

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра програмного забезпечення комп'ютерних систем

"На правах рукопису"
УДК 004.021

«До захисту допущено»
Науковий керівник кафедри

_____ Дичка І.А.
(підпис)

“ ____ ” _____ 2017 р.

Магістерська дисертація

зі спеціальності 8.05010302 “Інженерія програмного забезпечення”

на тему: МЕТОД ДИНАМІЧНОГО РОЗПОДІЛЕННЯ ЗАВДАНЬ В
КЛАСТЕРНИХ СИСТЕМАХ

Виконав: студент 6 курсу, групи КП-52м

Шершнів Олексій Володимирович

_____ (підпис)

Науковий керівник доц., к.т.н. Жабіна В.В.

_____ (підпис)

Рецензент доц., к.т.н., доц. Ткаченко В. В.

_____ (підпис)

Рецензент доц., к.т.н., доц. Орлова М. М.

_____ (підпис)

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць інших
авторів без відповідних посилань.

Студент _____
(підпис)

Київ – 2017

РЕФЕРАТ

Актуальність теми. Більшість з сучасних технологій паралельного програмування відносяться до засобів статичного розпаралелювання процесів. Завдання розпаралелювання в цьому випадку вирішуються на етапі розробки програм. При статичному аналізі алгоритмів не завжди вдається виявити паралельні гілки, тобто прихований паралелізм, що пояснюється браком інформації про динаміку процесів у системі.

Одним з перспективних підходів, що дозволяють усунути ряд недоліків статичного планування, є розробка засобів динамічного розпаралелювання обчислень. У цьому разі призначення завдань на обчислювальні вузли здійснюється системою в процесі вирішення завдань. Такий підхід дає можливість досягти більшої ступеня паралелізму, так як дозволяє виявити паралельні гілки, які виникають безпосередньо в процесі обчислень.

Тривалість вирішення задач в паралельних системах визначається ефективністю розподілу завдань між обчислювальними модулями для їх рівномірного завантаження. Статичне планування обчислень не завжди дозволяє виявити паралелізм задач, що знижує можливість ефективного завантаження модулів кластеру. Для ефективного розпаралелювання обчислень запропоновані методи динамічного розподілу завдань між виконуючими модулями як для дрібнозернистих алгоритмів, так і для крупнозернистих структур з використанням гібридних методів паралелізму. Однак, проблеми динамічного розпаралелювання обчислювальних процесів вимагають подальших досліджень.

Об'єктом дослідження є процес динамічного планування в паралельній обробці даних на кластерних системах.

Предметом дослідження : методи формування та оптимізації роботи паралельних потоків команд в кластерних системах.

Мета роботи: є розроблення методу для реалізації динамічного розподілення обчислень для крупнозернистих та дрібнозернистих алгоритмів з використанням технологій прискорення вирішення задач за рахунок зменшення об'ємів інформації, що пересилається між вузлами системи.

Методи дослідження. В роботі використовуються методи математичного моделювання, методи надання рекомендацій, методи ідеалізації в порівняннях.

Наукова новизна роботи полягає в наступному:

1. Розроблено новий метод динамічного розпаралелювання обчислень при виконанні задачі, що задана графом, на кластері та базується на ідеях гібридного паралелізму та використанні дескрипторної передачі даних.

2. Створена методологія для підвищення швидкодії та зменшення обсягу передачі даних між модулями кластера, що базується на групуванні процесорів та використанні нано-потоків.

Практична цінність отриманих в роботі результатів полягає в тому, що ми отримали новий метод динамічного планування, що дозволяє побачити цілісну картину проблеми паралелізму та методів його оптимізації в сучасних кластерних системах з використанням цілого спектру технологій мінімізації надлишкової передачі даних.

