



**HAL**  
open science

# Solving the Response Time Variability Problem by means of the Electromagnetism-like Mechanism

Alberto García-Villoria, Rafael Pastor

## ► To cite this version:

Alberto García-Villoria, Rafael Pastor. Solving the Response Time Variability Problem by means of the Electromagnetism-like Mechanism. International Journal of Production Research, 2009, pp.1. <10.1080/00207540902862545>. <hal-00545667>

**HAL Id: hal-00545667**

**<https://hal.science/hal-00545667v1>**

Submitted on 11 Dec 2010

**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.



HAL Authorization



**Solving the Response Time Variability Problem by means of the Electromagnetism-like Mechanism**

Journal:	<i>International Journal of Production Research</i>
Manuscript ID:	TPRS-2008-IJPR-0622.R1
Manuscript Type:	Original Manuscript
Date Submitted by the Author:	24-Dec-2008
Complete List of Authors:	García-Villoria, Alberto; Institute of Industrial and Control Engineering, Department of Management Pastor, Rafael; Institute of Industrial and Control Engineering, Department of Management
Keywords:	SEQUENCING, META-HEURISTICS, SCHEDULING, JIT
Keywords (user):	response time variability, electro-like mechanism



**Author 1 (corresponding author)**

- Name: Alberto García-Villoria
- Affiliations: Institute of Industrial and Control Engineering (IOC),  
Department of Management, Technical University of Catalonia  
(UPC)
- Postal address: Av. Diagonal 647 (Edif. ETSEIB), 11th floor, 08028 Barcelona,  
Spain
- Telephone: +34 93 405 4010
- Fax: +34 93 401 6605
- Email: alberto.garcia-villoria@upc.edu

**Author 2**

- Name: Rafael Pastor Moreno
- Affiliations: Institute of Industrial and Control Engineering (IOC),  
Department of Management, Technical University of Catalonia  
(UPC)
- Postal address: Av. Diagonal 647 (Edif. ETSEIB), 11th floor, 08028 Barcelona,  
Spain
- Telephone: +34 93 401 1701
- Fax: +34 93 401 6605
- Email: rafael.pastor@upc.edu

# Solving the Response Time Variability Problem by means of the Electromagnetism-like Mechanism\*

The Response Time Variability Problem (RTVP) is an NP-hard combinatorial scheduling problem that has been recently formalised in the literature. The RTVP has a wide range of real-life applications such as in the automobile industry, when models to be produced on a mixed-model assembly line have to be sequenced under a just-in-time production. The RTVP occurs whenever products, clients or jobs need to be sequenced so as to minimize variability in the time between the instants at which they receive the necessary resources. In two previous studies, three metaheuristic algorithms (a multi-start, a GRASP and a PSO algorithm) were proposed to solve the RTVP. We propose solving the RTVP by means of the electromagnetism-like mechanism (EM) metaheuristic algorithm. The EM algorithm is based on an analogy with the attraction-repulsion mechanism of the electromagnetism theory, where solutions are moved according to their associated charges. In this paper we compare the proposed EM metaheuristic procedure with the three metaheuristic algorithms aforementioned and it is shown that, on average, the EM procedure improves strongly on the obtained results.

**Keywords:** response time variability; fair sequences; scheduling; just-in-time; metaheuristics; electromagnetism-like mechanism

## 1. Introduction

The Response Time Variability Problem (RTVP) is a scheduling problem that has recently been defined in the literature (Corominas *et al.* 2007). The RTVP occurs whenever products, clients or jobs need to be sequenced so as to minimize variability in the time between the instants at which they receive the necessary resources. Although this combinatorial optimisation problem is easy to formulate, it is very difficult to solve (it is NP-hard, Corominas *et al.* 2007).

The RTVP has a broad range of real-life applications. For example, it can be used to sequence regularly models in the automobile industry (Ding and He 2007), to resource allocation in computer multi-threaded systems and network servers (Waldspurger and Weihl 1995), to broadcast video and sound data frames of applications over asynchronous transfer mode networks (Dong *et al.* 1998), in the periodic machine maintenance problem when the distances between consecutive services of the same machine are equal (Anily *et al.* 1998) and in the collection of waste (Herrmann 2007).

One of the first problems in which has appeared the importance of sequencing regularly is at the sequencing on the mixed-model assembly production lines at Toyota Motor Corporation under the just-in-time (JIT) production system. One of the most important JIT objectives is to get rid of all kinds of waste and inefficiency and, according to Toyota, the main waste is due to the stocks. To reduce the stock, JIT production systems require to producing only the necessary models in the necessary quantities at the necessary time (Aigbedo 2004). To achieve this, one main goal, as Monden (1983) says, is scheduling the units to be produced to keep a constant

\* An earlier version of this paper has been published on-line at <http://hdl.handle.net/2117/2013>

1  
2  
3 consumption rates of the components involved in the production process. Miltenburg  
4 (1989) deals with this scheduling problem considering only the demand rates of the  
5 models. He proposed four metrics to measure the regularity of a sequence based on the  
6 discrepancies, for each model, between the real production rate and the ideal one (i.e.,  
7 the one that would correspond to a constant rate of production). This problem is known  
8 as the Product Rate Variation (PRV) problem (Kubiak 1993). The PRV problem has  
9 been reformulated by Kubiak and Sethi (1994) as an assignment problem and, therefore,  
10 it can be solved efficiently.  
11  
12

