A DEA-GA multi-objective scheduling algorithm for Chip-Multiprocessor

Song Chai[1, a], Yubai Li[2,b], Chang Wu[3,c] and Jian Wang

[1-4] School of Communication and Information Engineering, University of Electronic Science and Technology of China, Chengdu, China

[a]s.tschai@gmail.com, [b]ybli@uestc.edu.cn, [c]changwu@uestc.edu.cn, [d]wangjian3630@uestc.edu.cn

Keywords: DEA, efficiency, Genetic Algorithm, multi-objective, scheduling, Chip Multiprocessor.

Abstract. In this paper, a Data Envelopment analysis based Genetic Algorithm (DEA-GA) is proposed for multi-objective scheduling on Chip-Multiprocessor. The proposal adopts modified GA as the searching heuristic to explore the solution space, and the fitness of each individual (schedule) is evaluated using the DEA approach. Three of the schedule metrics, namely makespan, energy and load balance are used to construct the multi-input multi-output Decision Making Units in the DEA, and the BCC super efficiency of each schedule is calculated. In the modified genetic algorithm, the metapopulation is divided into three subpopulations each optimizing a single metric. The top performance individuals in each subpopulation are then regrouped and applied DEA evaluation. Comparing to other multi-objective scheduling algorithm in simulations, our proposal always produces more efficient schedule solutions.

Introduction

The task scheduling problem on Chip Multiprocessors (CMPs) has been a research hotspot in recent years [1]. Current scheduling algorithm always performs optimization on multiple metrics, such as makespan [2], energy [3], workload balance [4] and etc. Some of these optimizations are even in conflict with each other, like makespan and energy optimization [5]. In order to make sure that the scheduling algorithm makes the ‘right’ tradeoff between the observed metrics; we introduce the Data Envelopment Analysis (DEA) to the algorithm design, and propose a DEA-based Genetic Algorithm (GA) for the real-time task scheduling problem on the CMP.

DEA is a non-parametric analytic method for measuring the relative efficiency of Decision Making Units (DMUs) [6]. The target objects, which in our case are the schedule solutions, are modeled as multi-input multi-output DMU, and the efficiency of each DMU, which representing the performance of the DMU, are calculated using the weighted sum of all the outputs divided by the weight sum of it inputs. The essence of DEA is that it allows the DMU to choose a set of weight coefficients that favors itself, under the constraint that the efficiencies of all DMUs calculated by this set of coefficients are not exceed one.

In this paper, we focus on the real-time task scheduling problem on CMP and propose a multi-metric scheduling algorithm using both super efficiency analyses in DEA and GA technique.

The rest of this paper is organized as follow: Section 2 summarizes the related works of multi-objective scheduling algorithms on CMP and the DEA evaluation; our proposal is presented in Section 3; the simulations and results are given in Section 4; Section 5 concludes the paper.

Related Work

The multi-objective scheduling algorithms have been widely researched in recent years. In [7], a modified algorithm which combines bacteriological algorithm and genetic algorithm is proposed to maximize the system reliability and reduce makespan. A two-phase cellular genetic scheduling algorithm is proposed in [8] to reduce both energy consumption and makespan. An NSGA-II based schedule algorithm is proposed in [9] to simultaneously optimize makespan and workload balance. In

[10], the problem of joint optimization of performance, energy, and temperature is addressed, and multi-objective evolutionary algorithm (MOEA) based schedule heuristic is proposed.

Proposed Algorithm

DEA evaluation of schedule solutions. The multi-input multi output DMU model is constructed using schedule metrics. The observed metric in our proposal are makespan, energy consumed and workload balance.

Makespan (M), or the schedule length, is the time length of the CMP finishing all the tasks.

The *energy metric* (E), is the amount of energy consumed during the execution of the tasks.

The last metric, *workload balance* (B), is defined to be the inverse coefficient of variant of the total workload on each processor. The larger metric value suggests better balanced schedule.

$$balance = \frac{ave\_load}{\left(\sum_{n=1}^{prc\_num}(load\_prc(n) - ave\_load)^2\right)^{\frac{1}{2}}} \tag{1}$$

The construction of the DMU model is the classification of these metrics as input or output, and follows a simple rule of thumb [11]: If the metrics is larger-the-better, then it is an output of DMU; otherwise it is an input. The result of this classification is $x = (M\ E)^T$ are the inputs, and $y = B$ is the output.

