупности параллельных независимых запусков имеет следующее обоснование:

- параллельные независимые запуски метаэвристических алгоритмов, производимые на одном и том же поисковом пространстве, с одной и той же целевой функций, но с произвольно назначенными начальными решениями и на ограниченном числе вызовов целевой функции позволяют получить результаты различного качества;
- таким образом, ограничивая время выполнения блоков метаэвристик, за счет стохастического характера выполнения алгоритмов и за счет увеличения количества параллельных независимых запусков имеется возможность сокращения времени выполнения с приемлемыми потерями результата;
- при этом преимуществом является отсутствие необходимости частых информационных обменов, что производится при распараллеливании метаэвристик по целевой функции внутри итерации, например.

Таким образом, будем далее рассматривать блок вычислений метаэвристического алгоритма как некоторое количество вызовов целевых функций, подлежащих распределению на вычислительное устройство. Однако, определение размеров таких блоков для имеющихся в наличии вычислительных ресурсов является также нетривиальной задачей по следующей причине: качество получаемого решения растет с количеством затраченного для решения времени. Следовательно, с одной стороны, требуется минимизировать время решения задачи оптимизации в гетерогенном сообществе устройств, с другой стороны, необходимо максимизировать размеры блоков вычислений с целью повышения качества решения.

Пусть имеется набор блоков вычислений некоторой метаэвристики $G = \{g_i\}$, где g_i – априори неизвестное количество вызовов ЦФ блока i. Мощность множества G соответствует количеству устройств, на которые предполагается распределение блоков.

Также пусть имеется множество вычислительных устройств, которые характеризуются значениями производительности: $M = \{m_j\}$. Поскольку мы предполагаем обмен данными в незначительных объемах, например, описание исходных данных для задачи для инициализации оптимизационного алгоритма и результат, а кроме того, считаем коммуникационную среду полносвязной, то накладными расходами на пересылку данных можно пренебречь. Также будем полагать, что задачу формирования и распределения блоков решает одно из устройств, наиболее мощное.

Поскольку вычислительная среда предполагается гетерогенной, следующие ограничения и частные критерии следует принять во внимание:

- Частные критерии оптимизации, свойственные отдельным устройствам (например, ограничение на загруженность, на энергопотребление и т.д. $P_0 = \{p_l\}$,
- Частные и общие ограничения $constr = \{constr_k\}$,

- Вычислительную сложность формирования вычислительных блоков метаэвристик gr.

Решением задачи является кортеж, включающий вектор A следующего вида:

$$A = (a_1, a_2, \dots a_{|G|}), \quad (1)$$

где a_i – объем блока, назначенного на i-е устройство, и g_r:

$$C = < A, g_r >. \quad (2)$$

Критериями оптимизации являются следующие:

$$T = \min_{A, g_r} T_{max}; \quad (3)$$

$$S = max_{A, g_r} \min(g_i). \quad (4)$$

Критерии (3,4) составляют векторную целевую функцию:

$$F(A, g_r) = (T, S). \quad (5)$$

Решение сформулированной задачи направлено на формирование и распределение блоков вычислений метаэвристики, такое, что будет выбрано одно из недоминируемых решений задачи двухкритериальной оптимизации.

2. Разработка метода распределенного решения вычислительно сложных оптимизационных задач с использованием гетерогенного коллектива вычислительных устройств

Сформулированная таким образом задача сама по себе является *NP*-сложной задачей оптимизации и, следовательно, ее решение может быть ресурсозатратным: если процесс формирования и распределения блоков по устройствам будет занимать долгое время с целью получения лучшего распределения, то в итоге преимущества, полученные при распределенном решении задачи оптимизации, будут утрачены. Следовательно, необходим подбор такого значения g_r, чтобы за меньшее время получить лучшее по качеству распределение, иными словами, задача снова сводится к решению двухкритериальной задачи (5), которая также является вычислительно сложной.

Поэтому, выбор значения g_r будем осуществлять следующим образом: выберем допустимый в контексте решаемой задачи интервал вызовов ЦФ, и для него выберем такой метаэвристический алгоритм решения задачи (5), который на заданном отрезке дает наилучший результат. Выбор эффективного алгоритма решения задачи реализуется посредством поиска в заранее подготовленной базе данных эффективных алгоритмов оптимизации [9].

Метод распределенного решения вычислительно сложных задач оптимизации с использованием гетерогенного коллектива вычислительных устройств состоит в последовательности следующих этапов:

- Основываясь на допустимом времени для формирования блоков метаэвристик и распределения их по устройствам, из БД эффективных алгоритмов устройство-лидер выбирает тот алгоритм, который на принятом интервале времени дает наилучший результат.
- Посредством выбранного алгоритма лидер формирует блоки вычислительных задач и их закрепление за доступными устройствами.
- Лидер инициализирует запуск вычислений на подчиненных узлах в соответствии с распределенной трудоемкостью вычислительных блоков.
- Лидер собирает полученные результаты и выбирает лучший.

Последовательность этапов предлагаемого метода показана на рис. 1.

