



## A Batch Optimization Solver for diffusion area scheduling in semiconductor manufacturing

Claude Yugma, Christian Artigues, Stéphane Dauzère-Pérès, Alexandre Derreumaux, Olivier Sibille

### ► To cite this version:

Claude Yugma, Christian Artigues, Stéphane Dauzère-Pérès, Alexandre Derreumaux, Olivier Sibille. A Batch Optimization Solver for diffusion area scheduling in semiconductor manufacturing. 2007. hal-00135012v1

**HAL Id: hal-00135012**

**<https://hal.science/hal-00135012v1>**

Preprint submitted on 6 Mar 2007 (v1), last revised 2 May 2007 (v2)

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

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

# A Batch Optimization Solver for diffusion area scheduling in semiconductor manufacturing

Claude Yugma<sup>1 \*</sup>      Christian Artigues<sup>2</sup>      Stéphane Dauzère Pères<sup>1</sup>  
Alexandre Derreumaux<sup>1</sup>      Olivier Sibille<sup>3</sup>

<sup>1</sup>École des Mines de Saint-Etienne, Centre Microelectronique de Provence - Site Georges Charpak, Avenue des Anémones, Quartier Saint-Pierre, F-13541 Gardanne, France

<sup>2</sup>LAAS-CNRS, 7 avenue du Colonel Roche, 31077 Toulouse, France

<sup>3</sup>ATMEL, Zone industrielle, 13790 Rousset, France

## Abstract

This paper presents a method and a software for solving a batching and scheduling problem in the diffusion area of a semiconductor plant, the ATMEL fabrication unit in Rousset, France. The diffusion area is one of the most complex area in the fab. A significant number of lots has to be processed while satisfying complex equipment process and line management constraints. The purpose of this study is to investigate approaches to group lots in batches, to assign the batches on the equipment and to sequence these batches. Three indicators are used to evaluate the quality of a solution: the total number of moves, the batching coefficient and the X-factor. The problem is modeled through the disjunctive graph formulation. A constructive algorithm is proposed and improvement procedures based on local search and simulated annealing are developed. Computational experiments, carried out on actual industrial problem instances, show the ability of local search to improve significantly the initial solution. The proposed simulated annealing method brings in turn important enhancements to the results of local research. The software based on these methods, named Batch Optimization Solver (BOS), is currently used in the diffusion area of ATMEL. The disjunctive graph model allows in addition a high level of interactivity with the decision makers.

**keyword:** Batch scheduling, disjunctive graph, multi criteria, local search, simulated annealing, wafer fabrication.

## 1 Introduction

Semiconductor wafer fabrication can be described as a multistage process with re-entrant flows. The processing is done layer by layer. Each layer requires several steps of processing such as chemical-mechanical polishing, diffusion, film deposition, photolithography, implant (doping) and etching. For each of the product types, and depending on the technology, a wafer goes through more than 400 process steps over a period of a few weeks. Wafer fabrication planning and scheduling is a complex task due to the large number of products and machines involved. It is further complicated by additional constraints such as re-entrant flow of operations [Kum94], setup issues, preventive maintenances, random machine breakdowns, etc.

---

\*Corresponding author e-mail: Claude.Yugma@emse.fr

The importance of scheduling on the performance of semiconductor wafer fabrication facilities is known for many years [Wei88, VS06]. Recently [TIHC05] have also shown the importance of using predictive scheduling in load balancing methods among multiple semiconductor wafer fabrication lines in order to make a better estimation of the lot cycle times. The cycle time is the total time necessary to manufacture a lot of wafers. It measures the duration time spent since the lot enters the fab and the moment it leaves it. According to experimental studies, decreasing the cycle time results in an increase of the productivity in a fab [JH90].

The complexity of the fabrication process is such that most previous existing work on scheduling in semiconductor manufacturing focus on dispatching rules [Wei88, DF03, VS06]. Other approaches deal with Integer Linear Programming formulations of the problem and apply lagrangean relaxation of the equipment limitation constraints [LCPC96, HC03]. For a given set of lagrangean multipliers, the relaxation can be solved with minimal cost network flow methods. Subgradient optimization is used to solve the dual linear program. A feasible solution is obtained by adjusting heuristically the solution of the relaxation.

In this paper, instead of scheduling the entire fab, we propose to focus on a bottleneck part of the manufacturing process. Among the complex operations involved in the fabrication of a wafer, the diffusion phase is of critical importance since the batching decisions that are involved may affect the performance of the entire wafer fab [ICN<sup>+</sup>03, MH03]. This is also the case in the ATMEL fabrication unit of Rousset (France). The diffusion phase is used primarily to alter the type and level of conductivity of semiconductor materials. It is used to form bases, emitters and resistors in bipolar devices, and also to dope polysilicon layers. The Batch Optimization Software (BOS) project aims at developing a method and a software for solving efficiently the batching and scheduling problem of the diffusion area while taking into account numerous constraints and optimizing various criteria. Another objective of the BOS is to provide an easy way to build, modify and use the results of the software. As for the global problem, batching and scheduling in diffusion areas is mainly performed in the literature by dispatching rules. In [ICS<sup>+</sup>03] and [MH03], the authors propose to use a dispatching rule to batch the operations and a genetic algorithm to assign the batches to equipment. In [MFC02], a scheduling problem in semiconductor manufacturing related to the one tackled in this paper is solved by a modified shifting bottleneck heuristic [ABZ88].

In this paper, we propose to model solutions of the considered batching and scheduling problem through a variant of the disjunctive graph proposed in [MFC02]. A constructive heuristic is designed that builds an initial solution as well as improvement procedures based on local search and simulated annealing.

This paper is organized as follows. In Section 2, we give the problem description. In Section 3, we present the proposed disjunctive graph. The method for computing an initial solution and a basic improvement procedure are described in Section 4 and experiment results are given on real problem instances issued from the Rousset ATMEL fabrication unit. A simulated annealing algorithm and its results on the considered instances are then presented in section 5. The BOS software is briefly presented in Section 6. The Section 7 concludes the paper with recommendations for further research.

## 2 Problem description

The diffusion area defines a batching and scheduling problem of wafer lots on two types of equipment: furnaces and cleaning machines. These resources are able to perform several lots simultaneously. Each lot requires one or more consecutive operations on the equipments and each operation has a recipe which determines its duration and the set of machines that are able to process it. On a 24-hour basis, each operation has to be assigned to an equipment, and included

into a batch, i.e. a set of operations of the same recipe that are processed simultaneously by the equipment. The constraints in the diffusion area are divided into 3 types: equipment constraints, process constraints and line management constraints.

## 2.1 Constraints

Some of these constraints are common to the two types of resources while others are dedicated to furnaces.

### 2.1.1 Equipment constraints

- **Common constraints**

*Dedicated equipment:* any equipment is able to process a limited set of recipes.

*Maximum batch size:* any equipment defines a maximum batch size corresponding to the capacity of that equipment.

*Loading and unloading times:* a time may be needed to load and unload a batch on the equipment.

*Unavailability periods:* the equipment may be unavailable during some periods (defined by time windows) due to qualification, repair, maintenance etc.

*In process jobs:* the equipment may be occupied at the beginning of the time horizon by in-process operations that have to be completed before the equipment becomes available.

- **Specific furnace constraints**

*Minimum time between two batches on an furnace:* some furnaces must be inspected at the end of each processed batch.

### 2.1.2 Process constraints

- **Common constraints**

*Precedence:* lot operations must be performed following the manufacturing process of the lot. The operations of a lot are chained and no operation can start before the end of its predecessor, except for the first operation of each lot.

*Minimum time lag:* there is a fixed handling and transport time between every two successive operations of a lot.

*Release dates:* they correspond to the arrival times of lots at the cleaning machines and furnaces. It means that a lot cannot be scheduled before its release date. Because the diffusion area is a stage of the complex global production process, the release date are precomputed by a simulation package designed by ATMEL.

*Fixed recipe:* each operation of a lot is associated with a recipe, i.e. the lot should be processed on resources that are qualified for the corresponding recipe. This implies that all lots in the same batch must be processed with the same recipe.

*Process time:* the process time of a batch on an equipment depends on the recipe.

*Maximum time lag:* a time limit is given for two successive operations  $x$  and  $y$  of a lot. The difference between the starting time of  $y$  and the completion time of  $x$  cannot exceed this limit. Maximum time lags depend on the operations  $x$  and  $y$ . They can correspond to the maximum time limit between cleaning and furnace equipment.

### 2.1.3 Line management constraints

- **Specific furnace constraints** *Minimum distance between two batches of same recipe:* there is a minimum time between the beginning of two batches of the same recipe on two

different furnaces.

## 2.2 Objectives

In semiconductor factories, several indicators are used to measure the performance of a fab. We describe thereafter the indicators that are relevant in our study. The interested reader can refer to [MT06] for more details.

- *The number of moves* (to maximize): It corresponds to the number of completed operations on the planning horizon, which can be compared to the target number fixed by the production managers.
- *Batching coefficient* (to maximize): Defined on the planning horizon, it is calculated as the number of moves divided by the sum of the number of batches performed on each machine, times the maximum capacity of that machine. Note that the denominator is the number of lots that could be performed if the equipment was loaded up to their maximum capacities.
- *The X-factor* (to minimize): This indicator is used to evaluate the waiting times of lots in the diffusion area in order to reduce the cycle times. For a given lot, this factor is calculated as the total staying time of the lot in the diffusion area (i.e. waiting time from its arrival in the diffusion area and processing time) over its processing time.

These three indicators will be used to evaluate solutions in the fab.

Others indicators not considered in this paper are of interest. Among them, the *Work-In-Process* (WIP), is defined as the number of wafers being in the fab at a given period, either in a production state or in a non-production state (e.g. transport and waiting). The law of Little [Lit61] establishes that if a system is stable and stationary then the average WIP is proportional to the average cycle time. The *throughput* is defined as the outgoing number of wafers of the fab per unit of time. The evolution of the throughput in time makes it possible to know if the system is stable i.e. to know if there is no accumulation of product in the fab or if the system evolves/moves according to forecasts. Authors, such as [GR88], observed that there is a relation between the increasing of the throughput and the output of a fab. The cycle time drastically increases when throughput is close to the maximal capacity of the fab. Hence for both WIP and throughput indicators, the X-factor indicator is adequate.

The goal is to optimize the various performance measures, while taking into account the numerous complex constraints. Another objective is to make the proposed schedule easily modifiable by the decision makers at the fabrication level. Machines may be declared as down, operations may be moved, lots may be removed or inserted. Such requirements can be achieved thanks to the flexibility of the disjunctive graph representation.

The next subsection described the mathematical formulation used to model the problem.

## 2.3 Problem Formulation

The scheduling problem can be formulated as follows. For sake of clarity, the above-described *Unavailability periods*, *In-process jobs* and *Minimum distance between batches of the same recipe* constraints are not included. In Section 3, we describe how we tackle these additional characteristics.