Апробація роботи. Основні положення і результати роботи були представлені та обговорювались на ІХ науковій конференції магістрантів та аспірантів «Прикладна математика та комп'ютинг» ПМК-2017 (Київ, 19–21 квітня 2017 р.) та опубліковані у збірнику тез за результатами конференції; результати дослідження існуючих рішень для збільшення швидкості обробки запитів були представлені на VII МІЖНАРОДНА КОНФЕРЕНЦІЯ СТУДЕНТІВ І МОЛОДИХ УЧЕНИХ "СУЧАСНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ 2017" (Одеса, 22 – 24 травня 2017 р.) та опубліковані у збірнику тез доповідей.

Структура та обсяг роботи. Магістерська дисертація складається з вступу, чотирьох розділів, висновків та додатків.

У вступі надано загальну характеристику роботи, виконано оцінку сучасного стану проблеми, обґрунтовано актуальність напрямку досліджень, сформульовано мету і задачі досліджень.

У першому розділі розглянуто основні технології та рекомендаційні системи, які можуть бути використані в створенні власного методу, проведений їх аналіз та описано рекомендації для подальшого застосування.

У другому розділі проаналізовано теоретичні основи реалізації, а саме методологію реалізації динамічного планування обчислень, алгоритми створення черг, а також методи, що будуть використовуватись для оптимізованої роботи методу та підвищення його відмово стійкості. Розглянутий гібридний метод представлення архітектури системи.

У третьому розділі показано обрані методи реалізації запропонованих підходів та їхні модифіковані реалізації для впровадження гармонійної взаємодії всіх елементів між собою.

У четвертому розділі описано тестування та аналіз описаного методу за допомогою додатку, що реалізує запропонований рекомендаційний метод та реалізації алгоритмів підвищення швидкодії та процесу додаткового планування.

У п'ятому розділі наведено опис стартап-проекту та його сильних і слабих сторін.

У висновках проаналізовано отримані результати роботи.

Робота виконана на 80 аркушах, містить 4 додатки та посилання на список використаних літературних джерел з 51 найменування. У роботі наведено 12 рисунків та 8 таблиць.

Ключові слова: мультипроцесорна система, паралельні обчислення, планування завдань, дескрипторна передача даних, гібридна модель паралелізму.

ABSTRACT

The relevance of the topic. Most of the modern technologies of parallel programming are static parallelization. Task parallelization in this case are solved at the development stage programs. For the static analysis of the algorithms is not always possible to identify parallel branches, then there is a hidden parallelism, due to the lack of information about the dynamics of processes in the system.

One of the promising approaches to overcome a number of deficiencies of the static planning is the development of dynamic parallelization of computations. In this case, assign tasks to the compute nodes is carried out by the system in the process of solving problems. This approach makes it possible to achieve higher degrees of parallelism, as it allows to identify parallel branches, which arise directly in the computation process.

The duration of decision tasks in parallel systems is determined by the efficiency of the allocation of tasks among computational modules for uniform load. Static scheduling of computations is not always possible to identify task parallelism, which reduces the effective loading of modules in the cluster. For effective parallelization methods are proposed for the dynamic allocation of tasks between the executing modules as algorithms for fine-grained and coarse-grained structures using hybrid methods of parallelism. However, the problem of the dynamic parallelization of computational processes require further research.

The object of research is the process of dynamic scheduling in parallel data processing on cluster systems.

The subject of research : methods of formation and optimization of parallel threads of commands in a clustered system.

The aim of this work is to develop a method for implementation of dynamic parallelization for coarse-grained and fine-grained algorithms using technology to speed up the solution of the task by reducing the volume of data sent between nodes in the system.

Research methods. We use the techniques of mathematical modeling, methods, recommendations, methods of idealization in comparisons.

The scientific novelty of the work lies in the following:

1. Developed its own method of dynamic parallelization of computations in the task which is given by the graph on the cluster and is based on the ideas of hybrid parallelism and use descriptive data transmission.

2. Created a proprietary methodology to reduce the amount of data transfer between modules of the cluster based on group processes and use of nano-flows, with the aim of improving the performance shown in the result graphs and tables.

Practical value the obtained results lies in the fact that we got a new method of dynamic scheduling that allows you to see the big picture concurrency problems and methods of its optimization in the modern cluster systems using a range of technologies to reduce excess data transfer.

