# An adaptive Search for the Response Time Variability Problem[1]

Said SALHI[a] and Alberto GARCÍA-VILLORIA[b*]
[a] The Centre for Logistics & Heuristic Optimisation (CLHO), Kent Business School, University of Kent at Canterbury, Canterbury CT2 7PE, UK
[b] Institute of Industrial and Control Engineering (IOC), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
s.salhi@kent.ac.uk, alberto.garcia-villoria@upc.edu

**Abstract**. The Response Time Variability Problem (RTVP) is an NP-hard combinatorial scheduling problem which has recently been reported and formalised in the literature. This problem has a wide range of real-world applications in mixed-model assembly lines, multi-threaded computer systems, broadcast of commercial videotapes and others. The RTVP arises whenever products, clients or jobs need to be sequenced in such a way that the variability in the time between the points at which they receive the necessary resources is minimised. We propose a greedy but adaptive heuristic that avoids being trapped into a poor solution by incorporating a look ahead strategy suitable for this particular scheduling problem. The proposed heuristic outperforms the best existing methods, while being much faster and easier to understand and to implement.

**Keywords:** response time variability, heuristics, adaptive search, scheduling, fair sequences

## 1. Introduction

The concept of a *fair sequence* has emerged independently from scheduling problems of diverse environments. The common aim of these scheduling problems, as defined in Kubiak (2004), is to build a fair sequence using $n$ symbols, where symbol $s$ ($s = 1,...,n$) must occur $d_s$ times in the sequence. The fair sequence is the one which allocates a fair share of positions to each symbol $s$ in any subsequence. This fair or ideal share of positions allocated to symbol $s$ in a subsequence of length $k$ is proportional to the relative importance ($d_s$) of symbol $s$ with respect to the total number of copies of competing symbols (equal to $\sum_{s=1}^{n} d_s$ ). There is no universal definition of fairness because several reasonable metrics can be defined according to the specific problem considered.

Among the different definitions of fairness, several fair sequencing problems have emerged, among them the Response Time Variability Problem (RTVP). This problem has been reported for the first time by Waldspurger and Weihl (1994) but formalised several years later by Corominas *et al*. (2007). In the RTVP, the fair sequence is the one which minimises the sum of the variability in the distances between any two consecutive copies of the same symbol. In other words, the distance between any two

[1] Supported by the Spanish Ministry of Education and Science under project DPI2007-61905; co-funded by the ERDF, also supported by the Department of Innovationm, Universities and Enterprise of Generalitat de Catalunya under grant BE-DGR-2008.
[*] Corresponding author: Alberto García-Villoria, Institute of Industrial and Control Engineering (IOC), Av. Diagonal 647 (Edif. ETSEIB), 11th floor, 08028 Barcelona, Spain; tel.: +34 93 4054010; e-mail: alberto.garcia-villoria@upc.edu. The research was conducted while visiting CLHO at Kent.

consecutive copies of the same symbol should be as regular as possible (i.e., ideally constant).

In practice, the RTVP arises whenever products, clients or jobs need to be sequenced so as to minimise the variability in the time between the instants at which they receive the necessary resources (Corominas *et al*., 2007). This problem has a broad range of real-world applications. These include, for instance, the sequencing of mixed-model assembly lines under JIT (Kubiak, 1993; Miltenburg, 1989), the resource allocation in computer multi-threaded systems such as operating systems, network servers and media-based applications (Dong *et al*., 1998; Waldspurger and Weihl, 1994, 1995), the periodic machine maintenance problem when the times between consecutive services of the same machine are equal (Anily *et al*., 1998; Wei and Liu, 1983), the collection of waste (Herrmann, 2007), the schedule of commercial videotapes for television (Bollapragada *et al*., 2004; Brusco, 2008) and the design of sales catalogues (Bollapragada *et al*., 2004).

Corominas *et al*. (2007) showed that the RTVP is NP-hard and produced an analytical formulae for finding the lower bound. The problem can be formulated as a mixed integer linear programming as shown by Corominas *et al*., (2007,2010). The only exact procedure that is able to generate optimal solutions for small instances up to 50 units is the Branch and Bound given by García-Villoria *et al*., (2009). For larger instances, several heuristic and metaheuristic algorithms have been proposed for its solution. Waldspurger and Weihl (1994) propose an algorithm that generates a solution randomly. The same authors (Waldspurger and Weihl, 1995) improve their previous results using the Jefferson method of apportionment (Balinski and Young, 1982), a greedy heuristic algorithm which they renamed as the stride scheduling technique. Herrmann (2007) solved the RTVP by applying a heuristic algorithm based on the stride scheduling technique. Corominas *et al*. (2007) proposed the Jefferson method together with other four constructive type heuristic algorithms. Seven new heuristics are also given by Corominas *et al*. (2009). Metaheuristics for the RTVP were recently proposed in García-Villoria and Pastor (2010a, 2010b, 2010c) and these include an electromagnetism-like mechanism (EM) algorithm, a psychoclonal algorithm and a genetic algorithm (GA) respectively.

The best five classical heuristics are described by (Corominas *et al*., 2009) and known as *Oc*, *AWe/dg*, *We/dg*, *Je/dg* and *In*. On the other hand, the best results recorded to date using relatively a larger computing time have been obtained with a GA (García-Villoria and Pastor, 2010c).

In this paper, an adaptive search based on a look ahead strategy is proposed. The reasoning behind this approach and the two theorems that support it are put forward. An extensive computational experiment is carried out to assess the superiority of this heuristic over the aforementioned classical heuristics for both solution quality and computational effort. Moreover, the solutions obtained with the proposed heuristic are also found competitive when compared to the GA while requiring a fraction of its cpu time.

In this study, we also introduce a new but related scheduling problem for the first time that we refer to as the *minmax* RTV problem. In this problem, the objective is to minimise the maximum absolute discrepancy in the distances between any two consecutive copies of the same symbol. Although the heuristic introduced in this paper has been specifically designed to solve the RTVP, the way the look ahead strategy is defined led itself to solve the *minmax* RTVP as well. The obtained results are reported here to provide a platform for benchmarking purposes in the future.

The remainder of the paper is organised as follows: First, Section 2 presents a formal definition of the RTVP. The next section represents the main body of the research and it covers the new heuristic algorithm, the supporting theorems and the proposed enhancements. The results of our computational experiment are presented in Section 4. A new but related problem, the *minmax* RTVP, is briefly described and its results summarised in Section 5. Finally, some conclusions and suggestions for future research are provided in the last section.

