Trabajo de grado en modalidad de aplicación

[183017] Designing a tabu search algorithm for the distributed permutation flow shop scheduling + capacitated vehicle routing integrated problem to minimize makespan and tardiness.

Carlos Andrés Giraldo D'Achiardi[a,c] , Juan Luis Pardo Reyes[a,c] , Héctor Adolfo Granados Vera[a,c],

Eliana María González Neira[b, c]

[a]*Industrial Engineering Students*
[b]*Teacher, Thesis Director, Industrial Engineering Department*
[c]*Pontificia Universidad Javeriana, Bogotá, Colombia*

Summary/Abstract

The present paper exposes a way to solve two individual problems, the classic Distributed Permutation Flowshop Scheduling Problem and capacitated vehicle routing problem as a new combined NP- Hard Problem. To simulate a more realistic environment a multi-objective optimization to minimize makespan and total tardiness was proposed. In order to solve the joint problem a Tabu Search metaheuristic algorithm is proposed and evaluated by comparing its performance to (Naderi & Ruiz, 2010) DPFSP benchmark and (Augerat, 1995) CVRP benchmark as individual problems. The joint DPFSP+CVRP problem was also evaluated comparing the TS solutions to a mixed integer linear model proposed as a benchmark for the joint problem. The results obtained demonstrate that the model produces solutions close to the optimal values reported in the literature for individual problems (10% or less deviation for DPFSP and 20% or less deviation for CVRP). Additionally, the proposed TS showed better solutions using less computational time than the MILP model, highlighting that the best results are obtained when using NEH as the starting solution for each instance.

---

1. Problem Statement

Production and distribution are two key functions of the Supply Chain (SC). Nowadays, industries aim at synchronizing the previously mentioned activities for various reasons such as having greater control, improving performance, and decreasing costs by generating a vertical type of integration in the SC. That is why, since the study carried out by (Potts, 1980), researchers have been intrigued by the study of the integrated production and distribution problem. In fact, multiple studies of this problem have been performed, such as it is compiled in the literature review presented by (Chen, 2010). As seen in this state of the art, the most studied problems in this field are: the *Single machine* and *Vehicle Routing Problem* (VRP), and amongst them the multiple client with a *batch delivery* problem (Chen & Vairaktarakis, 2005) or the *single machine + batch delivery* for a single client (Pundoor & Chen, 2005).

After (Chen, 2010) compilation there have been more and more complex problems researched. The most common are the ones including *Permutation Flow Shop Problem* (PFSP) joint with VRP environments that minimize tardiness (Mohammadi et al., 2018; Ta et al., 2016; Wang et al., 2017). To fulfill such objective (Ta et al., 2016) solve the problem through a *Genetic algorithm* (GA); (Mohammadi et al., 2018) implement a Tabu search (TS) and (Wang et al., 2017) employ a *variable neighborhood search* (VNS). In addition, (Chen, Yang, & Guo, 2015) studied a *Parallel Flow Shop* joint with VRP problem that minimizes the weighted costs in the SC by using a naïve heuristic that overcomes a GA. However, it is necessary to analyze more complex problems that consider real-life characteristics related with production plants configurations and the transport of the finished products.

On one hand, when it comes to production scheduling problems, one of the environments that are present in some industries is the Distributed *Permutation Flow Shop Problem* (DPFSP) that consists on a company that owns several identical factories that follow a *Permutation Flow Shop Problem* (PFSP) scheme. These types of environments are gaining popularity due to globalization, small locations for factories and to facilitate client distribution that force the enterprises to have more than one factory. More specifically, the DPFSP was investigated for the first time by (Naderi & Ruiz, 2010), and it is classified as an NP-Hard problem whose characteristics are:

- *F* number of identical factories that contain the same *m* number of machines.
- *n* number of Jobs that will be processed in one of the *F* Factories. All Jobs must be processed, each one in only one factory.
- Machines configuration at the inside of each fabric corresponds to a PFSP. PFSP consists on a series of m machines in which the jobs pass from one machine to another always in the same order. That is, all jobs are processed first on machine 1, then on machine 2 and so on until machine m. PFSP looks forward to determining the sequence of jobs to be processed, having a total of n_f! Possible solutions (where n_f it's the number of jobs processed in factory *f*)
- When a job has started its processing in one factory it cannot be changed to another one.
- Each machine can only process one job, and a job can be processed only by one machine at the same time
- Machines are always available, there is not maintenance or eventualities.

Amongst the research made for DPFSP environments (Gao, Chen, & Deng, 2013) minimized makespan using a TS. Additionally, (Li et al., 2016) considered different transport timetables and loading capacities for each factory to minimize makespan though a simulated annealing (SA) algorithm.

On the other hand, the transportation phase of the jobs is considered. In the case of DPFSP each factory could have one limited capacity vehicle to deliver the jobs to customers, which is close to real cases. In literature, the limited capacity vehicle routing problem is called *Capacitated Vehicle Routing Problem* (CVRP). CVRP was first studied by (Dantzig & Ramser, 1959).

Some CVRP studies have looked towards minimizing transport costs. It is the case of (Zhu et al., 2012) where the solution approach is a TS which outperforms the best algorithms proposed until that date in 20 of the 27 instances tested. Another recent case of study analyzes the same objective function diverging in the use of due dates restrictions (Cassettari et al., 2018). This research is applied to a real case study of a natural gas distribution vehicles network showing the effectiveness of the *Saving Algorithm + 2-Opt m*ethod.

Considering that the DPFSP+CVRP integrated problem has never been studied, and that the great majority of works in scheduling + distribution have considered single objective functions, this project aims to solve the DPFSP+CVRP integrated problem with one limited capacity vehicle at each factory. It is intended to find the Pareto Frontier of *Makespan and Tardiness*. Makespan will allow best utilization of resources and minimizing tardiness will increase client satisfaction by meeting a higher number of due dates.

Considering that both problems (DPFSP and CVRP) have been proved to be NP-hard, the use of a metaheuristic approach is necessary. The TS, well-known metaheuristic, was selected to solve the DPFSP+CVRP problem. TS, proposed by (Glover, 1989), is a metaheuristic that avoids repeating solution combinations through the use of taboo lists, which save results already explored. It has also been found that the TS metaheuristic has generated better solutions than the ones generated by a GA in a DPFSP problem (Gao et al., 2013). Going further, TS has also been used for solving distribution problems such as CVRP (Zhu et al., 2012) and as it has been mentioned before, it overcome some other algorithms. TS stand for its simplicity, adaptability, speed and sturdiness.

This work answers the following research question: How to design and implement the TS metaheuristic to solve the DPFSP problem along with the CVRP distribution problem for each factory that minimizes the makespan and tardiness?

2. Background

Considering that the problem to be addressed is the DPFSP+CVRP and that it will be solved with a TS metaheuristic, the following section its divided in four parts: i) literature review related with the DPFSP scheduling problem, ii) literature review related with CVRP, iii) literature review of works that integrated scheduling and transport problems, and, iv) studies related with TS implementation in scheduling or transport problems.

2.1. Problems related to DPFSP

DPFSP studies started in the current decade. The firsts to study this problem were (Naderi & Ruiz, 2010), where the problem and its implications were originally explained. The objective of this research was to minimize the makespan, goal searched by the proposal of 6 different mixed integer programming models and 14 dispatching rule-based heuristics.

Later, there was some research solving the same problem by the combination of a GA and a local search (Gao & Chen, 2011). Two years after (Lin et al., 2013) tried to simplify the solution by the implementation of an *Iterated Greedy Algorithm* (IGA) metaheuristic and at the same time (Gao et al., 2013) used a TS metaheuristic for its resolution.

Additionally, (Li et al., 2016) tried solving a DPFSP with different timetables and limited capacity for the transport of raw material to each factory, minimizing makespan. Finally (Bargaoui et al., 2017) propose a *novel chemical reaction optimization* metaheuristic based in NEH heuristic that also minimizes makespan, presenting better results in comparison with other studies. As future opportunities, these authors highlighted the importance of studying multiple objectives for making the problem closer to reality.

2.2. CVRP related Literature

In the CVRP related literature review was found that the most common objective function was to minimize transport costs, even though there were found several heuristics for its solution.

As shown in (Soto et al., 2017) and (Zhu et al., 2012) the TS improved several heuristics such as *Multiple neighborhood search* (MNS) and *Deepest-Bottom Left Fill heuristic* (DBLF). Additionally, (Iswari & Asih, 2018a) brought forward the minimization of distance objective function and compare a GA with a *Particle Swarm Optimization* (PSO). Results showed GA performed better than PSO. On the other hand, (Pinto et al., 2018) proposed a *Column generation algorithm* to make the problem size smaller and therefore easier to solve (computationally). (Yang & Ke, 2018) applied a *FireWorks discrete* algorithm to solve this problem and due to the new parametrization, some competitive solutions against other *Swarm* algorithms were found.

