Performance of Zinder-Roper algorithm for unitary RCPSP with constant precedence latencies *

Abir Benabid[1,2], Claire Hanen[2,3]

[1] Laboratoire LIP6, 104 av Kennedy 75016 Paris
[2] University Pierre and Marie Curie Paris
[3] University Paris Ouest Nanterre la Défense
Abir.Benabid@lip6.fr, Claire.Hanen@lip6.fr

Mots-Clés : *s* cheduling theory, resource constraints, precedence latencies, maximum lateness, worst-case analysis.

Scheduling problems on multiprocessors systems with precedence constraints are among the most difficult problems, in particular for the design of good approximation algorithms. Until now, much work has been done considering a model with identical processors and unit execution time tasks subject to precedence relations. However, one major architectural feature on the modern microprocessors is the employment of multiple specialized and pipelined functional units. Each functional unit executes only a specific type of tasks, and it is pipelined, it can start a new task every time unit, although the whole computation of a task takes a time which might be greater than one.

In order to model such realistic situations, we define the unitary RCPSP. In the three-field notation (see for example [2]), the above problem, with k types of processors is denoted by

$$\Sigma^k P|prec, \delta_{ij}, p_i = 1|L_{max}$$

where the terms *prec* and δ_{ij} indicate the presence of precedence latencies, associating with each precedence constraint a constant amount of time which must elapse between the completion and start times of the corresponding tasks. This instance has a set of n tasks $\mathcal{T}$. Each task $i \in \mathcal{T}$ requires one unit of processor's time and it has a due-date d_i and a boolean type vector $\{b_r\}_{1\leq r\leq k}$ where $b_r = 1$ when i is processed on processor of type r.

A schedule σ assigns a completion time $C_i(\sigma)$ to each task i. The maximum lateness of σ is defined as follows

$$L_{max}(\sigma) = \max_{i\in\mathcal{T}} C_i^\sigma - d_i$$

A well-studied special case is the makespan problem on identical processors. In fact, if all due dates are equal to zero and all tasks and all processors have the same type, then the maximum lateness problem becomes the makespan problem on identical processors

$$P|prec, \delta_{ij}, p_i = 1|C_{max}$$

with the criterion $C_{max}(\sigma) = \max_{i\in\mathcal{T}} C_i(\sigma)$.

* *ORCYAE project supported by awards from DIGITEO, a research park in Ile-de-France dedicated to information science and technology.*

It is well known that even $P|prec, p_j = 1|C_{max}$ is NP-hard in the strong sense [5], [4]. Moreover, as has been shown in [6], the $P|prec, p_j = 1|C_{max}$ problem remains NP-hard in the strong sense even if the partially ordered set of tasks is a bipartite graph. These NP-hardness results boost the interest in the worst-case performance of various approximation algorithms for this problem [2],[1], [6], [7].

When due-dates are involved, and maximum lateness is optimized, the first version of Garey-Johnson algorithm solves optimally the problem when $m = 2$, $\delta = 0$. In [3], the generalization of this algorithm to unitary RCPSP with constant latency δ leads to the following worst case performance :

$$L^{GJ}_{max} \leq \left(k + 1 - \frac{\alpha(m_x)}{\delta + 1}\right) L^{opt}_{max} + \left(k - \frac{\alpha(m_x)}{\delta + 1}\right) d_{max} - \frac{\alpha(m_x)}{\delta + 1}$$

with $d_{max} = \max_{i \in \mathcal{T}} d_i$, $m_x = \max_{1 \leq r \leq k} m_r$ and

$$\alpha(m_x) = \begin{cases} \frac{2}{m_x} & for\ even\ m_x \\ \frac{2}{m_x+1} & for\ odd\ m_x \end{cases} .$$

The best known performance guarantee for the L_{max} on parallel processors without latencies is given by [8] :

$$L^{ZR}_{max} \leq \left(2 - \frac{2}{m}\right) L^{opt}_{max} + \left(1 - \frac{2}{m}\right) d_{max}$$

In this presentation, we study the Zinder-Roper extension for the maximum lateness minimization on unitary RCPSP with precedence latencies and we prove that the performance guarantee obtained,

$$L^{ZR}_{max} \leq \left(k + 1 - \frac{2}{m_x(\delta + 1)}\right) L^{opt}_{max} + \left(k - \frac{2}{m_x(\delta + 1)}\right) d_{max} - 1 + \frac{3}{m_x(\delta + 1)}$$

with $\delta = \max_{(i,j) \in G} \delta_{ij}$, outperforms the bounds for general list schedules, and coincides with the Zinder-Roper bound for parallel processors without latencies.

Références

[1] B. Braschi and D. Trystram. A new insight into the coffman-graham algorithm. *SIAM Journal on Computing*, 23 :662–669, 1994.

[2] P. Brucker. *Scheduling algorithms*. Springer, 2001.

[3] C. Hanen and A. Benabid. Performance of garey-johnson algorithm for pipelined typed tasks systems. In *ALIO-EURO conference on combinatorial optimization, Buenos Aires Argentine*, 2008.

[4] J.K. Lenstra and A.H.G. Rinnooy Kan (1978). Complexity of scheduling under precedence constraints. *Operations Research*, 26 :22–35, 1978.

[5] J.D. Ullman. Np-complete scheduling problems. *Journal Comput. System Sci.*, 10 :384–393, 1975.

[6] Y.Zinder and D. Roper. A minimax combinatorial optimisation problem on an acyclic directed graph : polynomial-time algorithms and complexity. *in : Proceedings of the A.C. Aitken Centenary Conference, Dunedin*, pages 391–400, 1995.

[7] Y. Zinder. An iterative algorithm for scheduling uet tasks with due dates and release times,. *European Journal of Operational Research*, 149 :404–416, 2003.

[8] Y. Zinder and D. Roper. An iterative algorithm for scheduling unit-time operations with precedence constraints to minimise the maximum lateness. *Annals of Operations Research*, 81 :321–340, 1998.