## 2. The Response Time Variability Problem (RTVP)

The RTVP is formulated as follows. Let *n* be the number of symbols, $d_s$ the number of copies to be sequenced of symbol *s* (*s* = 1,…,*n*) and $D = \sum_{s=1}^{n} d_s$ the total number of copies. Let *seq* be a solution of an instance in the RTVP that consists of a circular sequence of these *D* copies ($seq = s_1 s_2 \ldots s_D$), where $s_j$ is the copy sequenced in position *j* of sequence *seq*. For each symbol *s* in which $d_s \geq 2$, let $t_k^s$ be the distance between the positions in which the copies *k* + 1 and *k* of symbol *s* are found. We consider the distance between two consecutive positions to be equal to 1. Since the sequence is circular, position 1 comes immediately after position *D*; therefore, $t_{d_s}^s$ is the distance between the first copy of symbol *s* in a cycle and the last copy of the same symbol in the preceding cycle. Let $\overline{t_s}$ be the ideal average distance between two consecutive copies of symbol *s* ($\overline{t_s} = D/d_s$). Note that for each symbol *s* in which $d_s = 1$, $t_1^s$ is equal to $\overline{t_s}$. The objective is to minimise the metric called response time variability (RTV), which is defined by the sum of the square errors with respect to the $\overline{t_s}$ distances. This is defined as $RTV = \sum_{s=1}^{n} \sum_{k=1}^{d_s} (t_k^s - \overline{t_s})^2$.

The lower bound introduced in Corominas *et al.* (2007) is defined as follows:

$$LB = \sum_{s=1}^{n} \left( (D \bmod d_i) \cdot \left( \left\lceil \frac{D}{d_i} \right\rceil - \overline{t_i} \right)^2 + (d_i - D \bmod d_i) \cdot \left( \left\lfloor \frac{D}{d_i} \right\rfloor - \overline{t_i} \right)^2 \right).$$

For an illustration, consider the following example. Let $n = 3$ with symbols *H*, *I* and *J*. Also consider $d_H = 3$, $d_I = 2$ and $d_J = 2$; thus, $D = 7$, $\overline{t_H} = 7/3$, $\overline{t_I} = 7/2$ and $\overline{t_J} = 7/2$. The corresponding lower bound, LB=

$$\left(1 \cdot (3 - 7/3)^2 + 2 \cdot (2 - 7/3)^2\right) + \left(1 \cdot (4 - 7/2)^2 + 1 \cdot (3 - 7/2)^2\right) + \left(1 \cdot (4 - 7/2)^2 + 1 \cdot (3 - 7/2)^2\right) = 5/3.$$

It can be shown that any sequence such that contains symbol $s$ $(\forall s)$ exactly $d_s$ times is a feasible solution. For instance, the sequence (H, I, H, J, I, J, H) is a feasible solution, which has an RTV value equal to

$$\left(\left(2-\tfrac{7}{3}\right)^2+\left(4-\tfrac{7}{3}\right)^2+\left(1-\tfrac{7}{3}\right)^2\right)+\left(\left(3-\tfrac{7}{2}\right)^2+\left(4-\tfrac{7}{2}\right)^2\right)+\left(\left(2-\tfrac{7}{2}\right)^2+\left(5-\tfrac{7}{2}\right)^2\right)=\tfrac{29}{3}.$$

## 3. An adaptive algorithm for the RTVP

In this section we propose a constructive adaptive heuristic, which uses a look ahead strategy, to solve the RTVP. The algorithm consists of $D$ steps and at each step $p$ ($p = 1,\ldots,D$) a symbol is selected to be sequenced at position $p$ of the sequence. In fact, it could be considered that this method has $D-1$ steps since the symbol to be sequenced at the last step will be automatically determined. The reasoning behind the strategy to select the symbol to be sequenced at each step is discussed in subsection 3.1 which also contains two theorems to support this selection process. We first describe our initial implementation in subsection 3.2 as this will serve as a basis for making the explanation of the proposed adaptive heuristic in subsection 3.3 relatively easier.

### *3.1. The basic idea of the heuristic*

Let first introduce some additional nomenclature:

- $seq_p$: The partial sequence obtained at step $p$; $p=0,\ldots,D-1$. Initially $seq_0$ is a void sequence
- $\hat{d}(s,p)$: The number of times left for symbol $s$ to be sequenced in $seq_p$; $s=1,\ldots,n$, $p=0,\ldots,D-1$
- $SS(p)$: The set of symbols that have been sequenced in $seq_p$ at least once; $p=0,\ldots,D-1$
- $lsp(s,p)$: The last position in which symbol $s$ has been sequenced in $seq_p$; $s\in SS(p)$, $p=0,\ldots,D-1$
- $t(s,p)$: $p-lsp(s,p-1)$; $s\in SS(p-1)$, $p=1,\ldots,D$
- $S^{+}(p)$: The set of symbols $\left\{s\in SS(p-1)\mid t(s,p)\geq \bar{t}_s \wedge \hat{d}(s,p-1)\geq 1\right\}$; $p=1,\ldots,D$
- $S^{-}(p)$: The set of symbols $\left\{s\in SS(p-1)\mid t(s,p)< \bar{t}_s \wedge \hat{d}(s,p-1)\geq 1\right\}$; $p=1,\ldots,D$

Given a partial solution sequence $seq_{p-1}$, the aim is to decide which symbol to be sequenced at position $p$ ($p = 1,\ldots,D$). The symbols that still have copies to be sequenced at step $p$ (that is, all symbol $s$ ($s = 1,\ldots,n$) such as $\hat{d}(s,p-1)\geq 1$) can be grouped into either the set $S^{+}(p)$ or the set $S^{-}(p)$. Given a symbol $s\in S^{+}(p)$ and a symbol $s'\in S^{-}(p)$, if one of them has to be sequenced at step $p$, then the decision that gives the lowest increment to the RTV value of the partial solution for the symbols $s$ and $s'$ is to sequence the symbol $s$ in position $p$ and to sequence the symbol s' in a later position. The validity of this claim is shown in Theorem 1. The reasoning behind this argument is that we try to avoid accumulating an excessive future increase in the distance between the next copy to be sequenced of symbol $s$ and its last sequenced copy. This is important as the square error between ideal distances and real distances is used and this can be amplified very quickly. On the other hand, we allow that the distance between the next copy to be sequenced of symbol *s*' and its last sequenced copy increases. In other words, it is better to select the symbol that will be late than early with respect to its average distance. This is obviously a straightforward but important observation. Note that such a discrepancy between this distance and $\overline{t}_{s'}$ will be reduced as shown in Figure 1.

