European Journal of Operational Research 000 (2018) 1–5

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Short Communication

On the existence of dominating 6-cyclic schedules in four-machine robotic cells

Vladimir Kats[a], Eugene Levner[b,*]

[a] Institute for Industrial Mathematics, Beer-Sheva 84641, Israel
[b] Holon Institute of Technology, Holon 58102, Israel

ARTICLE INFO

Article history:
Received 23 June 2017
Accepted 17 January 2018
Available online xxx

Keywords:
Scheduling
Robotic cells
Cyclic scheduling
Dominating schedule

ABSTRACT

We consider a four-machine robotic cell producing identical parts and served by a single robot. We study the no-wait multi-cyclic scheduling problem. Using the forbidden-intervals method, we show that in such a cell the optimal schedule can be k-cyclic with minimum $k \geq 6$. This fact refutes Agnetis' conjecture (Agnetis, 2000) stating that the minimum k for the optimal k-cyclic m-machine schedules does not exceed $m-1$. In particular, we construct a counter-example to Agnetis' conjecture.

1. Introduction and previous work

We consider the problem of no-wait scheduling of robotic cells that produce identical parts and which are served by a single robot. We are interested in maximizing the long-term average throughput of parts. The schedule with the maximum average throughput is called *optimal*. The problem of finding such an optimal schedule was studied, among others, by Agnetis (2000), Brauner (2008), Brauner and Finke (1999, 2001), Crama and van de Klundert (1999), Dawande, Geismar, and Sethi (2005), Geismar, Dawande, and Sriskandarajah (2005), Sethi, Sriskandarajah, Sorger, Blazewicz, and Kubiak (1992), and Che and Chu (2009).

Let us define a *cyclic schedule* as one where, within a fixed time interval, referred to as the *cycle*, the number of parts that enter the cell is equal to the number of parts leaving it. At the end of the cycle, the cell returns to its original state. This process is repeated an infinite number of times. The cyclic schedule is called *k-cyclic* if, in each cycle, exactly k parts enter and k parts leave the cell.

A perennial open question in scheduling theory is whether or not an optimal *cyclic* schedule exists from among all possible infinite schedules. Tanaev (1964) and Blokh and Tanaev (1966) showed that if the input data is integer (or, more generally, rational) then an optimal cyclic schedule exists from among all possible, infinitely-long schedules. Dawande et al. (2005) independently obtained a similar result. However, despite their arguments, the question still remains open for real input data.

The proof of the existence of optimal cyclic schedules for integer and rational data is essentially based on the notion of *state*, which is defined as a set of rational parameters describing the physical location of the robot; the parts; the status of the robot (i.e., unloaded, loaded); and each processed part, i.e. the time remaining to unload the part from machine. Accordingly, the schedule can then be presented as a sequence of states and the transitions between them. Since the number of different states becomes finite, some states will appear infinitely many times in any infinite schedule. Consequently, a section of the schedule between the repeated state occurrences can be duplicated infinitely many times without decreasing the production throughput.

If the sequence of states is optimal, then any sequence segment between two equal states has the same throughput rate. If we infinitely replicate such segment we obtain the k-cyclic schedule with the same optimal throughput rate as in the initial schedule. The estimated value of k depends on the input data values and is, in fact, a very large number.

In recent years, numerous studies have focused on special cases of the above question, for different small k. A k-cyclic schedule is called *optimal* if it has maximum throughput or, equivalently, a minimum average cycle time from among all cyclic schedules. Considering an m-machine cell and following Brauner (2008), let us introduce the *cycle function* $K(m)$, which, for a given number of machines m, is the smallest value of K such that the set of all k-cyclic schedules up to size K contains an optimal cyclic schedule for *all* the problem instances. (That is, for a given m, the set of all k-cyclic schedules up to size K contains an optimal cyclic schedule for *all* the problem instances; the set of all (K-1)-cyclic schedules does not have that property).