Table 1 presents a collection of problems involving CRVP, the objective function and its solution methods.

Table 1. Review of CVRP studies

	Transport (deliveries)	**Objective Function**		**Solution Method**							
Author/s	**CVRP**	**Distances**	**Costs**	**Column generation**	**FireWorks algorithm**	**Saving Algorithm**	**DBLF**	**MNS**	**GA**	**PSO**	**TS**
(Zhu, Qin, Lim, & Wang, 2012)	X		X				X				X
(Soto et al., 2017)	X		X					X			X
(Iswari & Asih, 2018b)	X	X							X	X	
(Cassettari et al., 2018)	X		X			X					
(Yang & Ke, 2018)	X		X		X						
(Pinto et al., 2018)	X		X	X							

2.3. Scheduling-Transport related problems

A systematic literature review was performed for papers indexed in SCOPUS database, searching for the scheduling + transport integrated problems by using the following exclusion and inclusion criteria:

•Inclusion Criteria : Title–abstract–keywords (“single machine” OR “parallel machines” OR flowshop OR "flow shop" OR jobshop OR “job shop” OR openshop OR “open shop” OR scheduling) AND Title–abstract–keywords (VRP OR TSP OR transport OR delivery OR deliveries)

•Exclusion criteria: from the articles previously found, the ones with a transport problem different than delivering products to clients after being produced, such as transport between machines were discarded. Additionally, the problems that didn’t have scheduling were also excluded.

Amongst the review, a PFSP joint to a TSP problem was found were there was only one vehicle with infinite capacity available (Ta et al., 2016). For its solution different methods were applied, such as: a GA, a TS and finally a combined metaheuristic that uses the previously mentioned ones; with the goal of minimizing total Tardiness.

It was also found a single machine + VRP combined problem with a limited number of vehicles available (Zou et al., 2017a). The authors used a GA and, due to comparison reasons, a 2-part algorithm that solves both problems simultaneously, its objective was to minimize the *makespan.*

Table 2 presents the works found in this topic. As it can be seen, the quantity of papers that studied the integrated scheduling and transportation problem is few in comparison to the *scheduling* and transport problems reviewed individually. This shows the importance of studying this type of joint problems for the benefit of the entire SC.

Table 2. Scheduling-Transport review

	Topics (Scheduling)								**(trans port)**		**Objective Function**			
Author/s	**Flow Shop**				**Job Shop**	**Open shop**	**Single Machine**	**Parallel Machine**	**TSP**	**VRP**	**Makespan**	**Quality**	**Weighted Costs**	**Tardiness**
	Flo w Sho p	DPFSP	Permutation Flow Shop	Parallel Flow Shop										
(Pundoor & Chen, 2005)							X			X			X	X
(Z.-L. Chen & Vairaktarakis, 2005)							X			X				X
(Z.-L.Chen, 2010)	X				X		X	X	X	X			X	
(L. Chen et al., 2015)				X						X			X	
(Ta et al., 2016)			X							X				X
(Zou et al., 2017b)							X			X	X			
(Wang et al., 2017)			X							X				X
(Mohammadi et al., 2018)			X							X				X

2.4. TS applied to Scheduling or Transport Problems

TS metaheuristic looks for improving the performance of an initial solution obtained in this project by a Greedy type heuristic. The improvement of the initial solution is reached by variating the initial solution through defined movements and avoiding repetition using a tabu list that registers past results (Gupta et al., 1999).

The TS metaheuristic contains the following elements: an initial solution; a search area or neighborhood; the tabu list, that prevents being stuck in a local solution; an aspiration criteria, which allows movements that found an immediate base solution in order to find a long term better one and; a stopping criteria, which ends the TS algorithm.

Recent research has used TS as a solution method for *scheduling* or transport (*deliveries*) problems. Past year, a VRP with order divided deliveries research was done (Xia et al., 2018). Authors showed that its proposed TS presents a good efficiency level.

Only one work was found in DPFSP that uses the TS metaheuristic as solution approach. (Gao et al., 2013) implemented the TS to solve the minimization of makespan. Authors showed that their proposed TS outperforms a GA.

Another TS application, like the scheduling + transport combination, is the PFSP + VRP (Ta et al., 2016). The objective was to minimize total tardiness of a chemotherapy production center. Using computational experiments comparison, the TS improved the solution given by the initial *Greedy Algorithm*.

Table 3 presents the review of the characteristics concerned with TS implementation for solving *scheduling*, transport or a combination of such problems.

zzz

Table 3. Works in scheduling or transport that have implemented TS

	Topics (Scheduling)					**(transport)**		**Objective Function**		
Author/s	**Flow Shop**			**Job Shop**	**Open shop**	**TSP**	**VRP**	**Makespan**	**Weighted Costs**	**Tardiness**
	Flow Shop	DPFSP	Permutation Flow Shop							
(Gupta et al., 1999)	X							X		
(Eren, 2007)	X							X		X
(Ismail et al., 2008)						X				
(Gao et al., 2013)		X						X		
(Ta et al., 2016)			X				X			X
(Xia et al., 2018)							X		X	

3. Objectives

General Objective

Designing a TS algorithm to solve the DPFSP and CVRP integrated problem with one vehicle per factory.

Specific Objectives

1. Propose the DPFSP+CVRP mathematic model
2. Design and implement a TS metaheuristic for solving the DPFSP+CVRP problem
3. Evaluate the proposed metaheuristic for DPSFP literature review instances.
4. Evaluate the proposed metaheuristic for CVRP literature review instances.
5. Evaluate the proposed metaheuristic performance for small instances of the integrated DPFSP+CVRP problem in comparison with mixed integer linear programming model.

4. Mixed integer linear programming model (MILP) of the DPFSP+CVRP

In this section, a MILP model is proposed for the solution of the DPFSP+CVRP for minimizing tardiness and makespan.

Set	Description
N	Jobs
M	Machines
F	Factories
C	Clients
R	Routes

Parameters:

$D_j : Due\ date\ of\ work\ j \in J$

$Time_{il} : Time\ of\ the\ route\ between\ client\ i \in C\ and\ cliente\ l \in C$

$$\boldsymbol{B_{jl}} \begin{cases} \mathbf{1}\ \boldsymbol{if\ the\ client\ l \in C\ owns\ job\ j \in J} \\ \mathbf{0}\ \boldsymbol{In\ other\ cases} \end{cases}$$

$TFC_{fl}: Time\ of\ the\ travel\ between\ factory\ f\ \in F\ and\ client\ l\ \in\ C$

$TCF_{fl}: Time\ of\ the\ travel\ between\ client\ c\ \in C\ and\ factory\ l\ \in F$

$H_f : Volume\ capacity\ of\ the\ vehicle\ assigned\ to\ factory\ f \in F$

$V_j: Volume\ of\ job\ j \in J$

$P_{jm} : Processing\ time\ of\ job\ j \in J\ on\ machine\ m \in M$

$GM: Big\ M, a\ sufficiently\ large\ positive\ number$

Variables:

$$X_{jkf}\begin{cases}\mathbf{1\ if\ job\ j \in J\ its\ processed\ before\ k \in J\ on\ factory\ f \in F} \\ \mathbf{0\ In\ other\ cases}\end{cases}$$

$$W_{jkrf}\begin{cases}\mathbf{1\ if\ job\ j \in J\ its\ delivered\ before\ k \in J\ on\ route\ r \in R\ dispatched\ from\ factory\ f \in F} \\ \mathbf{0\ In\ other\ cases}\end{cases}$$

C_{jmf} : $Completion\ time\ of\ job\ j \in J\ on\ machine\ m \in J\ of\ factory\ f \in F$
S_{jmf} : $Start\ time\ of\ job\ j \in J\ on\ machine\ m \in J\ of\ factory\ f \in F$
T_j: $Job\ j \in J\ tardiness$
$Cmax$: $Makespan$
TE_j : $delivery\ time\ of\ Job\ j \in J\ to\ client$
SR_{rf}: $Starting\ time\ of\ route\ r \in R$ $\mathbf{dispatched\ from}$ $factory\ f \in F$
CR_{rf}: $completion\ time\ of\ route\ r \in R$ $\mathbf{dispatched\ from}$ $factory\ f \in F$