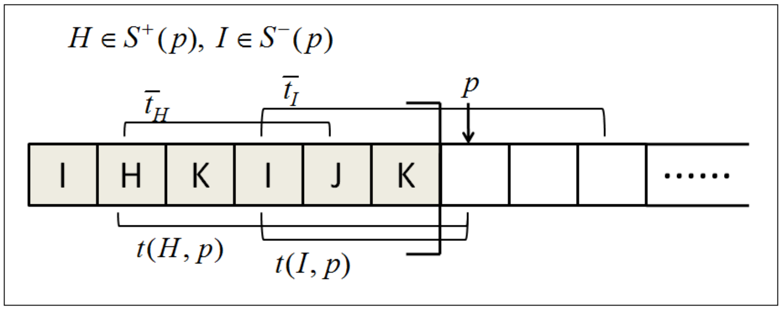


**Figure 1**. A graphical illustration of the sequencing distance concept

**Theorem 1** Let $seq_{p-1}$ be a partial sequence solution obtained at step *p*-1 (*p* = 1,…,*D*). Given a symbol $s \in S^+(p)$ and a symbol $s' \in S^-(p)$, if one of them has to be sequenced at step *p*, then the less RTV increment is obtained by sequencing the symbol *s* in position *p* and the symbol s' in a later position *p*' (*p*' > *p*).

**Proof.** By definition of the sets $S^+(p)$ and $S^-(p)$, we have that $t(s,p)-\overline{t}_s > t(s',p)-\overline{t}_{s'}$ or, equivalently, $t(s,p)-\overline{t}_s = t(s',p)-\overline{t}_{s'}+u$ where $u>0$. Analogously, $p'=p+q$ where $q \geq 1$. Consider the two possible options for sequencing the two symbols.

*Option 1*: The symbols *s* and *s*' are sequenced in the positions *p* and *p*', respectively. The increment of the RTV value ( $\Delta^1_{RTV}$ ) is the following: $\Delta^1_{RTV} = \left(t(s,p)-\overline{t}_s\right)^2+\left(t(s',p')-\overline{t}_{s'}\right)^2 = \left(t(s',p)-\overline{t}_{s'}+u\right)^2+\left(t(s',p+q)-\overline{t}_{s'}\right)^2 = \left(t(s',p)-\overline{t}_{s'}+u\right)^2+\left(t(s',p)+q-\overline{t}_{s'}\right)^2$.

*Option 2*: The symbols *s* and *s*' are sequenced in the positions *p*' and *p*, respectively. The increment of the RTV value ( $\Delta^2_{RTV}$ ) is the following: $\Delta^2_{RTV} = \left(t(s,p')-\overline{t}_s\right)^2+\left(t(s',p)-\overline{t}_{s'}\right)^2 = \left(t(s,p+q)-\overline{t}_s\right)^2+\left(t(s',p)-\overline{t}_{s'}\right)^2 = \left(t(s,p)+q-\overline{t}_s\right)^2+\left(t(s',p)-\overline{t}_{s'}\right)^2 = \left(t(s',p)+q-\overline{t}_{s'}+u\right)^2+\left(t(s',p)-\overline{t}_{s'}\right)^2$.

Let $\theta = t(s'p)-\overline{t}_{s'}$. Thus, $\Delta^1_{RTV} = (\theta+u)^2+(\theta+q)^2 = 2\theta^2+2\theta u+2\theta q+u^2+q^2$ and $\Delta^2_{RTV} = \left(\theta+(q+u)\right)^2+\theta^2 = 2\theta^2+2\theta u+2\theta q+u^2+q^2+2qu$.
Therefore, $\Delta^2_{RTV} = \Delta^1_{RTV}+2qu$. Since $q \geq 1$ and $u>0 \Rightarrow \Delta^1_{RTV} < \Delta^2_{RTV}$. ■

We can generalize Theorem 1 by extending it for any pair of symbols *s* and *s*' without considering if they are included in the set $S^+(p)$ or in the set $S^-(p)$.

**Theorem 2** Let $seq_{p\text{-}1}$ be a partial sequence solution obtained at step $p$-1 ($p$ = 1,…,$D$). Given the symbols $s, s' \in SS(p)$, when one of them has to be sequenced at step $p$, then the less RTV increment is obtained by sequencing the symbol $s^* = \arg\max_{i\in\{s,s'\}} \left(t(i,p) - \overline{t}_i\right)$ in position $p$ and the other symbol $s^{\#}$ $\left(s^{\#} = \{s,s'\} - \{s^*\}\right)$ in a later position $p'$ ($p' > p$).

**Proof.** By hypothesis, we have that $t(s^*,p) - \overline{t}_{s^*} \geq t(s^{\#},p) - \overline{t}_{s^{\#}}$. If $t(s^*,p) - \overline{t}_{s^*} > t(s^{\#},p) - \overline{t}_{s^{\#}}$ then we can apply Theorem 1. In the other hand, if $t(s^*,p) - \overline{t}_{s^*} = t(s^{\#},p) - \overline{t}_{s^{\#}}$ then it is indifferent which of the two symbols is sequenced first. ■

**Lemma.** When all symbols have been sequenced at least once, the symbol $s^* = \arg\max_{s\in SS(p) | \hat{d}(s,p)\geq 1} \left\{t(s,p) - \overline{t}_s\right\}$ is sequenced at step $p$.

The above lemma constitutes the cornerstone idea in which the proposed algorithm will be based upon.

### *3.2. An initial implementation*

We propose an initial heuristic based on Theorem 2 and the above lemma. This will also serve as a basis as well as a build-up for our enhancements which will be presented in the next subsection. At each step $p$ ($p$ = 1,…,$D$) of the heuristic, the symbols that still have copies to be sequenced are classified into the following three sets:

$S_1(p)$: The set of symbols $\left\{s \in \{1,\ldots,n\} \mid (d_s = 1) \wedge \left(\hat{d}(s,p-1) = 1\right)\right\}$; $p = 1,\ldots,D$

$S_2(p)$: The set of symbols $\left\{s \in \{1,\ldots,n\} \mid (d_s \geq 2) \wedge \left(\hat{d}(s,p-1) = d_s\right)\right\}$; $p = 1,\ldots,D$

$S_3(p)$: The set of symbols $\left\{s \in \{1,\ldots,n\} \mid (d_s \geq 2) \wedge \left(0 < \hat{d}(s,p-1) < d_s\right)\right\}$; $p = 1,\ldots,D$

Note that the symbols with only one copy to be sequenced have the following interesting property. All symbol $s$ of $S_1(p)$ (and, therefore, $t_1^s = \overline{t}_s$), will never increase the RTV value of the solution (this is explained in Section 2). The heuristic will sequence these symbols (i.e., those in which $d_s = 1$) whenever it is *not suitable* to sequence any other symbol $s$ from $S_2(p)$ or $S_3(p)$.

