Les cahiers Leibniz

Cyclic scheduling in robotic cells: about Agnetis' conjecture for the classical case

Nadia Brauner and Gerd Finke

Laboratoire Leibniz-IMAG, 46 av. Félix Viallet, 38000 GRENOBLE, France -
ISSN : 1298-020X

n° 120
Avr 2005

Site internet : http://www-leibniz.imag.fr

Cyclic scheduling in robotic cells: about Agnetis' conjecture for the classical case

Nadia Brauner and Gerd Finke *

April 13, 2005

Abstract

Robotic cells consists of a flow-shop with a circular layout and a single transporter, a robot, for the material handling. A single part is to be produced and the objective is to minimize the production rate. Different cell configurations have been studied, depending on the travel times of the empty robot: additive, constant or just triangular.

A k-cycle is a production cycle where exactly k parts enter and leave the system. Ideally, one would like to determine, for a given instance, an optimal k-cycle. Consider the set $S_{\mathcal{K}}$ of all k-cycles up to size $\mathcal{K}$ where $S_{\mathcal{K}}$ contains, for every instance, an optimal solution and $\mathcal{K}$ is minimal. The cycle function $\mathcal{K} = \mathcal{K}(config, m)$ depends on the cell configuration and the number of machines.

Some of these functions are known and there are conjectures about others. We give new results invalidating in particular the so-called Agnetis' Conjecture for the classical robotic cell configuration.

1 Robotic cells

Robotic flow-shops consist of m machines arranged in a circular layout and served by a single central robot [Figure 1]. It is known that the robotic scheduling problem is already NP-hard for a flowshop with $m \geq 3$ machines and two or more different part types [8]. It remains the interesting case of the m-machine robotic cell in which one wants to produce identical parts. Then the problem reduces to finding the optimal strategy for the robot moves in order to obtain the maximal throughput rate for this unique part. A survey on general robotic cells can be found in [6].

The m machines of a robotic cell are denoted by M_1, $M_2 \ldots M_m$ and we add two auxiliary machines, M_0 for the input station IN and M_{m+1} for the output station OUT. The raw material for the parts to be produced is available in unlimited quantity at M_0. The central robot can handle a single unit at a time. A part is picked up at M_0 and transferred in succession to M_1, $M_2 \ldots M_m$, where it is machined in this order until it finally reaches the output station M_{m+1}. At M_{m+1}, the finished parts can be stored in unlimited amounts. We focus on the classical case as in [11], where the machines M_1, $M_2 \ldots M_m$ are without

*{nadia.brauner, gerd.finke}@imag.fr. Laboratoire Leibniz-IMAG, 46, av. Félix Viallet 38031 GRENOBLE Cedex, France.

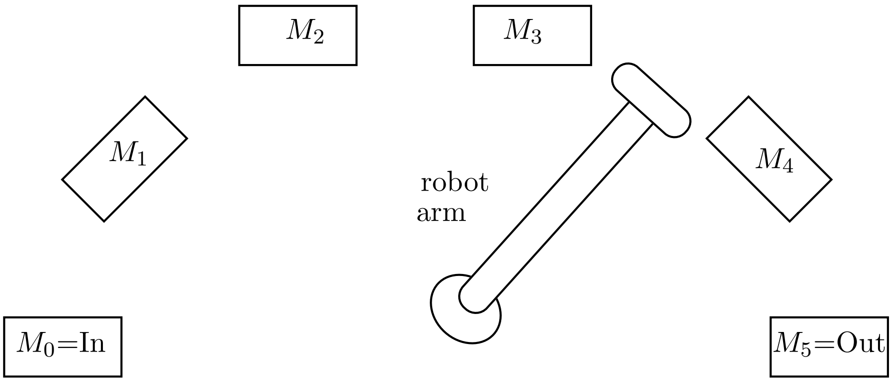

Figure 1: Robotic cell with $m = 4$ machines

buffer facility. In this case, the robot has to be empty whenever it wants to pick up a part at M_h (h = 0,1 ... m).