Approbation of the work. The main provisions and the results were presented and discussed at the IX scientific conference of students and postgraduates "Applied mathematics and computing" PMK-2017 (Kyiv, 19-21 April 2017) and published in the book of abstracts after the conference; the results of a study of existing solutions to increase the speed of query processing were presented at the VII INTERNATIONAL CONFERENCE of STUDENTS AND YOUNG SCIENTISTS "MODERN INFORMATION TECHNOLOGIES in 2017" (Odessa, may 22 – 24, 2017) and published in the collection of abstracts.

The structure and volume of work. Master thesis consists of introduction, four chapters, conclusions and applications.

The introduction includes a General description of the work performed assessment of the current state of the problem, the urgency of the referral

research goals the purpose and objectives of the research.

The first section describes the core technologies and Recommender systems, which can be used in the creation of their own method, their analysis and recommendations are described for further use.

The second section analyzes the theoretical foundations of the implementation, namely the implementation methodology dynamic scheduling of

computation, algorithms, Queuing, and methods that will be used for optimized operation of the method and increase its resiliency.

The third section shows selected methods for the implementation of the proposed approaches and their modified implement to implement the harmonious interaction of all the elements together.

The fourth section describes the testing and analysis method described above using the application that implements the proposed recommendation method and proprietary algorithms improve the performance.

In the fifth section we present the business idea and business plan creation and development of startups based on the developed method.

In conclusion, analyzed the obtained results.

The work is performed on 80 pages, contains 4 applications and references to the list of used literary sources of the 51 names. The paper presents 12 figures and 8 tables.

Keywords: multiprocessor system, parallel computing, task scheduling, descriptory data transmission, the hybrid model of parallelism.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Silva J.G.D., Wood J.V. Design of processing subsystems for Manchester data flow computer // IEEE Proc. N.Y. – 1981. – Vol. 128, N 5. – P. 218 – 224.
2. Watson R., Guard J. A practical data flow computer// Computer. – 1982. – Vol. 15, N 2. – P. 51 – 57.
3. Жабин В.И. Метод распараллеливания процессов в вычислительных системах// Вісник Національного технічного університету України “Київський політехнічний інститут”. Інформатика, управління та обчислювальна техніка. – 2000. – № 34. – С. 136 – 142.
4. Жабин В.И. Реализация параллельных процессов в вычислительных системах// Искусственный интеллект, №3, 2002, с. 235 – 241.
5. Жабин В.И. Организация параллельных вычислений в распределенных системах// Вісник Національного технічного університету України “Київський політехнічний інститут”, “Інформатика, управління та обчислювальна техніка”, № 38, 2002, с. 10 – 15.
6. Жабин В.И. Реализация вычислений под управлением потока дескрипторов данных в мультипроцессорных системах// Электронное моделирование. – 2003. – Т.25, №1. – С. 35 – 47.
7. Li. K., Hudak. P. Memory Coherence in Shared Virtual Memory Systems// ASM Trans. On Computer Systems. – 1989. – Vol.7, N11. – P. 321 – 359.
8. Авиженис А. Отказоустойчивость – свойство, обеспечивающее постоянную работоспособность цифровых систем// ТИИЭР. – 1978. – Т.66, №10. – С. 5 –25.
9. M. Acceta, R. Baron, D. Golub, R. Rashid, A. Tevanian and M.Young, “Mach: A New Kernel Foundation for UNIX Development”, Proceedings of the USENIX Conference, July 1986.
10. A. Agarwal, R. Bianchini, D. Chaiken, K.L. Johnson, D. Kranz, J. Kubiawicz, B.H. Lim, K. Mackenzie and D. Yeung, “The MIT Alewife Machine: Architecture and Performance”, Proceedings of the 22nd

International Symposium on Computer Architecture, Santa Margherita Ligure, Italy, June 1995.

