Mapping in heterogeneous systems with heuristical methods

Juan-Pedro Martínez-Gallar[1], Francisco Almeida[2], and Domingo Giménez[3]

[1] Departamento de Estadística, Matemáticas e Informática
Universidad Miguel Hernández, 03202, Alicante, Spain
jp.martinez@uhm.es
[2] EIOC-ULL. Edificio Física-Matemáticas
38271, La Laguna, Spain
falmeida@ull.es
[3] Universidad de Murcia, Departamento de Informática y Sistemas
30071 Murcia, Spain
domingo@dif.um.es

Abstract. During recent years a large number of parallel routines and libraries have been developed. Those routines were conceived for homogeneous systems. Thanks to the evolution of technology, now it is quite usual to have heterogeneous systems. Thus, these routines and libraries need to be adapted to the new environment. There are at least two options. The routines could be rewritten, but this means spending too much time and money. Alternatively, the processes of a homogeneous routine can be mapped into the processors in the heterogeneous system. For that pourpose, the development efficient mapping techniques is necessary. Our approach to obtain a satisfactory mappings consists of modelling the execution time of the parallel routine, and obtaining the mapping that gives the minimum modelled execution time. Exact solutions to this problem are very time consumming. As an alternative, we are researching the application of heuristic techniques (scatter search, tabu search, ...) to solve this problem and we will show how they can be applied to parallel iterative schemes.

1 The problem

In the field of parallel computing, assignation problems must be solved to assign processes to processors in order to obtain a mapping with which a reduced execution time is obtained. The assignation problem considered here is the one in which homogeneous processes are assigned to a parallel system, that may be homogeneous or heterogenous. In the case of a heterogeneous system, the type of algorithms to work with is some times called HeHo [1] (Heterogeneous assignation of processes which have homogeneous assignation of data). The general assignation problem is widely known to be NP-complete [2, 3]. Thus, efficient and optimal solutions have been found only for some particular assignation problems [4–6]. For other problems approximate solutions are obtained with heuristic methods [2, 7–9].

In previous works, the assignation of processes of a homogeneous program to a heterogeneous system was studied for a parallel dynamic programming scheme [10] and for dense linear algebra factorizations [11]. The problem is solved by searching through an assignation tree which includes all the possible processes to processors mappings. In this paper we propose to search through the tree with Scatter Search.

2 The assignation tree

To obtain the mapping, an exact model of the execution time of the routine must be developed, and a search is performed in a tree representing the mappings. Each node in the tree represents one of the possible assignations, and it has the corresponding modelled execution time associated [10]. The search through the tree is made with backtracking or branch and bound techniques, which consume a large amount of time, even when strategies to prune nodes in the tree are applied. Thus, the method does not scale well and it is only valid for small systems. For large systems, approximation methods (greedy) can be used to obtain the mapping in a reduced time, but the assignation obtained is far from the optimum in some cases. This can be seen in the folloging table, where the times to decide the mapping (mapping time) using an exact (backtracking) and a greedy method are compared. The table also showns the execution time (exec. time) obtained with these mappings, in a system with 20 processors, and in a simulated configuration with 40 processors.

method	System, 20 proc.		Simulation, 40 proc.	
	mapping time	exec. time	mapping time	exec. time
backtracking	0.122	2.96	111.369	5.07
greedy	0.001	5.09	0.009	5.72

The theoretical model requires a precise knowledge of system parameters like basic arithmetic operations time, or sending and start-up times of a communication. Frequently it is very difficult to find an equation to represent the problem.

We consider parallel iterative schemes, that can be used in the solution of a large variety of problems: dynamic programming, graph algorithms, genetic algorithms, iterative methods for the solution of linear systems, Fox and Cannon methods for matrix multiplication, Jacobi's relaxation, ... Additionally, the method can be applied to other problems and schemes, as for example linear algebra factorizations [11].

The case in which the data to work with are assigned to the different processes by blocks of equal size is considered. Each iteration consists of a computational part, and a communication step appears between two consecutive iterations. The total execution time is the addition of the execution times of the different iterations. Thus, it suffices to estimate the execution time of only one iteration, that has a theoretical execution time of the form:

$$t(s, D) = t_c t_{comp}(s, D) + t_{comm}(s, D) \tag{1}$$

where s represents the problem size, D the number of processes used in the solution to the problem, t_c the cost of a basic arithmetic operation, t_{comp} the number of arithmetic basic operations, and t_{comm} the communication cost (it includes a term with t_s, start-up time, and another with t_w, word-sending time).

3 Heuristic approaches

The previous techniques are either not fast enough or not efficient enough to get a satisfactory mapping. As an alternative, heuristics techniques sucessfully applied to solve other hard optimization problems [12–14] have been studied.