Рис. 1. Этапы метода распределенного решения вычислительно сложных задач оптимизации с использованием гетерогенного коллектива вычислительных устройств

В следующем подразделе приведены результаты экспериментальных исследований разработанного метода.

3. Результаты экспериментального исследования

Будем рассматривать следующие исходные данные: решается задача распределения миссий для гетерогенного в смысле вычислительных мощностей и скорости перемещения коллектива летающих спасательных роботов. Предполагается, что группа роботов имеет представление о координатах потенциальных целевых объектов, также существует лидер в группе, осуществляющий управление. Непосредственно сама задача распределения миссий имеет следующую формулировку: необходимо таким образом распределить миссии (цели) среди участников группы роботов, чтобы общее время достижения целей роботами было минимальным, с максимальной эффективностью и максимальным покрытием целей.

Решением задачи является матрица назначений роботов на цели:

$$A = \begin{bmatrix} a_{ij} & \cdots & \dots \\ \vdots & \ddots & \vdots \\ \dots & \cdots & a_{nm} \end{bmatrix}, \tag{6}$$

где

$$a_{ij} = \begin{cases} 1, \text{если робот } i \text{ назначен на объект } j, \\ 0, \text{в противном случае.} \end{cases}$$

Кроме того, роботы не должны сталкиваться в процессе достижения целей, и в самом простейшем случае, если траектории линейны, они не должны пересекаться в пространстве выше поверхности земли. Непересечение траекторий является главным ограничением при назначении роботов на объекты. Количество таких ограничений при этом будет (nm^2), где n – количество роботов, m – количество целей.

Критерии оптимизации имеют следующий вид:

- Время достижения всех целей $T = \max\limits_{A} T_{destination} \rightarrow min$;
- Эффективность взаимодействия роботов и целей $E = \prod_{i,j} em_{ij} \rightarrow max, i, j: a_{ij} \neq 0$, где em_{ij} – количественное выражение эффективности взаимодействия i-го робота и j-й цели.
- Количество покрытых целей $C = \sum_{i=1,j=1}^{n,m} a_{ij} \rightarrow max$.

Для значений $n = 50$ (роботов), $m = 100$ (целей) при использовании алгоритма роя частиц были получены следующие результаты (см. рис. 2)

Рис. 2. Решение задачи назначения миссий методом роя частиц

Предположим, что выполняется распределенное выполнение блоков вычислений, схожее с параллельными независимыми запусками, т.е. лидер осуществляет рассылку исходных данных соседним роботам, инициирует вычисления, затем собирает полученные данные и выбирает лучший результат.

Вполне очевидно, что, распределяя по гетерогенным устройствам блоки вызовов ЦФ при условии исследования одного и того же поискового пространства, имеется возможность распределения блоков, например, размерностью 2500 вызовов ЦФ.

При этом по результатам вычислительного эксперимента получаем следующие значения критериев: время выполнения миссий (среднее) – 8.43 с.; эффективность взаимодействия – 0.44, количество достигнутых целей – 10. Размер блока 4000 вызовов ЦФ позволяет получить следующие результаты: время выполнения миссий (среднее) – 6.68 с.; эффективность взаимодействия – 0.15, количество достигнутых целей – 13. Размер блока 8000 вызовов ЦФ позволяет получить следующие результаты: время выполнения миссий (среднее) – 3.34 с.; эффективность взаимодействия – 0.116, количество достигнутых целей – 13. Некоторые ухудшения отдельно взятых параметров связаны с использованием свертки критериев, однако, в целом целесообразность увеличения размера блока подтверждается результатами эксперимента.

Далее будем полагать, что решаем задачу формирования блоков вычислений и их распределения по устройствам. В качестве эффективного алгоритма решения задачи двухкритериальной оптимизации выбран NSGA, как наиболее эффективный на интервале (0,1500) вызовов ЦФ и получены следующие результаты.

Таблица 1. Зависимость размеров сформированных блоков и их распределение в зависимости от вычислительной сложности процедуры формирования и распределения блоков

Количество вызовов ЦФ в NSGA при формировании и распределении блоков вычислений	**Время вычислений при решении задачи распределения миссий**	**Сформированные и распределенные блоки вычислений**
100	353	2900; 4350; 3500; 1250; 1500; 2600; 4500; 950; 2600; 1700
400	170	600; 1850; 1250; 3100; 1600; 5000; 2900; 5000; 5000; 3600
1000	147	1600; 5000; 800; 2350; 1750; 5000; 4600; 5000; 5000; 700

По результатам моделирования наиболее мощный вычислительный узел имеет производительность в рамках моделирования – 35 ЦФ/с, наименее мощный 2,7 ЦФ/с. Целевые функции для задач распределения миссий и задачи формирования блоков имеют близкую к аналогичной вычислительную сложность. Следовательно, имеется возможность оценить временные затраты на формирование и распределение блоков, а также на непосредственное решение распределенных блоков.

