

ТЕХНОЛОГИЯ ИНТЕЛЛЕКТУАЛЬНОЙ ОРГАНИЗАЦИИ ПРОЦЕССА ВЫПОЛНЕНИЯ НЕОДНОРОДНЫХ КОМПОЗИТНЫХ ПРИЛОЖЕНИЙ В РАСПРЕДЕЛЕННОЙ ВЫЧИСЛИТЕЛЬНОЙ СРЕДЕ

М. А. МЕЛЬНИК, Д. А. НАСОНОВ, А. В. БУХАНОВСКИЙ

Университет ИТМО, 197101, Санкт-Петербург, Россия

E-mail: michael.melnik@niuitmo.ru

Сформулировано понятие неоднородных композитных приложений и идентифицированы вычислительные режимы: пакетный, потоковый и итеративный. Обусловлена необходимость разработки алгоритмов планирования для каждого из режимов. Разработаны алгоритмы планирования на базе методов машинного обучения, эволюционных подходов и методов искусственного интеллекта. Разработана технология, обеспечивающая совместное выполнение разнородных композитных приложений на основе каскада разработанных методов оптимизации.

Ключевые слова: планирование композитных приложений, облачные вычисления, суперкомпьютер, потоковая обработка данных, совместное проектирование

Расширение области применения вычислительных приложений в научных и промышленных исследованиях определяет необходимость интенсивного роста вычислительных систем (ВС). В настоящее время проектируются ВС эксафлопсной производительности, однако потребность в вычислительных мощностях растет значительно быстрее. Эффективное использование такого рода комплексных ВС осложняет необходимость обеспечения глобального взаимодействия разнородных режимов вычислений и обработки данных (пакетный, потоковый и итеративный режимы) в рамках решения одной или нескольких прикладных задач с использованием разнородных ресурсов (графические процессоры, облачные среды, суперкомпьютеры). Это обусловило формирование понятия неоднородного композитного приложения (НКП), которое имеет блоки с различными вычислительными архитектурами и принципами организации процесса выполнения, требующими индивидуального технологического подхода на уровне единой вычислительной платформы.

Целью работы является повышение эффективности выполнения в глобальных распределенных вычислительных средах неоднородных композитных приложений, совмещающих пакетный, потоковый и итеративный режимы исполнения. Одним из основных компонентов ВС для оптимизации процесса выполнения приложений является планировщик. Разработка методов и алгоритмов планирования активно ведется как российскими командами [1—4], так и по всему миру [5—7]. В ходе анализа существующих решений определены перспективы применения технологий машинного обучения для решения задачи планирования (пакетный режим) [8—10], а также обнаружены недостатки существующих методов планирования для потокового [11—13] и итеративного режимов [14, 15], как и отсутствия общей схемы совмещения этих режимов.

В ходе исследований разработаны алгоритмы NNS (Neural Network Scheduler), SSGA (Stream Scheduling Genetic Algorithm), IMSGA (Iterative Multiagent Scheduling Genetic Algorithm), позволяющие оптимизировать планирование выполнения композитных приложений в пакетном, потоковом и итеративном режимах. NNS основан на принципах обучения с подкреплением. SSGA и IMSGA в качестве ядра оптимизации используют эволюционные подходы.

При разработке алгоритма NNS решена проблема представления дискретной задачи планирования в терминологии машинного обучения за счет идентификации входного (состояния) и выходного (действия) слоев нейронной сети, а также предложен способ перехода к относительным метрикам для обобщения схемы планирования.

Семейство разработанных алгоритмов SSGA позволяет учитывать данные мониторинга системы потоковой обработки и перераспределять нагрузку по узлам облачной инфраструктуры с возможностью добавления и исключения узлов.

Алгоритм IMSGA основан на графовой модели размещения вычислительной нагрузки по узлам суперкомпьютера. В ходе исследования разработана схема совместного проектирования, обеспечивающая взаимодействие ВС и многомасштабного приложения.