Let the function $\Delta(s,p)$ $\forall s \in S_1(p) \cup S_3(p)$ and $\forall p$ ($p$ = 1,…,$D$) be defined as follows: $\Delta(s,p) = \begin{cases} t(s,p) - \overline{t}_s & \text{,if } d_s \geq 2 \\ 0 & \text{,if } d_s = 1 \end{cases}$

Note that, by definition, the symbols of the sets $S^+(p)$ have $\Delta \geq 0$, whereas those symbols of the sets $S^-(p)$ have $\Delta < 0$. Ideally, the remaining copies of the symbols that have been sequenced at least once should be next sequenced at step $p$ in which their $\Delta$

value is 0. In general, however, this is not always possible, so the idea is to sequence the symbols with the highest Δ value according to Theorem 2.

The pseudo-code of the proposed heuristic is shown in Figure 2. The algorithm has two phases. Let $R$ be the number of steps used by the algorithm to sequence all symbols $s$ in which $d_s \geq 2$ at least once. That is, $R$ is the step in which $S_2(R+1) = \varnothing$ and $S_2(R) \neq \varnothing$. The first phase applies during the first $R$ steps (lines 2 to 4 of the pseudo-code) and the second phase uses the remaining $D-R$ steps (lines 5 and 6 of the pseudo-code).

0. Let $seq_0$ be a void sequence
1. For $p = 1$ to $D$ do:
2.     If $S_2(p) \neq \varnothing$ then:
3.         If $\exists s : s \in S_3(p) \mid \Delta(s,p) \geq 0$ then $s_p^*$ is the symbol $s \in S_3(p)$ with the highest $\Delta(s,p)$ value. In case of tie, use the tie breaker of Figure 3.
4.         Otherwise $s_p^*$ is the symbol $s \in S_2(p)$ with the highest $d_s$ value. If there is a tie, use the lexicographical order.
5.     Otherwise ( $S_2(p) = \varnothing$ ):
6.         $s_p^*$ is the symbol $s \in S_1(p) \cup S_3(p)$ with the highest $\Delta(s,p)$ value. In case of a tie, use the tie breaker of Figure 3.
7.     $seq_p$ is obtained by sequencing $s_p^*$ in $seq_{p\text{-}1}$
8. Next $p$
9. Return $seq_D$

**Figure 2**. The pseudocode of the initial heuristic

- If there is a tie, select the symbol with the highest $\hat{d}(s,p)$ value.
- If there is again a tie, select the symbol with the highest $d_s$ value.
- Finally, if a tie still occurs, use the lexicographical order.

**Figure 3**. The tie breaker

*Phase I*. In this phase, all symbols $s$ in which $d_s \geq 2$ are sequenced at least once. At each step $p$ ($p = 1,\ldots,R$), only symbols of $S_2(p)$ or $S_3(p)$ are considered to be sequenced. The symbols of $S_1(p)$ are not considered in this phase because they are kept for the second phase to fill the positions which are not suitable for any other symbols. All symbols $s$ in which $d_s = 1$ can be used as a *wild card*. The main objective of this phase is to sequence at least the first copy of all symbols $s$ in which $d_s \geq 2$. However, if there is one or more symbols of $S_3(p)$ that have $\Delta \geq 0$, then the symbol with the highest value is selected.

*Phase II*. In this phase, all symbols $s$ in which $d_s \geq 2$ have been sequenced at least once. Thus, according to Theorem 2, at each step $p$ ($p = R+1,\ldots,D$), the symbol which has the highest Δ value is chosen. Note that if all symbols of $S_3(p)$ have a negative Δ value, then a symbol of $S_1(p)$ is sequenced (if $S_1(p)$ is not void), since its Δ value is 0. This scheme is introduced to stop the Δ values of the symbols of $S_3(p)$ to be increased at the next steps.

### *3.3. The adaptive heuristic*

In this section three modifications are introduced to improve the performance of the initial heuristic. These are given in subsections (3.3.1) to (3.3.3). These enhancements are then put together to make up our overall adaptive approach whose pseudo-code is given in the last subsection (3.3.4).

#### *3.3.1. Effect of the distances between the first and last copies of the symbols*

When the last copy of symbol $s$ remains to be sequenced, only the distance between this copy and its second to the last copy (i.e., $t^s_{d_s-1}$) is taken into account. However, the distance between its last copy and its first copy in the preceding cycle (i.e., $t^s_{d_s}$) should also be taken into consideration. The function $\Delta(s,p)$ is therefore redefined to overcome this discrepancy:

$$\Delta(s,p)=\begin{cases} t(s,p)-\overline{t}_s & \text{if } (d_s\geq 2)\wedge\left(\hat{d}(s,p-1)\geq 2\right) \\ \left[t(s,p)-\overline{t}_s\right]+\left[\overline{t}_s-\left(D+fsp(s)-p\right]\right. & \text{if } (d_s\geq 2)\wedge\left(\hat{d}(s,p-1)=1\right) \\ 0 & \text{if } d_s=1 \end{cases}$$

where $fsp(s)$ returns the first position in which symbol $s$ has been sequenced.

#### *3.3.2. Effect of the competition for the same position*

The initial heuristic sequences, at each step $p$, a symbol of $S_2(p)$ (during the first phase) or a symbol of $S_1(p)$ (during the second phase) when all symbols of $S_3(p)$ have negative Δ values. However, there are situations in which it is better to sequence a symbol of $S_3(p)$ though its Δ value is negative.

*A counter-example*

Let $n=5$ with symbols $H$, $I$, $J$, $K$ and $L$ in which $d_H=1$, $\overline{t}_I=5.7$, $\overline{t}_J=3.9$, $\overline{t}_K=2.6$ and $\overline{t}_L=2.8$, and let suppose that at step $p$ the sequence $seq_p$ shown in Figure 4a has been generated.