Objective Function:

$$\min Z1: \sum_{j \in J} T_j \quad (1)$$

$$\min Z2: C_{max} \quad (2)$$

Subject to:

$$\sum_{k \in J, j \neq k} \sum_{f \in F} X_{jkf} = 1 \quad \forall\ j \in J, j \neq 0 \quad (3)$$

$$\sum_{j \in J, j \neq k} \sum_{f \in F} X_{jkf} = 1 \quad \forall\ k \in J, k \neq 0 \quad (4)$$

$$\sum_{k \in J} X_{0kf} = 1 \quad \forall\ f \in F \quad (5)$$

$$\sum_{j \in J} X_{j0f} = 1 \quad \forall\ f \in F \quad (6)$$

$$\sum_{j \in J} X_{jkf} = \sum_{j \in J} X_{kjf} \quad \forall\ f \in F, \forall\ k \in J, k \neq 0 \quad (7)$$

$$C_{jmf} = S_{jmf} + P_{jm} \quad \forall\ j \in J, \forall\ m \in M, \forall\ f \in F, j \neq 0 \quad (8)$$

$$S_{kmf} \geq C_{jmf} - GM(1 - X_{jkf}) \quad \forall\ j, k \in J, \forall\ m \in M, \forall\ f \in F, j \neq k \quad (9)$$

$$S_{jmf} \geq C_{j(m-1)f} \quad \forall\ j \in J, \forall\ m \in M, \forall\ f \in F, m > 1 \quad (10)$$

$$SR_{rf} \geq C_{j|M|f} - M\left(1 - \sum_{k \in J} W_{jkrf}\right) \quad \forall\ j \in J, \forall\ r \in R, \forall\ f \in F, j \neq 0 \quad (11)$$

$$\sum_{k \in J} X_{jkf} = \sum_{k \in J} \sum_{r \in R} W_{jkrf} \quad \forall\ j \in J, \forall f \in F \quad (12)$$

$$\sum_{j\in J} X_{jkf} = \sum_{j\in J}\sum_{r\in R} W_{jkrf} \quad \forall\, k \in J, \forall\, f \in F \tag{13}$$

$$\sum_{k\in J} W_{0krf} = 1 \quad \forall\, r \in R, \forall\, f \in F \tag{14}$$

$$\sum_{j\in J} W_{j0rf} = 1 \quad \forall\, r \in R, \forall\, f \in F \tag{15}$$

$$\sum_{j\in J} W_{jkrf} = \sum_{j\in J} W_{kjrf} \quad \forall\, f \in F, \forall\, k \in J, \forall\, r \in R, k \neq 0 \tag{16}$$

$$TE_k \geq TE_j + \sum_{i\in C}\sum_{\substack{l\in C \\ \neq 0}} Time_{il} B_{ji} B_{kl} - GM(1 - W_{jkrf}) \quad \forall\, j,k \in J, \forall\, r \in R, \forall\, f \in F, j \neq k, k \tag{17}$$

$$TE_j \geq SR_{rf} + \sum_{i\in C} TFC_{fi} B_{ji} - GM(1 - W_{0jrf}) \quad \forall\, j \in J, \forall\, r \in R, \forall\, f \in F, j \neq 0 \tag{18}$$

$$SR_{rf} \geq CR_{(r-1)f} \,\forall\, r \in R, \forall\, f \in F, r > 1 \tag{19}$$

$$CR_{rf} \geq TE_j + \sum_{i\in C} TCF_{fi} B_{ji} - GM(1 - W_{j0rf}) \quad \forall\, j \in J, \forall\, r \in R, \forall\, f \in F \tag{20}$$

$$T_j \geq TE_j - d_j \quad \forall\, j \in J \tag{21}$$

$$Cmax \geq TE_j \quad \forall\, j \in J \tag{22}$$

$$\sum_{j\in J}\sum_{k\in J} \frac{V_j W_{jkrf}}{2} \leq H_f \quad \forall\, r \in R, \forall\, f \in F \tag{23}$$

$$W_{jjrf} = 0 \quad \forall\, j \in J, \forall\, r \in R, \forall\, f \in F, j \neq 0 \tag{24}$$

$$X_{jjf} = 0 \quad \forall\, j \in J, \forall\, f \in F, j \neq 0 \tag{25}$$

$$Cmax \geq C_{jmf} \quad \forall\, j \in J, \forall\, m \in M, \forall\, f \in F \tag{26}$$

$$C_{jmf} \geq 0 \quad \forall\, j \in J, \forall\, m \in M, \forall\, f \in F \tag{27}$$

$$T_j \geq 0 \quad \forall\, j \in J \tag{28}$$

$$S_{jmf} \geq 0 \quad \forall\, j \in J, \forall\, m \in M, \forall\, f \in F \tag{29}$$

$$TE_j \geq 0 \,\forall\, j \in J \tag{30}$$

$$SR_{rf} \geq 0 \,\forall\, r \in R, \forall\, f \in F \tag{31}$$

$$CR_{rf} \geq 0 \,\forall\, r \in R, \forall\, f \in F \tag{32}$$

The objective functions (1) and (2) are the tardiness minimization and makespan minimization respectively. The constraint set are explained below: Eq. (3) and (4) ensure that each job has only one position and is assigned to one factory. A dummy job 0 is included in the constraint sets (5)-(6) to indicate that the sequence of each factory starts and ends with job 0. Constraint set (7) guarantees that every job has a successor and predecessor in the factory to which it is assigned. Eq. (8) calculates the completion time of each job at each machine as its starting time plus the processing time on that machine. The constraint sets (9)-(10) guarantee that the start time of job j on a machine m is after the job finishes processing on the previous machine or the same machine m ends with the previous job. Constraint set (11) calculates the starting time of a route. Constraint sets (12)-(13)

ensure that that there is only one job after any other and one job before any other respectively (for the distribution and production processes respectively). Eq. (14)-(15) ensure that the route begin and end with the dummy job. Constraint set (16) guarantees that every job can have a successor and predecessor in the route delivery to which it is assigned. Eq. (17) determines that the time of delivery of job k is greater than or equal to the time of delivery of job j, plus time between clients who own job j and k, if job j it is assigned for delivering before job k. Constraint set (18) calculates the delivery time of the first job of each route. Due to the problem consider that there is only one vehicle per factory, the constraint set (19) ensures that a route does not start before the end of the previous one and constraint set (20) indicates that the completion time of route r, is the delivery time of the last job delivered on this route plus the time to return to the origin factory. Eq. (21) define the tardiness, (22) and (26) define the makespan. Constraint set (23) ensures not to exceed the capacity of the vehicle on route r. Constraint sets (24)-(25) assure that the same job is not assigned in 2 continuous places of the precedence. Finally, the constraint sets (27)-(32) represent the non-negativity of the variables.

5. Proposed TS

In this section we describe the proposed TS to find solutions for the DPFSP+CVRP problem (Figure 1). The following elements of the TS are explained in detail: initial solution, neighborhood structure, tabu list and stopping criteria.

5.1 Initial solution

Two dispatching rules were selected as initial solutions. The ENS2 proposed by (Kim et al., 1996) was selected for tardiness objective, and NEH (Nawaz et al., 1983) for makespan objective. The resultant sequences of the DPFSP after the application of these dispatching rules are the initial solution for the CVRP stage of the problem. It is important to note that the assignment to the factories is that the next job in the sequence is going to be processed in the first available factory. In addition, the vehicles are loaded with the first jobs that are already produced in their respective factory until the capacity of the vehicle is fulfilled.

At first, the initial DPFSP solution was modeled by the NEH dispatching rule (Figure 2) starts with ordering the jobs from longest to lowest total processing time. From that ordered list of jobs, the two first jobs are taken to find the best partial sequence of them in terms of makespan. Then, the third job of the initial list is taken and placed in the third possible positions of the partial sequence being allocated in the position that best makespan proportionate. The same procedure is performed with all remaining jobs of the first ordered list until the total sequence is completed.

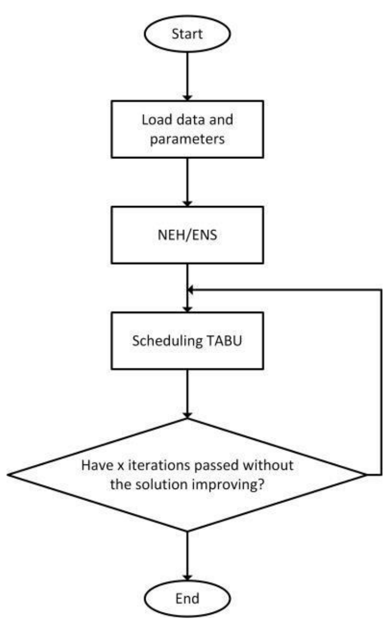

Figure 1. General scheme of TS

The second initial solution tested was modeled by an ENS-2 dispatching rule (Figure 3). It starts by using the EDD (Earliest Due Date) rule that form a list of the jobs in ascending order of the due dates. Then the insertion procedure used in NEH is performed but in this case evaluating not the makespan but the tardiness criterion. Once the insertion procedure is finished and a complete sequence is obtained, 2-optimal interchanges of jobs are done to improve the sequence according to the total tardiness.

After the process of obtaining a sequence of jobs with ENS2 or NEH is done, the allocation of jobs in the factories is completed by sequencing each job to the first available factory. This process is done in order to diminish the makespan of each factory and therefore the tardiness of the whole set. Once these steps are completed the initial solution of the problem has been developed and it is ready to enter the "Scheduling tabu" part.