* Corresponding author.
E-mail addresses: vkats@iimath.com (V. Kats), levner@hit.ac.il (E. Levner).

https://doi.org/10.1016/j.ejor.2018.01.034

Agnetis (2000) proved that, for the no-wait robotic cells, $K(2)=1$, $K(3)=2$ and conjectured that $K(m)=m-1$, for $m>3$. Let us consider the definition of $K(m)$ in more details. Let the cell contain two machines. The notation $K(2)=1$ means that in any two-machine cell, independently of what the input data of a problem instance is, there exists a one-cyclic optimal schedule for that instance. $K(3)=2$ means that in any three-machine cell, independently of a problem instance (input data), there exists a certain 1-cyclic or 2-cyclic optimal cyclic schedule for that instance. That is, if one finds, for instance, some k-cyclic optimal schedule with $k \geq 3$, for any sample three-machine problem, then there always exists (and can be found) a certain one- or two-cyclic optimal schedule for this instance and, in addition, there exist three-machine problem samples for which a 1-cyclic optimal schedule does not exist.

In this paper, we continue this line of research and study multi-cyclic, no-wait schedules for $m>3$. We present an example for the four-machine problem for which a six-cyclic schedule has an average cycle time less than any other 1-, 2-, 3-, 4- and 5-cyclic feasible schedule. We can therefore say that, in such a case, the 6-cyclic schedule *strictly dominates* all k-cyclic schedules with $k \leq 5$. According to the above definition of the cycle function $K(m)$, this proves that $K(4) \geq 6$.

As a by-product, the latter result refutes Agnetis' conjecture. Indeed, for $m=4$, the latter conjecture implies that $K(4)$ should be equal to 3. However, in fact, we shall see that $K(4) \geq 6$. In order to prove this fact, we apply the forbidden-intervals technique introduced by Aizenshat (1963) and further developed by Levner, Kats, and Levit (1997), Che, Chu, and Levner (2003) and Che and Chu (2009). Notice that the question remains open as to whether $K(4)$ is strictly larger than 6; the example presented in this note does not provide an answer to this question.

Instead of the guess that Agnetis makes, we conjecture that $K(4)=6$ and, more generally, $K(m)=(m-1)!$

2. Optimal cyclic schedules for robotic cells

In this section we briefly describe a m-machine robotic cell producing identical parts and served by a single robot. Denote m processing machines in the technological sequence of operations by $M_1, M_2, \ldots, M_m$. Let machines M_0 and M_{m+1} denote the input and output station, respectively. The robot transports the parts between machines, no more than one part at time. Each machine can process only one part at time. This means that the operations on the machine do not overlap in time and the robot must remove the finished part from the machine before the next part will be loaded onto it. The *no-wait constraint* which is imposed requires that after a part is processed at a machine M_i, this part must be immediately transferred by the robot to the next machine M_{i+1}. Denote the following parameters (which are the known constants):

- p_i – the processing time on machine M_i, $i=1,\ldots, m$;
- d_i – the time required for the robot to deliver a part from machine M_i to machine M_{i+1}, $i=0, 1,\ldots, m$;
- r_{ij} – the time required for the empty robot to run from machine M_i to machine M_j; $r_{ii}=0$, $i=1, \ldots, m+1$, $j=0, 1,\ldots, m$.

We assume that the robot's moves satisfy the following triangle inequalities:

$$R_{ik} \leq R_{ij} + R_{jk} \tag{1}$$

where $R_{ij}=d_i+r_{i+1,j}$ and $i, j, k=0, 1,\ldots, m$.

Let part 0 be moved by the robot from machine M_0 into the processing system at time $t_0=0$. Denote by Z_j the completion time of the parts' jth operation on machine M_j. In the no-wait case, all Z_j are defined by the time t_0 and the processing/delivering times, as follows:

$$Z_0 = t_0 = 0, \quad Z_j = t_0 + \sum_{i=1}^{j} (d_{i-1} + p_i), \quad j = l, \ldots, m. \tag{2}$$