A set of jobs (lots)  $\mathcal{J} = \{J_i | i = 1, \dots, n\}$  has to be processed during a period  $T$  by a set of machines  $\mathcal{M} = \{M_k | k = 1, \dots, m\}$ . Each job  $J_i$  is made of  $n_i$  operations such that each operation  $O_{ij}$  has a duration  $p_{ij} > 0$  and a set  $\mathcal{M}_{ij} \subseteq \mathcal{M}$  of machines (the furnaces or the

cleaning equipments of the diffusion area) able to process it. Let  $\mathcal{O}_k = \{O_{ij} \in \mathcal{O} | M_k \in \mathcal{M}_{ij}\}$  denote the set of operations that can be assigned to machine  $M_k$ . The value of  $p_{ij}$  and the elements of the set  $\mathcal{M}_{ij}$  are determined by the recipe of operation  $O_{ij}$  denoted  $\rho_{ij}$ . In general we have  $\mathcal{M}_{ij} \subset \mathcal{M}$  since each machine cannot be configured for all recipes. Each operation  $O_{ij}$  has to be included in a batch on a resource  $k \in \mathcal{M}_{ij}$ . Each machine has a finite capacity  $R_k$  which gives the maximal number of lots in the same batch. On each machine  $k$ ,  $S_k$  denote the setup time needed before starting a new batch,  $D_k$  denote the removal time needed after the completion of a batch and  $s_k$  denote the constant setup time needed between two different batches.  $s_k^0$  denotes the initial setup time on machine  $k$ , depending on the state of the resource at time 0. Two consecutive operations  $O_{ij}$  and  $O_{i(j+1)}$  of the same job are linked by minimal and maximal time lags. Once the batch of  $O_{ij}$  is completed and removed from  $k$ , the setup for the batch of  $O_{i(j+1)}$  cannot start before a minimal time lag  $\tau_{ij}^{\min}$  and has to start before a maximal time lag  $\tau_{ij}^{\max}$ . Let  $\mathcal{O} = \{O_{ij} | i = 1, \dots, n; j = 1, \dots, n_i\}$  denote the set of all operations. Each job  $J_i$  has a relative priority  $c_i$  ( $c_i < c_j \implies J_i$  is more urgent than  $J_j$ ). Each job corresponds to a number  $w_i$  of wafers produced when the job is completed. Finding a feasible solution for the problem lies in making four types of decisions:

- D1 - Partition the operations into batches,
- D2 - Select a resource to process each batch,
- D3 - Order the batches on each resource,
- D4 - And assign a start time to each batch.

Decisions D1-D3 can all be represented by a family of batches  $\mathcal{B} = \{B_{kq}\}_{k \in \{1, \dots, m\}, q \in \{1, \dots, \nu_k\}}$  where  $B_{kq}$  is the batch sequenced at position  $q$  on machine  $M_k$ .  $\nu_k \in \{0, \dots, |\mathcal{O}_k|\}$  denote the number of batches assigned to machine  $M_k$ . Decision D4 lead to a family of start times  $\mathcal{T} = \{t_{ij}\}_{O_{ij} \in \mathcal{O}}$  assigned to the operations. Once a solution  $\{\mathcal{B}, \mathcal{T}\}$  is determined, we can deduce a machine assignment  $\{m_{ij}\}_{O_{ij} \in \mathcal{O}}$  where  $m_{ij}$  denote the machine  $O_{ij}$  is assigned to, i.e. verifying  $\exists q \in \{1, \dots, \nu_k\}$  such that  $O_{ij} \in B_{m_{ij}q}$ . To be feasible, a solution  $\{\mathcal{B}, \mathcal{T}\}$  and its corresponding assignment  $\{m_{ij}\}_{O_{ij} \in \mathcal{O}}$  have to satisfy the following constraints. The operations of the same batch must have the same recipe, i.e.

$$\rho_{ij} = \rho_{xy} \quad \forall B \in \mathcal{B}, \forall O_{ij}, O_{xy} \in B \quad (1)$$

Each operation must be assigned to a machine able to process its recipe.

$$m_{ij} \in \mathcal{M}_{ij} \quad \forall O_{ij} \in \mathcal{O} \quad (2)$$

The batch capacity cannot be exceeded and each batch includes at least one operation.

$$1 \leq |B_{kq}| \leq R_k \quad \forall B_{kq} \in \mathcal{B} \quad (3)$$

An operation appears in only one batch i.e.

$$B \cap B' = \emptyset \quad B, B' \in \mathcal{B}, B \neq B' \quad (4)$$

All operations are included in a batch

$$\cup_{B \in \mathcal{B}} B = \mathcal{O} \quad (5)$$

The start time of the first operation of each lot cannot exceed the lot release date:

$$t_{i1} \geq r_i \quad \forall J_i \in \mathcal{J} \quad (6)$$

Each operation  $O_{ij}$ ,  $j > 1$ , cannot start before a minimal time lag after the end of preceding operation  $O_{i(j-1)}$ , which takes account of the removal time of the batch of  $O_{i(j-1)}$ , the minimal time lag  $\tau_{i(j-1)}^{\min}$  and the setup time of batch of  $O_{ij}$ :

$$\begin{aligned} t_{ij} - t_{i(j-1)} &\geq D_{m_{i(j-1)}} + p_{i(j-1)} + \tau_{i(j-1)}^{\min} + S_{m_{ij}} \\ \forall i &\in \{1, \dots, n\}, \forall j \in \{2, \dots, n_i\} \end{aligned} \quad (7)$$

Each operation  $O_{ij}$ ,  $j > 1$ , has to start before a maximal time lag after the end of preceding operation  $O_{i(j-1)}$ , which takes account of the removal time of the batch of  $O_{i(j-1)}$ , the maximal time lag  $\tau_{i(j-1)}^{\max}$  and the setup time of batch of  $O_{ij}$ :

$$\begin{aligned} t_{ij} - t_{i(j-1)} &\leq D_{m_{i(j-1)}} + p_{i(j-1)} + \tau_{i(j-1)}^{\max} + S_{m_{ij}} \\ \forall i &\in \{1, \dots, n\}, \forall j \in \{2, \dots, n_i\} \end{aligned} \quad (8)$$

The start times of two operations of the same batch must be equal:

$$t_{ij} = t_{xy} \quad \forall B \in \mathcal{B}, \forall O_{ij}, O_{xy} \in B \quad (9)$$

An operation of a batch which is not at the first position on its machine cannot start before the end of the preceding batch on the machine, plus the necessary removal time of the preceding batch, plus the minimal setup time on the machine between two batches, plus the necessary setup time for the next batch.

$$\begin{aligned} t_{ij} - t_{xy} &\geq p_{xy} + D_k + s_k + S_k \\ \forall B_{kq, q>1} &\in \mathcal{B}, \forall O_{ij} \in B_{kq}, \forall O_{xy} \in B_{k(q-1)} \end{aligned} \quad (10)$$

An operation of a batch in the first position on its machine cannot start before the initial setup time for this batch (we assume  $S_k^0 \leq S_k + D_k + s_k, \forall M_k \in \mathcal{M}$ ):

$$t_{ij} \geq s_{m_{ij}}^0 \quad \forall O_{ij} \in \mathcal{O} \quad (11)$$

By definition, a feasible solution includes each operation inside a batch. In the ATMEL Rousset production unit, the scheduling horizon is in fact limited to a scheduling period  $T$ . Hence only those batches released in the interval  $[0, T[$  must be taken into account. Several criteria are used to measure the quality of a feasible solution. The number of moves is the number of wafers produced during the time period  $T$ :

$$f_{\text{mov}} = \sum_{J_i \in \mathcal{J}} w_i \theta_i \quad (12)$$

where  $\theta_i = 1$  when the job is completed before  $T$  and  $\theta_i \in [0, 1[$  denotes the completion ratio of job  $i$  before time  $T$  otherwise.

$$\theta_i = \frac{\sum_{O_{ij}, t_{ij} + p_{ij} \leq T} p_{ij} + \sum_{O_{ij}, t_{ij} < T < t_{ij} + p_{ij}} (T - t_{ij})}{\sum_{j=1}^{n_i} p_{ij}}$$

The batching coefficient is the average ratio of the actual size of each batch over to its maximal size. Let  $\mathcal{B}^T = \{B \in \mathcal{B} | t_{ij} < T, \forall O_{ij} \in B\}$  denote the set of batches started before time period  $T$ .

$$\text{batching coefficient} = \frac{\sum_k \sum_{B_{kq} \in \mathcal{B}^T} |B_{kq}| / R_k}{|\mathcal{B}^T|} \quad (13)$$

$$f_{\text{batch}} = \frac{\sum_k \sum_{B_{kq} \in \mathcal{B}^T} |B_{kq}| / (R_k + \frac{Z_k}{100})}{|\mathcal{B}^T|} \quad (14)$$

where  $Z_k$  means the number of total qualified recipe on the machine  $k$ .

The weighting of the coefficient of batching by the number of qualified recipes on the concerned machine makes it possible to use the least general-purpose machines initially.

The average X-factor is the average of the X-factor of each job weighted by the job priority. All the jobs do not have the same priority. Certain jobs are more important than others (important customers, tests to be carried out quickly, delay to catch up with, etc). Thus, a weight is affected to the job to reflect the priority. Let  $\mathcal{J}^T = \{J_i \in \mathcal{J} | t_{in_i} + p_{in_i} \leq T\}$  denote the set of jobs completed before  $T$ .

$$\text{X-fac} = \frac{\sum_{J_i \in \mathcal{J}^T} (t_{in_i} - r_i) / p_{in_i}}{|\mathcal{J}^T|} \quad (15)$$

$$f_{\text{X-fac}} = \frac{\sum_{J_i \in \mathcal{J}^T} c_i (t_{in_i} - r_i)}{|\mathcal{J}^T|} \quad (16)$$

The choice made together with the decision makers in the ATMEL Rousset unit is to combine these different objectives into a single one by minimizing the weighted sum

$$f = \alpha f_{\text{mov}} + \beta f_{\text{batch}} - \gamma f_{\text{X-fac}} \quad (17)$$

where  $\alpha$ ,  $\beta$  and  $\gamma$  are adjustable weights allocated to each objective function. The objective functions have been designed to integrate some requests of ATMEL managers. On the one hand, in the calculation of  $f_{\text{X-fac}}$ , the delay of each lot is multiplied by its priority, in order to support the urgent lots. On the second hand, in the calculation of the total batching coefficient, the batching coefficient of each equipment is multiplied according to the number of qualified recipes on this equipment. This leads to occupy in priority the equipments able to treat less recipes, and thus to maintain the availability of the most flexible equipments.

The objective of the problem is to search for a feasible selection  $\{\mathcal{B}, \mathcal{T}\}$  such that  $f$  is maximized. Note that given a (feasible) solution  $\{\mathcal{B}, \mathcal{T}\}$ ,  $f$  can be computed in  $O(N)$  time where  $N = \sum_{i=1}^n n_i$  is the total number of operations.

### 3 The disjunctive graph representation

The considered problem can be seen as an extension of the job-shop problem and, consequently, the disjunctive graph model can be used for batching and scheduling problems as proposed by [MFC02]. In their article, the authors consider a different objective function (namely total weighted tardiness). Sequence-dependant setup times and reentrant flows are also tackled but there are no maximum time lags. Unfortunately, these latter constraints increase considerably the difficulty of the problem [GNST04]. Let us explain how we modeled the problem using disjunctive graphs.

We define the disjunctive graph  $G = (V, C, E)$  as follows.

- $V$  is a set of nodes where there is one node per operation, denoted  $V_{ij}$ , plus a dummy start node denoted 0.
- $C$  is the set of conjunctive arcs representing the release dates and minimal and maximal time lags. There is an arc from node 0 to node  $V_{i1}$  of each job  $J_i$ . There is an arc from  $V_{ij}$  to  $V_{i(j+1)}$  and an arc from  $V_{i(j+1)}$  to  $V_{ij}$ , for each consecutive operations  $O_{ij}$  and  $O_{i(j+1)}$  of each job  $J_i$ .



- $E$  is the set of disjunctive arcs which represent the decisions of the problem. There are two opposite disjunctive arcs  $(V_{ij}, V_{xy})^k$  and  $(V_{xy}, V_{ij})^k$  for any machine  $k \in \mathcal{M}$  and for any pair of operations  $O_{ij}, O_{xy} \in \mathcal{O}_k$ ,  $O_{ij} \neq O_{xy}$ . These arcs represent the sequencing or the batching decision concerning  $O_{ij}$  and  $O_{xy}$  on machine  $k$ .