Исследование эффективности разработанных алгоритмов показало прирост производительности до 40, 11 и 55 % для NNS, SSGA и IMSGA соответственно. На базе разработанных семейств алгоритмов разработана интеллектуальная технология планирования НКП.

СПИСОК ЛИТЕРАТУРЫ

1. *Афанасьев А. П.* и др. Увеличение вычислительной мощности распределенных систем с помощью грид-систем из персональных компьютеров // Параллельные вычислительные технологии. 2011. С. 6—14.
2. *Козинев Е. А., Гергель В. П.* Параллельные вычисления на графических процессорах в задачах многокритериальной оптимизации // Суперкомпьютерные дни в России. Москва, 24—25 сентября 2018 г. Т. 24. С. 25.
3. *Гервич Л. Р., Кравченко Е. Н., Штейнберг Б. Я., Юрушкин М. В.* Автоматизация распараллеливания программ с блочным размещением данных // Сибирский журнал вычислительной математики. 2015. Т. 18, № 1. С. 41—53.
4. *Сухорослов О. В., Назаренко А. М.* Сравнительная оценка методов планирования приложений в распределенных вычислительных средах // Программные системы: теория и приложения. 2017. Т. 8, № 1. С. 63—81.
5. *Rodriguez M. A., Buyya R.* A taxonomy and survey on scheduling algorithms for scientific workflows in IaaS cloud computing environments // *Concurr. Comput.* 2017. Vol. 29, N 8.
6. *Deelman E. et al.* Pegasus, a workflow management system for science automation // *Futur. Gener. Comput. Syst.* 2015. Vol. 46. P. 17—35.
7. *Foster I., Kesselman C.* The grid 2: Blueprint for a new computing infrastructure. Morgan Kaufmann, 2004. 748 p.
8. *Tong Z. et al.* QL-HEFT: a novel machine learning scheduling scheme base on cloud computing environment // *Neural Comput. Appl.* 2019. <https://doi.org/10.1007/s00521-019-04118-8>.
9. *Tong Z. et al.* A scheduling scheme in the cloud computing environment using deep Q-learning // *Information Sciences.* 2020. Vol. 512, February. P. 1170—1191.
10. *Rashmi S., Basu A.* Q learning based workflow scheduling in hadoop // *Intern. J. Appl. Eng. Res.* 2017. Vol. 12, N 12. P. 3311—3317.
11. *Peng B. et al.* R-storm: Resource-aware scheduling in storm // *Middlew. 2015. Proc. 16th Annu. Middlew. Conf.* 2015. P. 149—161.
12. *Dwarakanathan S.* S-Flink: Schedule for QoS in Flink Using SDN // *Proc. Intern. Comput. Softw. Appl. Conf.* 2016. Vol. 2. P. 620—621.
13. *Cheng D. et al.* Adaptive scheduling parallel jobs with dynamic batching in spark streaming // *IEEE Trans. Parallel Distrib. Syst.* 2018. Vol. 29, N 12. P. 2672—2685.
14. *Hoekstra A. G. et al.* Multiscale computing for science and engineering in the era of exascale performance // *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* 2019. Vol. 377, N 2142.
15. *Ben Belgacem M., Chopard B.* MUSCLE-HPC: A new high performance API to couple multiscale parallel applications // *Futur. Gener. Comput. Syst.* 2017. Vol. 67. P. 72—82.

- Михаил Алексеевич Мельник** — *Сведения об авторах*
— Университет ИТМО, национальный центр когнитивных разработок; инженер; E-mail: michael.melnik@niuitmo.ru
- Денис Александрович Насонов** — канд. техн. наук; Университет ИТМО, национальный центр когнитивных разработок; E-mail: dnasonov@itmo.ru
- Александр Валерьевич Бухановский** — д-р техн. наук; Университет ИТМО, национальный центр когнитивных разработок; E-mail: avb_mail@mail.ru

Поступила в редакцию
31.01.2020 г.