The initial proposed heuristic will produce the partial sequence shown in Figure 4b as follows:

- At step $p$, $\Delta(H,p)=0$, $\Delta(I,p)=-1.7$, $\Delta(J,p)=-0.9$, $\Delta(K,p)=-0.6$ and $\Delta(L,p)=-1.8$; thus, the symbol $H$ is sequenced since it has the highest Δ value.
- At step $p+1$, $\Delta(I,p+1)=-0.7$, $\Delta(J,p+1)=0.1$, $\Delta(K,p+1)=0.4$ and $\Delta(L,p+1)=-0.8$, so symbol $K$ is sequenced.

- At step $p+2$, $\Delta(I, p+2)=0.3$, $\Delta(J, p+2)=1.1$, $\Delta(K, p+2)=-1.6$ and $\Delta(L, p+2)=0.2$, so symbol $J$ is sequenced.
- At step $p+3$, $\Delta(I, p+3)=1.3$, $\Delta(J, p+3)=-2.9$, $\Delta(K, p+3)=-0.6$ and $\Delta(L, p+3)=1.2$, so symbol $I$ is sequenced.
- At step $p+4$, $\Delta(I, p+4)=-4.7$, $\Delta(J, p+4)=-1.9$, $\Delta(K, p+4)=0.4$ and $\Delta(L, p+4)=2.2$, so symbol $L$ is sequenced.

The increment of the RTV value obtained from the copies of the symbols *I*, *J*, *K* and *L* sequenced from step $p+1$ to step $p+4$ is $(7-5.7)^2+(5-3.9)^2+(3-2.6)^2+(5-2.8)^2$ = 7.9.

On the other hand, a lower RTV increment could be obtained with the sequence shown in Figure 4c, which is $(6-5.7)^2+(4-3.9)^2+(2-2.6)^2+(4-2.8)^2$ = 1.9. In this case, the symbol $K$ has been sequenced at step *p* although $\Delta(K, p)=-0.6$.

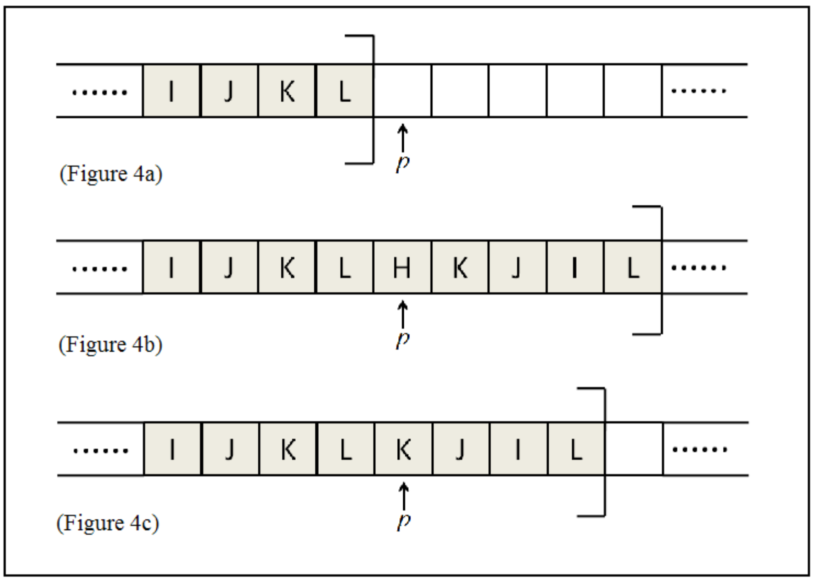


**Figure 4**. Different ways of sequencing

The proposed condition for sequencing at step *p* a symbol of $S_3(p)$ though all its symbols have a negative Δ value is that there could be *too many* symbols that would be sequenced during the next immediate positions of *p*. To overcome this shortcoming, the following condition is introduced:

$$\exists q \in \{p+1 \ldots D\} : \left|\tilde{S}_3(p,q)\right| \geq (q-p+M),$$

where $M$ ($M \geq 1$) is a parameter that quantifies the effect of the cardinality of the set $\tilde{S}_3(p,q)=\{s \in S_3(p) : \Delta(s,p)+(q-p) \geq 0\}$. The value of $M$ that obtains the best performance was found empirically to be 2.

### 3.3.3. Effect of dynamic ideal distances

In the initial heuristic, the ideal distance between two copies of symbol $s$ is considered to be equal to $\overline{t_s}$ in all steps of the construction of the solution. On the other hand, it seems better to adjust dynamically the ideal distance of symbol $s$ according to the current partial solution. This aims to sequence the remaining copies of $s$ more regularly among the remaining positions. The adjusted ideal distances $\hat{t}(s,p)$ are then defined for all $s \in S_3(p)$ and for all steps $p$ ($p = 1,\ldots,D$) as follows:

$$\hat{t}(s,p) = \frac{D - lsp(s,p-1) + fsp(s)}{\hat{d}(s,p-1)+1}$$

### 3.3.4. The enhanced heuristic

The pseudo-code of our proposed adaptive heuristic is shown in Figure 5, with the summary of the modifications as explained in the last three subsections:

0. Let $seq_0$ be a void sequence
1. For $p=1$ to $D$ do:
2. If $S_2(p) \neq \varnothing$ then:
3. If $(\exists s : s \in S_3(p) \mid \Delta(s,p) \geq 0) \vee (\exists q \in \{p+1 \ldots D\} : |\tilde{S}_3(p,q)| \geq (q-p+M))$ then $s_p^*$ is the symbol $s \in S_3(p)$ with the highest $\Delta(s,p)$ value. In case of a tie, use the tie break procedure of Figure 3.
4. Otherwise $s_p^*$ is the symbol $s \in S_2(p)$ with the highest $d_s$ value. If there is a tie, use the lexicographical order.
5. Otherwise $S_2(p) = \varnothing$:
6. If $(\exists q \in \{p+1 \ldots D\} : |\tilde{S}_3(p,q)| \geq (q-p+M))$ then $S' = S_3(p)$; otherwise, $S' = S_1(p) \cup S_3(p)$
7. $s_p^*$ is the symbol $s \in S'$ with the highest $\Delta(s,p)$ value. In case of a tie, use the tie break procedure of Figure 3.
8. $seq_p$ is obtained by sequencing $s_p^*$ in $seq_{p-1}$
9. Next $p$
10. Return $seq_D$