Let  $\mathcal{B}$  denote a partial or complete batching for the problem satisfying at least constraints (1), (2), (3) and (4).  $\mathcal{B}$  is a complete batching if constraint (5) is also verified, otherwise it is a partial batching. Recall  $\mathcal{B}$  also defines the machine assignment  $m_{ij}$ , for all  $O_{ij} \in \cup_{B \in \mathcal{B}} B$ . We assume  $m_{ij} = 0$  if  $O_{ij}$  is not batched in  $\mathcal{B}$ , i.e. if we have  $O_{ij} \notin \cup_{B \in \mathcal{B}} B$ .

$\mathcal{B}$  unambiguously defines a selection  $\mathcal{S}$  as follows. For each distinct operations  $O_{ij}$  and  $O_{xy}$  such that  $m_{ij} = m_{xy} = k \neq 0$ , select arc  $(V_{xy}, V_{ij})^k$  if  $O_{xy}$  is in a batch sequenced before the batch of  $O_{ij}$ , select arc  $(V_{ij}, V_{xy})^k$  if  $O_{ij}$  is in a batch sequenced before the batch of  $O_{xy}$ , and select both arcs  $(V_{ij}, V_{xy})^k$  and  $(V_{xy}, V_{ij})^k$  if  $O_{ij}$  and  $O_{xy}$  are assigned to the same batch.

Once a selection is computed, we define a graph  $G(\mathcal{S}) = (V, C \cup \mathcal{S})$  where arcs  $C \cup \mathcal{S}$  are valued as follows (we assume  $s_0^0 = s_0 = S_0 = D_0 = 0$ ):

- Each arc from 0 to the first operation  $O_{i1}$  is valued by  $\max(r_i, S_{m_{ij}}^0)$ , the maximal value between the release date of job  $i$  and the initial setup time for machine  $m_{i1}$ .
- Each arc from  $V_{ij}$  and  $V_{i(j+1)}$  is valued by  $D_{m_{ij}} + p_{ij} + \tau_{ij}^{\min} + S_{m_{i(j+1)}}$ , the value of the minimal time lag between  $O_{ij}$  and  $O_{i(j+1)}$  plus the setup and removal times linked to the assignment of the operations.
- Each arc from  $O_{i(j+1)}$  and  $O_{ij}$  is valued by  $-(D_{m_{ij}} + p_{ij} + \tau_{ij}^{\max} + S_{m_{i(j+1)}})$ , the (negative) value of the maximal time lag between  $O_{ij}$  and  $O_{i(j+1)}$  plus the necessary setup and removal times.
- Each arc  $(V_{ij}, V_{xy})^k$  is valued either by  $p_{ij} + D_k + s_k + S_k$  if the opposite arc is not selected or by 0 if the opposite arc is selected. Indeed, in the first case this arc represent the decision to sequence  $O_{xy}$  after  $O_{ij}$  on machine  $k$  whereas in the second case, both arcs  $(V_{ij}, V_{xy})^k$  and  $(V_{xy}, V_{ij})^k$  represent the synchronisation of the operations included in the same batch.

We can state that the (partial) solution represented by the (partial) batching  $\mathcal{B}$  and its selection  $\mathcal{S}$  is feasible if and only if all longest path problems from node 0 to each node  $V_{ij}$  in  $G(\mathcal{S})$  have a solution. In the positive case and if  $\mathcal{B}$  is complete, a feasible schedule  $\mathcal{T}$  can be obtained by setting  $t_{ij}$  to the length of the longest path from 0 to  $V_{ij}$ . Furthermore  $\mathcal{T}$  is the best schedule compatible with  $\mathcal{B}$  one can obtain when the objective is to maximize  $f$ .

The problem can be formulated as follows: Find the batching  $\mathcal{B}$  verifying constraints (1), (2), (3), (4) and (5) such that the corresponding selection  $\mathcal{S}$  is feasible and maximizes  $f$ .

As stated in the previous sections, the actual data issued from the fab has other characteristics that have been tackled, such as in-process jobs and machine down times. We can model easily these characteristics thanks to the disjunctive graph formulation. They can be both represented by operations with fixed start times on the involved machine. A start time  $t$  can be fixed by linking the node with node 0 by two opposite arcs valued by  $t$  and  $-t$ . As stated in Section 2, the actual problem also includes an important line management constraint: a minimum distance between the start time of two batches of the same recipe scheduled on two different batches has to be respected. This can also be tackled through the disjunctive graph representation. A fictitious machine can be associated to each recipe and disjunctive arcs linking two operations of the same recipe can be defined. Then, whenever the two operations are batched on two different machines (furnaces), the disjunctive arc has to be orientated. Once orientated the arc is valued by the minimum required distance.

## 4 Solution Methods

We propose a two-phase heuristic method to solve the problem. The first phase is a constructive heuristic based on successive job insertions. The second phase is a local search method which aims at improving the initial solution. Both phases are based on the evaluation of a (complete or partial) selection through longest path calculations.

### 4.1 Evaluation of a partial or complete batching

Any partial or complete batching  $\mathcal{B}$  and its selection  $\mathcal{S}$  can be evaluated through the calculation of start time  $t_{ij}$ , equal to the longest path from 0 to  $V_{ij}$  in  $G(\mathcal{S})$  of each operation  $O_{ij}$ . To compute such longest paths, since the graph includes arcs with negative weights, we use the Bellman-Ford algorithm which has a  $O(N|\mathcal{S} \cup \mathcal{C}|)$  time complexity. If the algorithm finds a path of positive length, then the partial or complete solution is unfeasible. Otherwise the algorithm returns start times  $t_{ij}$  and the objective function value  $f$  can be determined.

### 4.2 Computing an initial solution

The initial solution (selection) is computed by a simple job insertion method. The jobs are first sorted in a list  $L$  according to the order: jobs involving maximal time lags first, then increasing release dates, then job priority.

The method starts with an empty batching. Then the jobs are taken in the order given by the list and inserted one by one in the current batching. The insertion of  $J_i$  is made as follows. Let  $\mathcal{B} = \{B_{kq}\}_{k \in \mathcal{M}, q \in 1, \dots, \nu_k}$  denote the current batching including jobs located before  $J_i$  in  $\mathcal{L}$ . For each operation  $O_{ij}$  of  $J_i$  and for each resource  $k \in \mathcal{M}_{ij}$ , there are  $2\nu_k + 1$  insertion positions of  $O_{ij}$  in the batch sequence of machine  $M_k$ : indeed, for each batch  $B_{kq}$ , we may insert  $O_{ij}$  inside batch  $B_{kq}$  or create a new batch at any position. Each of these insertion positions is evaluated with the algorithm described in the previous section and the one that maximizes  $f$  is kept to update  $\mathcal{B}$ . If none of the insertion positions is feasible for an operation  $O_{ij}$  then all these partial solutions violate a maximal time lag and there exists insertion positions that violate only maximal time lag between  $O_{i(j-1)}$  and  $O_{ij}$  (the last position on each resource of  $\mathcal{M}_{ij}$ , for instance). Hence we have to remove  $O_{i(j-1)}$  from  $\mathcal{B}$ , the previous operation of job  $J_i$ , and insert it at a later insertion position. If this is not feasible, they are not scheduled and delayed from the list of operations. Note that at most  $\sum_{j=1}^{n_i} \sum_{k \in \mathcal{M}_{ij}} 2\nu_k + 1$  insertion positions are tested per inserted lot.

### 4.3 Improving the solution

The initial solution computed by the job insertion algorithm can be improved by local search. In a straightforward local search (steepest descent method), a set of moves is tested from an initial solution, each one issuing a new neighbor solution. The best neighbor solution is kept as the new starting solution if it strictly improves the objective function and the algorithm is iterated until there is no more improvements of the objective function. We have designed several types of moves for the problem as described as follows:

- "Merge batches". In order to improve the batching coefficient, the merge move aims at merging two batches (within the limit of the equipment capacity) having the same recipe.
- "Remove a batch" removes a batch and replaces its operations in other batches. This move makes it possible to decrease the batching coefficient.

- The "Job reinsertion" move selects a job and reinserts it in the solution by using the same method i.e. as for computing the initial solution, as if this job was the last inserted job.
- "Reverse two successive batches" on the same equipment, may help to improve waiting times.
- "Move a lot", consists in moving all operations of the lot. Indeed, when there are maximum and minimum time lag, the fact of moving an operation task can only be done correctly by moving the preceding and following operations.

The local search method applies iteratively a steepest descent method based on these moves.

#### 4.4 Computational experiments

We have coded the proposed algorithm in Java and we have tested it on a 1 gigaoctet RAM and 3.4 gigahertz processor computer. These tests have been done on real instances of the problem issued for several production days in the ATMEL Rousset factory. There are 700 jobs yielding a total number of 1400 operations with about 50 different recipes to schedule on about 70 furnaces, and each of them has a capacity between 4 and 6 lots. There are 12 cleaning equipment with capacity about 2 to 4 lots. The time needed to load and unload a batch varies between 10 and 30 minutes. The minimum and maximum time lags varie from 10 minutes to 4 hours. The target time horizon is 24 hours.

We study thereafter the results of the initial solution computation procedure followed by local search improvements.

We have tested three different initial solution computation procedures. The first one (sorted list) computes a single solution with the method described in Section 4.2. The second one (random list) computes a solution by initializing randomly the input list of the lot insertion method. The third one (pseudo random list) computes the list by making light random modifications on the sorted list. It consists in dividing the list into intervals and, for each interval, in making a random modification following a uniform law.

Each computed solution is then improved by the local search procedure described in Section 4.3.

From Table 1 to Table 10, we display the results (Objective function value, number of moves, batching coefficient, X-factor and CPU time) for the sorted list method, the random list method and the pseudo random list method on the 10 considered instances. For the random and pseudo random methods the mean, best, worst solutions (w.r.t. the objective function) are given as well as the standard deviation for 1000 runs. The row "best" ("worst") means the best (worst) value obtained on the considered instance among all computed solutions (random or pseudo random list) of the global objective function  $f$  (equation 17), the number of moves (equation 12), the batching coefficient (equation 13), the X-factor (equation 15). Row "best solution obtained" corresponds to the best solution obtained when we consider the objective function (17) only and displays the results of the three criteria to evaluate the distance with the best values. The results of the improvement procedure (applied to the best solution obtained in the case of random and pseudo random methods) are also displayed. The row "improvement" corresponds to the relative improvement of the best solution obtained compared to the sorted list (in percentage). It is calculated according to the formula  $\frac{\text{Best solution} - \text{Sorted list}}{\text{Sorted list}}$ .

The weights of the components of the objective function have been computed both to normalize the different indicators and to take account of the user preferences. For the experiments, we have selected  $\alpha = 601$ ,  $\beta = 1500001$ ,  $\gamma = 41$ .

The analysis of Tables 1-10 shows that we get the initial solution from the sorted list in a very fast calculation time: the smallest time is 2 seconds. The largest time is 17 seconds.

For the managers of the diffusion area, it was important to obtain a solution in a fast time. We underline that the sorted list method provides solutions which are evaluated as satisfactory by the ATMEL managers. These first results are also relevant since they are coherent with the operator practices. This was an important point for the ATMEL managers who would not accept solutions changing too deeply the organization of the diffusion area. Consequently the first results we obtained were well accepted in the fab and constituted good start points for further improvement.