Consider an instance I of an m-machine robotic cell. The *processing time* p_h represents the minimum time a part must remain on machine M_h ($h = 1, 2 \ldots m$). Once the part is finished, two policies may apply. In the *no-wait* case, the part must be removed immediately from the machine after p_h time units and transferred to the following machine. In the *classical* case, the part can remain on the machine waiting for the robot. Let ϵ be the time to load a part onto a machine from the robot or to unload a part from a machine onto the robot. We shall consider two classical metrics for the travel times of the robot. For *additive* travel times [11], let δ be the time for the robot to travel (idle or loaded) between two consecutive machines. The travel times are additive. Hence, the trip of the idle robot from M_h to $M_{h'}$ ($h \neq h'$) takes $\delta_{hh'} = |h - h'|\delta$. For *constant* travel times [7], δ is the time for the robot to travel between any two machines M_h and $M_{h'}$: $\delta_{hh'} = \delta$.

We consider cyclic robot moves for the production process of the parts and define a k-*cycle* as a production cycle of exactly k parts. It can be described as a sequence of robot moves where exactly k parts enter the system at M_0, k parts leave the system at M_{m+1} and each time the robot executes the k-cycle, the system returns to the same state, *i.e.* the same machines are loaded, the same machines are empty and the robot returns to the starting position. To describe k-cycles we use the concept of *activities* [4]. The activity A_h ($h = 0, 1 \ldots m$) consists of the following sequence:

- The idle robot takes a part from M_h.
- The robot travels with this part from M_h to M_{h+1}.
- The robot loads this part onto M_{h+1}.

In [4], the authors characterize the k-cycles as follows: A k-cycle C_k is a sequence of activities, in which each activity occurs exactly k times and between two consecutive (in a cyclic sense) occurrences of A_h ($h = 1, 2 \ldots m - 1$) there is exactly one occurrence of A_{h-1} and exactly one occurrence of A_{h+1}.

A ρ-cycle C_ρ is *optimal* if it minimizes $\frac{T(C_k)}{k}$ over all possible k-cycles $k = 1, 2 \ldots$, where $T(C_k)$ denotes the cycle time of C_k. A set of cycles S is said to be *dominant* if, for any instance, there exists a cycle of S that is optimal.

2 Dominant sets of cycles

In [11] the authors conjectured that the 1-cycles are dominant. This conjecture is valid for classical two- and three-machine cells. However it is false for four-machine cells [3, 4, 7]. It has been replaced by the following conjecture:

Agnetis' Conjecture[1]: The set of k-cycles with $k \leq m-1$ is dominant.

Note that this conjecture was originally formulated by Agnetis for additive no-wait cells. Let $S_{\mathcal{K}}$ be the set of all k-cycles with $1 \leq k \leq \mathcal{K}$. We are interested in the minimal dominant set $S_{\mathcal{K}}$, *i.e.*, $S_{\mathcal{K}}$ is dominant and no $S_{\mathcal{K}'}$ is dominant with $\mathcal{K}' < \mathcal{K}$. We can expect that $\mathcal{K} = \mathcal{K}(m)$ is a function of the number of machines m. We denote $\mathcal{K}(m)$ by $\mathcal{K}_{nw}(m)$ in the no-wait case and by $\mathcal{K}_c(m)$ in the classical case. Agnetis' Conjecture claims that $\mathcal{K}_{nw}(m) = m-1$. In additive no-wait robotic cells, we know that

$$\mathcal{K}_{nw}(m) \begin{cases} = m-1 & \text{for } m = 2,3 \quad [1] \\ \geq m-1 & \text{for } m \geq 4 \quad [10] \end{cases}$$

In classical robotic cells (with additive or constant travel times), one has

$$\mathcal{K}_c(m) \begin{cases} = 1 & \text{for } m = 2,3 \\ \geq m & \text{for } m = 4 \end{cases}$$

For $m = 2$, and 3 this result was proven in [3, 4] for the additive case and in [7] in the constant case. It was shown in [3] that $\mathcal{K}_c(4) \geq 3$. We now strengthen this result to $\mathcal{K}_c(4) \geq 4$. We exhibit a cycle which proves that Agnetis' Conjecture is false in classical 4-machine robotic cells:

