

Load Balancing Algorithms for Parallel Computing

Subjects: Computer Science, Theory & Methods

Contributor: Yong Li, Jinxing Li, Yu Sun, Haisheng Li

Computational fluid dynamics (CFD) is a discipline that solves and analyzes fluid dynamics problems through computer and numerical simulation methods. In the process of CFD numerical simulations, the complexity of the problem is gradually increasing, the accuracy of the numerical simulation is becoming more and more demanding, and the network size is continuously expanding. Load-balancing algorithms for CFD parallel computing are being gradually and extensively researched in the world.

Keywords: firefly algorithm ; bio-inspired design ; hybrid methods

1. Introduction

Computational fluid dynamics (CFD) ^[1] is a discipline that solves and analyzes fluid dynamics problems through computer and numerical simulation methods. In the process of CFD numerical simulations, the complexity of the problem is gradually increasing, the accuracy of the numerical simulation is becoming more and more demanding, and the network size is continuously expanding. In the process of numerical simulations, the computational area is discretized into multiple grid blocks of different sizes, and these blocks are assigned to different processors for parallel computation. Therefore, how to make the load task of each processor reasonably balanced is the main problem to be solved, and it is also an important technology to improve the processing efficiency. As the scale of computation increases, the impact of the communication overhead of the processors is also gradually increasing on the parallel processing efficiency. The mismatch between the number of grid blocks and the number of processes, as well as the mismatch between the computational capacity of grid blocks and the computational capacity of processes, makes the traditional partitioning or combination strategies unable to meet the demand of load balancing well ^[2]. Therefore, the research on algorithms for large-scale load balancing is crucial.

At present, load-balancing algorithms for CFD parallel computing are being gradually and extensively researched in the world. For example, the load-balancing algorithm of greedy method and the recursive pairwise edge splitting load-balancing algorithm proposed by Streng ^[3] and Ytterström ^[4] are classical load-balancing algorithms based on a geometric level. In addition, Hendrickson et al. ^[5] proposed a multi-level algorithm for partitioning graphs, which is a method of mapping to a refined graph by coarsening the graph dissection. In recent years, heuristic algorithms have also been increasingly used in the research of load-balancing algorithms. Most of them are swarm intelligence algorithms that imitate natural bodies, such as firefly algorithm, ant colony algorithm, and bee colony algorithm. They are applied to solve various optimization scheduling problems and are widely used in industry ^[6], network transmission ^[7], biology ^[8] and cloud computing ^[9]. Among them, the combination of neural networks and other algorithms has achieved considerable research success in image capture and retrieval ^{[10][11][12][13]}. With the development of parallel computing, the improvement and optimization of heuristic algorithms to solve the load-balancing problem of parallel computing have also achieved considerable results. For example, Kernighan et al. ^[14] proposed a heuristic algorithm for graph segmentation. The hybrid load-balancing algorithm proposed by Yang Chengfu et al. ^[15] and the ant colony optimization based MrLBA algorithm proposed by Arfa Muteeh ^[16] are typical methods of heuristic algorithms applied to load balancing. However, swarm intelligence algorithms generally easily fall into local optimum, and the optimization effect is uneven. This research uses the multi-level graph segmentation algorithm of a structural grid to obtain a network coarsening graph and proposes a FaCO algorithm based on the firefly and ant colony algorithm to subdivide the coarsening graph. Based on the fusion of the two in the optimization rule, a new optimization rule is established to adjust the position update, which avoids falling into the local optimal situation to a certain extent.

2. Load Balancing Algorithms for Parallel Computing

In order to solve the time and cost problems of large-scale computation, parallel computing decomposes complex problems into several parts with certain regularity and assigns each part to a separate processor for simultaneous

computation of multiple instructions. It not only reduces the time cost but also improves the overall computational performance. With the development of CFD numerical simulation research, the scale of computation is increasing, and parallel computing becomes an effective method to solve this problem.