11. S. Aldomà, “Mecanismes en el Kernel d’un Sistema Operatiu pel Control de l’Execució d’Aplicacions Paral·leles”, M.S. Thesis, Departament d’Arquitectura de Computadors - UPC, February 1999, written in Catalan.
12. T. E. Anderson, “FastThreads User’s Manual”, Department of Computer Science and Engineering, University of Washington, January 1990.
13. T. Anderson, B. Bershad, E. Lazowska and H. Levy, “Scheduler Activations: Effective Kernel Support for the User-Level Management of Parallelism”, Proceedings of the 13th. ACM Symposium on Operating System Principles (SOSP), October 1991.
14. T. E. Anderson, E. D. Lazowska, “Quartz: A Tool for Tuning Parallel Program Performance”, Department of Computer Science and Engineering, University of Washington, September 1989.
15. E. Ayguadé, X. Martorell, J. Labarta, M. González and N. Navarro, “Exploiting Multiple Levels of Parallelism in Shared-memory Multiprocessors: a Case Study”, Dept. d’Arquitectura de Computadors - Universitat Politècnica de Catalunya, Technical Report: UPC-DAC-1998-48, November 1998.
16. E. Ayguadé, X. Martorell, J. Labarta, M. González and N. Navarro, “Exploiting Parallelism Through Directives on the Nano-Threads Programming Model”, Proceedings of the 10th. Language and Compilers for Parallel Computing (LCPC), Minneapolis, USA, August 1997.
17. M. Bach, “The Design of the UNIX Operating System”, Prentice-Hall, 1986.
18. D. Bailey, T. Harris, W. Saphir, R. Wijngaart, A. Woo and M. Yarrow, “The NAS Parallel Benchmarks 2.0”, Technical Report NAS-95-020, NASA, December 1995.
19. J. Barton, N. Bitar, “A Scalable Multi-Discipline Multiple-Processor Scheduling Framework for IRIX”, Proceedings of the IPPS ‘95 Workshop on Job Scheduling Strategies for Parallel Processing”, April 1995.

20. B. J. Beckmann, "Hardware and Software for Functional and Fine Grain Parallelism", Ph.D. thesis, Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, 1993.
21. C. D. Polychronopoulos, "Microarchitecture Support for Dynamic Scheduling of Acyclic Task Graphs", CSRD Technical Report 1207, August 1992.
22. B. Bershad, E. Lazowska, H. Levy, "Presto: A System for Object-oriented Parallel Programming", *Software - Practice and Experience*, vol. 18, no. 8, pp. 713-732, 1988.
23. D. L. Black, "Scheduling and Resource Management Techniques for Multiprocessors", Ph.D. Thesis, CMU-CS-90-152, Carnegie Mellon University, July 1990.
24. R.D. Blumofe, "Managing Storage for Multithreading Computations", M.S. Thesis, Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 1992.
25. R. D. Blumofe, C. F. Joerg, B. C. Kuszmaul, C. E. Leiserson, K. H. Randall and Y. Zhou, "Cilk: An Efficient Multithreaded Runtime System", *Proceedings of the Fifth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP '95)*, Santa Barbara, California, July 19-21, 1995.
26. J. Boykin, D. Kirschen, A. Langerman, S. LoVerso, "Programming Under Mach", Addison-Wesley Publishing Company, 1993.
27. D.R. Butenhof, "Programming with POSIX threads", *Professional Computing Series*, Addison-Wesley, ISBN 0-201-63392-2, May 1997.
28. C.R. Calidonna, M. Giordano, M. Mango Furnari, "A Graphic Parallelizing Environment for User-Compiler Interaction", *Proceedings of the 13th ACM International Conference on Supercomputing (ICS'99)*, Rhodes, Greece, June 1999.
29. J. Caubet, "Paral.lelització Automàtica i Serveis de l'Entorn del Sistema", M.S. Thesis, Departament d'Arquitectura de Computadors - UPC, February 1999, written in Catalan.

