



HAL
open science

A heap of pieces model for the cyclic job shop problem

Laurent Houssin

► **To cite this version:**

Laurent Houssin. A heap of pieces model for the cyclic job shop problem. International Conference on Project Management and Scheduling, Mar 2014, Munich, Germany. pp.112-115. hal-01026514

HAL Id: hal-01026514

<https://hal.science/hal-01026514v1>

Submitted on 30 Nov 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A heap of pieces model for the cyclic job shop problem

Laurent Houssin¹

CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France
 Univ de Toulouse, UPS, LAAS, F-31400 Toulouse, France
 laurent.houssin@laas.fr

Keywords: cyclic scheduling, max-plus algebra.

1 Introduction

In classical scheduling, a set of tasks is executed once while the determined schedule optimizes objective functions such as the makespan or earliness-tardiness. In contrast, cyclic scheduling means performing a set of generic tasks infinitely often while minimizing the time between two occurrences of the same task. Cyclic scheduling has several applications, e.g. in robotic industry (see Kats and Levner [1997]), in manufacturing systems (see Pinedo [2005] and Hillion and Proth [1989]) or multiprocessor computing (see Hanen and Munier [1995b]). It has been studied from multiple perspectives, since there exist several possible representations of the problem such as graph theory, mixed integer linear programming, Petri nets or $(max, +)$ algebra. An overview about cyclic scheduling problems and the different approaches can be found in Hanen and Munier [1995a].

However, a less known tool is the *heaps of pieces* approach. A first paper considering this modelling for cyclic scheduling is Gaubert and Mairesse [1999]. The authors introduce a method to represent safe Petri nets as particular automata which compute the height of heaps of pieces. This approach enables to compute efficiently the throughput of a cyclic schedule. In this paper, we use the heaps of pieces approach for solving cyclic scheduling problems with resource constraints.

2 Cyclic scheduling problems

The basic cyclic scheduling problem (BCS) involves generic tasks and precedence constraints between tasks but no resource constraints are considered. In this problem, a set of n generic operations are processed in parallel by an unbounded number of machines and there is a set of q precedence constraints. We denote this set $\mathcal{A} = \{a_1, \dots, a_q\}$ where a_l corresponds to a constraint represented by a triple (i, j, h) . More precisely, the *uniform constraint* (i, j, h) means that

$$s_i(k) + p_i \leq s_j(k + h), \quad \forall k \geq 1$$

where s_i denotes the beginning of the operation i and p_i is the processing time of i . In this framework, the asymptotic cycle time $\alpha(S)$ is usually minimized (with S a feasible schedule). Equivalently we can aim at maximizing the throughput $r(S) = \frac{1}{\alpha(S)}$.

We can distinguish two categories of cyclic schedules: the 1-periodic schedules and the K -periodic schedules. The first category is characterized by a period α such that :

$$s_i(k + 1) = \alpha + s_i(k), \quad \forall i \in \mathcal{T}, \quad \forall k \geq 1,$$

where \mathcal{T} is the set of tasks to perform. Whereas K -periodic schedules are also defined by period which corresponds to a fixed interval time between any K consecutive occurrences of i :

$$s_i(k + K) = \alpha_K + s_i(k), \quad \forall i \in \mathcal{T}, \quad \forall k \geq 0.$$

For our concern, we are interested in the cyclic job shop problem. In this case, tasks are *a priori* mapped onto machines and the number of machines is smaller than the number of tasks to perform. More precisely, a cyclic job shop is defined by :

- a set \mathcal{T} of elementary tasks,
- a set \mathcal{R} of machines,
- for each task $t \in \mathcal{T}$, a processing time p_t and a machine $m_t \in \mathcal{R}$ on which the task has to be performed,
- a set \mathcal{A} of uniform constraints (as defined above),
- a set of jobs \mathcal{J} corresponding to a production sequence of elementary tasks. More precisely, a job J_1 defines a sequence $J_1 = t_{11} \dots t_{1k}$ to be executed in this order.

Previous studies of this problem have shown that K -periodic schedules are dominant (Hanan and Munier [1995a]) and the problem is NP-hard (Hanan [1994]) for throughput maximization.