13 Although the sequencing on the mixed-model assembly production lines is usually  
14 considered in the literature as a PRV problem (Miltenburg 1989, Kubiak 1993, Steiner  
15 and Yeomans 1993), in our experience with practitioners of manufacturing industries  
16 we noticed that they usually refer to a good mixed-model sequence not in terms of ideal  
17 production, but in terms of having distances between the units for the same model as  
18 regular as possible. Therefore, the metric used in the RTVP reflects the way in which  
19 practitioners refer to a desirable regular sequence.  
20

21 In this paper, the electromagnetism-like mechanism (EM) metaheuristic is proposed  
22 to solve the RTVP. EM is a recent population-based metaheuristic that was first  
23 proposed by Birbil and Fang (2003). It is based on an analogy with the attraction-  
24 repulsion mechanism of electromagnetism theory. Each solution is considered as a point  
25 with an electrical charge that is measured by the objective function. This charge  
26 determines the magnitude of attraction or repulsion of the other points for applying the  
27 electromagnetism equations and EM iteratively calculates the movement of the points.  
28

29 The EM metaheuristic has yielded good results when it has been used to solve  
30 several combinatorial optimisation problems (Debels and Vanhoucke 2006, Debels et  
31 al. 2006, Yuan et al. 2006, Maenhout and Vanhoucke 2007, Chang et al. 2009). The EM  
32 algorithm proposed to solve the RTVP is compared with efficient procedures for  
33 solving non-small instances: the multi-start and the GRASP algorithms proposed in  
34 García et al. (2006) and the PSO algorithm called *DPSOpoi-c<sub>p</sub>dyn* proposed in García-  
35 Villoria and Pastor (2009).  
36

37 The remainder of this paper is organized as follows. Section 2 presents a formal  
38 definition of the RTVP. Section 3 briefly exposes three metaheuristic algorithms  
39 presented in García et al. (2006) and García-Villoria and Pastor (2009). Section 4  
40 describes the basic scheme of the EM. Section 5 proposes a procedure an EM algorithm  
41 for solving the RTVP. Section 6 provides the computational experiments and the  
42 comparison with the other metaheuristic algorithms. Finally, some conclusions and  
43 suggestions for future research are given in Section 7.  
44  
45  
46  
47  
48

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

50 The aim of the Response Time Variability Problem (RTVP) is to minimise variability in  
51 the distances between any two consecutive units of the same model to be scheduled.  
52

53 The RTVP is formulated as follows. Let  $n$  be the number of models,  $d_i$  the number  
54 of units to be scheduled of model  $i$  ( $i = 1, \dots, n$ ) and  $D$  the total number of units  
55 ( $D = \sum_{i=1..n} d_i$ ). Let  $s$  be a solution of a RTVP instance that consists of a circular  
56 sequence of units ( $s = s_1 s_2 \dots s_D$ ), where  $s_j$  is the unit sequenced in position  $j$  of  
57 sequence  $s$ . For all unit  $i$  in which  $d_i \geq 2$ , let  $t_k^i$  be the distance between the positions in  
58 which the units  $k + 1$  and  $k$  of the model  $i$  are found (i.e. the number of positions  
59  
60

between them, where the distance between two consecutive positions is considered equal to 1). Since the sequence is circular, position 1 comes immediately after position  $D$ ; therefore,  $t_{d_i}^i$  is the distance between the first unit of the model  $i$  in a cycle and the last unit of the same model in the preceding cycle. Let  $\bar{t}_i$  be the average distance between two consecutive units of the model  $i$  ( $\bar{t}_i = D/d_i$ ). For all symbol  $i$  in which  $d_i = 1$ ,  $t_1^i$  is equal to  $\bar{t}_i$ . The objective is to minimize the metric Response Time Variability (RTV) which is defined by the following expression:

$$RTV = \sum_{i=1}^n \sum_{k=1}^{d_i} (t_k^i - \bar{t}_i)^2 \quad (1)$$

For example, let  $n=3$ ,  $d_A=2$ ,  $d_B=2$  and  $d_C=4$ ; thus,  $D=8$ ,  $\bar{t}_A=4$ ,  $\bar{t}_B=4$  and  $\bar{t}_C=2$ . Any sequence is a feasible solution. For example, the sequence (C, A, C, B, C, B, A, C) is a solution, where

$$RTV = ((5-4)^2 + (3-4)^2) + ((2-4)^2 + (6-4)^2) + ((2-2)^2 + (2-2)^2 + (3-2)^2 + (1-2)^2) = 12.$$

### 3. Three metaheuristic algorithms for the RTVP

Corominas *et al.* (2007) proposed a mixed integer lineal programming (MILP) model to solve the RTVP. Corominas *et al.* (2006) improved the previous MILP model but the practical limit to obtain optimal solutions is 40 units to be scheduled. Thus, the use of heuristic or metaheuristic methods for solving real-life instances of the RTVP is justified. Corominas *et al.* (2007) proposed five heuristic algorithms. García *et al.* (2006) proposed six metaheuristic algorithms: a multi-start, a GRASP (Greedy Randomized Adaptive Search Procedure) and four PSO (Particle Swarm Optimisation) algorithms. Eleven variants of the PSO metaheuristic were also used to solve the RTVP in García-Villoria and Pastor (2009).