Assume that identical parts 1, 2, … are introduced by the robot from machine M_0 into the process at times $Y_1, Y_2, \ldots$ respectively. Then the completion time of the jth operation of part q is Y_q+Z_j, ($q=0, 1, 2, \ldots$; $j=0, 1, 2, \ldots, m$), where $Y_0 = t_0=0$. The sequence $Y_1, Y_2, \ldots$ fully determines the starting and completion times of processing parts on each machine and the sequence of all robot moves. We shall call the sequence $S=(Y_1, Y_2, \ldots)$ a *robotic schedule*. Let $T_{q+1}=Y_{q+1}-Y_q$ ($q=0, 1, 2, \ldots$) be the time between two consecutive robot moves transferring the parts from machine M_0. Then, the schedule S can be also presented as a sequence of times T_q, $S=(T_1, T_2, \ldots)$. The schedule is k-cyclic if there exists a constant (called the cycle time) $C=Y_{q+k}-Y_q$ for all $q=0, 1, 2, \ldots$. The k-cyclic schedule S_k can be defined by k time values, $S_k=\{T_1, T_2, \ldots, T_k\}=(T_1, T_2, \ldots, T_k, T_1, T_2, \ldots, T_k, \ldots)$ and its cycle time is $C=\Sigma_{i=1}{}^{k} T_i$. Schedule's *average cycle time* is the mean time required to produce a part, or equivalently, is the mean time between two consecutive robot moves with parts from machine M_0.

$$T_{avr} = {}^{Sup}_{q\to\infty} \left(\sum_{i=1}^{q} (T_i/q) = {}^{Sup}_{q\to\infty} (Y_q/q). \tag{3}$$

In the k-cyclic case, we have

$$T_{k-cyclic\ avr} = C/k = \sum_{i=1}^{k} T_i/k. \tag{3a}$$

Our objective is to find an optimal schedule, that is, one displaying the minimum average cycle time.

3. Method of forbidden intervals

For the reader's convenience, in this section we first briefly outline the so-called *method of forbidden intervals*, an algebraic approach to analyzing and solving scheduling problems. Consider the infinite production process. Denote the sequence of the robot moves by $Q=\{q(0), q(1), q(2), \ldots, q(k), \ldots\}$ meaning that the robot sequentially unloads and transports parts from machines $\{M_{q(0)}, M_{q(1)}, M_{q(2)}, \ldots, M_{q(k)}, \ldots\}$, where $q(0)=0$. Let move $q(k)$ start at time t_k, ($k=0, 1, 2, \ldots$), where $t_0=0$. The necessary and sufficient conditions for the robot to serve machines without delays in sequence $\{M_{q(0)}, M_{q(1)}, M_{q(2)}, \ldots, M_{q(k)}, \ldots\}$ completing their operations at times $\{t_0, t_1, t_2, \ldots, t_k, \ldots\}$, respectively, are

$$t_k + R_{q(k),q(k+1)} \leq t_{k+1}, \ (k = 0, 1, 2, \ldots) \tag{4}$$

Consider two consequent inequalities in (4):

$$t_k + R_{q(k),q(k+1)} \leq t_{k+1} \ and \ t_{k+1} + R_{q(k+1),q(k+2)} \leq t_{k+2}$$

Taking into account the triangle inequalities (1) we obtain:

$$t_k + R_{q(k),q(k+2)} \leq t_{k+2}.$$

Continuing the summing-up along the sequence Q and taking into account the triangle inequalities (1), we obtain

$$t_k + R_{q(k),q(l)} \leq t_l \tag{5}$$

for all $k=0, 1, 2, \ldots$; $l=1, 2, \ldots$; and $t_k<t_l$.

Assume that at time t_k, part s finishes its ith operation on machine $M_{q(k)}$, i.e. $i=q(k)$. Substitute the completion times t_k, ($k=0, 1, 2, \ldots$) in (5) by their expressions Y_s+Z_i, ($s=0, 1, 2, \ldots$; $i=0, 1, 2, \ldots, m$). Depending on values Y_s+Z_i, two forms of inequalities (5) are possible:

$$Y_s + Z_i + R_{ij} \leq Y_q + Z_j, \quad if \ Y_s + Z_i < Y_q + Z_j \tag{6}$$

Please cite this article as: V. Kats, E. Levner, On the existence of dominating 6-cyclic schedules in four-machine robotic cells, European Journal of Operational Research (2018), https://doi.org/10.1016/j.ejor.2018.01.034