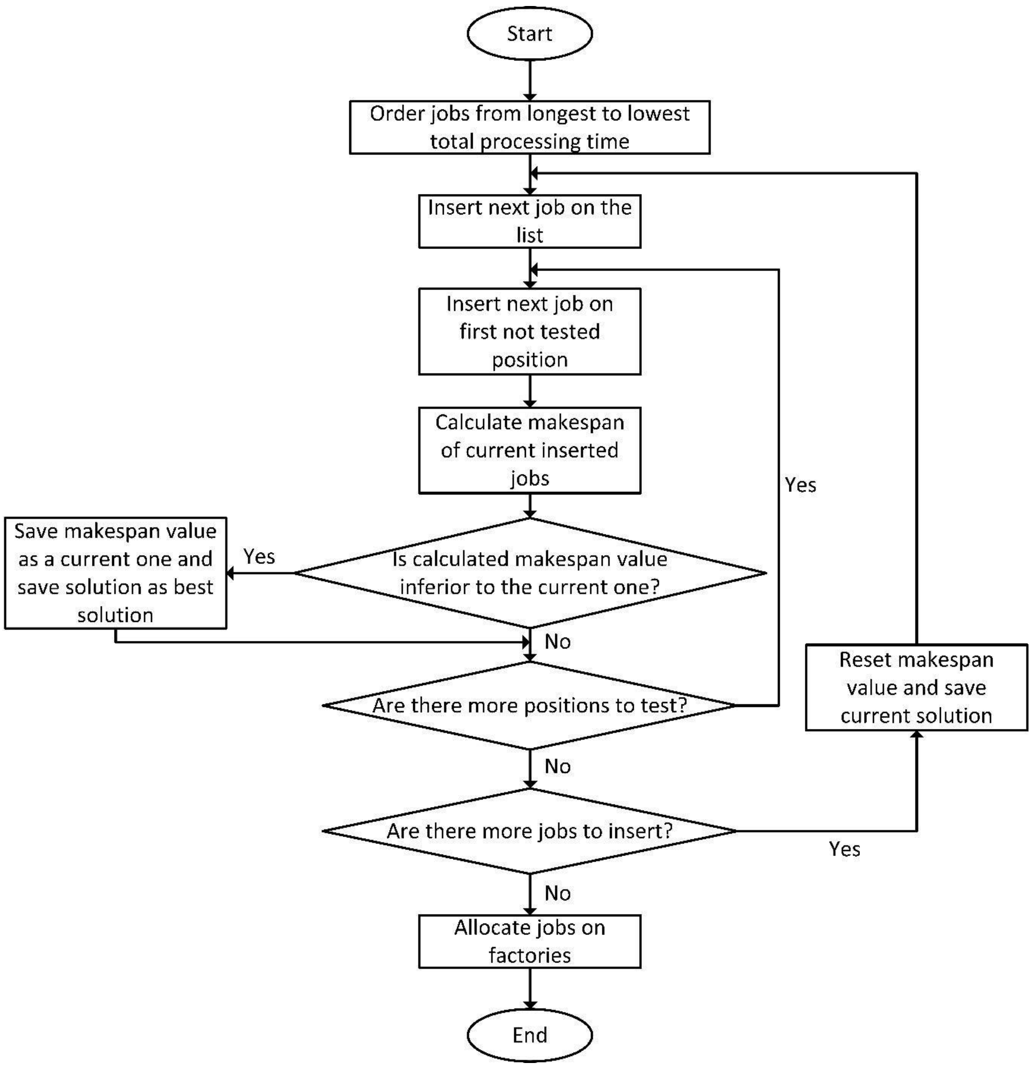

Figure 2. NEH initial solution procedure

5.2 Scheduling Tabu

The Scheduling Tabu consists of two phases. First, the "Production tabu" phase consists in the generation of different solutions by moving a vector that comes from the first part of the problem (DPFSP). After the "production tabu" it comes the "distribution tabu" phase that generates different solutions from the best "production tabu" solution by moving the routes generation for the CVRP piece of the problem. At last, both solutions are evaluated through the Pareto Archive Evolution Strategy (PAES) procedure (Knowles & Corne, 2000) to evaluate if the solutions should be added to the pareto frontier and update the frontier.

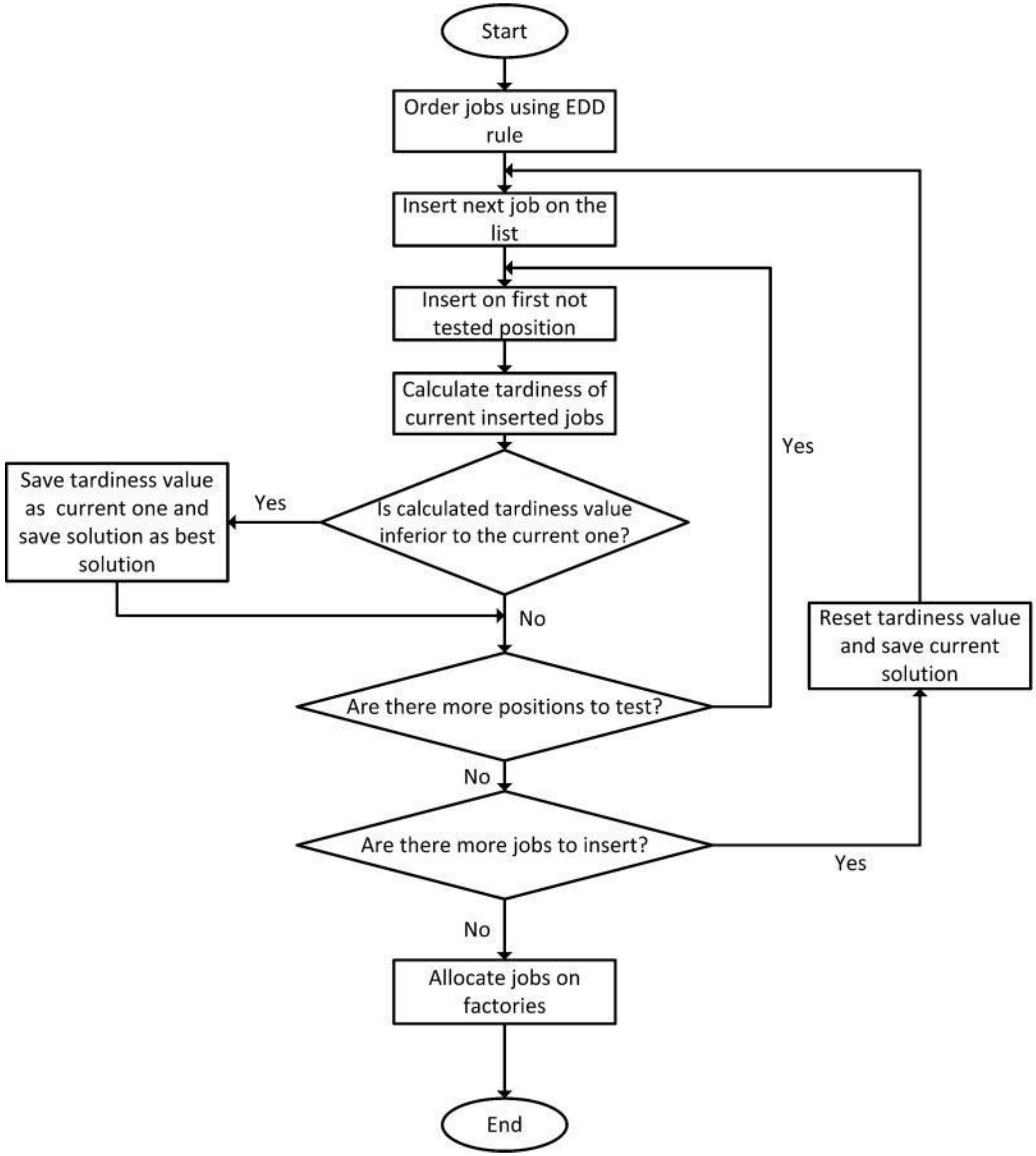

Figure 3. ENS-2 initial solution procedure

The "Production Tabu" flowchart is presented in Figure 4. It works as a standard TS where the neighborhood is defined by the scheduling order vector in which all allowed sequences are tested. A sequence is allowed if it comes from a movement that is not in the tabu list vector. The jobs which positions were interchanged are marked and makespan is calculated. If the makespan is improved with the movement of jobs, then the new makespan and new solution are saved, and another iteration of the TS begins. In the case that there is no improvement after the movement of jobs, the TS continues with the previous solution, but the counter of iterations without improvement increases. This counter serves as a stopping criterion of the TS. Once the stopping criteria is reached, the best solution found is subject to the allocation procedure in which jobs are assigned one by one to the first factory available. This final solution (with the jobs assigned to factories) is an input of the "distribution tabu" phase.