The three most effective aforementioned procedures are the multi-start and the GRASP algorithm proposed in García *et al.* (2006) and the PSO algorithm called *DPSPoi-c<sub>p</sub>dyn* proposed in García-Villoria and Pastor (2009). Next, the algorithms are briefly explained (for more details of the three algorithm procedures, see García *et al.* 2006 and García-Villoria and Pastor 2009).

#### 3.1. The Multi-start algorithm

The multi-start method is based on generating initial random solutions and on improving each of them to find a local optimum, which is usually done by means of a local search procedure (Martí 2003). Random solutions are generated as follows. For each position, a model to be sequenced is randomly chosen. The probability of each model is equal to the number of units of this model that remain to be sequenced divided by the total number of units that remain to be sequenced. The local search procedure used is applied as follows. A local search is performed iteratively in a neighbourhood that is generated by interchanging each pair of two consecutive units of the sequence that represents the current solution; the best solution in the neighbourhood is chosen; the optimisation ends when no neighbouring solution is better than the current solution.

### 3.2. The GRASP algorithm

GRASP, designed by Feo and Resende (1989), can be considered to be a variant of the multi-start method in which the initial solutions are obtained using directed randomness. The solutions are generated by means of a greedy strategy in which random steps are added and the choice of elements to be included in the solution is adaptive. The random step in the GRASP proposed by García *et al.* (2006) consists of selecting the next model to be added to the solution; the probability of each candidate model is proportional to the value of its Webster index, which is based on the parametric method of apportionment with parameter  $\delta = \frac{1}{2}$  (Balinski and Young 1982). The Webster index for model  $i$  ( $i = 1, \dots, n$ ) is evaluated as  $\frac{d_i}{(x_{it} + \delta)}$ , where  $x_{it}$  is the number of units of model  $i$  in the sequence of length  $t = 0, \dots, D$  (assume  $x_{i0} = 0$ ). The local search procedure applied to the initial solutions is the same local search as in the multi-start method.

### 3.3. The PSO algorithm

PSO is a population-based metaheuristic designed by Kennedy and Eberhart (1995), which is based on an analogy of the social behaviour of flocks of birds when they search for food. The population or swarm is composed of particles (birds), which have an  $m$ -dimensional real point (which represents a feasible solution) and a velocity (the movement of the point in a  $m$ -dimensional real space). The velocity of a particle is typically a combination of three kinds of velocities: 1) inertia velocity; 2) velocity to the best point found by the particle; and 3) velocity to the best point found by the swarm. These components of the particles are iteratively modified by the PSO algorithm as it looks for an optimal solution.

In the *DPSO<sub>poi-c<sub>p</sub>dyn</sub>* algorithm (García-Villoria and Pastor 2009), random modifications to the points of the particles are introduced. The frequency of the modifications changes dynamically according to the homogeneity of the swarm. The aim is to prevent premature convergence and to enable the PSO algorithm to escape a local optimum. Although the PSO metaheuristic was originally designed for working in a  $m$ -dimensional real space, *DPSO<sub>poi-c<sub>p</sub>dyn</sub>* is adapted to work with a sequence that represents the solution. In this adaptation, a point is now the sequence of units of the models that represents a solution, and the velocity is an ordered list of transformations that must be applied to the particle so that it changes from its current point to another point; each transformation consists of a pair of positions of the point (sequence) that are to be swapped. The velocity to the best point found by the particle is the list of transformations needed to obtain the best particle point from the current position; the same applies for the velocity to the best point found by the swarm.

## 4. The EM metaheuristic

The electromagnetism-like mechanism (EM) metaheuristic is a new population-based metaheuristic created by Birbil and Fang (2003). The EM metaheuristic has been applied successfully to the following problems: the project scheduling problem (Debels and Vanhoucke 2006; Debels *et al.* 2006), neural network training (Wu *et al.* 2004), the permutation flowshop scheduling problem (Yuan *et al.* 2006), the nurse scheduling

problem (Maenhout and Vanhoucke 2007), the single machine scheduling problem (Chang et al. 2009) and multi-objective optimisation problems (Tsou and Kao 2008). On the other hand, in the Birbil's PhD thesis (Birbil 2002), the EM metaheuristic is compared with other methods and shown to have substantial performance.

The EM metaheuristic operates basically as follows. EM starts with an initial population of solutions that will be attracted to the deep valleys and repulsed from the steep hills (if we wish to minimise the value of the solutions). Each solution can be thought of as a particle charged according to its objective function value. Then, an analogy of the attraction-repulsion mechanism of the electromagnetism theory can be applied. Moreover, some solutions are improved by a local search.

Next, we present the framework of the EM metaheuristic; for further details, see Birbil and Fang (2003). This algorithm works with a special class of optimisation problems with bounded variables in the following form:

$$\begin{aligned} & \min (\max) && f(x) \\ & \text{subject to } && x \in \mathfrak{R}^m \mid l_j \leq x_j \leq u_j, j = 1, \dots, m \end{aligned}$$

where  $f$  is the function that evaluates a point (which represents a solution),  $m$  is the dimension of the problem (in the case of the RTVP,  $m$  would be equal to  $D$ , which is the total number of units) and  $x_j$  is the coordinate of the  $j$ th dimension, which is lower bounded by  $l_j$  and upper bounded by  $u_j$ .