One can find the description of two classical methods of performance evaluation of cyclic job shop in Hanan and Munier [1995a]. The first one is based on graph theory. In this framework, the throughput computation problem of a cyclic job shop schedule becomes the search of maximum circuit ratio in a graph. The second approach considers the job shop as a $(max, +)$ -linear system. Then, the spectrum of the evolution matrix of the system gives the cycle time. In both approaches, the complexity of the evaluation of the cycle time is the same (polynomial).

3 The heap of pieces model

We first recall some algebraic tools concerning $(max, +)$ algebra. The $(max, +)$ semiring is the set $\mathbb{R} \cup \{-\infty\}$ endowed with the max operator, written $a \oplus b = max(a, b)$, and the usual sum written $a \otimes b = a + b$. The sum (resp. product) admits a neutral element denoted $\varepsilon = -\infty$ (resp. $e = 0$), it leads to $a \oplus \varepsilon = a$ and $a \otimes e = a$. For matrices, additions and products give $(A \oplus B)_{ij} = A_{ij} \oplus B_{ij}$ and $(A \otimes B)_{ij} = \bigoplus_{k=1}^n A_{ik} \otimes B_{kj}$.

An exhaustive presentation of $(max, +)$ algebra can be found in Baccelli et al. [1992]. We now present the heap model structure that was introduced in Gaubert and Mairesse [1999].

Definition 1 (Heap model). *A heap model is composed by :*

- \mathcal{P} a finite set of pieces,
- \mathcal{S} a finite set of slots,
- R gives the subset of slots occupied by a piece,
- $l : \mathcal{P} \times \mathcal{S} \rightarrow \mathbb{R} \cup \{-\infty\}$ gives the lower contour of a piece,
- $u : \mathcal{P} \times \mathcal{S} \rightarrow \mathbb{R} \cup \{-\infty\}$ gives the upper contour of a piece.

For each piece p (possibly non connected) of a heap model, we define the matrix $\mathcal{M}(p)$ of dimension $|\mathcal{S}| \times |\mathcal{S}|$ by

$$\mathcal{M}(p)_{sr} = \begin{cases} 0 & \text{if } s = r, r \notin R(p), \\ u(p)_r - l(p)_r & \text{if } r \in R(p), s \in R(p), \\ \varepsilon & \text{otherwise.} \end{cases}$$

Calculus in the heap model are based on $(max, +)$ -algebra. More precisely, if a piece p_1 is piled up in an empty heap, the upper contour of the heap is given by $x(p_1) = I \otimes \mathcal{M}(p_1)$ where I is a $1 \times |\mathcal{S}|$ matrix defined by $I_j = e, \forall j \in \{1, \dots, |\mathcal{S}|\}$. In the same manner, the pile of two pieces p_1 and p_2 give the following upper contour of the heap : $x(p_1 p_2) = I \otimes \mathcal{M}(p_1) \otimes \mathcal{M}(p_2)$.

4 Application to cyclic job shop

In this section, we show how to model the cyclic job shop problem as a heap of pieces. We take the assumption that constraints $a_k \in \mathcal{A}$ are such that $h = 0$ (see §2). As shown in Gaubert and Mairesse [1999], the job shop problem admits a heap realization described under

$$\begin{aligned} \mathcal{P} &= \mathcal{T} \\ \mathcal{S} &= \mathcal{R} \cup \mathcal{J} \\ R(t) &= m(t) \cup \{\mathcal{J}_i | t \in \mathcal{J}_i\} \\ l(t, r) &= 0 \text{ if } m(t) = r, \quad l(t, r) = \varepsilon \text{ otherwise} \\ u(t, r) &= p(t) \text{ if } m(t) = r, \quad u(t, r) = \varepsilon \text{ otherwise.} \end{aligned}$$

In this framework, each elementary task t is represented by a matrix $\mathcal{M}(t)$. Considering the objective of the cyclic job shop problem, the problem is to find the sequence of pieces piled up in the heap that maximize the throughput. The following theorem indicates how to compute the cycle time

Theorem 1. *The throughput of a job J_i for the cyclic sequence $p_1 \dots p_n$ in a heap model representing a job shop system is given by*

$$\lambda_{J_i} = |p_1 \dots p_n|_{J_i} \otimes (\rho(\mathcal{M}(p_1) \otimes \dots \otimes \mathcal{M}(p_n)))^{-1}. \quad (1)$$

where $|p_1 \dots p_n|_{J_i}$ denotes the number of jobs J_i completed under the sequence $p_1 \dots p_n$ and $\rho(X)$ is the unique eigenvalue of the irreducible matrix X (see [Baccelli et al., 1992, chap.2]).