Proposition 1 *The 4-cycle* $C_4 = (A_0A_1A_0A_3A_4A_2A_1A_0A_3A_2A_1A_4A_3A_2A_0A_1A_4A_3A_4A_2)$ *strictly dominates all k-cycles for $k = 1,2,3$ for the following instance:*

$$m = 4; \qquad \delta = 1; \qquad \epsilon = 0; \qquad p_1 = 0; \qquad p_4 = 0;$$

in the additive case: $p_2 = 10;$ $p_3 = 10;$
in the constant case: $p_2 = 6;$ $p_3 = 6.$

Proof for the additive case

One has $\frac{T(C_4)}{4} = 15$. We shall prove that for all k-cycles C_k ($k = 1,2,3$), one has $\frac{T(C_k)}{k} > 15$.

$k = 1$: For the instance I, the best 1-cycle has cycle time 16.

$k = 2$: For the instance I, the 1-cycles dominate the 2-cycles (regular or equidistant case) [2, 3]. Therefore the best 2-cycle has cycle time greater or equal to 16.

$k = 3$: Let C_3 be a 3-cycle. Let m_i be the number of times the robot travels between machines M_i and M_{i+1} in both directions during one execution of C_3 and let $|S|$ be the number of occurrences of the sequence of activities S in C_3 and let $u_i = |A_{i-1}A_i|$. If the robot never makes any dummy moves, one has [3, 2]:

$$\begin{aligned} m_0 &= 2k \\ m_m &= 2k \\ m_1 &= 4k - 2u_1 \\ m_{m-1} &= 4k - 2u_m \\ m_2 &\geq 4k - 2u_2 - 2|A_1A_0A_2| \\ m_{m-2} &\geq 4k - 2u_{m-1} - 2|A_{m-2}A_mA_{m-1}| \end{aligned}$$

Moreover we know that the sequences $A_{i-1}A_i$ generate a waiting time of p_i and the sequence $A_1A_0A_2$ generates a waiting time of $\max(0, p_2 - 4\delta - 2\epsilon)$ and the sequence $A_{m-2}A_mA_{m-1}$ generates a waiting time of $\max(0, p_{m-1} - 4\delta - 2\epsilon)$. All those sequences do not interfere for the calculation of the waiting time. Therefore, one has

$$\begin{aligned} T(C_3) &= \sum_{i=0}^{4} m_i\delta + \text{waiting times} \\ &\geq 16k\delta + u_1(p_1 - 2\delta) + u_4(p_4 - 2\delta) + u_2(p_2 - \delta) + u_3(p_3 - \delta) \\ &\quad + |A_1A_0A_2|(\max(0, p_2 - 4\delta - 2\epsilon) - \delta) \\ &\quad + |A_2A_4A_3|(\max(0, p_3 - 4\delta - 2\epsilon) - \delta) \end{aligned}$$

and hence, for the instance I, the cycle time of C_3 satisfies

$$T(C_3) \geq 48 - 2u_1 - 2u_4 + 5|A_1A_0A_2| + 5|A_2A_4A_3| + 9u_2 + 9u_3 \tag{1}$$

Suppose $T(C_3) \leq \frac{T(C_4)}{4} \times 3 = 15 \times 3 = 45$. In C_3, between two consecutive occurrences of A_2, there is exactly one occurrence of A_1 and one occurrence of A_3. If somewhere A_3 happens before A_1 between two consecutive occurrences of A_2, then C_3 is of the form described on Figure 2. In this case, the cycle time of C_3 satisfies $T(C_3) \geq 54$ (contradiction).

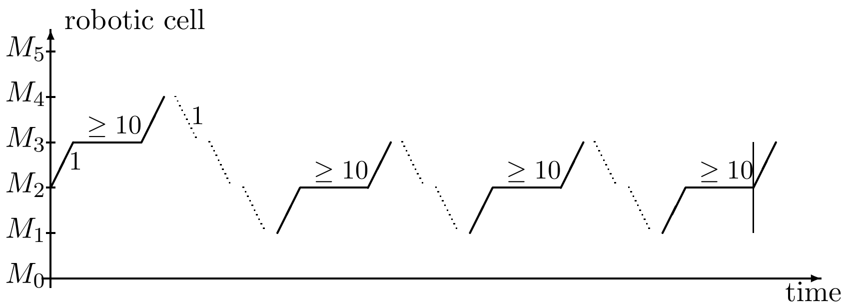

Figure 2: Position of the robot in the cell for the case with A_3 before A_1

Therefore, C_3 is of the form

$$C_3 = A_2 \cdots A_1 \cdots A_3 \cdots A_2 \cdots A_1 \cdots A_3 \cdots A_2 \cdots A_1 \cdots A_3 \ldots$$