The EM metaheuristic is divided into four phases (which are explained in Subsections 4.1 to 4.4): 1) the initialization of the population of the points; 2) the application of the local search; 3) the calculation of the total force vector; and 4) the movement according to the total force. The pseudocode of the metaheuristic is shown in Figure 1.

[Insert Figure 1 here]

#### 4.1. Initial population

The metaheuristic starts generating randomly the initial population, which consists of  $p$  points of the feasible domain. Each coordinate of each point is uniformly distributed between their upper and lower bounds.

#### 4.2. Local search

The local search procedure provides the EM algorithm with a good balance between the exploration and exploitation of the feasible region. Birbil and Fang (2003) propose two approaches according to the points to which the local search can be applied: local search applied to all points and local search applied only to the current best point.

Local search applied to all points promotes a more meticulous examination of the region around the points. However local search applied only to the best point usually gives as good results and less time is spent on the local search.

In both cases, a simple local search is recommended rather than a powerful one because it is enough for a good convergence (Birbil and Fang 2003). The local search is not applied until a local optimal point is reached; the local search stops when a number of iterations (let it be called  $lsiter$ ) is executed.

#### 4.3. Calculation of the total force vector

The charge of each point  $x$  belonging to the population  $P$  (let it be called  $q_x$ ), which determines the intensity of attraction or repulsion of the point, changes at each iteration of the EM **metaheuristic**. The charge is first evaluated as follows:

$$q_x = \exp \left( -m \frac{f(x) - f(x^{best})}{\sum_{y \in P} (f(y) - f(x^{best}))} \right) \quad (2)$$

Note that, unlike electrical charges, no signs are associated with the charges. The direction of a particular force between two points is determined when their objective values have been compared. The total force for each point belonging to the population  $P$  (let it be called  $F_x$ ) is evaluated as follows:

$$F_x = \sum_{y \in P | y \neq x} \left\{ \begin{array}{l} (y-x) \frac{q_x q_y}{\|y-x\|^2} \quad \text{if } f(y) < f(x) \quad (\text{Attraction}) \\ (x-y) \frac{q_x q_y}{\|y-x\|^2} \quad \text{if } f(y) \geq f(x) \quad (\text{Repulsion}) \end{array} \right\} \quad (3)$$

where  $\|y-x\|$  is the euclidean distance between the two points.

#### 4.4. Movement according to the total force

Each point  $x$  belonging to the population  $P$  is moved according to the next equation:

$$x = x + \lambda \frac{F_x}{\|F_x\|} \quad (4)$$

where  $\lambda$  denotes a random number uniformly distributed between 0 and 1 and  $\|F_x\|$  is the norm of the force vector. The parameter  $\lambda$  is used to ensure that the points have a nonzero probability of moving to the unvisited regions in this direction. Furthermore, the force applied to each point is normalized, so the feasibility is maintained (i.e., each coordinate of each point will be between  $l_j$  and  $u_j$ ).

### 5. The EM algorithm for the RTVP

The objective function and the equations of the EM metaheuristic work with points of a region of the  $m$ -dimensional real space. Others procedures such as PSO algorithms or other **optimisation** algorithms of real variables are also designed for working in an  $m$ -dimensional real space. However, a solution of many combinatorial **optimisation** problems is usually represented as an ordered sequence of integer numbers (as in the RTVP), so these metaheuristics (EM, PSO and others) are incompatible with this

1  
2  
3 representation of the solution as an ordered sequence of integer numbers. There are two  
4 ways of applying algorithms of this kind to the RTVP: to adapt the algorithm to work  
5 with a sequence of integer numbers or to adapt the representation of the solution as an  
6  $m$ -dimensional real point.  
7

8 To adapt the PSO algorithm to a sequence of integer numbers for the RTVP is done  
9 in García-Villoria and Pastor (2009), as explained in Section 2. As would happen in the  
10 EM algorithm, this way involves redefining several mathematical operators used by the  
11 algorithm. For example, the difference between two points  $((y-x)$  and  $(x-y)$  in  
12 Equation 3) would now be the difference between two sequences of integer numbers  
13 and this new **different** operator **should** be defined.  
14

15 On the other hand, a sequence of integer numbers can be represented by an  $m$ -  
16 dimensional real point using random key representation (RK) (Bean 1994). The main  
17 advantage of using RK is that each solution corresponds to a real point, so that  
18 geometric operations (for example, the evaluation of a point charge (Equation 2)) can be  
19 performed on its components. Since geometric operations are the cornerstone of several  
20 metaheuristics (such as EM), RK allows a straightforward application of this type of  
21 metaheuristics to solve combinatorial optimisation problems. Although there is  
22 empirical evidence that Genetic Algorithms that applies RK may obtain slight worst  
23 results than Genetic Algorithms adapted to the combinatorial problem (Bean 1994), in  
24 other metaheuristics there is no conclusion about which approach is better. In the case  
25 of EM algorithms, to the best of our knowledge, most of the papers in which an EM  
26 algorithm is proposed to solve a combinatorial optimisation problem RK is used  
27 (Debels and Vanhoucke 2006; Debels et al. 2006; Yuan et al. 2006; Chang et al. 2009)  
28 and only one paper adapts EM (Maenhout and Banhoucke, 2007). Because RK have  
29 been effectively applied to the EM metaheuristic to solve several combinatorial  
30 problems, in this paper RK is also used in the EM algorithm for solving the RTVP.  
31  
32  
33