For sake of comparison the "pseudo random list" and "random list" part of the tables start from the "sorted list" solution. The best objective function of the random list solutions is better than the best one of the pseudo random list solutions. The random list covers a greater solution space, meaning that there are better solutions, but also worse. By considering the row "Improvement", we obtain an increase of 6,77% in the best case (table 7) and of 0,00% (table 3) in the worst case for random lists. For pseudo random lists, the increase is of 6,18% (table 7) in the best case and of 0,85% (table 5) in the worst case. Looking separately at the objective function components, the move indicator increases in the best case by 4,49% for the random list and by 1,89% for the pseudo random list. The batching coefficient is improved up to 2,73% for the random list (table 3) and to 1,45% for the pseudo random list (table 9). In the best case, the X-factor indicator decreases by 4,45% for the random list and by 2,45% for the pseudo random list (table 6). The pseudo random list allows to not lose totally the structure of the initial list. Thus, it is more difficult to improve a solution from the pseudo random list as the initial list (sorted list) yet provides good results.

As already explained in the subsection 2.3, the objective function is not a simple combination of X-fac, batching coefficient and number of moves, but uses "weighted" versions. In comparison of X-fac and  $f_{X-fac}$ , the weighting of the delay by the priority of the batches makes it possible to schedule the most urgent batches first. Comparing the Batching coefficient and  $f_{batch}$ , the weighting of the batching coefficient by the number of qualified recipes on the concerned equipment makes it possible to fill the least general-purpose machines first. In table 5 (random list) and in table 1 and table 10 (pseudo random list), we can notice the objective function is improved even if the indicators remain unchanged or are slightly worsened.

In general, local search significantly improves the quality of the initial solution in less than 20 seconds. To still increase the quality of the obtained solutions, a simulated annealing algorithm, which allows to cover a larger search space, has been developed.

## 5 Simulated annealing

Simulated annealing (SA) belongs to a class of randomized local search algorithms and has been developed by [KGV83] to handle hard combinatorial problems. The use of simulated annealing supposes the definition of a combinatorial optimization problem and a neighborhood. We are here looking for a solution to the considered batching and a scheduling problem with the maximal objective function value. The neighborhood we consider is defined by the following moves:

1. "Batch move". A batch is moved randomly (both equipment and position can change).
2. "Job move". An operation is moved randomly in an existing batch or a new batch is created (both equipment and position can change).
3. "Job switch". Two batches with the same recipe are selected randomly. Then two randomly selected jobs are switched.

50% of the considered moves are batch moves, 25% are job moves and 25% are job switches. According to the move, randomly generated, we proceed as follows. All random selections are

made according to the uniform law. If a move is impossible due to a violation of a constraint, we restore the current solution, we try another move and we reiterate.

The algorithm starts with an initial solution  $s_0$  and then continually tries to find better solutions by searching in its neighborhood (obtained by the moves described above), and applying a stochastic acceptance criterion. When a neighbor (a solution  $s$ ) of the current solution is selected we compare the difference  $\Delta f = f(s) - f(s_0)$ . If the difference  $\Delta f$  is negative, the neighbor replaces the current solution. Otherwise, if the difference is positive the neighbor is accepted with a probability based upon the Boltzmann distribution  $P_{accept}(\Delta f) \approx \exp(\frac{-\Delta f}{kT})$  where  $k$  is a constant and the temperature  $T$  is a control parameter. This temperature is gradually lowered following a geometrical law function  $g(T) : g(T) = \alpha T$  with  $\alpha < 1$ .

## 5.1 Computational results

We have coded the simulated annealing algorithm in Java. The design of experience is the same as in subsection 4.4. The initial solution is taken according to the best solution obtained by the steepest descent method (see subsection 4.3) on the random list and on the pseudo random list. The running time for each simulation is 360 seconds. The results of the simulated annealing are reported from table 11 to table 50. It should be noted that there is no significant difference between starting from a high temperature or a low temperature. Considering we start from an initial solution which was well improved by local improvements, the fact of starting from a high temperature does not have a significant influence on the objective function. Starting from the initial solution, the objective function can be improved up to 46,19 % table 22 (random list) and 43,14% table 27 (pseudo random list). This confirms what we had already suspected that considering random lists, we obtain the best solutions and also the worst ones. Indeed, the random list covers a greater neighborhood than the pseudo random one. We can also improve the objective function while slightly worsening some indicators table 11, table 12, table 13, table 14, table 18 for random list and table 18 for pseudo random list.

The “number of moves” indicator is rather difficult to improve. The majority of the tests slightly worsen it but we can reach an improvement on it up to 1,33% (table 41, random list) and 1,27% (table 42, pseudo random list). The batching coefficient can be improve up to 3,54% (table 44, random list) and 2,53 (table 39, pseudo random list). The X-factor is the indicator where the improvements are the more significant. There are improvements which go up to 33,74% (table 48, random list) and 33,27% (table 49, pseudo random list).

## 6 The BOS software

The use of a disjunctive graph brings significant improvements for interactive scheduling at the fab level. A prototype software (Figure 1 gives the graphical user interface of the software), the Batch Optimization Software, has been developed jointly by ATMEL, the University of Avignon (LIA) and the École des Mines de Saint-Etienne (CMP Gardanne). It includes the off-line batching and scheduling phase, as described above, and also an interactive module that allows the decision-makers to make modifications and to test options directly on the proposed plan. The longest path calculations in the disjunctive graph allow to almost immediately give the impact of a modification such as job removal, operation move, job or operation insertion. The software is currently in phase of intensive test in the ATMEL diffusion area.

## 7 Concluding remarks

We have proposed a model and a method based on a disjunctive graph for batching and scheduling problem in a semiconductor manufacturing factory while taking into account complex constraints and optimizing multiple measures. A constructive algorithm has been proposed to solve the problem, local search improvements based on the disjunctive graph representation have been defined and a simulated annealing algorithm has been developed. The computational tests made on real instances of ATMEL showed that we obtain quality solutions in fast calculation times. The simulated annealing procedure improve significantly the different criteria (number of moves, batching coefficient and X-factor) in comparison with a simple local search method.

This research can be extended in many ways. Other types of moves such as the simultaneous move of 2 jobs linked by a maximum time lag should be tested. The maximal time lags are currently hard constraints. However in practice some of them can be relaxed and treated as soft constraints or objectives. Another important issue can be in a multicriteria analysis study. Computing several Pareto solutions are useful particularly when the situation frequently changes as it is usually the case of semiconductor factories.

## Acknowledgements

This work is part of the MEDEA+ European project HYMNE (High Yield driven MaNuufacturing Excellence in sub 65 nm CMOS), partly funded by the “Ministère de l’Économie, des Finances et de l’Industrie” (French Ministry of Economy, Finance and Industry).

## References

- [ABZ88] J. Adams, E. Balas, and D. Zawack. The shifting bottleneck procedure for job shop scheduling. *Management Science*, 34:391–401, 1988.
- [DF03] R. M. Dabbas and J. W. Fowler. A new scheduling approach using combined dispatching criteria in wafer fabs. *IEEE Transactions on Semiconductor Manufacturing*, 16(3):501–510, 2003.
- [GNST04] K. Gentner, K. Neumann, C. Schwindt, and N. Trautmann. Batch production scheduling in the process industries. In J.Y.T. Leung, editor, *Handbook of Scheduling: Algorithms, Models and Performance Analysis*, pages 48.1–48.21. CRC Press, Boca Raton, 2004.
- [GR88] C. Glassey and M. Resende. Closed-loop job release control for vlsi circuit manufacturing. *IEEE Transactions on Semiconductor manufacturing*, 1(1):36–46, 1988.
- [HC03] T.-K. Hwang and S.-C.g Chang. Design of a lagrangian relaxation-based hierarchical production scheduling environment for semiconductor wafer fabrication. *IEEE Transactions on Robotics and Automation*, 19(4):566–578, 2003.
- [ICN<sup>+</sup>03] K. Ibrahim, M. A. Chik, W. S. Nizam, N. L. Fem, and N. F. Za’bah. Efficient lot batching system for furnace operation. In *IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, pages 322–324, 2003.
- [ICS<sup>+</sup>03] K. Ibrahim, M. A. Chik, W. S.Nizam, N. L. Fem, and N. F. Za’bah. Efficient lot batching system for furnace operation. In *IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, pages 322–324, 2003.

- [JH90] G.S. JR and T. Hout. Competing against time: how time-based competition is reshaping global markets. Technical report, The Free Press, 1990.
- [KGV83] S. Kirkpatrick, C. Gelatt, and M. Vecchi. Optimization by simulated annealing. *Science*, tome 220(4598):671–680, 1983.
- [Kum94] P.R. Kumar. Scheduling semiconductor manufacturing plants. *IEEE Control Systems Magazine*, 14(6):30–40, 1994.
- [LCPC96] D.-Y. Liao, S.-C. Chang, K.-W. Pei, and C.-M. Chang. Daily scheduling for r&d semiconductor fabrication. *IEEE Transactions on Semiconductor Manufacturing*, 9(4):550–560, 1996.
- [Lit61] J. Little. A proof for the queuing formula  $l = \lambda w$ . *Operations Research*, 9:383–387, 1961.
- [MFC02] S.J. Mason, J.W. Fowler, and W.M. Carlyle. A modified shifting bottleneck heuristic for minimizing total weighted tardiness in complex job shops. *Journal of Scheduling*, 5(3):247–262, 2002.
- [MH03] L. Monch and I. Habenicht. Simulation-based assessment of batching heuristics in semiconductor manufacturing. In S. Chick, P.J. Sanchez, D. Ferrin, and D. J. Morrice, editors, *Winter Simulation Conference*, pages 1338–1345. ACM, 2003.
- [MT06] J. Montoya-Torres. Manufacturing performance evaluation in wafer semiconductor factories. *International Journal of Productivity and Performance management*, 55(3/4):300–310, 2006.
- [TIHC05] H. Toba, H. Izumi, H. Hatada, and T. Chikushima. Dynamic load balancing among multiple fabrication lines through estimation of minimum inter-operation time. *IEEE Transactions on Semiconductor Manufacturing*, 18(1):202–213, 2005.
- [VS06] A. Varadarajan and S. C. Sarin. A survey of dispatching rules for operational control in wafer fabrication. In A. Dolgui, G. Morel, and C. E. Pereira, editors, *12th IFAC Symposium on Information Control Problems in Manufacturing*, pages 715–726, 2006.
- [Wei88] L. M. Wein. Scheduling semiconductor wafer fabrication. *IEEE Transactions on Semiconductor Manufacturing*, 1(3):115–130, 1988.