We know that between A_1 and the following A_2 there are at least $p_2 = 10$ time units and between A_2 and the following A_3, there are at least $p_3 = 10$ time units.

From inequality (1), $T(C_3) \leq 45$ implies $2(u_1 + u_4) \geq 3$, which means $u_1 + u_4 \geq 2$ since u_1 and u_4 are integers.

Consider now what happens between two consecutive activities A_2: there is exactly one occurrence of activity A_1 and between the end of A_1 and the following A_2 the time spent is at least $p_2 = 10$. Moreover, between the end of A_2 and the end of the following A_1, the robot has to travel from machine M_3 to machine M_1 and then from M_1 to M_2 (execution of A_1). Therefore, the time spent between two consecutive occurrences of A_2 is at least 14 plus the time for the execution of the activities between A_2 and A_1 (at least 2 for each activity). The same reasoning applies for A_3: the time spent between two consecutive occurrences of A_2 is at least 14 plus the time for the execution of the activities between A_3 and A_2.

We know that the number u_1 of A_0A_1 plus the number u_4 of A_3A_4 is greater than two. If those two occurrences do not appear between the same consecutive A_2, then each generates an additive time of 2 and one has $T(C_3) \geq 14*3+2*2 = 46$ which leads to a contradiction. Therefore, C_3 is of the form:

$$C_3 = A_2A_1 \cdots A_3A_2A_0A_1 \underbrace{\cdots}_{S} A_3A_4A_2A_1 \cdots A_3$$

Indeed, no other activity can happen between A_2 and the following A_1 or between A_3 and the following A_2. Between two consecutive occurrences of A_1, there is exactly one occurrence of A_0. Therefore, the sequence S contains an occurrence of A_0. The same reasoning implies that S contains an occurrence of A_4. The sequences $A_2A_0A_1A_0A_4A_3A_4$ or $A_2A_0A_1A_4A_0A_3A_4$ imply a travel time of at least 20 before the next A_2. Since between the two other consecutive activities A_2 the total time is at least 14, one has $T(C_3) \geq 14*2+20 = 48$ which leads to a contradiction. Therefore, $T(C_3) > 45$ which is a contradiction. □

Proof in the constant case

One has $\frac{T(C_4)}{4} = 9.5$. The proof for the constant case is almost the same as for the additive case. It is a little bit simpler since, in the constant case, one has the following equality for the travel time $T_T(C_k)$ of the k-cycle C_k:

$$T_T(C_k) = 2k(m+1)\delta - \sum_{i=1}^{m} u_i\delta \qquad (2)$$

The intuition for this equality is that between two activities, one has a time δ if and only if the two activities are not consecutive, *i.e.* they do not participate in a u_i. This equality is proven for $k = 1$ in [7].

$k = 1$: For the instance I, the best 1-cycle has cycle time 10.

$k = 2$: Consider a 2-cycle C_2. Suppose that $T(C_2) \leq \frac{T(C_4)}{4} \times 2 = 19$. For the instance I in the constant case, inequality (2) becomes: $T_T(C_2) \geq 20 - (u_1 + u_4) - (u_2 + u_3)$ and the total cycle time is equal to the travel times plus the waiting times:

$$T(C_2) \geq 20 - (u_1 + u_4) + 5(u_2 + u_3).$$

Supposing $T(C_2) \leq 19$ implies that $u_1 + u_4 \geq 1$. Hence, at least one of the two is greater or equal to 1. Without loss of generality, we take $u_1 \geq 1$ (the case $u_4 \geq 1$ is similar). In this case, C_2 is of the form

$$C_2 = A_0 A_1 \overbrace{\cdots}^{\geq 6} A_2 \underbrace{\cdots}_{S} A_1 \overbrace{\cdots}^{\geq 6} A_2 \underbrace{\cdots}_{S'}$$