With the wide application of CFD numerical simulations, such as point cloud sampling [17], multi-view image retrieval [18], etc., the research on load-balancing algorithms for CFD parallel computing can be basically divided into two main categories: geometry-based and graph-based [3][4][5]. However, with the development of research on heuristic algorithms, there is a great potential for their application to solve load-balancing problems. Researchers mainly focus on the improvement of heuristic algorithms and their graph-based optimization applications to load-balancing problems for parallel computing.

Kannan et al. proposed a multi-objective load-balancing method using a bio-inspired algorithm [19]. It solves the pre-convergence problem by a micro-genetic algorithm and proposed a stable method of combining cat swarm optimization for process load distribution (MG-CSO). Better results are obtained on the time cost problem of load balancing for cloud computing.

The hybrid discrete artificial bee colony algorithm (ABC) proposed by Junqing Li and Yunqi Han investigated and solved the task scheduling problem in cloud computing systems [20]. It designs an improved scout bee using different local search methods to obtain the best food source or waste solution that can improve the convergence of the proposed algorithm.

Ahmad M. Manasrah et al. proposed a hybrid algorithm based on genetic algorithm (GA) and particle swarm optimization (PSO) to solve the multitask scheduling problem [21]. In the field of multitask scheduling, this algorithm converges to the optimal solution much faster and with higher quality.

The firefly load-balancing algorithm was proposed by Manisha et al. It reduces the computational cycles and the degree of load imbalance while exhibiting better working performance [22]. Both the genetic ant colony algorithm proposed by Cheng Cheng et al. [23] and the ACO focusing algorithm proposed by Skinderowicz Rafał [24] improved the ant colony algorithm to a new level and obtained better computational performance. Tang Bo et al. proposed the idea of applying genetic algorithms to the mapping process of grid blocks and processors and then performing intelligent allocation [2], but genetic algorithms have more space for optimization than other algorithms.

In the optimization process of a series of heuristic algorithms, although they also solve the load-balancing problem to a certain extent, they have the limitation of a single mechanism of finding the best and the tendency to fall into the local optimum. In this research, the proposed FaCO algorithm based on the fusion of firefly algorithm and ant colony algorithm with merit-seeking mechanism is free from the constraints of a single mechanism. Additionally, the previous heuristic algorithm is introduced to fuse the allocation mapping process of parallel computing, and the innovative FaCO algorithm is applied to solve the simple tsp graph structure after multi-level graph dissection. The load-balancing problem of massively parallel computing is solved with the goal of optimal allocation.

References

1. Kusmayadi, A.; Suyono, E.A.; Nagarajan, D.; Chang, J.S.; Yen, H.W. Application of computational fluid dynamics (CFD) on the raceway design for the cultivation of microalgae: A review. *J. Ind. Microbiol. Biotechnol.* 2020, 47, 373–382.
2. Tang, B.; Wang, Y. A novel task load balancing algorithm in the large-scale CFD with multi-zone structured grids. *Comput. Eng. Sci.* 2014, 36, 1213–1220.
3. Streng, M. *Load Balancing for Computational Fluid Dynamics Calculations*; Springer: Dordrecht, The Netherlands, 1996.
4. Ytterström, A. A Tool for Partitioning Structured Multiblock Meshes for Parallel Computational Mechanics. *Int. J. High Perform. Comput. Appl.* 1997, 11, 336–343.
5. Hendrickson, B.A.; Leland, R.W. A Multi-Level Algorithm for Partitioning Graphs. *Comput. Eng. Sci.* 2014, 36, 1213–1220.
6. Oh, B.S.; Cho, J.; Choi, B.; Choi, H.W.; Kim, M.S.; Lee, G. Application of heuristic algorithms for design optimization of industrial heat pump. *Int. J. Refrig.* 2022, 134, 1–15.
7. Castillon, L.F.; Bedriñana, M.F. Transmission Network Reconfiguration in Restoration Process Based on Constructive Heuristic Algorithms. *J. Control Autom. Electr. Syst.* 2022, 33, 929–938.