The “distribution tabu” phase (Figure 5) uses the final “production tabu” solution and after allocating it in the factories and generating routes, it moves de jobs inside each route, meaning changes between factories are forbidden. Each route means a unique neighborhood.

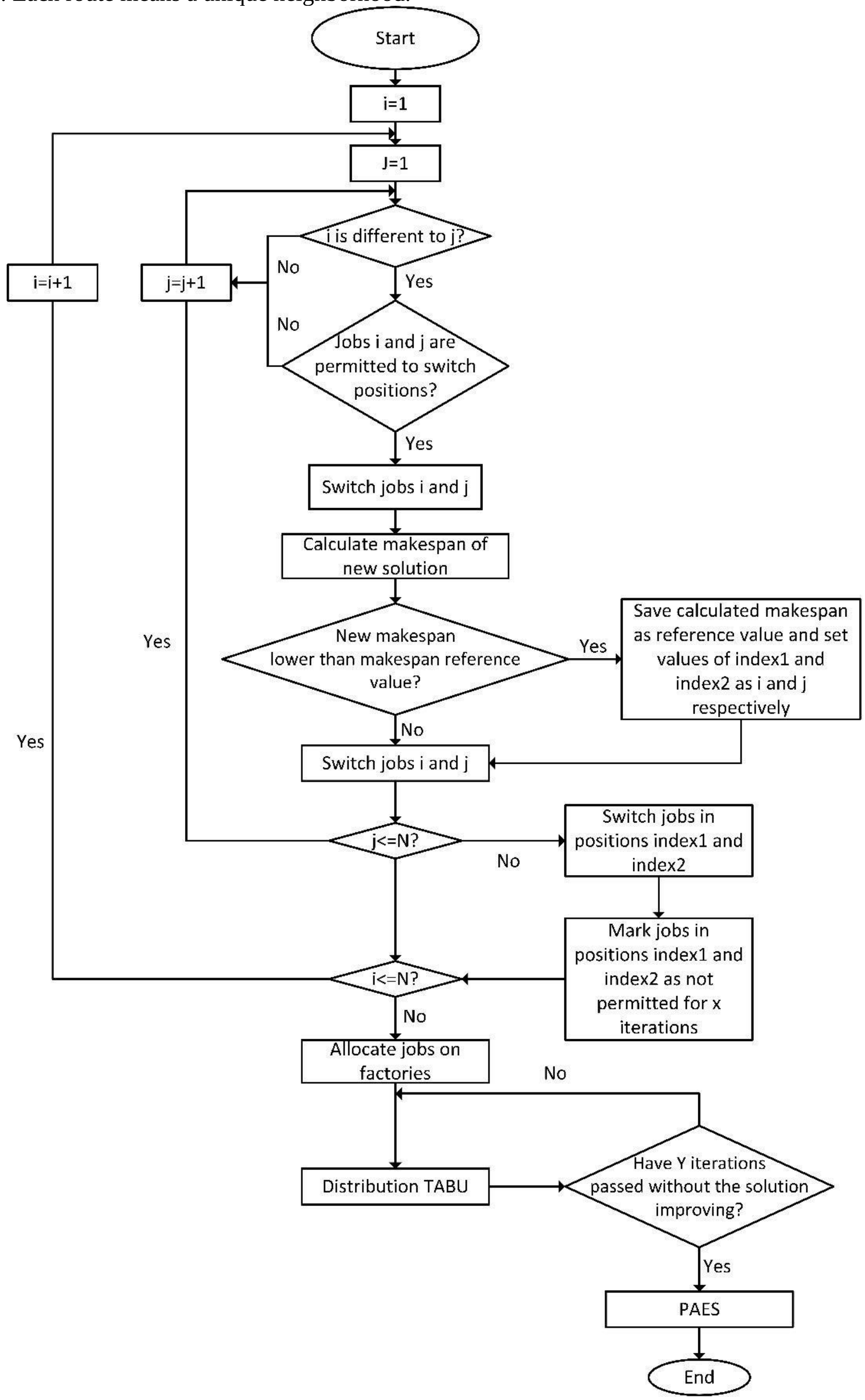

Figure 4. Scheduling Tabu flowchart

Like the "production tabu", the "distribution tabu" works as a standard TS with the difference of moving an R number of routes, and testing tardiness after it. A sequence is allowed if it comes from a movement that is not in the tabu list vector. If the tardiness improved with the movement of jobs, then the new makespan (calculated as the maximum completion time + the distribution time of the final job of all factories) and new solution are saved, and another iteration of the TS begins. If the solution does not improve, then the counter of iteration without improvement increases. This counter serves as a stopping criterion of the TS. Once the stopping criteria is reached. Both, the "production tabu" and "distribution tabu" best solutions are saved on the PAES.

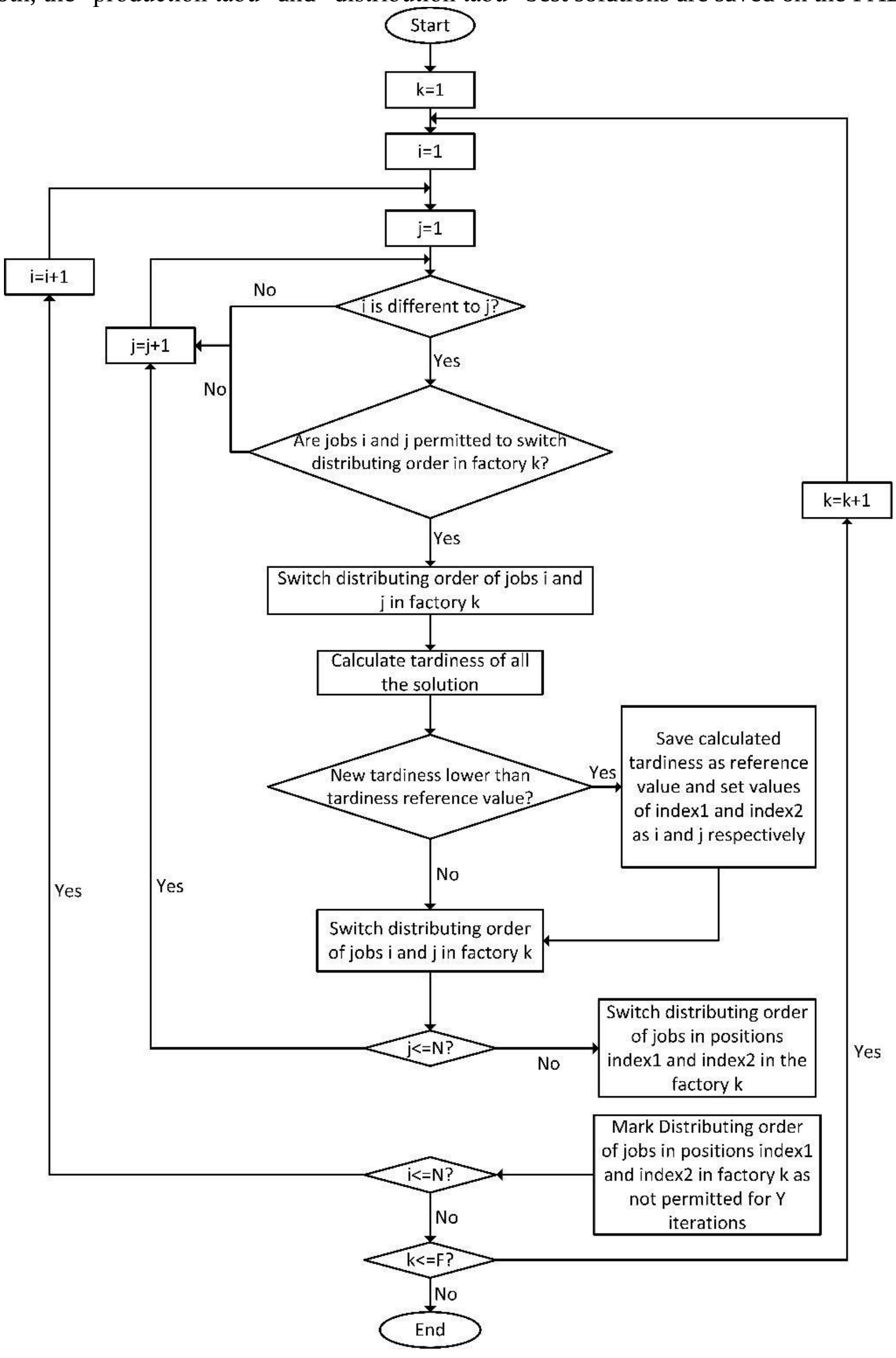

Figure 5. Distribution tabu flowchart

Finally, the last step of the TS consists in applying the PAES procedure which saves the pareto solutions, that are the non-dominated ones.