34 Random key representation for the RTVP is explained in Subsection 4.1. How the  
35 initial population is generated is described in Subsection 4.2. The local search used in  
36 our EM algorithm is explained in Subsection 4.3. The calculations of the total force  
37 vectors and the movements according to the total force are directly implemented  
38 according to Equations (2), (3) and (4). Finally, Subsection 4.4 explains the fine-tuning  
39 of the parameter values of the EM algorithm: the size of the initial population ( $p$ ) and  
40 the maximum number of iterations of the local search procedure ( $l_{siter}$ ).  
41  
42  
43

#### 44 45 **5.1. Random key representation**

46  
47 Random key representation (Bean 1994) consists of an  $m$ -length sequence of different  
48 real numbers called *keys*. Let the key sequence be  $r = r_1, \dots, r_m$ , where  $r_j$  is the key of  
49 the position  $j$ . In the context of the proposed EM algorithm, the key sequence has  $D$   
50 (number of units to be sequenced with  $d_i$  units of model  $i$ ) keys. As the EM  
51 metaheuristic works with bounded variables, the values of the keys are bounded  
52 between 0 and 1.  
53

54 Given a key sequence  $r$ , the solution  $s$  (sequence of models) that is represented by  $r$   
55 is as follows. First, for each position  $j = 1, \dots, D$  of  $r$ , key  $r_j$  is associated with a model.  
56 The association is done in a way that, for each model  $i$ , there are  $d_i$  consecutive keys  
57 associated with model  $i$ . For each key sequence, the association for key  $r_j$  will be always  
58 done with the same model, i.e., if, for example, key  $r_1$  is associated with model A in  
59 every key sequence  $r$ ,  $r_1$  will be associated with this model. Next, a new key sequence,  
60  $r'$ , is obtained by putting  $r$  (and therefore their associated models) in descending order

according to the values of the keys. Then, for each position  $j = 1, \dots, D$ , model  $s_j$  is the model associated with key  $r'_j$ , i.e., the model sequenced in position  $j$  is the model associated with key  $r_j$  that is in the position  $j$  in the key sequence  $r'$ .

For example, let a RTVP instance be  $n = 3$ ,  $d_A = 2$ ,  $d_B = 2$  and  $d_C = 4$ . Given the key sequence  $r = (0.12, 0.26, 0.67, 0.08, 0.14, 0.45, 0.87, 0.62)$ , each key  $r_j$  ( $j = 1, \dots, 8$ ) is associated with a model as follows:

models	A	A	B	B	C	C	C	C
	↑	↑	↑	↑	↑	↑	↑	↑
keys	0.12	0.26	0.67	0.08	0.14	0.45	0.87	0.62

So, the descending ordered key sequence is  $r' = (0.87, 0.67, 0.62, 0.45, 0.26, 0.14, 0.12, 0.08)$  and, therefore, the solution represented is (C, B, C, C, A, C, A, B).

## 5.2. Initial population

The initial population of points consists of  $p$  solutions generated randomly. As has been introduced previously, each solution is represented by a key sequence where each key value is bounded between 0 and 1. To get a solution, we generate a random value uniformly distributed in  $[0,1]$  for each key.

## 5.3. Local search

The local search procedure used in the EM algorithm is as follows. A local search is performed iteratively in a neighbourhood that is generated by interchanging two units of different consecutive and non-consecutive models; the first solution found in the neighbourhood that is better than the current solution is selected; the **optimisation** ends when the maximum number of iterations is reached or no neighbouring solution is better than the current solution.

Local search applied to all points and local search applied only to the best point were tested by an initial experiment. To apply the local search only to the best point provided much better solutions for the RTVP, so the local search applied only to the best point is used in the EM algorithm.

## 5.4. Fine-tuning of the EM parameters

Fine-tuning the parameters of a metaheuristic algorithm is almost always a difficult task. Although parameter values are extremely important because the results of the metaheuristic for each problem are highly sensitive to them, the selection of parameter values is commonly justified in one of the following ways (Eiben *et al.* 1999, Adenso-Díaz and Laguna 2006): 1) “by hand”, on the basis of a small number of experiments that are not specifically referenced; 2) using the general values recommended for a wide range of problems; 3) using the values reported to be effective in other similar problems; or 4) choosing values without any explanation.

Adenso-Díaz and Laguna (2006) proposed a new technique called CALIBRA for fine-tuning the parameters of heuristic and metaheuristic algorithms. CALIBRA is

based on Taguchi's fractional factorial experimental designs coupled with a local search procedure. The local search is applied to promising regions of the parameter values and the promising regions are found by means of Taguchi's experimental designs. One assumption of Taguchi's experimental designs is the linear interdependence between the parameters, whereas the interdependence is usually non-linear (Adenso-Díaz and Laguna 2006). CALIBRA uses the analysis of the factorial experiment results only as a guideline to narrow the search and to initiate the next round of experiments. Because the search focuses on narrower ranges for each parameter value, the linear assumption becomes less restrictive and the predicted optimal values approach to the true optimal values.

CALIBRA was chosen for fine-tuning the EM algorithm parameters. A set of 60 representative training instances, which were generated as explained in Section 5, was used. The values obtained for the size of the population ( $p$ ) and the maximum number of iterations of the local search ( $l_{siter}$ ) are shown in Table 1.