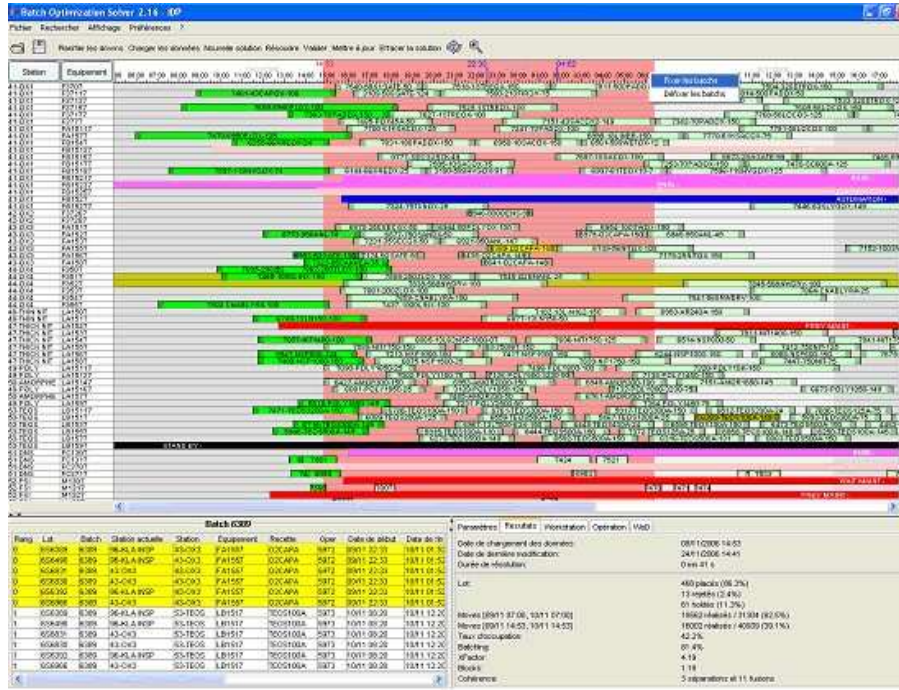


Figure 1: Interface of the prototype

	Objective	Moves	Batching Coeff.	X-factor	Time
<b>(Random list)</b>					
Sorted list	-3,38E+07	8158	0,70	2,47	2 s
Mean value	-3,41E+07	8147	0,70	2,42	2 s
St. deviation	-1,03%	1,45%	1,43%	2,09%	0,00%
Best	-3,32E+07	8333	0,72	2,28	2 s
Worst	-3,53E+07	7708	0,67	2,61	2 s
Best solution obtained	-3,32E+07	8254	0,71	2,36	2 s
Improvement	1,77%	1,17%	1,42%	4,45%	—
<b>(Pseudo random list)</b>					
Sorted list	-3,38E+07	8158	0,70	2,47	3 s
Mean value	-3,37E+07	8140	0,70	2,47	2 s
St. deviation	-0,20%	0,13%	0,12%	0,10%	0,00%
Best	-3,35E+07	8158	0,70	2,47	2 s
Worst	-3,39E+07	8133	0,70	2,48	2 s
Best solution obtained	-3,35E+07	8133	0,70	2,47	2 s
Improvement	0,88%	-0,30%	0,00%	0,00%	—

Table 1: Results on instance-1(060221): computation results and local improvements



	Objective	Moves	Batching Coeff.	X-factor	Time
<b>(Random list)</b>					
Sorted list	-2,38E+07	8201	0,74	2,78	3 s
Mean value	-2,43E+07	8128	0,73	2,79	3 s
St. deviation	-1,51%	0,93%	1,39%	1,76%	0,00%
Best	-2,26E+07	8226	0,76	2,64	3 s
Worst	-2,56E+07	7828	0,69	2,93	3 s
Best solution obtained	-2,26E+07	8126	0,74	2,70	3 s
Improvement	5,04%	-0,91%	0,00%	2,87%	–
<b>(Pseudo random list)</b>					
Sorted list	-2,38E+07	8201	0,74	2,78	3 s
Mean value	-2,37E+07	8201	0,74	2,77	3 s
St. deviation	-0,36%	0,00%	0,00%	0,38%	0,00%
Best	-2,36E+07	8201	0,74	2,75	3 s
Worst	-3,38E+07	8201	0,74	2,78	3 s
Best solution obtained	-2,36E+07	8201	0,74	2,75	3 s
Improvement	1,26%	0,00%	0,00%	1,07%	–

Table 2: Results on instance-2(060224): computation results and local improvements

	Objective	Moves	Batching Coeff.	X-factor	Time
<b>(Random list)</b>					
Sorted list	-6,13E+07	15115	0,73	3,13	12 s
Mean value	-6,55E+07	15178	0,72	3,20	12,72 s
St. deviation	-1,59%	1,92%	1,26%	1,62%	7,13%
Best	-6,13E+07	16012	0,76	3,05	12 s
Worst	-6,97E+07	14222	0,69	3,39	18 s
Best solution obtained	-6,13E+07	15289	0,75	3,10	12 s
Improvement	0,00%	1,15%	2,73%	0,95%	–
<b>(Pseudo random list)</b>					
Sorted list	-6,13E+07	15115	0,73	3,13	12 s
Mean value	-5,79E+07	14927	0,74	3,12	11,47 s
St. deviation	-0,47%	0,51%	0,35%	0,52%	4,35%
Best	-5,73E+07	15041	0,74	3,08	11 s
Worst	-5,87E+07	14715	0,73	3,17	12 s
Best solution obtained	-5,73E+07	15015	0,74	3,10	11 s
Improvement	6,52%	-0,66%	1,36%	0,95%	–

Table 3: Results on instance-3(060303): computation results and local improvements

	Objective	Moves	Batching Coeff.	X-factor	Time
<b>(Random list)</b>					
Sorted list	-4,16E+07	15193	0,79	2,98	13 s
Mean value	-4,28E+07	15461	0,76	3	13,97 s
St. deviation	-1,78%	1,76%	1,06%	1,53%	9,93%
Best	-4,05E+07	16053	0,78	2,86	12 s
Worst	-4,71E+07	14042	0,73	3,16	25 s
Best solution obtained	-4,05E+07	15642	0,76	2,91	13 s
Improvement	2,64%	2,96%	-3,80%	2,35%	–
<b>(Pseudo random list)</b>					
Sorted list	-4,16E+07	15193	0,79	2,98	13 s
Mean value	-4,21E+07	15335	0,78	2,94	12,94 s
St. deviation	-0,90%	1,40%	0,61%	1,56%	2,26%
Best	-4,11E+07	15716	0,79	2,84	12 s
Worst	-4,36E+07	14818	0,76	3,04	13 s
Best solution obtained	-4,11E+07	15419	0,77	2,98	13 s
Improvement	1,20%	1,49%	-2,53%	0,00%	–

Table 4: Results on instance-4(060310): computation results and local improvements

	Objective	Moves	Batching Coeff.	X-factor	Time
<b>(Random list)</b>					
Sorted list	-5,88E+07	16383	0,75	3,29	17 s
Mean value	-5,99E+07	15781	0,74	3,38	15,68 s
St. deviation	-1,31%	1,69%	1,19%	1,32%	4,01%
Best	-5,79E+07	16441	0,76	3,26	14 s
Worst	-6,35E+07	14732	0,71	3,55	22 s
Best solution obtained	-5,79E+07	15995	0,75	3,33	16 s
Improvement	1,53%	-2,37%	0,00%	-1,22%	–
<b>(Pseudo random list)</b>					
Sorted list	-5,88E+07	16383	0,75	3,29	17 s
Mean value	-5,90E+07	16248	0,74	3,31	16,43 s
St. deviation	-0,41%	0,50%	0,47%	0,56%	3,01%
Best	-5,83E+07	16383	0,75	3,22	16 s
Worst	-5,96E+07	15917	0,73	3,34	17 s
Best solution obtained	-5,83E+07	16148	0,74	3,28	16 s
Improvement	0,85%	-1,43%	-1,33%	0,61%	–

Table 5: Results on instance-5(060317): computation results and local improvements

	Objective	Moves	Batching Coeff.	X-factor	Time
<b>(Random list)</b>					
Sorted list	-4,50E+07	14804	0,75	2,45	11 s
Mean value	-4,60E+07	14575	0,73	2,50	10,55 s
St. deviation	-1,38%	1,71%	1,16%	1,82%	5,29%
Best	-4,43E+07	15190	0,76	2,36	10 s
Worst	-4,91E+07	13419	0,70	2,64	14 s
Best solution obtained	-4,43E+07	14415	0,74	2,43	10 s
Improvement	1,56%	-2,63%	-1,33%	0,82%	–
<b>(Pseudo random list)</b>					
Sorted list	-4,50E+07	14804	0,75	2,45	11 s
Mean value	-4,49E+07	14834	0,75	2,43	10 s
St. deviation	-0,44%	0,41%	0,55%	0,95%	0,32%
Best	-4,43E+07	14929	0,76	2,38	10 s
Worst	-4,56E+07	14581	0,73	2,50	11 s
Best solution obtained	-4,43E+07	14854	0,74	2,39	10 s
Improvement	1,56%	0,34%	-1,33%	2,45%	–

Table 6: Results on instance-6(060331): computation results and local improvements

	Objective	Moves	Batching Coeff.	X-factor	Time
<b>(Random list)</b>					
Sorted list	-5,02E+07	13112	0,70	3,39	9 s
Mean value	-5,04E+07	13326	0,69	3,48	9,30 s
St. deviation	-2,17%	2,42%	1,32%	1,42%	7,90%
Best	-4,68E+07	14035	0,72	3,31	8 s
Worst	-5,71E+07	11929	0,66	3,67	16 s
Best solution obtained	-4,68E+07	13701	0,71	3,45	10 s
Improvement	6,77%	4,49%	1,43%	-1,77%	–
<b>(Pseudo random list)</b>					
Sorted list	-5,02E+07	13112	0,70	3,39	9 s
Mean value	-5,02E+07	13266	0,70	3,40	9,01 s
St. deviation	-1,57%	1,38%	0,57%	0,54%	0,99%
Best	-4,71E+07	13526	0,71	3,35	9 s
Worst	-5,22E+07	12665	0,69	3,46	10 s
Best solution obtained	-4,71E+07	13360	0,71	3,40	9 s
Improvement	6,18%	1,89%	1,43%	-0,29%	–

Table 7: Results on instance-7(060407): computation results and local improvements

	Objective	Moves	Batching Coeff.	X-factor	Time
<b>(Random list)</b>					
Sorted list	-3,53E+07	13912	0,76	2,91	9 s
Mean value	-3,61E+07	13861	0,74	2,95	9,01 s
St. deviation	-2,40%	1,71%	1,29%	1,44%	0,99%
Best	-3,34E+07	14411	0,76	2,82	8 s
Worst	-3,99E+07	12839	0,70	3,08	10 s
Best solution obtained	-3,34E+07	13952	0,75	2,92	9 s
Improvement	5,38%	0,29%	-1,32%	-0,34%	–
<b>(Pseudo random list)</b>					
Sorted list	-3,53E+07	13912	0,76	2,91	9 s
Mean value	-3,50E+07	13810	0,76	2,94	9 s
St. deviation	-2,10%	0,88%	0,75%	0,86%	0,00%
Best	-3,35E+07	14077	0,78	2,87	9 s
Worst	-3,67E+07	13357	0,74	3,02	9 s
Best solution obtained	-3,35E+07	14051	0,76	2,88	9 s
Improvement	5,10%	1,00%	0,00%	1,03%	–

Table 8: Results on instance-8(060414): computation results and local improvements

	Objective	Moves	Batching Coeff.	X-factor	Time
<b>(Random list)</b>					
Sorted list	-5,57E+07	13845	0,69	2,95	11 s
Mean value	-5,65E+07	13908	0,70	3,10	11,45 s
St. deviation	-1,88%	2,38%	1,28%	1,75%	10,03%
Best	-5,28E+07	14662	0,72	2,92	10 s
Worst	-6,14E+07	12229	0,67	3,29	19 s
Best solution obtained	-5,28E+07	14070	0,70	3,06	10 s
Improvement	5,21%	1,99%	1,45%	-3,73%	–
<b>(Pseudo random list)</b>					
Sorted list	-5,57E+07	13845	0,69	2,95	11s
Mean value	-5,56E+07	13740	0,70	3,01	11 s
St. deviation	-0,30%	0,67%	0,31%	0,35%	0,00%
Best	-5,52E+07	13994	0,70	2,98	11 s
Worst	-5,61E+07	13643	0,69	3,04	11 s
Best solution obtained	-5,52E+07	13919	0,70	3	11 s
Improvement	0,90%	0,53%	1,45%	-1,69%	–

Table 9: Results on instance-9(060421): computation results and local improvements