The BCC efficiency of a schedule γ_i in a schedule set Γ is the optimal value of the following *Linear Programing* (LP):

$$\begin{aligned}&\max_{u,v,u_0}\ Z = u^T \cdot y_i - u_0\\ &s.t\begin{cases} v \cdot x_i = 1\\ v \cdot x_l - u \cdot y_l + u_0 \ge 0,\ l = 1,2,...,n\\ v,u \ge \varepsilon\end{cases}\end{aligned} \tag{2}$$

Where $x_i = (M_i\ E_i)^T$ and $y_i = B_i$ is the input vector and output scalar of γ_i ; *n* is the number of schedules in Γ; *u* and *v* is the output and input coefficients; ε is a non-Archimedean infinitesimal constant; and θ is the efficiency of γ_i; u_0 is a scalar; and *Z* is the BCC efficient value of γ_i.

The BCC super efficiency is derived by relaxing the constraint of current DMU's efficient does not exceed 1 in the second constraint in (2). The BCC super efficiency is defined to be the optimal value of LP in (3).

$$\begin{aligned}&\max_{u,v,u_0}\ Z = u^T \cdot y_i - u_0\\ &s.t\begin{cases} v \cdot x_i = 1\\ v \cdot x_l - u \cdot y_l + u_0 \ge 0,\ l = 1,2,...,n,\ l \ne i\\ v,u \ge \varepsilon\end{cases}\end{aligned} \tag{3}$$

One inherent drawback of super efficiency analysis is that LP in (3) may have infeasible solution [12]. This happens when the target DMU has the largest and unique output in one of its outputs, which in our case, the schedule with the largest and unique *Workload Balance* metric. Our solution to this problem is when the infeasible schedule happens, the BCC efficiency of the schedule is used to replace the infeasible super efficiency.

DEA-GA scheduling algorithm. In our proposal, GA is adopted as the searching heuristic to explorer the solution space, and the fitness evaluation process in GA is finished using BCC super efficiency analysis. The main problem of incorporate GA with DEA is the high computational complexity. For example, in a genetic algorithm with 1000 population, the computation requirement of calculating the BCC super efficiency for a single individual (schedule) is solving a LP in (3) with

$n = 1000$, which has 3001 variables and 1000 constraints. The evaluation of all the individuals in each generation is solving 1000 such LPs.

Our solution to reduce the computational complexity is to divide the original population (called the *metapopulation*) into three separated subpopulations, and each of these subpopulations operates as an individual GA to optimize one metrics, as shown in Fig. 1.