The GRASP and PSO algorithms (the multi-start algorithm does not have parameters) were also fine-tuned using CALIBRA and the same 60 training instances. GRASP only has one parameter, which is the size of the candidate list ( $CL$ ).  $DPSO_{poi-c_p dyn}$  has five parameters: the size of the population ( $p$ ), the coefficient that weights the inertia velocity ( $\omega$ ), the coefficient that weights the velocity to the best particle point ( $c_1$ ), the coefficient that weights the velocity to the best swarm point ( $c_2$ ) and the factor of the degree of the random modifications introduced ( $K$ ). The parameter values obtained are shown in Table 1.

[Insert Table 1 here]

## 6. Computational experiment

The computational experiment for the EM algorithm is carried out for the same instances and conditions used in García *et al.* (2006) and in García-Villoria and Pastor (2009). That is, the algorithms ran 740 instances, which were grouped into four classes (from  $CAT1$  to  $CAT4$  with 185 instances in each class) according to their size. The instances were generated using the random values of  $D$  (number of units) and  $n$  (number of models) shown in Table 2. For all instances and for each model  $i = 1, \dots, n$ , a random

value of  $d_i$  (number of units of model  $i$ ) is between 1 and  $\left\lfloor \frac{D-n+1}{2.5} \right\rfloor$  such that

$\sum_{i=1..n} d_i = D$ . All algorithms were coded in Java and the computational experiment was carried out using a 3.4 GHz Pentium IV with 512 Mb of RAM.

[Insert Table 2 here]

For each instance, the four algorithms were run for 50 seconds. Table 3 shows the averages of the RTV values to be minimized for the global of 740 instances and for each class of instances ( $CAT1$  to  $CAT4$ ).

[Insert Table 3 here]

For the global of all instances, the EM algorithm is 18.99% better than  $DPSO_{poi-c_p dyn}$ , 73.55% better than the GRASP algorithm and 82.48% better than the multi-start algorithm. Observing the results in Table 3 by class, we can see that a simple algorithm

such as the multi-start algorithm obtains the best averages for small and medium instances (*CAT1*, *CAT2* and *CAT3*) but a very poor average for large instances (*CAT4*). On the other hand, *DPSOpoi-c<sub>p</sub>dyn* works well for small and large instances (*CAT1*, *CAT2* and *CAT4*) but obtains bad results for medium instances (*CAT3*). Finally, the EM algorithm works fine for small and medium instances; and for large instances, which are the most difficult to solve, it obtains the best results.

To complete the analysis of the results, their dispersion is observed. A measure of the dispersion (let it be called  $\sigma$ ) of the RTV values obtained by each algorithm  $mh = \{EM, \text{multi-start, GRASP, } DPSOpoi-c_p\text{dyn} \}$  for a given instance,  $ins$ , is defined as follows:

$$\sigma(mh, ins) = \left( \frac{RTV_{ins}^{(mh)} - RTV_{ins}^{(best)}}{RTV_{ins}^{(best)}} \right)^2 \quad (5)$$

where  $RTV_{ins}^{(mh)}$  is the RTV value of the solution obtained with the algorithm  $mh$  for the instance  $ins$ , and  $RTV_{ins}^{(best)}$  is, for the instance  $ins$ , the best RTV value of the solutions obtained with the four algorithms. Table 4 shows the average  $\sigma$  dispersion for the global of 740 instances and for each class of instances.

[Insert Table 4 here]

For the global of all instances, the EM procedure has the second least average  $\sigma$  dispersion: 87.24% better than the GRASP algorithm and 79.33% better than the multi-start algorithm, but the *DPSOpoi-c<sub>p</sub>dyn*  $\sigma$  dispersion is better than the EM procedure  $\sigma$  dispersion. Observing the results in Table 4 by class, we see that the EM, multi-start and PSO algorithms have a very stable performance for small instances (*CAT1* and *CAT2*). For medium instances (*CAT3*), only the EM and multi-start algorithms shows a very stable performance. Finally, no algorithm has a very stable performance for the largest instances (*CAT4*). This may occur because 50 computing seconds are not be enough time for the algorithms to converge for the *CAT4* instances. Table 5 and Table 6 show the averages of the RTV values and the  $\sigma$  dispersion, respectively, for all instances and for each class of instance (*CAT1* to *CAT4*) obtained with the four algorithms when they are run for 1,000 seconds.

[Insert Table 5 here]

[Insert Table 6 here]

When a computing time of 1,000 seconds is used—which seems to be long enough for all algorithms to converge (see Figure 2)—the EM algorithm is clearly the best algorithm: it is 76.04%, 77.91% and 78.52% better than the multi-start, GRASP and PSO algorithm, respectively. Moreover, when the EM algorithm has converged, it has a very stable performance for all type of instances (*CAT1* to *CAT4*). That is, when the best solution is not obtained with the EM algorithm, the EM solution is always very close to the best solution.

[Insert Figure 2 here]

## 7. Conclusions and future research

The EM metaheuristic is a population-based metaheuristic for optimisation recently proposed by Birbil and Fang (2003). The method uses an attraction-repulsion mechanism to move the points of the population towards the optimality. In this paper, an EM algorithm is presented for solving the Response Time Variability Problem (RTVP), which has been recently appeared in the literature.