The main rules for saving one solution S to the Pareto Archive are:

- If the solution S dominates one or more solutions saved into the pareto Archive, S must be saved into the Pareto Archive and the dominated solution(s) must be deleted.
- If the solution S do not dominate and is not dominated by any solution saved on the Pareto Archive, the solution S must be saved into the Pareto Archive
- If the solution S is dominated by any of the solutions saved into the pareto Archive, the solution S is discarded

6. Parametrization

In our extensive review of the literature, there were not any instances that suited the type of problem described in this paper, therefore, joint problem instances had to be created. In order to create these instances, the following characteristics where considered:

- J= number of jobs {10, 30, 50}
- M= number of machines {5, 10, 15}
- F= number of factories {2, 4, 6}
- C= vehicle capacity {4, 8, 12}

Where the number of jobs (J) defines if the instance is small (10), medium (30) or large (50).

The combination of these elements leads to 81 different instance configurations. Additionally, four different instances were created for each one the configurations, based on two parameters for the generation of due dates:

- T= tardiness factor {0.2, 0.6}
- R= due date range {0.2, 0.4}

Therefore, four versions of due dates (V) for each of the 81 configurations:

- V1: (R = 0.2| T = 0.2)
- V2: (R = 0.2| T = 0.6)
- V3: (R = 0.4| T = 0.2)
- V4: (R = 0.4| T = 0.6)

Due dates for each instance were generated from a uniform distribution in the range [P(l - T- R/2), P(1- T + R/2)], where P is a lower bound of the makespan, as stated by (Potts and Van Wassenhove, 1982). Processing times and distances for each instance where generated following the procedures proposed by (Naderi & Ruiz, 2010) and (Augerat, 1995) for the scheduling and transportation stages respectively.

Based on all these combinations, a total of 324 instances were created for the joint problem (|J|*|M|*|F|*|C|*|V|=3*3*3*3*4=324). As an example of the nomenclature used: J10_M5_F4_C12_V1 breaks down to 10 jobs, 5 machines, 4 factories, capacity of vehicles equal to 12, tardiness factor of 0.2 and due date range of 0.2.

Finally, from the 324 instances that were created, 15 instances (5 small, 5 medium, 5 large) were selected at random to test different combination of parameters necessary to execute the proposed TS. The parameters and their values are described as follows:

- Initial Solution: NEH or ENS2
- Tabu Iterations without improvement: {75, 150, 225}
- Tabu List Prohibitions: {10, 20, 30}

Each of the 15 randomly selected instances had to be executed a total of 18 times to account for all the different combinations of parameters. Having the results, the MID (Mean Ideal Distance) of each instance was calculated using their respective non dominated pareto frontier solutions, as proposed by (Ebrahimi et al., 2014).

Using all the MID's from these 270 (15*18) sets of solutions a 4-way ANOVA with a blocking factor was calculated in order to determine the "best" possible combination for the TS parameters, factors of the ANOVA are:

Factor 1: Instance (Block factor)
Factor 2: Initial Solution
Factor 3: Tabu Iterations without improvement
Factor 4: Tabu List Prohibitions

The results are described as follows:

Tests of Between-Subjects Effects

Dependent Variable: MIDBook

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Model	2,071E+11[a]	32	6471285066	2604,783	,000
Instance	9,698E+10	14	6927405396	2788,378	,000
Iterations	16013586,55	2	8006793,275	3,223	,042
Prohibitions	1805619,999	2	902809,999	,363	,696
Initial_Solution	6852223,762	1	6852223,762	2,758	,098
Iterations * Prohibitions	564908,492	4	141227,123	,057	,994
Iterations * Initial_Solution	407052,685	2	203526,343	,082	,921
Prohibitions * Initial_Solution	12113841,06	2	6056920,532	2,438	,090
Iterations * Prohibitions * Initial_Solution	1114854,392	4	278713,598	,112	,978
Error	591283764,4	238	2484385,565		
Total	2,077E+11	270			

a. R Squared = ,997 (Adjusted R Squared = ,997)

Figure 6. Parametrization ANOVA

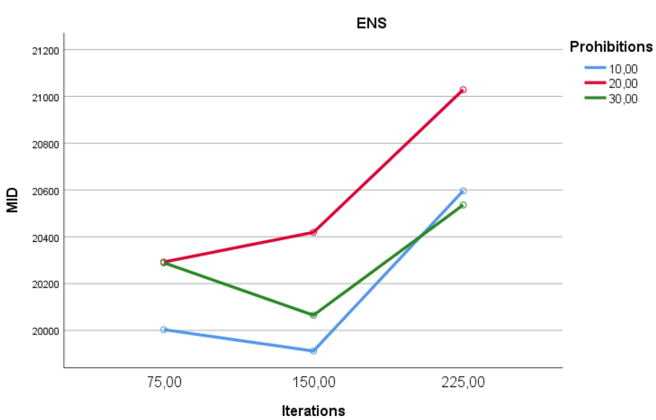

Figure 7. Marginal Average for ENS

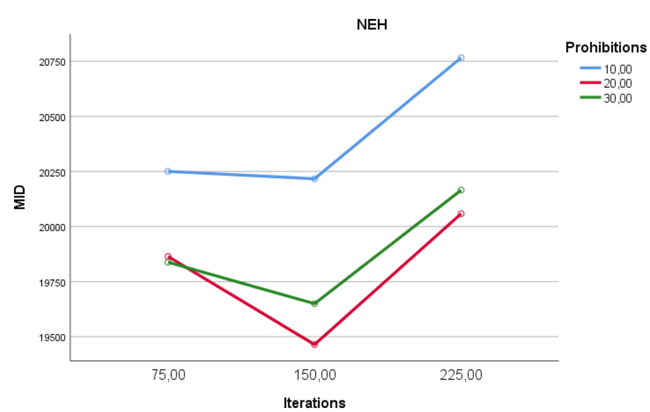

Figure 8. Marginal Average for NEH

Using a 95% confidence level, the "Iterations" factor proves to have a significant effect on the observations. By using LSD comparisons 225 was favored as the ideal value for minimizing the MID. Additionally, the effect the "Solution" factor as well as the effect of the combination of "Solution" & "Prohibitions" factors, while not significant at this confidence level, are close to have an effect in the resulting MID for any given instance, favoring NEH as the starting solution and 20 as the Tabu list Prohibitions.

Accounting the results of the ANOVA *Figure 6.* and the graphical comparisons obtained from *Figure 7* and *Figure 8* the combination of parameters chosen for this investigation were:

Starting solution: NEH
Iterations without Improvement: 225
Prohibitions in the Tabu list: 20

All the results presented in this document were based off this combination of parameters for the proposed TS. Additionally, a constraint that limits computational runtime of the metaheuristic model to 9000 seconds was added to make large instances of the joint problem more manageable.

7. Results

7.1. DPFSP

The proposed TS was executed using the DPFSP benchmark instances of (Naderi & Ruiz, 2010) and results were compared to the ones obtained by those authors. Small instances ranged from 2 to 4 factories (F), 4 to 16 jobs (n) and 2 to 5 machines (m). On the other hand, large instances ranged from 2 to 7 factories, 20 to 50 Jobs and 5 to 20 machines. All instances were completed within the already mentioned time limit.

Results for small instances show that the proposed TS reached the best solution found by (Naderi & Ruiz, 2010) in 25% of cases, and the average difference between the TS and benchmark results was 5%. Additionally, the Gap which is the percentual difference between our solution and the benchmark solution was calculated for each instance using the equation (33):

$$Gap = \left(\frac{TS\ result}{Best\ result} - 1\right) * 100 \qquad (33)$$

A confidence interval with a 95% confidence level was then calculated for the Gap showing that the proposed TS reaches solution ranging from 4% to 6% difference from the optimal benchmark results.

Table 4 presents the results for each executed small instance. The first three columns represent the instance size (factories, jobs and machines respectively). The fourth and fifth columns represent the results reported by (Naderi & Ruiz, 2010), and our TS results respectively. Finally, the last column represents the Gap calculated for each instance.