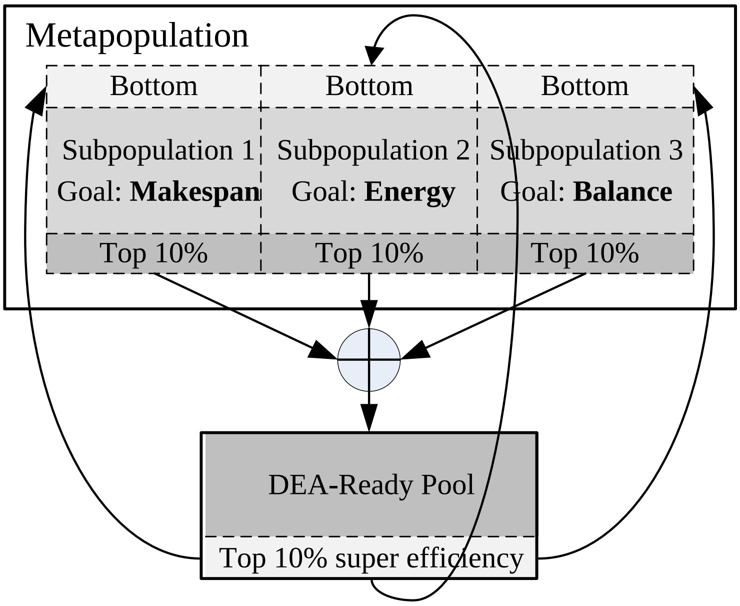


Figure 1. The population structure of DEA-GA

After calculating the metrics for every individual in the *metapopulation*, each subpopulation ranks the individual according to its own optimization goal. Then the top 10% individuals in each subpopulations are selected and regrouped into a DEA-ready pool. Then the BCC super efficiency analysis is performed for each individual in the DEA-ready pool, and all the individuals in the pool are sorted according to their super efficiency values. Top 10% individuals in the pool are duplicated to the three subpopulations to replace the same amount of individuals in the bottom, and then each subpopulation carries on its own evolving. The basic framework of GA is based on the work in [13].

Simulation Results

Simulation Setup. In this section, randomly generated task sets are used to test the performance of our proposal.In the simulation, the processor number is set to 8 and 12, and 1000 task sets are generated and simulated for each processor number using both our algorithm and three other multi-objective GAs. The task number of the generated task sets varies from 40 to 100.

Our DEA-GA scheduling algorithm is implemented under C, and the LPs are solved using GLPK [14]. Three GA scheduling algorithms with different multi-objective functions are also implemented under C as reference, and they are Multiplication-Division method, Weighted Sum method and Weighted Exponential Sum method based on global criterion [15]. The fitness functions are as follow:

$$fitness_{MD} = M^{-1} \cdot E^{-1} \cdot B$$

$$fitness_{WS} = \frac{1}{3}M^{-1} + \frac{1}{3}E^{-1} + \frac{1}{3}B$$

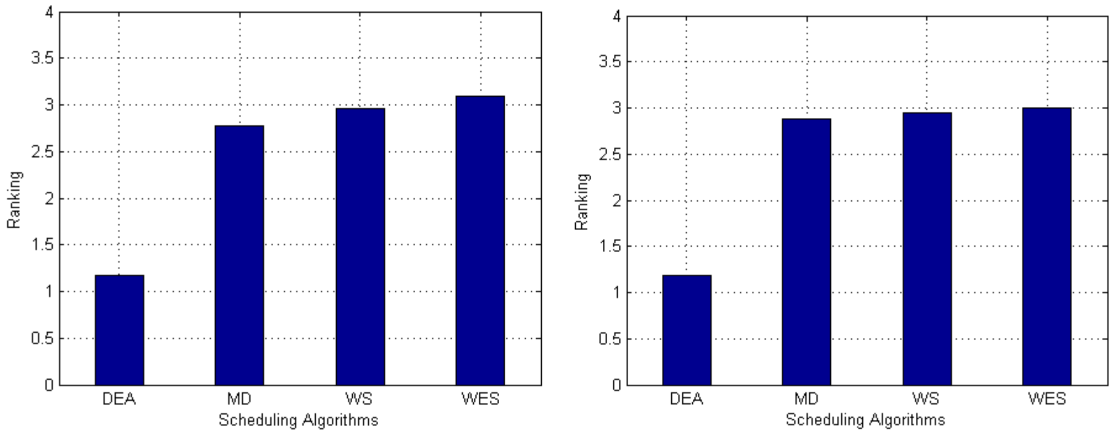
$$fitness_{WES} = \frac{1}{3}M^{-2} + \frac{1}{3}E^{-2} + \frac{1}{3}B^{2}$$

Comparative Results. The simulation results are grouped for each task set. Then the BCC Super Efficiency (SE) is calculated for each group, and four scheduling algorithms are ranked according to their efficiency values. The higher SE value means more efficient scheduling algorithm. Table 1 shows a sample simulation result of our simulation. The unit of Makespan is cycles, and Energy is measured in the unit energy, which is the amount of energy consumed in a cycle by the processor executing at the full speed.

Table 1. An example of simulation results

Algorithms	Makespan	Energy	Balance	SE	Rank
DEA-GA	423	28876	28.47	1.013	1
MD-GA	441	292560	17.52	0.987	3
WS-GA	448	29280	23.72	0.986	4
WES-GA	428	29078	30.97	1	2

The ranking of four scheduling algorithms in the 2000 simulation scenes are calculated, and the average ranking of each scheduling algorithm is shown in Figure 2.