**Figure 5**. The pseudocode of the enhanced heuristic

- $M = 2$
- $\Delta(s,p) = \begin{cases} t(s,p) - \hat{t}(s,p) & \text{if } (d_s \geq 2) \wedge \left(\hat{d}(s,p) \geq 2\right) \\ \left[t(s,p) - \hat{t}(s,p)\right] + \left[\hat{t}(s,p) - (D + fsp(s) - p\right] & \text{if } (d_s \geq 2) \wedge \left(\hat{d}(s,p) = 1\right) \\ 0 & \text{if } d_s = 1 \end{cases}$

  $p = 1, \ldots, D$, $\forall s \in S_1(p) \cup S_3(p)$
- $\tilde{S}_3(p,q) = \{s \in S_3(p) : \Delta(s,p) + (q - p) \geq 0\}$; $p = 1, \ldots, D$, $q = p+1, \ldots, D$

*Time Complexity-*

Note that the parameters $D$ and $n$ will bound the run time of the algorithm (see Figure 5). More specifically, the time complexity of our heuristic is polynomial and of the order $O(D \cdot n)$. In brief this is because for each iteration $p = 1, ..., D$, the time complexity of the operations to do is $O(n)$.

## 4. Computational results for the RTVP

To assess the performance of our proposed heuristic we conduct a large experiment of around 800 instances and compare our results against the best from the classical heuristics as well as the meta-heuristics. Lower bounds are also reported for completeness. All algorithms are coded in Java and executed on a 3.4 GHz Pentium IV with 1.5 GB of RAM.

### *4.1. Comparison vs. the best classical heuristics*

The proposed heuristic is compared with the five best existing classical heuristics proposed (Corominas *et al*., 2009). Those are known as *Oc*, *AWe/dg*, *We/dg*, *Je/dg* and *In*. In their study, 600 test instances were used, which were grouped into three classes according to size (classes *CAT1* to *CAT3*, with 200 instances in each class). In this study, we also add 200 other larger test instances under class *CAT4*. All instances were generated using the random values of $D$ (total number of copies) and $n$ (number of symbols) shown in Table 1. For all instances and for each symbol $s = 1, \ldots, n$, a random number of copies to be sequenced of model $s$ ($d_s$) is randomly generated between 1 and $\left|(D - n + 1)/2.5\right|$ such that $\sum_{s=1..n} d_s = D$. The 800 instances are available at http://www.ioc.upc.edu/EOLI/research.

**Table 1**. Uniform distribution for the $D$ and $n$ values of the test instances

| | *CAT1* | *CAT2* | *CAT3* | *CAT4* |
|---|---|---|---|---|
| ***D*** | U(25, 50) | U(50, 100) | U(100, 200) | U(200, 500) |
| ***n*** | U(3, 15) | U(3, 30) | U(3, 65) | U(3, 150) |

The results are analysed by considering all the sets of instances as well as in each class of instances (*CAT1* to *CAT4*). The average RTV values of the solutions obtained by the proposed adaptive heuristic (let it be called *ENH-H*) and those from other heuristics are given in Table 2. For simplicity we do not report the solutions generated by our initial heuristic, though these were better than most existing constructed heuristics, these were as expected outperformed by those found with our enhanced version namely *ENH-H*.

We can see in Table 2 that *Oc* was the best existing heuristic in the literature. This observation is valid for the overall RTV averages as well as in each class of instances (*CAT1* to *CAT4*). On the other hand, our heuristic (*ENH-H*) obtains, on average, better solutions than *Oc*. If we consider the results by class, *ENH-H* is 6.91%, 17.99%, 31.80% and 36.88% better than *Oc* for *CAT1*, *CAT2*, *CAT3* and *CAT4* instances, respectively. Thus, the results point that the larger the instance, the more competitive is our heuristic. Moreover, *ENH-H* is much faster than *Oc* as it is shown in Table 3. On average, *ENH-H* requires only 1.82 milliseconds to solve an instance, whereas *Oc* needs 1,479.99 milliseconds (i.e., nearly 810 times slower). In summary, our *ENH-H* is the best performer by far in terms of both solution quality and computational effort.

**Table 2**. Average RTV values obtained by the classical heuristics

| | **Global** | ***CAT1*** | ***CAT2*** | ***CAT3*** | ***CAT4*** |
|---|---|---|---|---|---|
| ***ENH-H*** | **144.30** | **26.96** | **60.85** | **135.45** | **353.92** |
| ***Oc*** | 215.61 | 28.96 | 74.20 | 198.61 | 560.68 |
| ***Awe/dg*** | 405.88 | 47.03 | 120.32 | 349.13 | 1,107.03 |
| ***We/dg*** | 434.56 | 50.93 | 129.62 | 376.27 | 1,181.43 |
| ***Je/dg*** | 594.51 | 57.52 | 164.19 | 499.72 | 1,656.61 |
| ***In*** | 778.51 | 121.16 | 308.45 | 658.21 | 2,026.21 |

**Table 3**. Average computing time (in milliseconds) used by the classical heuristics

| | **Global** | ***CAT1*** | ***CAT2*** | ***CAT3*** | ***CAT4*** |
|---|---|---|---|---|---|
| ***ENH-H*** | **0.72** | **0.12** | **0.22** | **0.43** | **2.12** |
| ***Oc*** | 1,479.99 | 13.38 | 83.32 | 511.66 | 5,311.62 |
| ***Awe/dg*** | 4.56 | 0.86 | 1.45 | 3.91 | 12.01 |
| ***We/dg*** | 4.42 | 0.65 | 1.35 | 4.27 | 11.41 |
| ***Je/dg*** | 3.47 | 0.15 | 0.55 | 4.06 | 9.12 |
| ***In*** | 0.48 | 0.30 | 0.30 | 0.40 | 0.90 |

*Robustness of the solutions*

The dispersion with respect to the best RTV value obtained is also recorded. A measure of the dispersion (let it be $\sigma$) of the RTV values obtained by each algorithm, say *alg*, for a given instance, say *ins*, is defined as $\sigma(alg,ins) = \left(\left(\mathrm{RTV}_{ins}^{(alg)} - \mathrm{RTV}_{ins}^{(best)}\right) \big/ \mathrm{RTV}_{ins}^{(best)}\right)^2$, where $\mathrm{RTV}_{ins}^{(alg)}$ is the RTV value of the solution obtained with the algorithm *alg* for the instance *ins*, and $\mathrm{RTV}_{ins}^{(best)}$ is, for the instance *ins*, the best RTV value of the solutions obtained with all heuristics. Table 4 shows the average $\sigma$ dispersion values.