Table 4. Comparison of proposed TS with small benchmark instances for DPFSP

F	n	m	Best	TS_{Result}	Gap
2	4	2	120	120	0%
2	4	3	228	228	0%
2	4	4	274	274	0%
2	4	5	314	314	0%
2	6	2	174	175	1%
2	6	3	282	303	7%
2	6	4	320	333	4%
2	6	5	358	368	3%
2	8	2	201	208	3%
2	8	3	294	309	5%
2	8	4	364	403	11%
2	8	5	376	405	8%

F	n	m	Best	TS_{Result}	Gap
2	10	2	345	345	0%
2	10	3	322	347	8%
2	10	4	415	420	1%
2	10	5	439	454	3%
2	12	2	354	355	0%
2	12	3	391	410	5%
2	12	4	493	516	5%
2	12	5	492	522	6%
2	14	2	385	396	3%
2	14	3	427	442	4%
2	14	4	555	587	6%
2	14	5	480	509	6%

F	n	m	Best	TS_{Result}	Gap
2	16	2	494	510	3%
2	16	3	449	480	7%
2	16	4	585	652	11%
2	16	5	532	592	11%
3	4	2	163	163	0%
3	4	3	176	176	0%
3	4	4	291	291	0%
3	4	5	390	390	0%
3	6	2	161	175	9%
3	6	3	185	185	0%

F	n	m	Best	TS_{Result}	Gap
3	6	4	275	284	3%
3	6	5	413	413	0%
3	8	2	169	177	5%
3	8	3	196	196	0%
3	8	4	311	315	1%
3	8	5	320	357	12%
3	10	2	202	213	5%
3	10	3	264	279	6%
3	10	4	364	382	5%
3	10	5	352	356	1%
3	12	2	236	261	11%
3	12	3	276	300	9%
3	12	4	364	382	5%
3	12	5	414	448	8%
3	14	2	258	278	8%
3	14	3	276	312	13%
3	14	4	405	431	6%

F	n	m	Best	TS_{Result}	Gap
3	14	5	491	521	6%
3	16	2	250	273	9%
3	16	3	340	383	13%
3	16	4	421	454	8%
3	16	5	473	526	11%
4	4	2	164	164	0%
4	4	3	271	271	0%
4	4	4	281	281	0%
4	4	5	287	287	0%
4	6	2	178	178	0%
4	6	3	227	227	0%
4	6	4	258	258	0%
4	6	5	331	338	2%
4	8	2	161	161	0%
4	8	3	230	230	0%
4	8	4	294	301	2%
4	8	5	310	315	2%

F	n	m	Best	TS_{Result}	Gap
4	10	2	162	176	9%
4	10	3	219	232	6%
4	10	4	289	291	1%
4	10	5	379	386	2%
4	12	2	189	215	14%
4	12	3	237	257	8%
4	12	4	360	384	7%
4	12	5	372	389	5%
4	14	2	223	237	6%
4	14	3	291	320	10%
4	14	4	323	355	10%
4	14	5	432	467	8%
4	16	2	249	278	12%
4	16	3	294	332	13%
4	16	4	360	398	11%
4	16	5	365	409	12%

Large instances, on the other hand, show that the proposed TS reached the best solution found by (Naderi & Ruiz, 2010) in 3% of cases, and the average difference between our TS and benchmark results was 9%. A Gap was also calculated for each of these solutions and another interval with a 95% confidence level was calculated where proposed TS reaches solutions with a 7% to 11% difference from optimal benchmark results. Table 5. Comparison of proposed TS with Large benchmark instances for DPFSP phase presents the results for each executed large instance.

Table 5. Comparison of proposed TS with Large benchmark instances for DPFSP phase

F	n	m	Best	TS_{Result}	Gap
2	20	5	676	709	5%
2	20	10	959	1013	6%
2	20	20	1694	1771	5%
2	50	5	1508	1522	1%
2	50	10	1902	2070	9%
2	50	20	2537	2813	11%
3	20	5	598	621	4%
3	20	10	905	973	8%
3	20	20	1509	1611	7%
3	50	5	1016	1102	8%
3	50	10	1408	1566	11%
3	50	20	2079	2338	12%

F	n	m	Best	TS_{Result}	Gap
4	20	5	465	522	12%
4	20	10	789	855	8%
4	20	20	1321	1415	7%
4	50	5	792	928	17%
4	50	10	1182	1390	18%
4	50	20	1844	2082	13%
5	20	5	469	480	2%
5	20	10	747	792	6%
5	20	20	1348	1388	3%
5	50	5	680	823	21%
5	50	10	1048	1203	15%
5	50	20	1730	1888	9%

F	n	m	Best	TS_{Result}	Gap
6	20	5	436	436	0%
6	20	10	670	698	4%
6	20	20	1313	1354	3%
6	50	5	656	765	17%
6	50	10	967	1119	16%
6	50	20	1550	1751	13%
7	20	5	430	437	2%
7	20	10	674	724	7%
7	20	20	1237	1297	5%
7	50	5	584	709	21%
7	50	10	944	1064	13%
7	50	20	1465	1613	10%

7.2. CVRP

The metaheuristic model was also evaluated with the CVRP instances proposed by (Augerat, 1995). These instances were modified to fit the parameters of the proposed model, it was possible to make the previously mentioned modification since the evaluated variable is the total distance traveled (equivalent to the makespan of the TS when the vehicle travels one distance unit per each time unit). The TS once again was able to finish processing all instances using the already mentioned parameters within the time limit.

A Gap was calculated for each of the proposed instances and with it, a confidence interval with a 95% confidence level. Results show that the TS reaches solutions with a 16% to 25% difference from the optimal solution, also, the solutions given by the model have an average difference of 20% from the optimal solution of the instances evaluated.

Table 6. represents the results for each instance. The first two columns represent the number of nodes and number of trucks respectively as described in the literature, column 3 represents the vehicle capacity used for the TS, defined to make the number of routes of the TS solution to be equivalent to the number of trucks in the literature. Columns 4 to 6 represent the optimal solution for the instance, the result of the TS for the instance and the Gap between the optimal solution and the TS solution respectively, all these results were rounded to the nearest integer.

Table 6. Comparison of proposed TS with benchmark instances for CVRP

n	k	c	Best	TSResult	Gap
16	8	2	450	451	0%
19	2	9	212	284	34%
20	2	10	216	279	29%
21	2	10	211	240	14%
22	2	11	216	313	45%
22	8	2	603	702	16%
23	8	2	529	630	19%
40	5	8	458	588	28%
45	5	9	510	629	23%
50	7	7	554	674	22%
50	8	7	631	819	30%
50	10	5	696	801	15%
51	10	5	741	818	10%
55	7	8	568	751	32%
55	10	6	694	795	15%
55	15	3	989	1096	11%
60	10	6	744	822	10%
60	15	4	968	1019	5%
65	10	7	792	871	10%
70	10	7	827	942	14%
76	4	19	593	729	23%
76	5	15	627	743	19%
101	4	25	681	983	44%

7.3. DPFSP + CVRP

As stated previously, all 324 instances where evaluated on the metaheuristic model with the chosen parameters and within the 9000 second CPU time limit. Each of them was also evaluated with the MILP model described before, nevertheless, only small instances (J10) were able to show results and only when using the Objective Function to minimize makespan. The complete makespan is calculated until the final job of all factories is effectively delivered to the costumer. The calculation of makespan for the joint problem, that is, until the last delivery occurs allows to improve resources utilization in both stages, scheduling and transportation.

Small instances using the objective function to minimize tardiness could not be handled in GUSEK due to instability in the base matrix and exceeded the maximum competition time in NEOS servers. Medium and large instances (J30, J50) ran out of memory in both GUSEK and NEOS servers before being able to show any result. A relaxation of the binary variables in the MILP was also attempted by making them continuous between 0 and 1, but this test only led to makespan and tardiness values of 0 using any of the objective functions. These difficulties are to be expected from NP-Hard problems.

7.3.1. Small Instances Gap Comparison

Since only small instances executed with the makespan objective function were able to be evaluated both in the metaheuristic model and the MILP model, we can only compare the makespan Gaps of 108 instances of the 324 that were created.

Table 7 represents the results for each of the 108 instances described previously, the first column contains the name of the instance tested, the second column represents the optimal solution for the MILP when using the minimize makespan objective function, the third column represents the best makespan value reached by the TS among the non-dominated solutions for each instance, finally the fourth column represents the GAP between the MILP optimal makespan solution and the best TS non dominated makespan solution.