30. R. Chandra, "COOL User Manual", Computer Systems Laboratory, Stanford University, November 1992.
31. R. Chandra, A. Gupta and J. L. Hennessy, "Data Locality and Load Balancing in COOL", Fourth ACM SIGPLAN Symposium on the Principles and Practice of Parallel Programming (PPoPP), pp. 249-259, May 1993.
32. R. Chandra, A. Gupta, J. L. Hennessy, "Integrating Concurrency and Data Abstraction in the COOL Parallel Programming Language", Technical Report CSL-TR-92-511, Computer Systems Laboratory, Stanford University, February 1992.
33. J. Chow, W. L. Harrison III, "Switch Stacks: A Scheme for Microtasking Nested Parallel Loops", Center for Supercomputing Research and Development (CSRD), University of Illinois at Urbana-Champaign, 1990.
34. A. C. Cooper and R. P. Draves, "CThreads", Technical Report CMU-CS-88-154, School of Computer Science, Carnegie Mellon University, February 1998.
35. D. Cortessi, A. Evans, W. Ferguson and J. Hartman, "Topics in IRIX Programming", Doc. num. 007-2478-004, Silicon Graphics, Inc., <http://techpubs.sgi.com>, 1996.
36. D. Cortessi, A. Evans, W. Ferguson and J. Hartman, "Topics in IRIX Programming", Doc. num. 007-2478-006, Silicon Graphics, Inc., <http://techpubs.sgi.com>, 1998.
37. N. Duran A. Hoeflinger J. Yuan Lin Massaioli F. Teruel X. Unnikrishnan P. Guansong Zhang Ayguade, E. Copty. The design of openmp tasks. *Parallel and Distributed Systems*, 20:404–418, March 2009.
38. J. M. Bull. Measuring synchronisation and scheduling overheads in openmp. In *Proceedings of First European Workshop on OpenMP (1999)*, pages 99–105, 1999.
39. Larry Snyder Calvin Lin. *Principles of Parallel Programming*. Addison Wesley, 2008.
40. Per Carlsson. Tutorial dieselFoam. Energy Technology Center, Lulea University of Technology, February 2009.

41. William D. Gropp David E. Keyes. A comparison of domain decomposition techniques for elliptic partial differential equations and their parallel implementation. *SIAM Journal on Scientific and Statistical Computing*, 8:162–202, March 1987.
42. Daniel Etiemble Franck Cappello. Mpi versus mpi+openmp on ibm sp for the nas benchmarks. *Supercomputing, ACM/IEEE 2000 Conference*, page 12, Nov 2000.
43. Gabriele Jost Georg Hager and Rolf Rabenseifner. Communication characteristics and hybrid mpi/openmp parallel programming on clusters of multi-core smp nodes. *2009 Parallel, Distributed and Network-based Processing*, pages 427–436, 2009. 75 76 BIBLIOGRAPHY
44. Wuilbert lopez Gustaf Hallberg, Per Hallgren. Parallelisering av algoritmer for flerkarniga processorer. Master's thesis, Chalmers University of Technology, 2010.
45. Mattijs Janssens Andy Heather Sergio Ferraris Graham Macpherson Helene Blanchonnet Henry Weller, Chris Greenshields and Jenya Collings. *OpenFOAM, The Open Source CFD Toolbox - User Guide*. OpenCFD, 1.7.1 edition, August 2010.
46. Gabriele Jost and Ferhat F. Hatay Haoqiang Jin, Dieter an Mey. Comparing the openmp, mpi, and hybrid programming paradigm on an smp cluster. *NAS Technical Report, NAS-03-019:10*, November 2003. Michael R. Marty Mark D. Hill. Amdahl's law in the multicore era. *IEEE Computer*, pages 33–38, July 2008.
47. T. Bosilca G. Fagg G.E. Gabriel E. Dongarra J.J. Pjesivac-Grbovic, J. Angskun. Performance analysis of mpi collective operations. *Parallel and Distributed Processing Symposium, 2005. Proceedings. 19th IEEE International*, 4:8, April 2005.
48. Bruno R. Preiss. *Data Structures and Algorithms with Object-Oriented Design Patterns in C++*. John Wiley & Sons, 1998.
49. George N. Barakos Rene Steijl and Ken J. Badcock. Parallellisation of cfd methods for multi-physics problems. *European Conference on Computational Fluid Dynamics, ECCOMAS CFD 2006:16*, 2006.