Proof. Considering the heap, the cyclic sequence $p_1 \dots p_n$ corresponds to a cyclic piled-up of these pieces. Let $v = p_1 \otimes \dots \otimes p_n$. The matrix $\mathcal{M}(v)$ can be seen as the matrix representing an unique piece $p_1 \dots p_n$. The evolution of the heap is now given by the pile up of v . As seen in section 3, the upper contour of the heap is given by $x(v \dots v) = I \otimes \mathcal{M}(v) \otimes \dots \otimes \mathcal{M}(v)$. Since the job shop is connected (and we assume it is), the matrix $\mathcal{M}(v)$ is irreducible. After a certain number of iterations (that is called the transient time), a cyclic behaviour appears for an irreducible matrix $X : X^{k+c} = \rho(X)^c \otimes X^k$ (see Cohen et al. [1983]). This property hold for $\mathcal{M}(v)$ and after a certain number of iterations, the growth rate of the heap become $\rho(\mathcal{M}(v))$ and the throughput is given by (1).

Considering the heap of pieces model, the cyclic job shop problem is equivalent to find a bounded sequence of pieces with the maximum throughput. Compared to classical approaches, this model takes benefit from K -cyclic schedules. Indeed, graph based approach have to consider $K \times n$ number of nodes. Regarding mixed integer linear programming, the number of variables grows substantially. To deal with this situation, the heap of pieces is the best model since the number of slots is independant of K . Nonetheless the drawback of this method is that we can only consider a work in progress of one and it can not represent a "nested" schedule since it can only represent pattern that can be piled up (no intertwine). We propose a branch and bound procedure to solve this problem. The branching rule chose the piece to pile up in the heap and the evaluation fonction is the cycle time computation of the pattern in the heap. We extend this model to consider several work in process inside the pattern (but not nested pattern) through an extension of the number of slots.

Bibliography

- F. Baccelli, G. Cohen, G. J. Olsder, and J. P. Quadrat. *Synchronization and Linearity*. Wiley, 1992.
- J. Cochet-Terrasson, G. Cohen, S. Gaubert, M. Mc Gettrick, and J. P. Quadrat. Numerical computation of spectral elements in max-plus algebra. In *Proceedings of the IFAC Conference on System Structure and Control*, Nantes, 1998.
- G. Cohen, D. Dubois, J. P. Quadrat, and M. Viot. Analyse du comportement périodique des systèmes de production par la théorie des diodes. Rapport de recherche 191, INRIA, Le Chesnay, France, 1983.
- Ali Dasdan and Rajesh K. Gupta. Faster maximum and minimum mean cycle algorithms for system-performance analysis. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 17(10), 1998.
- S. Gaubert and J. Mairesse. Modeling and analysis of timed petri nets using heap of pieces. *IEEE Trans. on Automatic Control*, 44(4), Apr. 1999.
- C. Hanen. Study of a np-hard cyclic scheduling problem: The recurrent job-shop. *European Journal of Operational Research*, 72(1):82 – 101, 1994.
- C. Hanen and A. Munier. *Scheduling Theory and Its Applications*, chapter Cyclic Scheduling on Parallel Processors: An Overview. Wiley, 1995a.
- C. Hanen and A. Munier. A study of the cyclic scheduling problem on parallel processors. *Discrete Applied Mathematics*, 57:167–192, 1995b.
- H. P. Hillion and J. M. Proth. Performance evaluation of job-shop systems using timed event graphs. *IEEE Transaction on Automatic Control*, 34(1):3–9, Jan. 1989.
- R. M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Math*, 23, 1978.
- V. Kats and E. Levner. A strongly polynomial algorithm for no-wait cyclic robotic flowshop scheduling. *Operations Research Letters*, 21:171–179, 1997.
- M. Pinedo. *Planning and Scheduling in Manufacturing and Services*. Springer, 2005.