One scheme of the Scatter Search technique is:

```
generate initial population
while convergence not reached
    select elements to be combined
    combine selected elements
    include the most promising elements in the population
    include in the population the elements most scattered with respect
        to the most promising elements
endwhile
```

In the case in question this technique has significant problems. Fixing the number of processes to be used previously leads to the problem being solved more easily than when this is not done. However, the number of possible assignations is limited. The structure to represent the population is an array $D = (d_1, d_2, \ldots, d_P)$, with P the number of processors in the system and d_i the number of processes assigned to processor i. Because scatter search is based in random decisions, two mappings are combined by assigning more probability to processors with more computational capacity. The individuals with the lowest modelled execution time are included in the population, and those with the greatest distance function to the best individuals are also included. We are working to tune all these aspect to our mapping problem. Results for several iterative schemes will be reported.

4 Conclusions and Future Work

The application of heuristics techniques to the processes to processors mapping problem has been studied. The technique is based on the exploration of an assignation tree where each node has a theoretical execution time associated. Heuristic techniques allow us to obtain a satisfactory mapping in a reasonable time, but some parameters of the basic heuristic search must be tuned.

In the future we plan to apply other heuristic techniques (tabu search, simulated annealing, genetic algorithms, ...) to the same problem, and to study the application of the method to other parallel algorithmic schemes.

Acknowledgment

This work has been funded in part by the Spanish MCYT and FEDER under Grants TIC2003-08238-C02-02 and TIN2005-09037-C02-01.

References

1. A. Kalinov and A. Lastovetsky, Heterogeneous distribution of computations solving linear algebra problems on networks of heterogenous computers, *Journal of Parallel and Distributed Computing* **61** (2001) 520-535.
2. W. Zhao and K. Ramamritham, Simple and Integrated Heuristic Algorithms for Scheduling Tasks with Time and Resource Constrains, *The Journal of Systems and Software* **7** (3) (1987) 195-205.
3. H. Lennerstad and L. Lundberg, Optimal Scheduling Results for Parallel Computing, *SIAM News* (1994) 16-18.
4. C. Banino, O. Beaumont, L. Carter, J. Ferrante, A. Legrand and Y. Robert, Scheduling Strategies for Master-Slave Tasking on Heterogeneous Processor Platforms, *IEEE Transactions on Parallel and Distributed Systems* **15** (4) (2004) 319-330.
5. A. Legrand, H. Renard, Y. Robert and F. Vivien, Mapping and Load-Balancing Iterative Computations, *IEEE Transactions on Parallel and Distributed Systems* **15** (6) (2004) 546-558.
6. F. Almeida, D. Gonzalez and L Moreno, Master-Slave Paradigm on Heterogeneous Systems: A Dynamic Programming Approach for the Optimal Mapping. 12th Euromicro Workshop on Parallel, Distributed and Network-Based Processing (2004).
7. S. Fujita, M. Masukawa and S. Tagashira, A Fast Branch-and-Bound Scheme for the Multiprocessor Scheduling Problem with Communication Time, in: *Proceedings of the 2003 International Conference on Parallel Processing Workshop (ICPPW'03)* (Kaohsiung, Taiwan, 2003) 104-111.
8. G. Sabin, R. Kettimuthu, A. Rajan and P. Sadayappan, Scheduling of Parallel Jobs in a Heterogeneous Multi-Site Environment, in: D.G. Feitelson, L. Rudolph and U. Schwiegelshohn, eds., *Job Scheduling Strategies for Parallel Processing, 9th International Workshop* (Seattle, WA, 2003), Lecture Notes in Computer Science, vol. 2862, 87-104.
9. F. Almeida, D. González, L. Moreno and C. Rodríguez, Pipelines on heterogeneous systems: models and tools, in: *Concurrency - Practice and Experience, 17, 9, 2005.*
10. J. Cuenca, D. Giménez and J. P. Martínez, Heuristics for work distribution of a homogeneous parallel dynamic programming scheme on heterogeneous systems, *Parallel Computing* **31** (2005) 771-735.
11. J. Cuenca, L.-P. García, D. Giménez and J. Dongarra, Heuristics for work distribution of a homogeneous parallel dynamic programming scheme on heterogeneous systems, in: *Proceedings of the 2005 International Conference on Cluster Computing, HeteroPar Workshop* (Boston, 2005).
12. Juraj Hromkovič. *Algorithmics for Hard Problems*, Second Edition, Springer, 2003.
13. M. Pérez, F. Almeida and J.M. Moreno-Vega, On the Use of Path Relinking for the p-Hub Median Problem, in: *Lecture Notes in Computer Science 3004* (2004) 155-164.
14. Dréo, Pétrowski, Siarry, Taillard. *Metaheuristics for Hard Optimization*, Springer, 2005