This scheduling problem arises in a variety of real-life environments including mixed-model assembly lines, multi-threaded computer systems, periodic machine maintenance, and waste collection. The aim of the RTVP is to minimize the variability in the distances between any two consecutive units of the same model. Since the RTVP is an NP-hard problem, heuristic and metaheuristic methods are needed to solve real-life instances. García *et al.* (2006) and García-Villoria and Pastor (2009) have proposed a multi-start, a GRASP and several PSO algorithms for solving the RTVP. A computational experiment was done and the results obtained with the EM algorithm are better than the results of the aforementioned algorithms. Moreover, the EM algorithm has a very stable performance when it has converged.

There are two approaches for applying a metaheuristic that works in a real space for solving combinatorial optimisation problems: to adapt the algorithm for working with a sequence of integer numbers or to adapt the representation of the solution as a real point with a random key representation. One of the best referenced procedures is a PSO algorithm (*DPSO<sub>poi-c<sub>p</sub>dyn</sub>*) that follows the first approach; on the other hand, the proposed EM algorithm follows the second approach. We propose as a future research to develop a version of *DPSO<sub>poi-c<sub>p</sub>dyn</sub>* following the second approach, and to develop a version of the EM algorithm following the first approach. The objective is to obtain better results for the RTVP.

## Acknowledgements

This paper was supported by the Spanish Ministry of Education and Science under project DPI2007-61905 and co-funded by the European Regional Development Fund (ERDF). The authors wish to express their gratitude to the anonymous reviewers for their valuable comments and suggestions, which have improved the quality of this paper.

## References

- Adenso-Díaz, B. and Laguna, M., 2006. Fine-tuning of algorithms using fractional experimental designs and local search. *Operations Research*, 54, 99-114.
- Aigbedo, Henry, 2004. Analysis of parts requirements variance for a JIT supply chain. *International Journal of Production Research*, 42, 417-430.
- 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: meeting the ideal of one man, one vote. *Yale University Press*. New Haven CT.
- Bean, J.C., 1994. Genetic Algorithms and Random Keys for Sequencing and Optimization. *ORSA Journal on Computing*, 6, 154-160.

- 1  
2  
3 Birbil, S.I., 2002. Stochastic global optimization techniques. *PhD Thesis*, North  
4 Carolina, State University, Raleigh, NC.
- 5  
6 Birbil, S.I. and Fang, S.C., 2003. An Electromagnetism-like Mechanism for Global  
7 Optimization. *Journal of Global Optimization*, 25, 263-282.
- 8 Chang, P.-C., Chen, S.-H and Fan, C.-Y., 2009. A hybrid electromagnetism-like  
9 algorithm for single machine scheduling problem. *Expert Systems with*  
10 *Applications*, 36, 1259-1267.
- 11 Corominas, A., Kubiak, W. and Pastor, R., 2006. *Mathematical programming modelling*  
12 *of the response time variability problem* [online]. IOC, Universitat Politècnica  
13 de Catalunya, Spain. Available from: <http://hdl.handle.net/2117/408>.
- 14 Corominas, A., Kubiak, W. and Moreno, N., 2007. Response time variability. *Journal*  
15 *of Scheduling*, 10, 97-110.
- 16 Debels, D. and Vanhoucke, M., 2006. The Electromagnetism Meta-heuristic Applied to  
17 the Resource-Constrained Project Scheduling Problem. *Lecture Notes in*  
18 *Computer Science*, 3871, 259-270.
- 19 Debels, D., De Reyck, B., Leus, R. and Vanhoucke, M., 2006. A hybrid scatter  
20 search/electromagnetism meta-heuristic for project scheduling Problem.  
21 *European Journal of Operational Research*, 169, 638-653.
- 22 Ding, F.-Y. and He, J., 2007. A heuristic procedure for the automobile assembly-line  
23 sequencing problem considering multiple product options. *International Journal*  
24 *of Production Research*, doi: 10.1080/00207540701381291. Available online, 13  
25 July 2007.
- 26 Dong, L., Melhem, R. and Mossel, D., 1998. Time slot allocation for real-time  
27 messages with negotiable distance constraint requirements. *Real-time*  
28 *Technology and Application Symposium*. RTAS, Denver.
- 29 Eiben, A.E., Hinterding, R. and Michalewicz, Z., 1999. Parameter control in  
30 evolutionary algorithms. *IEEE Transactions on evolutionary computation*, 3,  
31 124-141.
- 32 Feo, T.A. and Resende, M.G.C., 1989. A probabilistic heuristic for a computationally  
33 difficult set covering problem. *Operations Research Letters*, 8, 67-81.
- 34 García, A., Pastor, R. and Corominas, A., 2006. Solving the Response Time Variability  
35 Problem by means of metaheuristics. *Special Issue of Frontiers in Artificial*  
36 *Intelligence and Applications on Artificial Intelligence Research and*  
37 *Development*, 146, 187-194.
- 38 García-Villoria, A. and Pastor, R., 2009. Introducing dynamic diversity into a discrete  
39 particle swarm optimization. *Computers & Operations Research*, 36, 951-966.
- 40 Herrmann, J.W., 2007. *Generating Cyclic Fair Sequences using Aggregation and Stride*  
41 *Scheduling* [online]. University of Maryland, USA. Available from:  
42 <http://hdl.handle.net/1903/7082>
- 43 Kennedy, J. and Eberhart, R.C., 1995. Particle swarm optimization. *IEEE International*  
44 *Conference on Neural Networks*, 1942-1948.
- 45 Kubiak, W., 1993. Minimizing variation of production rates in just-in-time systems: A  
46 survey. *European Journal of Operational Research*, 66, 259-271.
- 47 Kubiak, W. and Sethi, S.P., 1994. Optimal just-in-time schedules for flexible transfer  
48 lines. *The International Journal of Flexible Manufacturing Systems*, 6, 137-154.
- 49 Maenhout, B. and Vanhoucke, M., 2007. An electromagnetic meta-heuristic for the  
50 nurse scheduling problem. *Journal of Heuristics*, 13, 359-385.
- 51 Marti, R., 2003. Multi-start methods. In: Glover and Kochenberger, eds. *Handbook of*  
52 *Metaheuristics*. Kluwer Academic Publishers, 355-368.
- 53  
54  
55  
56  
57  
58  
59  
60