	Objective	Moves	Batching Coeff.	X-factor	Time
<b>(Random list)</b>					
Sorted list	-4,86E+07	15748	0,70	3,53	13s
Mean value	-4,82E+07	15332	0,70	3,59	12,10 s
St. deviation	-1,66%	1,70%	1,16%	1,12%	6,24%
Best	-4,56E+07	16079	0,72	3,45	11 s
Worst	-5,09E+07	14252	0,68	3,72	20 s
Best solution obtained	-4,56E+07	15258	0,71	3,52	12 s
Improvement	6,17%	-3,11%	1,43%	0,28%	—
<b>(Pseudo random list)</b>					
Sorted list	-4,86E+07	15748	0,70	3,53	13s
Mean value	-4,85E+07	15599	0,70	3,54	12 s
St. deviation	-0,54%	0,46%	0,51%	0,28%	0,00%
Best	-4,78E+07	15748	0,71	3,51	12 s
Worst	-4,92E+07	15329	0,69	3,56	12 s
Best solution obtained	-4,78E+07	15592	0,70	3,53	12 s
Improvement	1,65%	-0,99%	0,00%	0,00%	—

Table 10: Results on instance-10(060428): computation results and local improvements

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-3,32E+07	8254	0,71	2,36
Mean value	-3,40E+07	8150	0,70	2,42
St. deviation	-1,27%	1,44%	1,43%	2,12%
Best	-3,27E+07	8333	0,72	2,28
Worst	-3,53E+07	7708	0,67	2,61
Best solution obtained	-3,27E+07	8179	0,71	2,36
Improvement	1,5%	-0,9%	0,00%	0,00%
<b>(Pseudo random list)</b>				
Initial solution	-3,35E+07	8133	0,70	2,47
Mean value	-3,41E+07	8150	0,70	2,42
St. deviation	-1,13%	1,44%	1,40%	2,12%
Best	-3,32E+07	8333	0,72	2,28
Worst	-3,53E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8183	0,70	2,43
Improvement	0,8%	0,6%	0,00%	1,61%

Table 11: Results on instance-1(060221): Simulated annealing, Initial Temperature 32000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-3,32E+07	8254	0,71	2,36
Mean value	-3,40E+07	8148	0,70	2,42
St. deviation	-1,28%	1,43%	1,42%	2,14%
Best	-3,27E+07	8333	0,72	2,28
Worst	-3,53E+07	7708	0,67	2,61
Best solution obtained	-3,27E+07	8154	0,70	2,36
Improvement	1,5%	-1,21%	-1,4%	0,00%
<b>(Pseudo random list)</b>				
Initial solution	-3,35E+07	8133	0,70	2,47
Mean value	-3,41E+07	8150	0,70	2,42
St. deviation	-1,14%	1,45%	1,40%	2,11%
Best	-3,32E+07	8333	0,72	2,28
Worst	-3,53E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8183	0,70	2,43
Improvement	0,89%	0,00%	0,00%	1,61%

Table 12: Results on instance-1(060221): Simulated annealing, Initial Temperature 32000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-3,32E+07	8254	0,71	2,36
Mean value	-3,40E+07	8149	0,70	2,42
St. deviation	-1,27%	1,44%	1,43%	2,12%
Best	-3,27E+07	8333	0,72	2,28
Worst	-3,53E+07	7708	0,67	2,61
Best solution obtained	-3,27E+07	8029	0,70	2,44
Improvement	1,5%	-2,72%	-1,4%	-3,38%
<b>(Pseudo random list)</b>				
Initial solution	-3,35E+07	8133	0,70	2,47
Mean value	-3,41E+07	8151	0,70	2,42
St. deviation	-1,13%	1,44%	1,40%	2,10%
Best	-3,32E+07	8333	0,72	2,28
Worst	-3,53E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8183	0,70	2,38
Improvement	0,89%	0,00%	0,00%	3,61%

Table 13: Results on instance-1(060221): Simulated annealing, Initial Temperature 65000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-3,32E+07	8254	0,71	2,36
Mean value	-3,40E+07	8148	0,70	2,42
St. deviation	-1,28%	1,43%	1,42%	2,14%
Best	-3,27E+07	8333	0,72	2,28
Worst	-3,53E+07	7708	0,67	2,61
Best solution obtained	-3,27E+07	8154	0,70	2,36
Improvement	1,5%	-1,21%	-1,4%	0,00%
<b>(Pseudo random list)</b>				
Initial solution	-3,35E+07	8133	0,70	2,47
Mean value	-3,41E+07	8150	0,70	2,42
St. deviation	-1,14%	1,45%	1,40%	2,11%
Best	-3,32E+07	8333	0,72	2,28
Worst	-3,53E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8183	0,70	2,43
Improvement	0,89%	0,00%	0,00%	1,61%

Table 14: Results on instance-1(060221): Simulated annealing, Initial Temperature 65000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-2,32E+07	8176	0,73	2,73
Mean value	-3,36E+07	8148	0,70	2,44
St. deviation	-6,66%	1,41%	1,92%	3,41%
Best	-2,29E+07	8333	0,75	2,28
Worst	-3,53E+07	7708	0,67	2,77
Best solution obtained	-2,29E+07	8176	0,73	2,69
Improvement	1,22%	0,00%	0,00%	1,30%
<b>(Pseudo random list)</b>				
Initial solution	-2,36E+07	8201	0,74	2,75
Mean value	-3,36E+07	8150	0,70	2,44
St. deviation	-6,42%	1,42%	1,89%	3,39%
Best	-2,33E+07	8333	0,74	2,28
Worst	-3,53E+07	7708	0,67	2,75
Best solution obtained	-2,33E+07	8201	0,74	2,73
Improvement	1,23%	0,00%	0,00%	1,06%

Table 15: Results on instance-1(060224): simulated annealing, initial temperature 32000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-2,30E+07	8102	0,73	2,77
Mean value	-3,35E+07	8144	0,70	2,44
St. deviation	-6,87%	1,42%	2,20%	3,70%
Best	-2,25E+07	8333	0,75	2,28
Worst	-3,53E+07	7708	0,67	2,81
Best solution obtained	-2,25E+07	8076	0,75	2,79
Improvement	2,05%	-0,32%	3,51%	-0,75%
<b>(Pseudo random list)</b>				
Initial solution	-2,36E+07	8201	0,74	2,75
Mean value	-3,36E+07	8150	0,70	2,44
St. deviation	-6,42%	1,42%	1,89%	3,37%
Best	-2,33E+07	8333	0,74	2,28
Worst	-3,53E+07	7708	0,67	2,74
Best solution obtained	-2,33E+07	8201	0,74	2,72
Improvement	1,24%	0,00%	0,00%	1,10%

Table 16: Results on instance (060224): Simulated annealing, Initial Temperature 32000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-2,26E+07	8126	0,74	2,70
Mean value	-3,35E+07	8146	0,70	2,43
St. deviation	-7,06%	1,41%	2,03%	2,93%
Best	-2,22E+07	8333	0,75	2,28
Worst	-3,53E+07	7708	0,67	2,69
Best solution obtained	-2,22E+07	8101	0,75	2,68
Improvement	1,83%	-0,31%	0,97%	0,71%
<b>(Pseudo random list)</b>				
Initial solution	-2,36E+07	8201	0,74	2,75
Mean value	-3,36E+07	8150	0,70	2,44
St. deviation	-6,41%	1,42%	1,90%	3,38%
Best	-2,33E+07	8333	0,74	2,28
Worst	-3,53E+07	7708	0,67	2,75
Best solution obtained	-2,33E+07	8201	0,74	2,71
Improvement	1,14%	0,00%	0,00%	1,77%

Table 17: Results on instance-1(060224): simulated annealing, initial temperature 65000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-2,31E+07	8151	0,73	2,74
Mean value	-3,35E+07	8147	0,70	2,43
St. deviation	-6,90%	1,41%	1,87%	3,21%
Best	-2,25E+07	8333	0,74	2,28
Worst	-3,53E+07	7708	0,67	2,74
Best solution obtained	-2,25E+07	8151	0,73	2,68
Improvement	2,70%	0,00%	0,00%	2,26%
<b>(Pseudo random list)</b>				
Initial solution	-2,36E+07	8201	0,74	2,75
Mean value	-3,36E+07	8150	0,70	2,44
St. deviation	-6,37%	1,42%	1,90%	3,55%
Best	-2,34E+07	8333	0,74	2,28
Worst	-3,53E+07	7708	0,67	2,77
Best solution obtained	-2,34E+07	8201	0,74	2,75
Improvement	0,74%	0,00%	0,00%	0,00%

Table 18: Results on instance-1(060224): simulated annealing, initial temperature 65000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-6,14E+07	15215	0,72	3,15
Mean value	-3,52E+07	8516	0,70	2,45
St. deviation	-12,61%	18,91%	2,33%	6,32%
Best	-3,32E+07	15764	0,76	2,28
Worst	-5,70E+07	7708	0,67	3,15
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	45,96%	-45,75%	-1,62%	25,07%
<b>(Pseudo random list)</b>				
Initial solution	-5,72E+07	15015	0,74	3,10
Mean value	-3,51E+07	8494	0,70	2,45
St. deviation	-11,29%	17,86%	2,17%	5,98%
Best	-3,32E+07	15389	0,76	2,28
Worst	-5,45E+07	7708	0,67	3,13
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	41,92%	-45,03%	-3,41%	23,86%

Table 19: Results on instance (060303): Simulated annealing, Initial Temperature 32000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-6,12E+07	15315	0,73	3,13
Mean value	-3,53E+07	8498	0,70	2,45
St. deviation	-13,04%	18,07%	1,87%	6,03%
Best	-3,32E+07	15639	0,74	2,28
Worst	-5,76E+07	7708	0,67	3,15
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	45,71%	-46,11%	-2,37%	24,60%
<b>(Pseudo random list)</b>				
Initial solution	-5,72E+07	15016	0,74	3,09
Mean value	-3,51E+07	8492	0,70	2,45
St. deviation	-11,37%	17,75%	2,24%	6,13%
Best	-3,32E+07	15240	0,76	2,28
Worst	-5,47E+07	7708	0,67	3,13
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	41,95%	-45,03%	-3,41%	23,75%

Table 20: Results on instance (060303): Simulated annealing, Initial Temperature 32000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-6,12E+07	15289	0,75	3,10
Mean value	-3,52E+07	8509	0,70	2,45
St. deviation	-12,56%	18,59%	2,11%	5,91%
Best	-3,32E+07	15567	0,75	2,28
Worst	-5,68E+07	7708	0,67	3,11
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	45,75%	-46,01%	-4,67%	23,91%
<b>(Pseudo random list)</b>				
Initial solution	-5,72E+07	15015	0,74	3,10
Mean value	-3,51E+07	8492	0,70	2,45
St. deviation	-11,37%	17,76%	2,25%	5,96%
Best	-3,32E+07	15315	0,76	2,28
Worst	-5,46E+07	7708	0,67	3,12
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	42,00%	-45,03%	-3,41%	23,94%

Table 21: Results on instance (060303): Simulated annealing, Initial Temperature 65000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-6,17E+07	15716	0,74	3,12
Mean value	-3,53E+07	8501	0,70	2,45
St. deviation	-13,22%	18,21%	1,84%	5,81%
Best	-3,32E+07	15687	0,74	2,28
Worst	-5,80E+07	7708	0,67	3,12
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	46,19%	-47,48%	-3,41%	24,44%
<b>(Pseudo random list)</b>				
Initial solution	-5,72E+07	15015	0,74	3,10
Mean value	-3,51E+07	8481	0,70	2,45
St. deviation	-11,54%	17,20%	2,03%	6,20%
Best	-3,32E+07	14865	0,75	2,28
Worst	-5,47E+07	7708	0,67	3,11
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	41,92%	-45,03%	-3,41%	23,84%