(a) processor number = 8 (b) processor number = 12

Figure 2. Average ranking of DEA-, MD-, WS- and WES-GA.

Figure 2 (a) and (b) shows the average ranking of DEA-GA, MD-GA, WS-GA and WES-GA under processor number of 8 and 12, respectively. As shown in the figure, MD-GA, WS-GA and WES-GA have the same level of ranking which is around 2.8~3.1. On the other hand, the average ranking of DEA-GA is 1.17 in Figure 2 (a) and 1.18 in Figure 2 (b), which is remarkable better than the rest of the algorithms. That means the schedules generated by our proposal are always the most efficient ones among all the schedules produced.

Conclusion

In this paper, a DEA-GA is proposed for multi-objective scheduling on Chip-Multiprocessor. The proposal adopts modified GA as the searching heuristic to explore the solution space, and the fitness is evaluated using the DEA approach. By comparing to other multi-objective scheduling algorithm in simulations, our proposal always produces more efficient schedule solutions.

References

[1] Sarkar A., Mueller F., Ramaprasad H., Predictable task migration for locked caches in multi-core systems, ACM SIGPLAN Notices, ACM. p. 131 (2011).

[2] Kong F., Guan N., Deng Q., Yi W., Energy-efficient scheduling for parallel real-time tasks based on level-packing, Proceedings of the 2011 ACM Symposium on Applied Computing, ACM. p. 635 (2011).

[3] Wu G., Xu Z., Xia Q., Ren J., An Energy-Aware Multi-Core Scheduler based on Generalized Tit-For-Tat Cooperative Game, in: Journal of Computers Vol.7 (2012), p.106.

[4] Chen Q., Chen Y., Huang Z., Guo M., Wats: Workload-aware task scheduling in asymmetric multi-core architectures, Parallel & Distributed Processing Symposium (IPDPS), 2012 IEEE 26th International, IEEE. p. 249 (2012).

[5] Ahmad I., Arora R., White D., Metsis V., Ingram R., Energy-Constrained Scheduling of DAGs on Multi-core Processors, in: Contemporary Computing (2009), p.592.

[6] Charnes A, in: Data envelopment analysis: theory, methodology, and application, Kluwer Academic Pub(1994).

[7] Sathappan O., Chitra P., Venkatesh P., Prabhu M., Modified genetic algorithm for multiobjective task scheduling on heterogeneous computing system, in: International Journal of Information Technology, Communications and Convergence Vol.1 (2011), p.146.

[8] Pinel F., Dorronsoro B., Pecero J.E., Bouvry P., Khan S.U., A two-phase heuristic for the energy-efficient scheduling of independent tasks on computational grids, in: Cluster Computing (2012), p.1.

[9] Abad J.M.N., Shekofteh S.K., Tabatabaee H., Mehrnejad M., CoreIIScheduler: Scheduling Tasks in a Multi-core-Based Grid Using NSGA-II Technique, in: Intelligent Informatics (2013), p.507.

[10] Sheikh H.F., Ahmad I., Simultaneous optimization of performance, energy and temperature for DAG scheduling in multi-core processors, Green Computing Conference (IGCC), 2012 International, IEEE. p. 1 (2012).

[11] Ruiz-Torres A.J., López F.J., Using the FDH formulation of DEA to evaluate a multi-criteria problem in parallel machine scheduling, in: Computers & Industrial Engineering Vol.47 (2004), p.107.

[12] Seiford L.M., Zhu J., Infeasibility of super-efficiency data envelopment analysis models, in: Infor-Information Systems and Operational Research Vol.37 (1999), p.174.

[13] Sinnen O., Sousa L.A., Sandnes F.E., Toward a realistic task scheduling model, in: Parallel and Distributed Systems, IEEE Transactions on Vol.17 (2006), p.263.

[14] Makhorin A., GLPK (GNU linear programming kit) (2006).

[15] Marler R.T., Arora J.S., Survey of multi-objective optimization methods for engineering, in: Structural and multidisciplinary optimization Vol.26 (2004), p.369.