As in the preceding proof, between an occurrence of A_1 and the following occurrence of A_2 the time is greater than $p_2 = 6$ and each activity takes a time of $\delta = 1$ and, in S and S', the travel time is greater than $\delta = 1$. This leads to a total time of 19. Moreover, if the sequences S or S' are not empty, then the time to execute the activity they may contain, adds $\delta = 1$ to the total time (which contradicts the hypothesis that $T(C_2) \leq 19$). Therefore, $S = S' = \emptyset$ and C_2 is of the form

$$C_2 = A_0 A_1 \overbrace{\cdots}^{\geq 6} A_2 \overbrace{A_1 \cdots}^{\geq 6} A_3 \underbrace{\cdots}_{T} A_2$$

For the same reason as before, the sequence T has to be empty and C_2 is of the form

$$C_2 = A_0 A_1 \cdots A_3 \underbrace{\cdots}_{U} A_2 A_1 \underbrace{\cdots}_{U'} A_3 A_2$$

The characterization of k-cycles indicates that $U \cup U'$ contains an occurrence of A_4 and U' cannot contain an activity A_0 since between two occurrences of A_1 in a cyclic sense, there is at most one occurrence of A_0. Therefore either $U' = \emptyset$ or $U' = A_4$. If $U' = \emptyset$ then C_2 contains the sequence $A_1 A_3 A_2$ which generates a waiting time of $p_2 - 3\delta = 3$ and the form of C_2 leads to $u_1 + u_4 \leq 2$. In this case the travel time is greater than 18 and the waiting time is greater than 3 which contradicts the hypothesis $T(C_2) \leq 19$. In the other case, $A_4 \in U'$ and $u_1 + u_4 = 1$ which leads to a travel time of 19 time units. Moreover, the sequence $A_1 A_4 A_3 A_2$ generates a waiting time of 1 which also contradicts the hypothesis.

$k = 3$: For this case, the proof is analogous (and even simpler) to the proof for the additive case. However, for the sake of completeness we give the main ideas of the proof.

Let C_3 be a 3-cycle. Suppose $T(C_3) \leq 28.5$. Inequality (2) implies that $T(C_3) \geq 30 - (u_1 + u_4)$ and hence $u_1 + u_4 \geq 2$. Suppose $u_1 \geq 2$. In this case, C_3 is of the form

$$C_3 = A_0 A_1 \overbrace{\cdots}^{\geq 6} A_2 \overbrace{\cdots}^{\geq 1} A_0 A_1 \overbrace{\cdots}^{\geq 6} A_2 \overbrace{\cdots}^{\geq 1} A_1 \overbrace{\cdots}^{\geq 6} A_2 \overbrace{\cdots}^{\geq 1}$$

Since each of the eight activities adds a travel time of 1, this leads to an execution time greater than $3 \times 6 + 8 + 3 = 29$. Therefore, one has $u_1 = 1$ and $u_4 = 1$ and both sequences $A_0 A_1$ and $A_3 A_4$ are between the same two occurrences of A_2. Moreover, for the same reason as before, between A_2 and the following A_1 and between A_3 and the following A_2 (except for one of them each time) there can be no other activity. Therefore, C_3 is of the form

$$C_3 = A_0 A_1 \cdots A_3 A_4 A_2 A_1 \cdots A_3 A_2 A_1 \underbrace{\cdots}_{S} A_3 A_2$$

The sequence of activities S either contains only activity A_4 or is empty. In both case, it generates a waiting time greater than 1 and the total travel time is greater than 28 which contradicts $T(C_3) \leq 28.5$. $\square$

3 Remaining challenging questions

Finding the best 1-cycle can be done in polynomial time ([5] for the additive classical case, [7] for the constant case and [9] for no-wait additive cells). Three challenging questions remain:

- Determine $\mathcal{K}_c(m)$ or find at least an upper finite bound for $\mathcal{K}_c(m)$;
- Settle Agnetis' conjecture in the no-wait case;
- Describe the complexity of finding the best cycle with degree smaller than $\mathcal{K}_c(m)$ and/or $\mathcal{K}_{nw}(m)$.