Table 22: Results on instance (060303): Simulated annealing, Initial Temperature 65000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,03E+07	15318	0,76	2,92
Mean value	-3,43E+07	8516	0,70	2,44
St. deviation	-1,90%	18,91%	2,77%	4,15%
Best	-3,32E+07	15828	0,78	2,28
Worst	-3,72E+07	7708	0,67	2,91
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	17,67%	-46,12%	-6,28%	19,33%
<b>(Pseudo random list)</b>				
Initial solution	-4,12E+07	15518	0,78	2,99
Mean value	-3,43E+07	8520	0,70	2,44
St. deviation	-2,49%	19,12%	3,14%	4,48%
Best	-3,32E+07	15741	0,80	2,28
Worst	-3,83E+07	7708	0,67	2,95
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	19,41%	-46,81%	-9,21%	21,16%

Table 23: Results on instance (060310): Simulated annealing, Initial Temperature 32000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,01E+07	15479	0,76	2,94
Mean value	-3,43E+07	8519	0,70	2,44
St. deviation	-1,90%	19,09%	3,05%	4,55%
Best	-3,32E+07	15728	0,79	2,28
Worst	-3,71E+07	7708	0,67	2,97
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	17,16%	-46,68%	-6,40%	19,90%
<b>(Pseudo random list)</b>				
Initial solution	-4,10E+07	15628	0,78	2,97
Mean value	-3,43E+07	8525	0,70	2,44
St. deviation	-2,45%	19,37%	3,10%	4,32%
Best	-3,32E+07	15828	0,80	2,28
Worst	-3,82E+07	7708	0,67	2,93
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	19,02%	-47,18%	-8,64%	20,49%

Table 24: Results on instance (060310): Simulated annealing, Initial Temperature 32000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,05E+07	15642	0,76	2,91
Mean value	-3,43E+07	8515	0,70	2,44
St. deviation	-2,36%	18,90%	3,11%	4,47%
Best	-3,32E+07	15706	0,79	2,28
Worst	-3,80E+07	7708	0,67	2,94
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	18,05%	-47,23%	-6,18%	18,86%
<b>(Pseudo random list)</b>				
Initial solution	-4,12E+07	15416	0,78	2,99
Mean value	-3,43E+07	8522	0,70	2,44
St. deviation	-2,58%	19,20%	3,01%	4,34%
Best	-3,32E+07	15853	0,80	2,28
Worst	-3,85E+07	7708	0,67	2,96
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	19,41%	-46,46%	-8,55%	21,12%

Table 25: Results on instance (060310): Simulated annealing, Initial Temperature 65000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-3,92E+07	15366	0,78	2,89
Mean value	-3,42E+07	8525	0,70	2,44
St. deviation	-1,76%	19,38%	2,93%	3,90%
Best	-3,32E+07	15878	0,79	2,28
Worst	-3,69E+07	7708	0,67	2,84
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	15,40%	-46,28%	-8,84%	18,42%
<b>(Pseudo random list)</b>				
Initial solution	-4,11E+07	15591	0,78	2,95
Mean value	-3,43E+07	8528	0,70	2,44
St. deviation	-2,46%	19,53%	3,15%	4,22%
Best	-3,32E+07	15803	0,80	2,28
Worst	-3,81E+07	7708	0,67	2,87
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	19,12%	-47,06%	-8,99%	19,94%

Table 26: Results on instance (060310): Simulated annealing, Initial Temperature 65000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-5,70E+07	15242	0,71	3,37
Mean value	-3,51E+07	8533	0,70	2,46
St. deviation	-10,65%	19,74%	2,31%	7,65%
Best	-3,32E+07	16180	0,76	2,28
Worst	-5,33E+07	7708	0,67	3,30
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	41,74%	-45,85%	0,00%	29,94%
<b>(Pseudo random list)</b>				
Initial solution	-5,84E+07	16248	0,75	3,30
Mean value	-3,52E+07	8544	0,70	2,46
St. deviation	-12,42%	20,27%	2,43%	7,81%
Best	-3,32E+07	16248	0,77	2,28
Worst	-5,66E+07	7708	0,67	3,32
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	43,14%	-49,20%	-4,76%	28,58%

Table 27: Results on instance (060317): Simulated annealing, Initial Temperature 32000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-5,80E+07	16092	0,73	3,25
Mean value	-3,52E+07	8551	0,70	2,46
St. deviation	-12,14%	20,63%	2,13%	7,29%
Best	-3,32E+07	16342	0,76	2,28
Worst	-5,60E+07	7708	0,67	3,27
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	42,71%	-48,71%	-2,74%	27,41%
<b>(Pseudo random list)</b>				
Initial solution	-5,84E+07	16173	0,74	3,28
Mean value	-3,52E+07	8546	0,70	2,46
St. deviation	-12,41%	20,38%	2,47%	7,59%
Best	-3,32E+07	16300	0,77	2,28
Worst	-5,65E+07	7708	0,67	3,30
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	43,11%	-48,96%	-4,36%	28,01%

Table 28: Results on instance (060317): Simulated annealing, Initial Temperature 32000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-5,79E+07	15995	0,75	3,33
Mean value	-3,52E+07	8538	0,70	2,46
St. deviation	-12,09%	20,02%	2,44%	7,63%
Best	-3,32E+07	16119	0,77	2,28
Worst	-5,59E+07	7708	0,67	3,30
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	42,62%	-48,40%	-5,16%	29,24%
<b>(Pseudo random list)</b>				
Initial solution	-5,83E+07	16148	0,74	3,28
Mean value	-3,52E+07	8542	0,70	2,46
St. deviation	-12,45%	20,21%	2,37%	7,69%
Best	-3,32E+07	16248	0,76	2,28
Worst	-5,67E+07	7708	0,67	3,31
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	43,06%	-48,89%	-4,22%	28,12%

Table 29: Results on instance (060317): Simulated annealing, Initial Temperature 65000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-5,74E+07	15767	0,75	3,35
Mean value	-3,51E+07	8533	0,70	2,46
St. deviation	-11,30%	19,76%	2,36%	7,65%
Best	-3,32E+07	16141	0,77	2,28
Worst	-5,46E+07	7708	0,67	3,32
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	42,17%	-47,65%	-4,62%	29,59%
<b>(Pseudo random list)</b>				
Initial solution	-5,83E+07	16148	0,74	3,28
Mean value	-3,52E+07	8540	0,70	2,46
St. deviation	-12,44%	20,11%	2,43%	7,75%
Best	-3,32E+07	16050	0,76	2,28
Worst	-5,63E+07	7708	0,67	3,30
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	43,05%	-48,89%	-4,22%	28,12%

Table 30: Results on instance (060317): Simulated annealing, Initial Temperature 65000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,41E+07	14965	0,75	2,45
Mean value	-3,45E+07	8490	0,70	2,42
St. deviation	-5,01%	17,64%	2,19%	2,10%
Best	-3,32E+07	15103	0,76	2,28
Worst	-4,28E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	24,75%	-44,84%	-5,40%	3,87%
<b>(Pseudo random list)</b>				
Initial solution	-4,44E+07	14879	0,75	2,43
Mean value	-3,45E+07	8486	0,70	2,42
St. deviation	-5,04%	17,45%	2,31%	2,21%
Best	-3,32E+07	15227	0,76	2,28
Worst	-4,30E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	25,21%	-44,53%	-4,95%	2,84%

Table 31: Results on instance (060331): Simulated annealing, Initial Temperature 32000, step 25



	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,43E+07	14656	0,75	2,47
Mean value	-3,45E+07	8477	0,70	2,42
St. deviation	-4,79%	17,02%	2,27%	2,07%
Best	-3,32E+07	14865	0,76	2,28
Worst	-4,25E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	25,08%	-43,68%	-4,80%	4,43%
<b>(Pseudo random list)</b>				
Initial solution	-4,44E+07	14879	0,75	2,42
Mean value	-3,45E+07	8484	0,70	2,42
St. deviation	-4,98%	17,39%	2,33%	2,26%
Best	-3,32E+07	15202	0,76	2,28
Worst	-4,30E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	25,16%	-44,53%	-4,95%	2,56%

Table 32: Results on instance (060331): Simulated annealing, Initial Temperature 32000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,43E+07	14415	0,74	2,43
Mean value	-3,45E+07	8463	0,70	2,42
St. deviation	-4,51%	16,33%	2,06%	2,12%
Best	-3,32E+07	14628	0,75	2,28
Worst	-4,20E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	25,00%	-42,74%	-3,42%	2,84%
<b>(Pseudo random list)</b>				
Initial solution	-4,43E+07	14854	0,74	2,39
Mean value	-3,45E+07	8485	0,70	2,42
St. deviation	-4,99%	17,41%	2,25%	2,21%
Best	-3,32E+07	15152	0,76	2,28
Worst	-4,30E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	25,06%	-44,43%	-4,13%	1,48%

Table 33: Results on instance (060331): Simulated annealing, Initial Temperature 65000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,44E+07	14984	0,74	2,43
Mean value	-3,46E+07	8486	0,70	2,42
St. deviation	-5,05%	17,44%	2,18%	2,11%
Best	-3,32E+07	15106	0,76	2,28
Worst	-4,30E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	25,29%	-44,91%	-3,55%	2,89%
<b>(Pseudo random list)</b>				
Initial solution	-4,44E+07	14731	0,75	2,44
Mean value	-3,46E+07	8489	0,70	2,42
St. deviation	-5,09%	17,63%	2,16%	2,12%
Best	-3,32E+07	15007	0,75	2,28
Worst	-4,29E+07	7708	0,67	2,61
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	25,28%	-43,97%	-4,73%	3,15%

Table 34: Results on instance (060331): Simulated annealing, Initial Temperature 65000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,79E+07	13606	0,70	3,46
Mean value	-3,47E+07	8427	0,70	2,46
St. deviation	-6,67%	14,54%	1,65%	8,04%
Best	-3,32E+07	13860	0,73	2,28
Worst	-4,59E+07	7708	0,67	3,36
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	30,73%	-39,34%	1,04%	31,81%
<b>(Pseudo random list)</b>				
Initial solution	-4,69E+07	13398	0,71	3,42
Mean value	-3,46E+07	8433	0,70	2,46
St. deviation	-5,61%	14,86%	1,67%	7,88%
Best	-3,32E+07	13958	0,73	2,28
Worst	-4,41E+07	7708	0,67	3,34
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	29,16%	-38,39%	0,00%	30,95%

Table 35: Results on instance (060407): Simulated annealing, Initial Temperature 32000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,68E+07	13653	0,70	3,44
Mean value	-3,46E+07	8416	0,70	2,47
St. deviation	-5,74%	13,98%	1,58%	8,62%
Best	-3,32E+07	13831	0,73	2,28
Worst	-4,42E+07	7708	0,67	3,43
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	29,07%	-39,54%	1,69%	31,49%
<b>(Pseudo random list)</b>				
Initial solution	-4,68E+07	13397	0,71	3,41
Mean value	-3,46E+07	8427	0,70	2,47
St. deviation	-5,74%	14,53%	1,75%	8,22%
Best	-3,32E+07	13954	0,74	2,28
Worst	-4,42E+07	7708	0,67	3,39
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	28,98%	-38,39%	0,00%	30,74%

Table 36: Results on instance (060407): Simulated annealing, Initial Temperature 32000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,68E+07	13701	0,71	3,45
Mean value	-3,46E+07	8424	0,70	2,47
St. deviation	-5,82%	14,41%	1,75%	8,55%
Best	-3,32E+07	13981	0,73	2,28
Worst	-4,43E+07	7708	0,67	3,43
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	28,98%	-39,76%	-0,00%	31,73%
<b>(Pseudo random list)</b>				
Initial solution	-4,71E+07	13360	0,71	3,40
Mean value	-3,46E+07	8432	0,70	2,46
St. deviation	-5,65%	14,78%	1,66%	8,03%
Best	-3,32E+07	13982	0,73	2,28
Worst	-4,42E+07	7708	0,67	3,36
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	29,51%	-38,22%	0,00%	30,57%