8. Spirov, A.V.; Myasnikova, E.M. Heuristic algorithms in evolutionary computation and modular organization of biological macromolecules: Applications to in vitro evolution. *PLoS ONE* 2022, 17, e0260497.
9. Zuo, L.; Shu, L.; Dong, S.; Zhu, C.; Hara, T. A multi-objective optimization scheduling method based on the ant colony algorithm in cloud computing. *IEEE Access* 2015, 3, 2687–2699.
10. Li, H.; Zhao, T.; Li, N.; Cai, Q.; Du, J. Feature Matching of Multi-view 3D Models Based on Hash Binary Encoding. *Neural Netw. World* 2017, 27, 95–105.
11. Li, H.; Liu, X.; Lai, L.; Cai, Q.; Du, J. An Area Weighted Surface Sampling Method for 3D Model Retrieval. *Chin. J. Electron.* 2014, 23, 484–488.
12. Li, H.; Zheng, Y.; Wu, X.; Cai, Q. 3D Model Generation and Reconstruction Using Conditional Generative Adversarial Network. *Int. J. Comput. Intell. Syst.* 2019, 12, 697–705.
13. Li, H.; Sun, L.; Dong, S.; Zhu, X.; Cai, Q.; Du, J. Efficient 3D Object Retrieval Based on Compact Views and Hamming Embedding. *IEEE Access* 2018, 6, 31854–31861.
14. Kernighan, B.W.; Lin, S. An efficient heuristic procedure for partitioning graphs. *Bell Syst. Tech. J.* 1970, 49, 291–307.
15. Hou, X.; Yang, C.; Liu, D. Hybrid load balancing algorithm based on osmotic artificial bee colony and ant colony optimization. *Appl. Res. Comput.* 2021, 38, 440–443.
16. Muteeh, A.; Sardaraz, M.; Tahir, M. MrLBA: Multi-resource load balancing algorithm for cloud computing using ant colony optimization. *Cluster Comput.* 2021, 24, 3135–3145.
17. Zeng, G.; Li, H.; Wang, X.; Li, N. Point cloud up-sampling network with multi-level spatial local feature aggregation. *Comput. Electr. Eng.* 2021, 94, 107337.
18. Li, H.; Zeng, G.; Cao, J.; Cai, Q. Multi-view-based siamese convolutional neural network for 3D object retrieval. *Comput. Electr. Eng.* 2019, 78, 11–21.
19. Kannan, K.S.; Sunitha, G.; Deepa, S.N.; Babu, D.V.; Avanija, J. A multi-objective load balancing and power minimization in cloud using bio-inspired algorithms. *Comput. Electr. Eng.* 2022, 102, 108225.
20. Li, J.; Han, Y. A hybrid multi-objective artificial bee colony algorithm for flexible task scheduling problems in cloud computing system. *Cluster Comput.* 2020, 23, 2483–2499.
21. Manasrah, A.M.; Hanan, B.A. Workflow Scheduling Using Hybrid GA-PSO Algorithm in Cloud Computing. *Wirel. Commun. Mob. Comput.* 2018, 2018, 1934784.
22. Tapale, M.T.; Goudar, R.H.; Birje, M.N.; Patil, R.S. Utility based load balancing using firefly algorithm in cloud. *J. Data Inf. Manag.* 2022, 2, 215–224.
23. Cheng, C.; Xu, Y.; Daniels, G. Efficient Management and Application of Human Resources Based on Genetic Ant Colony Algorithm. *J. Sens.* 2022, 2022, 9903319.
24. Skinderowicz, R. Improving Ant Colony Optimization efficiency for solving large TSP instances. *Appl. Soft Comput.* 2022, 120, 108653.

Retrieved from <https://encyclopedia.pub/entry/history/show/77626>