References

[1] Agnetis A. Scheduling no-wait robotic cells with two and three machines. *European Journal of Operational Research*, 123(2): 303-314, 2000.

[2] N. Brauner. *Ordonnancement dans des cellules robotisées, analyse de la conjecture des un-cycles.* Thèse de doctorat, Université Joseph Fourier, Grenoble, France, 1999.

[3] N. Brauner and G. Finke. Cycles and permutations in robotic cells. *Mathematical and Computer Modelling*, 34:565–591, 2001.

[4] Y. Crama and J. van de Klundert. Cyclic scheduling in 3-machine robotic flow shops. *Journal of Scheduling*, 2:35–54, 1999.

[5] Y. Crama and J. van de Klundert. Cyclic scheduling of identical parts in a robotic cell. *Operations Research*, 45(6):952–965, 1997.

[6] Y. Crama, V. Kats, J. van de Klundert, and E. Levner. Cyclic scheduling in robotic flowshops. *Annals of Operations Research: Mathematics of Industrial Systems*, 96:97-124, 2000.

[7] M. Dawande, C. Sriskandarajah, and S. Sethi. On throughput maximization in constant travel-time robotic cells. *Manufacturing and Service Operations Management*, 4(4):296-312, 2002.

[8] N. G. Hall, H. Kamoun, and C. Sriskandarajah. Scheduling in robotic cells: Classification, two and three machine cells. *Operations Research*, 45(3):421–439, 1997.

[9] E. Levner, V. Kats, and V.E. Levit. An improved algorithm for cyclic flowshop scheduling in a robotic cell. *European Journal of Operational Research*, 97:500–508, 1997.

[10] F. Mangione, N. Brauner, and B. Penz. Optimal cycles for the robotic balanced no-wait flow shop. In *Proceedings IEPM'03, International Conference of Industrial Engineering and Production Management*, volume 2, pages 539-547, Porto, Portugal, 2003.

[11] S. P. Sethi, C. Sriskandarajah, G. Sorger, J. Blazewicz, and W. Kubiak. Sequencing of parts and robot moves in a robotic cell. *International Journal of Flexible Manufacturing Systems*, 4:331–358, 1992.

Les Cahiers Leibniz

Le **Laboratoire Leibniz** est fortement pluridisciplinaire. Son activité scientifique couvre un large domaine qui comprend des thèmes fondamentaux aussi bien en informatique qu'en mathématiques, avec une ouverture sur l'apprentissage machine, la modélisation de systèmes complexes adaptatifs, et les applications aux environnements informatiques pour l'apprentissage humain.

Les **Cahiers Leibniz** ont pour vocation la diffusion de rapports de recherche, de supports de cours, de textes de séminaires ou de projets de publications réalisés par des membres du laboratoire. Ils peuvent accueillir aussi des textes de chercheurs n'appartenant pas au laboratoire Leibniz mais qui travaillent sur des thèmes proches et ne disposent pas de tels supports de publication. Ces chercheurs sont priés de contacter un des membres du comité éditorial ; le comité décidera de l'acceptation du texte proposé.

Le contenu des textes publiés dans les **Cahiers Leibniz** relève de la seule responsabilité de leurs auteurs.

The research at **Laboratoire Leibniz** is multidisciplinary. It covers a large domain of fundamental and applied subjects in informatics an mathematics, with openings to machine learning, the modelisation of adaptive complex systems, and applications to teaching software.

The **Cahiers Leibniz** aim at diffusing research reports, lectures, and texts of conferences or pre-prints of the members of the laboratory. Moreover, the **Cahiers** welcome manuscripts of researchers belonging to other laboratories, working on subjects close to ours, but not disposing of such a medium. These researchers should contact one of the members of the editorial board; the latter will decide whether to accept the proposal.

The responsibility of the contents of the **Cahiers** lies exclusively with the authors.

Réalisation : Jacky Coutin

ISSN : 1298-020X - © laboratoire Leibniz