Table 37: Results on instance (060407): Simulated annealing, Initial Temperature 65000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,72E+07	13358	0,70	3,48
Mean value	-3,46E+07	8424	0,70	2,46
St. deviation	-5,66%	14,37%	1,60%	8,07%
Best	-3,32E+07	13835	0,73	2,28
Worst	-4,41E+07	7708	0,67	3,36
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	29,60%	-38,21%	2,15%	32,26%
<b>(Pseudo random list)</b>				
Initial solution	-4,70E+07	13368	0,71	3,38
Mean value	-3,46E+07	8434	0,70	2,46
St. deviation	-5,69%	14,90%	1,63%	7,90%
Best	-3,32E+07	13934	0,73	2,28
Worst	-4,41E+07	7708	0,67	3,33
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	29,36%	-38,26%	0,00%	30,32%

Table 38: Results on instance (060407): Simulated annealing, Initial Temperature 65000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-3,37E+07	14137	0,73	2,86
Mean value	-3,40E+07	8442	0,70	2,44
St. deviation	-1,79%	15,27%	2,29%	4,00%
Best	-3,14E+07	14261	0,76	2,28
Worst	-3,53E+07	7708	0,67	2,88
Best solution obtained	-3,14E+07	14087	0,75	2,77
Improvement	6,75%	-0,35%	2,57%	3,04%
<b>(Pseudo random list)</b>				
Initial solution	-3,35E+07	14052	0,76	2,88
Mean value	-3,40E+07	8437	0,70	2,44
St. deviation	-1,81%	15,04%	2,80%	4,02%
Best	-3,14E+07	14077	0,78	2,28
Worst	-3,53E+07	7708	0,67	2,88
Best solution obtained	-3,14E+07	13977	0,78	2,81
Improvement	6,39%	-0,53%	2,53%	2,47%

Table 39: Results on instance (060414): Simulated annealing, Initial Temperature 32000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-3,32E+07	14012	0,74	2,90
Mean value	-3,40E+07	8442	0,70	2,44
St. deviation	-2,04%	15,27%	2,27%	4,07%
Best	-3,09E+07	14112	0,76	2,28
Worst	-3,53E+07	7708	0,67	2,86
Best solution obtained	-3,09E+07	14017	0,75	2,83
Improvement	6,93%	0,04%	1,46%	2,52%
<b>(Pseudo random list)</b>				
Initial solution	-3,36E+07	13926	0,77	2,93
Mean value	-3,40E+07	8436	0,70	2,44
St. deviation	-1,78%	14,99%	2,77%	3,97%
Best	-3,14E+07	14076	0,78	2,28
Worst	-3,53E+07	7708	0,67	2,86
Best solution obtained	-3,14E+07	13976	0,78	2,80
Improvement	6,56%	0,36%	1,08%	4,53%

Table 40: Results on instance (060414): Simulated annealing, Initial Temperature 32000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-3,34E+07	13952	0,75	2,92
Mean value	-3,39E+07	8439	0,70	2,44
St. deviation	-2,25%	15,14%	2,45%	3,70%
Best	-3,05E+07	14138	0,77	2,28
Worst	-3,53E+07	7708	0,67	2,82
Best solution obtained	-3,05E+07	14138	0,76	2,74
Improvement	8,86%	1,33%	1,30%	6,37%
<b>(Pseudo random list)</b>				
Initial solution	-3,35E+07	14051	0,76	2,88
Mean value	-3,40E+07	8440	0,70	2,44
St. deviation	-1,78%	15,18%	2,70%	4,16%
Best	-3,15E+07	14033	0,77	2,28
Worst	-3,53E+07	7708	0,67	2,90
Best solution obtained	-3,15E+07	14026	0,77	2,80
Improvement	6,05%	-0,18%	1,33%	2,78%

Table 41: Results on instance (060414): Simulated annealing, Initial Temperature 65000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-3,35E+07	14211	0,75	2,88
Mean value	-3,40E+07	8445	0,70	2,44
St. deviation	-1,68%	15,45%	2,42%	3,94%
Best	-3,17E+07	14262	0,76	2,28
Worst	-3,53E+07	7708	0,67	2,84
Best solution obtained	-3,17E+07	14112	0,76	2,77
Improvement	5,46%	-0,70%	0,81%	3,91%
<b>(Pseudo random list)</b>				
Initial solution	-3,35E+07	13827	0,77	2,91
Mean value	-3,40E+07	8442	0,70	2,44
St. deviation	-1,76%	15,29%	2,76%	3,93%
Best	-3,16E+07	14077	0,77	2,28
Worst	-3,53E+07	7708	0,67	2,83
Best solution obtained	-3,16E+07	14003	0,77	2,80
Improvement	5,72%	1,27%	0,00%	3,93%

Table 42: Results on instance (060414): Simulated annealing, Initial Temperature 65000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-5,19E+07	14010	0,69	3,07
Mean value	-3,49E+07	8458	0,70	2,45
St. deviation	-8,47%	16,07%	1,41%	5,03%
Best	-3,32E+07	14531	0,72	2,28
Worst	-4,93E+07	7708	0,67	3,05
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	35,98%	-41,08%	3,44%	23,15%
<b>(Pseudo random list)</b>				
Initial solution	-5,52E+07	13918	0,70	3,01
Mean value	-3,50E+07	8454	0,70	2,45
St. deviation	-10,45%	15,87%	1,47%	4,88%
Best	-3,32E+07	14433	0,72	2,28
Worst	-5,28E+07	7708	0,67	2,98
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	39,79%	-40,70%	1,39%	21,69%

Table 43: Results on instance (060421): Simulated annealing, Initial Temperature 32000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-5,01E+07	13689	0,69	3,03
Mean value	-3,48E+07	8456	0,70	2,45
St. deviation	-7,44%	16,00%	1,52%	5,12%
Best	-3,32E+07	14495	0,72	2,28
Worst	-4,75E+07	7708	0,67	3,00
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	33,70%	-39,70%	3,54%	22,20%
<b>(Pseudo random list)</b>				
Initial solution	-5,51E+07	13919	0,70	3,00
Mean value	-3,50E+07	8456	0,70	2,45
St. deviation	-10,43%	15,96%	1,48%	4,90%
Best	-3,32E+07	14408	0,72	2,28
Worst	-5,27E+07	7708	0,67	2,98
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	39,69%	-40,70%	1,39%	21,47%

Table 44: Results on instance (060421): Simulated annealing, Initial Temperature 32000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-5,28E+07	14120	0,71	3,06
Mean value	-3,49E+07	8456	0,70	2,44
St. deviation	-8,99%	15,97%	1,45%	4,70%
Best	-3,32E+07	14533	0,72	2,28
Worst	-5,02E+07	7708	0,67	2,97
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	37,14%	-41,54%	0,00%	22,98%
<b>(Pseudo random list)</b>				
Initial solution	-5,52E+07	13919	0,70	3,00
Mean value	-3,50E+07	8456	0,70	2,45
St. deviation	-10,46%	15,98%	1,46%	4,83%
Best	-3,32E+07	14433	0,72	2,28
Worst	-5,28E+07	7708	0,67	2,96
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	39,81%	-40,70%	1,39%	21,50%

Table 45: Results on instance (060421): Simulated annealing, Initial Temperature 65000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-5,23E+07	14308	0,70	2,97
Mean value	-3,49E+07	8454	0,70	2,45
St. deviation	-8,86%	15,86%	1,44%	4,85%
Best	-3,32E+07	14595	0,72	2,28
Worst	-4,99E+07	7708	0,67	2,96
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	36,49%	-42,31%	2,21%	20,49%
<b>(Pseudo random list)</b>				
Initial solution	-5,51E+07	13968	0,70	3,01
Mean value	-3,50E+07	8453	0,70	2,44
St. deviation	-10,48%	15,82%	1,42%	4,76%
Best	-3,32E+07	14307	0,72	2,28
Worst	-5,27E+07	7708	0,67	2,94
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	39,74%	-40,91%	1,06%	21,55%

Table 46: Results on instance (060421): Simulated annealing, Initial Temperature 65000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,59E+07	15479	0,71	3,54
Mean value	-3,46E+07	8511	0,70	2,47
St. deviation	-5,38%	18,70%	1,50%	9,27%
Best	-3,32E+07	15623	0,72	2,28
Worst	-4,36E+07	7708	0,67	3,50
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	27,66%	-46,68%	0,00%	33,33%
<b>(Pseudo random list)</b>				
Initial solution	-4,78E+07	15573	0,70	3,53
Mean value	-3,47E+07	8523	0,70	2,47
St. deviation	-6,45%	19,28%	1,49%	9,52%
Best	-3,32E+07	15766	0,72	2,28
Worst	-4,56E+07	7708	0,67	3,53
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	30,59%	-47,00%	1,77%	33,18%

Table 47: Results on instance (060428): Simulated annealing, Initial Temperature 32000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,61E+07	15180	0,71	3,56
Mean value	-3,46E+07	8521	0,70	2,47
St. deviation	-5,64%	19,16%	1,60%	9,07%
Best	-3,32E+07	15679	0,73	2,28
Worst	-4,41E+07	7708	0,67	3,48
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	27,91%	-45,63%	-0,00%	33,74%
<b>(Pseudo random list)</b>				
Initial solution	-4,78E+07	15592	0,70	3,53
Mean value	-3,47E+07	8522	0,70	2,47
St. deviation	-6,45%	19,24%	1,50%	9,47%
Best	-3,32E+07	15817	0,73	2,28
Worst	-4,56E+07	7708	0,67	3,53
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	30,58%	-47,06%	1,77%	33,21%

Table 48: Results on instance (060428): Simulated annealing, Initial Temperature 32000, step 50

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,54E+07	15258	0,71	3,51
Mean value	-3,46E+07	8499	0,70	2,47
St. deviation	-5,38%	18,11%	1,78%	9,07%
Best	-3,32E+07	15398	0,74	2,28
Worst	-4,36E+07	7708	0,67	3,51
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	26,92%	-45,90%	0,00%	32,78%
<b>(Pseudo random list)</b>				
Initial solution	-4,78E+07	15592	0,70	3,53
Mean value	-3,47E+07	8522	0,70	2,47
St. deviation	-6,45%	19,22%	1,49%	9,55%
Best	-3,32E+07	15810	0,73	2,28
Worst	-4,58E+07	7708	0,67	3,52
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	30,59%	-47,06%	1,77%	33,27%

Table 49: Results on instance (060428): Simulated annealing, Initial Temperature 65000, step 25

	Objective	Moves	Batching coeff.	X-factor
<b>(Random list)</b>				
Initial solution	-4,60E+07	15454	0,72	3,54
Mean value	-3,46E+07	8521	0,70	2,47
St. deviation	-5,40%	19,17%	1,72%	9,23%
Best	-3,32E+07	15653	0,73	2,28
Worst	-4,36E+07	7708	0,67	3,51
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	27,77%	-46,59%	-0,40%	33,37%
<b>(Pseudo random list)</b>				
Initial solution	-4,78E+07	15592	0,70	3,53
Mean value	-3,47E+07	8525	0,70	2,47
St. deviation	-6,62%	19,36%	1,45%	9,74%
Best	-3,32E+07	15717	0,72	2,28
Worst	-4,56E+07	7708	0,67	3,52
Best solution obtained	-3,32E+07	8254	0,71	2,36
Improvement	30,56%	-47,06%	1,77%	33,17%

Table 50: Results on instance (060428): Simulated annealing, Initial Temperature 65000, step 50