- 1  
2  
3 Miltenburg, J., 1989. Level schedules for mixed-model assembly lines in just-in-time  
4 production systems. *Management Science*, 35, 192-207.  
5  
6 Monden, Y., 1983. Toyota Production Systems. *Industrial Engineering and*  
7 *Management Press*. Norcross, GA.  
8 Steiner, G. and Yeomans S., 1993. Level Schedules for Mixed-Model, Just-in-Time  
9 Processes. *Management Science*, 39, 728-735.  
10 Tsou, C.-S. and Kao, C.-H., 2008. Multi-objective inventory control using  
11 electromagnetism-like meta-heuristic. *International Journal of Production*  
12 *Research*, 46, 3859-3874.  
13  
14 Waldspurger, C.A. and Wehl, W.E., 1995. *Stride Scheduling: Deterministic*  
15 *Proportional-Share Resource Management* [online]. Institute of Technology,  
16 MIT Laboratory for Computer Science. Available from:  
17 <http://ebets.poly.edu/wein/cs6243/stride.pdf>.  
18  
19 Wu, P., Yang, K.-J. and Hung, Y.-Y., 2004. A study of electromagnetism-like  
20 mechanism for training fuzzy neural network. *In Proc. 1st Annu. Conf. OR*  
21 *Society at Taiwan*, 471-480.  
22 Yuan, K., Hennequin, S., Wang, X. and Gao, L., 2006. A new heuristic-EM for  
23 permutation flowshop scheduling. *12<sup>th</sup> IFAC Symposium on Information Control*  
24 *Problems in Manufacturing*.  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60

Table 1. Parameter values obtained with CALIBRA

EM	GRASP	DPSOpoi-c <sub>p</sub> dyn
$p = 25$	CL size = 3	$p = 13$
$lsiter = 5$		$\omega = 0.75$
		$c_1 = 0.13$
		$c_2 = 0.75$
		$K = 8.70$

Table 2. Uniform distribution for the  $D$  and  $n$  values of the test instances

	CAT1	CAT2	CAT3	CAT4
$D$	$\sim U(25, 50)$	$\sim U(50, 100)$	$\sim U(100, 200)$	$\sim U(200, 500)$
$n$	$\sim U(3, 15)$	$\sim U(3, 30)$	$\sim U(3, 65)$	$\sim U(3, 150)$

Table 3. Averages of the RTV values for 50 seconds

	EM	Multi-start	GRASP	DPSOpoi-c <sub>p</sub> dyn
Global	3,747.05	21,390.40	14,168.83	4,625.54
CAT1	19.14	12.08	15.47	16.42
CAT2	54.54	44.36	88.48	51.34
CAT3	260.79	226.90	510.44	610.34
CAT4	14,653.72	85,278.25	56,060.92	17,824.04

Table 4. Average  $\sigma$  dispersion regarding the best solution found for 50 seconds

	EM	Multi-start	GRASP	DPSOpoi-c <sub>p</sub> dyn
Global	10.45	50.56	81.88	4.79
CAT1	1.62	0.04	0.65	0.76
CAT2	0.49	0.09	4.38	0.34
CAT3	0.24	0.13	7.28	5.29
CAT4	39.45	201.96	315.20	12.78

Table 5. Averages of the RTV values for 1,000 seconds

	EM	Multi-start	GRASP	DPSOpoi-c <sub>p</sub> dyn
Global	330.29	1,378.58	1,495.12	1,537.34
CAT1	18.64	10.93	13.59	14.35
CAT2	52.97	35.48	75.08	46.55
CAT3	157.20	160.67	428.86	143.96
CAT4	1,092.36	5,307.25	5,462.95	5,944.51

Table 6. Average  $\sigma$  dispersion regarding the best solution found for 1,000 seconds

	EM	Multi-start	GRASP	DPSOpoi-c <sub>p</sub> dyn
Global	0.79	4.80	14.47	7.61
CAT1	1.84	0.04	0.48	0.52
CAT2	1.03	0.04	5.29	0.53
CAT3	0.21	0.12	10.98	0.07
CAT4	0.08	18.98	41.06	29.27

Figure 1. Pseudocode of the EM metaheuristic.

```

1: P = initial population
2: while the stopping criteria is not reached do
3:   for each point  $x$  do:  $F_x = \text{total force vector}(x, P)$ 
4:   for each point  $x$  do: Move( $x, F_x$ )
5:   Local search
6:    $x^{\text{best}} = \text{best point of } P$ 
7: end while

```

Figure 2. Average of the RTV values obtained during the execution time