Ссылка для цитирования: Мельник М. А., Насонов Д. А., Бухановский А. В. Технология интеллектуальной организации процесса выполнения неоднородных композитных приложений в распределенной вычислительной среде // Изв. вузов. Приборостроение. 2020. Т. 63, № 2. С. 191—193.

INTELLECTUAL TECHNOLOGY FOR ORGANIZING THE EXECUTION OF HETEROGENEOUS WORKFLOWS IN A DISTRIBUTED COMPUTING ENVIRONMENT

M. A. Melnik, D. A. Nasonov, A. V. Boukhanovsky

ITMO University, 197101, St. Petersburg, Russia
E-mail: michael.melnik@niuitmo.ru

The concept of heterogeneous workflows is formulated and following computational modes are identified: batch, streaming, and iterative. The necessity of developing of scheduling algorithms for each of modes is determined. Developed scheduling algorithms are based on machine learning methods, evolutionary approaches, and artificial intelligence methods. A technology has been developed that ensures the collaborative execution of heterogeneous workflows based on a cascade of developed optimization methods.

Keywords: workflow scheduling, cloud computing, supercomputer, stream data processing, collaborative design

REFERENCES

1. Afanasyev A.P. et al. *Parallel computational technologies*, 2011, pp. 6–14.
2. Kozinov E.A., Gergel' V.P. *Russian Supercomputing Days*, Moscow, September 24–25, 2018. Vol. 24. P. 25.
3. Gervich L.R., Kravchenko E.N., Steinberg B.Y., Yurushkin M.V. *Numerical Analysis and Applications*, 2015, no. 1(18), pp. 35–45.
4. Sukhoroslov O., Nazarenko A. *Program Systems: Theory and Applications*, 2017, no. 1(8), pp. 63–81 (in Russ.)
5. Rodriguez M.A., Buyya R. *Concurr. Comput.*, 2017, no. 8(29).
6. Deelman E. et al. *Futur. Gener. Comput. Syst.*, 2015, Vol. 46, pp. 17–35.
7. Foster I., Kesselman C. *The grid 2: Blueprint for a new computing infrastructure*, Morgan Kauffman, 2004, 748 p.
8. Tong Z. et al. *Neural Comput. Appl.*, 2019, <https://doi.org/10.1007/s00521-019-04118-8>.
9. Tong Z. et al. *Information Sciences*, 2020, vol. 512, February, pp. 1170–1191.
10. Rashmi S., Basu A. *Intern. J. Appl. Eng. Res.*, 2017, no. 12(12), pp. 3311–3317.
11. Peng B. et al. *Middlew. 2015. Proc. 16th Annu. Middlew. Conf.*, 2015, pp. 149–161.
12. Dwarakanathan S. *Proc. Intern. Comput. Softw. Appl. Conf.*, 2016, vol. 2, pp. 620–621.
13. Cheng D. et al. *IEEE Trans. Parallel Distrib. Syst.*, 2018, no. 12(29), pp. 2672–2685.
14. Hoekstra A.G. et al. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.*, 2019, no. 2142(377).
15. Ben Belgacem M., Chopard B. *Futur. Gener. Comput. Syst.* 2017, vol. 67, pp. 72–82.

Data on authors

- Mikhail A. Melnik** — ITMO University, National Center of Cognitive Research; Engineer
E-mail: michael.melnik@niuitmo.ru
- Denis A. Nasonov** — PhD; ITMO University, National Center of Cognitive Research;
E-mail: dnasonov@itmo.ru
- Alexander V. Boukhanovsky** — Dr. Techn. Sci., Professor; ITMO University, National Center of Cognitive Research; E-mail: avb_mail@mail.ru

For citation: Melnik M. A., Nasonov D. A., Boukhanovsky A. V. Intellectual technology for organizing the execution of heterogeneous workflows in a distributed computing environment. *Journal of Instrument Engineering*. 2020. Vol. 63, N 2. P. 191—193 (in Russian).

DOI: 10.17586/0021-3454-2020-63-2-191-193