*ENH-H* and *Oc* both obtain low averages of the σ dispersion values. This indicates that both algorithms are very stable especially our enhanced heuristic which besides outperforming all the other heuristics, it is found to be extremely robust and consistent in generating excellent results.

**Table 4**. Average σ dispersion values regarding the best solution found by the classical heuristics

| | Global | *CAT1* | *CAT2* | *CAT3* | *CAT4* |
|---|---|---|---|---|---|
| ***ENH-H*** | **0.11** | 0.26 | **0.07** | **0.03** | **0.07** |
| ***Oc*** | 0.27 | **0.21** | 0.23 | 0.35 | 0.28 |
| ***Awe/dg*** | 7.15 | 4.01 | 3.50 | 4.89 | 16.21 |
| ***We/dg*** | 9.09 | 4.44 | 4.11 | 6.45 | 21.36 |
| ***Je/dg*** | 22.18 | 8.28 | 8.57 | 15.66 | 56.22 |
| ***In*** | 48.26 | 54.51 | 85.66 | 21.82 | 31.06 |

### *4.2. Comparison vs metaheuristics*

We also compare the results of our heuristic with the best results obtained by the GA of García-Villoria and Pastor (2010c). In this scenario, a set of 740 test instances is used instead. This is a subset of the 800 test instances (the other 60 instances were used to calibrate the parameters of the GA in their study). As in the previous subsection, these 740 instances are also grouped into four classes according to size (classes *CAT1'* to *CAT4'*, with 185 instances in each class). Table 5 shows the averages of the RTV values obtained by our proposed heuristic and the GA with 10, 50, 200, 500 and 1,000 seconds of computing time.

**Table 5**. Average RTV values for a computing time of 10, 50, 200, 500 and 1,000 seconds

| | | Global | *CAT1'* | *CAT2'* | *CAT3'* | *CAT4'* |
|---|---|---|---|---|---|---|
| ***ENH-H*** | | 159.50 | 27.56 | 62.76 | 151.91 | 395.77 |
| | **10 secs** | 1,245.10 | 12.13 | 31.85 | 111.47 | 4,824.94 |
| | **50 secs** | 186.94 | 11.65 | 29.41 | 84.54 | 622.16 |
| **GA** | **200 secs** | 131.81 | 11.34 | 28.26 | 77.81 | 409.84 |
| | **500 secs** | 114.39 | 11.00 | 27.63 | 75.59 | 343.33 |
| | **1,000 secs** | 106.68 | 10.92 | 27.00 | 74.86 | 313.92 |

On average, the GA is able to improve *ENH-H*. Observing the results by class, the metaheuristic algorithm obtains, on average, better solutions for all type of instances (*CAT1'* to *CAT4'*), though these results are not directly comparable due to the large difference in the computing times. For instance, the GA needs more than 200 seconds to obtains better results for the largest instances (*CAT4'*) while our heuristic requires a tiny fraction of a second (0.72 milliseconds) only. As our heuristic is so fast and generates reasonably good solutions, it could be an invaluable tool to be incorporated within other powerful meta-heuristics for the generation of the initial solution.

*4.3. Comparison vs optimal solutions or lower bounds*

We have tried to find the optimal solutions using the proposed B&B procedure proposed in Garcia-Villoria *et al.,* (2009) to provide additional information regarding the effectiveness of the proposed heuristic. However, only the smallest instances (*CAT1'* instances) were optimally solved (using a limit execution time of 15 hours per instance). For the remaining instances, we used the lower bound proposed in Corominas *et al*. (2007) (see Section 2). Table 6 shows the averages of the optimal RTV values ( $\overline{OPT}$ , where *NA* indicates that no average is available), the averages of the lower bounds ( $\overline{LB}$ ) and the averages of the RTV values obtained by *ENH-H* ( $\overline{RTV}$ ).

**Table 6**. Compariosn vs the RTV lower bounds

| | *CAT1'* | *CAT2'* | *CAT3'* | *CAT4'* |
|---|---|---|---|---|
| $\overline{LB}$ | 5.35 | 10.95 | 21.15 | 48.15 |
| $\overline{OPT}$ | 10.24 | *NA* | *NA* | *NA* |
| $\overline{RTV}$ | 27.56 | 62.76 | 151.91 | 395.77 |

Table 6 indicates that these lower bounds are either too loose and hence not very informative or the solutions obtained by our heuristic are very poor. These were found to be nearly 100% above those lower bounds for the instances in the class CAT1' but very close to those obtained by the GA. In other words, the above lower bounds are too simplistic to be useful whereas it seems that those upper bounds found by the GA could be better used for comparison instead.

## 5. The *minmax* RTVP

As our approach is flexible enough to cater for other type of objective functions, in this paper we introduce a related RTVP which we refer to as the *minmax* RTVP. Here, the objective is to minimise the metric that we call the maximum response time variability (*max*RTV) instead of the RTV. This is defined by the maximum of the absolute errors with respect to the $\overline{t}_s$ distances, $maxRTV = \max_{s=1}^{n} \max_{k=1}^{d_s} \left( \left| t_k^s - \overline{t}_s \right| \right)$.

For an illustration, consider the following example. Let $n = 3$ with symbols *H*, *I* and *J*. Also consider $d_H = 2$ , $d_I = 2$ and $d_J = 4$ ; thus, $D = 8$ , $\overline{t}_H = 4$ , $\overline{t}_I = 4$ and $\overline{t}_J = 2$. Any sequence such that contains symbol $s$ $(\forall s)$ exactly $d_s$ times is a feasible solution. For example, the sequence (J, H, J, I, J, I, H, J) is a feasible solution. The *maxRTV* value of the illustrative example is, therefore,

$$\max\left(\max\left(|5-4|,|3-4|\right),\max\left(|2-4|,|6-4|\right),\max\left(|2-2|,|2-2|,|3-2|,|1-2|\right)\right)=12 .$$

*Computational results*

The *minmax* RTVP is solved for all 800 instances using our proposed *ENH-H* algorithm. Since this is the first time in the literature this related problem is presented, there is obviously no comparison with other existing results. In Table 7, we provide some basic statistics (the best, average and the worst *maxRTV* value) which can be used for future benchmarking purposes which hopefully will entice other researchers to investigate this particular scheduling problem.