Таблица 2. Сравнение временных затрат на распределенное решение задачи распределения миссий

	Количество вызовов ЦФ в NSGA при формировании и распределении блоков вычислений	**Время формирования блоков и их распределение**	**Время вычислений при решении задачи распределения миссий**	**Суммарное время распределенного решения задачи распределения целей**	**Результат решения (средние значения)**
1	100	2.9с	353	355.9	Эффективность: 6,28; время выполнения миссий – 7.37с, количество достигнутых целей – 13
2	400	11с	170	181	Эффективность: 0.93; время выполнения миссий – 6.8с, количество достигнутых целей – 13
3	1000	29 с	147	176	Эффективность: 0.93; время выполнения миссий – 6.8с, количество достигнутых целей – 13

На примере строк 2 и 3 таблицы 2 видно, что имеет место ситуация, когда в случае 3 времени на формирование и распределение блоков потрачено больше, чем в случае 2. Но при этом получен такой результат формирования и распределения блоков, что суммарно распределение блоков и распределенное решение задачи назначения миссий получился меньше, чем в случае 2 при одинаковых средних результатах.

Проведем сравнение полученных при моделировании интервалов времени с интервалами времени, которые бы понадобились той же системе роботов для решения блоков минимального и максимального размера, если бы каждый вычислитель получал блоки одинакового размера.

Таблица 3. Сравнение затрат времени при назначении равных блоков вычислений и блоков с варьируемым размером как результат решения задачи (5)

	Время решения задачи распределения целей	**Результат назначения миссий**
100 вызовов ЦФ/блок	37с	Эффективность – 0,16; Время выполнения миссий – 58 с., количество достигнутых целей – 1
5000 вызовов ЦФ/блок	1851с	Эффективность: 0.93; время выполнения миссий – 6.8 с., количество достигнутых целей – 13
Вариативный размер блоков в соответствии с разработанным методом	176с	Эффективность: 0.93; время выполнения миссий – 6.8 с., количество достигнутых целей – 13

Таким образом, по результатам таблицы 3 видно, что разработанный метод позволяет получить решение задачи распределения миссий с ухудшением по времени решения до 4 раз, однако, с

существенным улучшением результата назначения миссий. При этом также, если сравнивать с распределением блоков равного размера, позволяющих получить аналогичный по качеству результат, расход времени уменьшается до 10 раз.

4. Заключение

В данной статье предложен метод распределенного решения вычислительно сложных оптимизационных задач с использованием гетерогенного коллектива вычислительных устройств. Актуальность темы обусловлена интенсивными исследованиями, ведомыми в направлении распределенного решения сложных вычислительных задач. Разработанный метод опирается на использование параллельных независимых запусков метаэвристик в гетерогенной распределенной вычислительной среде с учетом производительности устройств. Формирование блоков вычислений и их распределение формализовано как двухкритериальная задача оптимизации, и для ее решения также выбирается эффективная метаэвристика. Достигается следующий результат: сокращение времени вычислений при сохранении качества результата до 10 раз, увеличение времени вычислений по сравнению с решением всеми устройствами самого короткого блока вычислений до 4 раз со значительным улучшением результата распределенного решения задачи оптимизации.

Литература

1. *Abu-Taleb N. A., Hasan F. Abdulrazzak, Zahary A. T. and Al-Mqdashi A. M.* Offloading Decision Making in Mobile Edge Computing: A Survey// 2nd International Conference on Emerging Smart Technologies and Applications (eSmarTA) –Yemen, 2022. – P. 1–8.
2. *Ahmed M. et al.* A survey on vehicular task offloading: Classification, issues, and challenges // J. King Saud Univ. – Comput. Inf. Sci. 2022. – Vol. 34, № 7. – P. 4135–4162.
3. *Deng Y. et al.* Task offloading in multi-hop relay-aided multi-access edge computing // IEEE Trans. Veh. Technol. 2023. – Vol. 72, № 1. – P. 1372–1376.
4. *Do T., Tran D.A., Vo A.* Edge assignment in edge federated learning // SN Appl. Sci. 2023. – Vol. 5, № 11.
5. *Yang H. et al.* Pruning-based Deep Reinforcement Learning for task offloading in end-edge-cloud collaborative Mobile Edge Computing // Journal of Computing and Electronic Information Management. 2024. – Vol. 13, № 1. – P. 1–9.
6. *Huang Z.-S., Lu C.-H., Hwang I.-S.* Direct edge-to-edge local-learning-assisted model-based transfer learning // IEEE Internet Things J. 2024. – Vol. 11, № 11. – P. 20223–20235.
7. *Li J., Gonsalves T.* A hybrid approach for metaheuristic algorithms using island model // Proceedings of the Future Technologies Conference (FTC) 2021, Volume 3. Cham: Springer International Publishing, 2022. – P. 311–322.
8. *Zhang M. et al.* Resource-efficient parallel split learning in heterogeneous edge computing // 2024 International Conference on Computing, Networking and Communications (ICNC). IEEE, 2024.
9. *Клименко А.Б., Баринов А.А*. Метод управления вычислительными ресурсами распределенных систем на основе «жадной» стратегии и онтологии эффективных алгоритмов // Моделирование, оптимизация и информационные технологии. 2024;12(1). URL: https: // moitvivt.ru/ru/journal /pdf?id=1508 (дата обращения: 20.07.2024).