**Table 7**. Basic statistics obtained by *ENH-H* on the *max*RTV

| | ***max*RTV** | **Global** | ***CAT1*** | ***CAT2*** | ***CAT3*** | ***CAT4*** |
|---|---|---|---|---|---|---|
| ***ENH-H*** | **Best** | 0.58 | 0.91 | 0.58 | 1.38 | 1.58 |
| | **Average** | 3.17 | 2.19 | 2.72 | 3.43 | 4.33 |
| | **Worst** | 10.18 | 4.00 | 5.14 | 7.55 | 10.18 |

## 6. Conclusions and future research

This paper proposes a new constructive greedy heuristic based on an adaptive search to solve the Response Time Variability Problem (RTVP). The RTVP is an NP-hard scheduling problem that appears in a broad range of real-life applications. Several heuristics and metaheuristic algorithms have been proposed in previous studies to solve the RTVP. The best solutions have been achieved by means of metaheuristics, but they need a lot of computing time (1,000 seconds). On the other hand, classical heuristics require a fraction of that amount only, but the solutions were usually found to be inferior.

The heuristic that we propose improves upon the performance of the best existing classical heuristics in terms of solution quality and computing time. Moreover, the solutions obtained are also competitive with the best solutions found by the existing metaheuristics while requiring a fraction of their computing time especially for the largest tested instances. In addition, we adopted this heuristic to tackle a related but a new scheduling problem namely the *minmax* RTVP with computational results for benchmarking purposes. The complexity of this new problem is unknown but it could be derived from the NP-hardness of the RTVP.

A promising line of research is to develop additional properties to make the heuristic even more powerful. Another simple way is to incorporate post optimisation. For instance partial enumeration can easily be implemented a few positions before the end, local search procedures as well as metaheuristics such as tabu search or simulated annealing can also be introduced.

Another interesting and exciting line of research is that, given the extremely reduced computing time of our heuristic, it can be incorporated as part of some exact methods such as the B&B algorithm proposed in Corominas *et al*. (2009) for providing tighter upper bounds. These extra information could be used either as an additional constraint as part of the formulation or within a B&B node with the aim of improving the search strategy used for selecting the next node to be explored. The integration of exact methods and heuristics is emphasized by Salhi (2006) as one of the promising future research avenues within heuristic search.

From a practical view point, other metrics to define the fairness could also be attempted for this exciting scheduling problem. The commonly used measure between two successive symbols is one unit of distances, but this could be generalised to be dependent on the type of symbols and their relationships. This additional feature will obviously make the problem more complex but practically interesting and academically challenging.

***Acknowledgments-*** The authors would like to thank both referees for their time and effort in improving the presentation as well as the content of the paper.

**REFERENCES**

Anily, S., Glass, C.A. and Hassin, R. (1998) 'The scheduling of maintenance service', *Discrete Applied Mathematics*, **82**, 27-42.

Balinski, M.L. and Young, H.P. (1982) *Fair Representation*, Yale University Press, New Haven.

Bollapragada, S., Bussieck, M.R. and Mallik, S. (2004) 'Scheduling Commercial Videotapes in Broadcast Television', *Operations Research*, **52**, 679-689.

Brusco, M.J. (2008) 'Scheduling advertising slots for television', *Journal of the Operational Research Society*, **59**, 1363-1372.

Corominas, A., Kubiak, W. and Moreno, N. (2007) 'Response time variability', *Journal of Scheduling*, **10**, 97-110.

Corominas, A., Kubiak, W. and Pastor, R. (2009) 'Heuristic algorithms for solving the Response Time Variability problem', *Technical report IOC-DT-P-2009-03*, Universitat Politècnica de Catalunya, Spain.

Corominas, A., Kubiak, W. and Pastor, R. (2010) 'Mathematical programming modeling of the Response Time Variability Problem', *European Journal of Operational Research*, **200**, 347-357.

Dong, L., Melhem, R. and Mosse, D. (1998) 'Time slot allocation for real-time messages with negotiable distance constrains requirements', *Fourth IEEE Real-Time Technology and Applications Symposium* (RTAS'98), Denver, CO., 131-136.

García-Villoria, A., Corominas, A., Delorme, X., Dolgui, A., Kubiak, W. and Pastor, R. (2009) 'A branch and bound approach for the response time variability problem', *Technical report IOC-DT-P-2009-05*, Universitat Politècnica de Catalunya, Spain.

García-Villoria, A. and Pastor, R. (2010a) 'Solving the Response Time Variability Problem by means of the Electromagnetism-like Mechanism', *International Journal of Production Research*, **48**, 6701-6714.

García-Villoria, A. and Pastor, R. (2010b) 'Solving the Response Time Variability Problem by means of a psychoclonal approach', *Journal of Heuristics*, **16**, 337-351.

García-Villoria, A. and Pastor, R. (2010c) 'Solving the response time variability problem by means of a genetic algorithm', *European Journal of Operational Research*, **202**, 320-327.

Herrmann, J.W. (2007) 'Generating Cyclic Fair Sequences using Aggregation and Stride Scheduling', *Technical Report TR 2007-12*, University of Maryland, USA. Available at http://hdl.handle.net/1903/7082.

Kubiak, W. (1993) 'Minimizing variation of production rates in just-in-time systems: A survey', *European Journal of Operational Research*, **66**, 259-271.

Kubiak, W. (2004) 'Fair Sequences', Chapter 19 in *Handbook of Scheduling: Algorithms, Models and Performance Analysis*, Chapman and Hall.

Miltenburg, J. (1989) 'Level schedules for mixed-model assembly lines in just-in-time production systems', *Management Science*, **35,** 192-207.

Salhi, S. (2006) 'Heuristic Search: the Science of Tomorrow', in Salhi S (Ed), *OR48 Key Note Papers*, Bath, ORS, 39-58.

Waldspurger, C.A. and Weihl, W.E. (1994) 'Lottery Scheduling: Flexible Proportional-Share Resource Management', *First USENIX Symposium on Operating System Design and Implementation*, Monterey, California.

Waldspurger, C.A. and Weihl, W.E. (1995) 'Stride Scheduling: Deterministic Proportional-Share Resource Management', *Technical Report MIT/LCS/TM-528*, Massachusetts Institute of Technology, MIT Laboratory for Computer Science. Available at https://eprints.kfupm.edu.sa/67117.

Wei, W.D. and Liu, C.L. (1983) 'On a periodic maintenance problem', *Operations Research Letters*, **2,** 90-93.