Table 7. Small instances comparison GAP

Instance	Best MK	TS MK	GAP
J10_M5_F4_C4_V1	330	344	4%
J10_M5_F4_C4_V2	450	478	6%
J10_M5_F4_C4_V3	383	398	4%
J10_M5_F4_C4_V4	372	377	1%
J10_M5_F4_C8_V1	399	404	1%
J10_M5_F4_C8_V2	347	354	2%
J10_M5_F4_C8_V3	328	342	4%
J10_M5_F4_C8_V4	335	354	5%
J10_M5_F4_C12_V1	399	404	1%
J10_M5_F4_C12_V2	399	404	1%
J10_M5_F4_C12_V3	399	404	1%
J10_M5_F4_C12_V4	399	404	1%
J10_M10_F4_C4_V1	657	702	7%
J10_M10_F4_C4_V2	657	685	4%
J10_M10_F4_C4_V3	657	682	4%
J10_M10_F4_C4_V4	673	708	5%
J10_M10_F4_C8_V1	680	733	8%
J10_M10_F4_C8_V2	643	701	9%
J10_M10_F4_C8_V3	627	667	6%
J10_M10_F4_C8_V4	623	643	3%
J10_M10_F4_C12_V1	670	692	3%
J10_M10_F4_C12_V2	653	673	3%
J10_M10_F4_C12_V3	633	713	13%
J10_M10_F4_C12_V4	633	713	13%
J10_M15_F4_C4_V1	908	910	0%
J10_M15_F4_C4_V2	902	957	6%
J10_M15_F4_C4_V3	821	857	4%
J10_M15_F4_C4_V4	820	857	4%
J10_M15_F4_C8_V1	934	942	1%
J10_M15_F4_C8_V2	934	950	2%
J10_M15_F4_C8_V3	934	950	2%
J10_M15_F4_C8_V4	904	947	5%
J10_M15_F4_C12_V1	945	981	4%
J10_M15_F4_C12_V2	976	1015	4%
J10_M15_F4_C12_V3	976	1015	4%
J10_M15_F4_C12_V4	1003	1015	1%
J10_M5_F2_C4_V1	438	466	6%
J10_M5_F2_C4_V2	454	470	4%
J10_M5_F2_C4_V3	466	498	7%
J10_M5_F2_C4_V4	463	489	6%
J10_M5_F2_C8_V1	439	464	6%
J10_M5_F2_C8_V2	456	487	7%
J10_M5_F2_C8_V3	448	471	5%
J10_M5_F2_C8_V4	481	515	7%
J10_M5_F2_C12_V1	457	473	4%
J10_M5_F2_C12_V2	448	454	1%
J10_M5_F2_C12_V3	469	507	8%
J10_M5_F2_C12_V4	462	486	5%
J10_M10_F2_C4_V1	716	764	7%
J10_M10_F2_C4_V2	728	751	3%
J10_M10_F2_C4_V3	742	779	5%
J10_M10_F2_C4_V4	811	828	2%
J10_M10_F2_C8_V1	781	840	8%
J10_M10_F2_C8_V2	827	892	8%
J10_M10_F2_C8_V3	831	870	5%
J10_M10_F2_C8_V4	820	861	5%
J10_M10_F2_C12_V1	787	799	1%
J10_M10_F2_C12_V2	769	803	4%
J10_M10_F2_C12_V3	792	801	1%
J10_M10_F2_C12_V4	726	754	4%
J10_M15_F2_C4_V1	1050	1080	3%
J10_M15_F2_C4_V2	1044	1070	3%
J10_M15_F2_C4_V3	1044	1070	3%
J10_M15_F2_C4_V4	990	1030	4%
J10_M15_F2_C8_V1	1048	1086	4%
J10_M15_F2_C8_V2	1067	1112	4%
J10_M15_F2_C8_V3	944	975	3%
J10_M15_F2_C8_V4	996	1013	2%
J10_M15_F2_C12_V1	944	975	3%
J10_M15_F2_C12_V2	996	1013	2%
J10_M15_F2_C12_V3	999	1027	3%
J10_M15_F2_C12_V4	1044	1091	4%
J10_M5_F6_C4_V1	353	354	0%
J10_M5_F6_C4_V2	364	364	0%
J10_M5_F6_C4_V3	308	348	13%
J10_M5_F6_C4_V4	318	336	6%
J10_M5_F6_C8_V1	391	391	0%
J10_M5_F6_C8_V2	391	391	0%
J10_M5_F6_C8_V3	391	391	0%
J10_M5_F6_C8_V4	391	391	0%
J10_M5_F6_C12_V1	357	376	5%
J10_M5_F6_C12_V2	328	328	0%
J10_M5_F6_C12_V3	354	357	1%
J10_M5_F6_C12_V4	278	278	0%

Instance	Best MK	TS MK	GAP
J10_M10_F6_C4_V1	630	631	0%
J10_M10_F6_C4_V2	630	631	0%
J10_M10_F6_C4_V3	630	648	3%
J10_M10_F6_C4_V4	630	645	2%
J10_M10_F6_C8_V1	737	737	0%
J10_M10_F6_C8_V2	599	627	5%
J10_M10_F6_C8_V3	615	615	0%
J10_M10_F6_C8_V4	682	713	5%

Instance	Best MK	TS MK	GAP
J10_M10_F6_C12_V1	807	807	0%
J10_M10_F6_C12_V2	807	807	0%
J10_M10_F6_C12_V3	807	807	0%
J10_M10_F6_C12_V4	807	807	0%
J10_M15_F6_C4_V1	947	985	4%
J10_M15_F6_C4_V2	947	950	0%
J10_M15_F6_C4_V3	932	970	4%
J10_M15_F6_C4_V4	940	968	3%

Instance	Best MK	TS MK	GAP
J10_M15_F6_C8_V1	951	951	0%
J10_M15_F6_C8_V2	931	931	0%
J10_M15_F6_C8_V3	924	929	1%
J10_M15_F6_C8_V4	992	992	0%
J10_M15_F6_C12_V1	1055	1055	0%
J10_M15_F6_C12_V2	1055	1055	0%
J10_M15_F6_C12_V3	925	948	3%
J10_M15_F6_C12_V4	925	970	5%

Results show that for any of the 108 small instances tested the average deviation between the TS solution and the MILP optimal solution was only 3%. A confidence interval for the Gap with a 95% confidence level was calculated and demonstrates that the proposed TS reaches solutions with a 3% to 4% difference from the optimal MILP solutions for small instances.

As stated before, the MILP model was not able to produce solutions for these instances using the minimize tardiness objective function so comparisons for the GAP between the optimal MILP tardiness value and the best TS tardiness value were not made.

7.3.2. All Instances MID Comparison

In order to compare all pareto solution sets we used the index called Mean Ideal Distance (MID) proposed by Ebrahimi et al. (2014), which is calculated with equations (34) and (35)

$$\text{MID} = \frac{\sum_{i=1}^{n} C_i}{n} \qquad (34)$$

$$C_i = \sqrt{{f_{1i}}^2 + {f_{2i}}^2} \qquad (35)$$

Where n is the number of non-dominated Pareto Frontier Solutions per instance and C_i is the distance between the ith non-dominated solution and the ideal point, while f_{1i} and f_{2i} are the values of ith non-dominated solution for first and second objective functions respectively.

As stated in the last-mentioned study, we will use (0,0) as the ideal point for this study so to evaluate the TS for instances that were not able to reach a MILP optimal solution. With these results a 4-way ANOVA was calculated in order to determine the effect of the size and characteristics of the instance on the Multi-Objective solution. factors of this ANOVA are:

Factor 1: Number of Jobs (J)
Factor 2: Number of Factories (F)
Factor 3: Number of Machines (M)
Factor 4: Capacity of vehicle (C)

Each combination had a total of 4 observations which correspond to the 4 versions (V) any given instance combination has. The results of the ANOVA are presented as follows:

Tests of Between-Subjects Effects

Dependent Variable: MIDBook

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Model	2,561E+11[a]	81	3162004175	170,330	,000
Jobs	8,517E+10	2	4,259E+10	2294,036	,000
Machines	2924307434	2	1462153717	78,763	,000
Factories	1,955E+10	2	9773944071	526,500	,000
Capacity	634206915,7	2	317103457,9	17,082	,000
Jobs * Machines	1247760897	4	311940224,3	16,804	,000
Jobs * Factories	1,420E+10	4	3550526224	191,259	,000
Jobs * Capacity	625790160,9	4	156447540,2	8,427	,000
Machines * Factories	99367005,11	4	24841751,28	1,338	,256
Machines * Capacity	49907306,39	4	12476826,60	,672	,612
Factories * Capacity	101380703,4	4	25345175,86	1,365	,247
Jobs * Machines * Factories	150523880,5	8	18815485,06	1,014	,426
Jobs * Machines * Capacity	83099050,53	8	10387381,32	,560	,810
Jobs * Factories * Capacity	118408550,2	8	14801068,78	,797	,605
Machines * Factories * Capacity	53773215,94	8	6721651,993	,362	,940
Jobs * Machines * Factories * Capacity	161956587,2	16	10122286,70	,545	,921
Error	4511051693	243	18563998,74		
Total	2,606E+11	324			

a. R Squared = ,983 (Adjusted R Squared = ,977)

Figure 8. ANOVA all instances

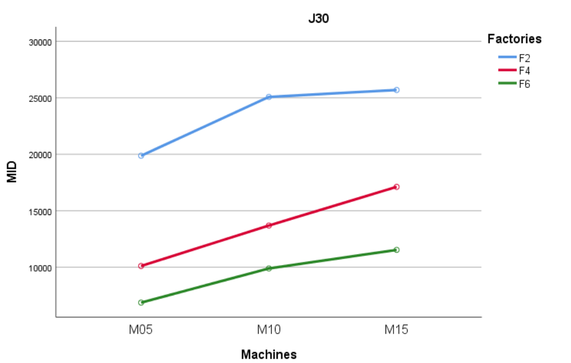

Figure 9. J10 Marginal Average

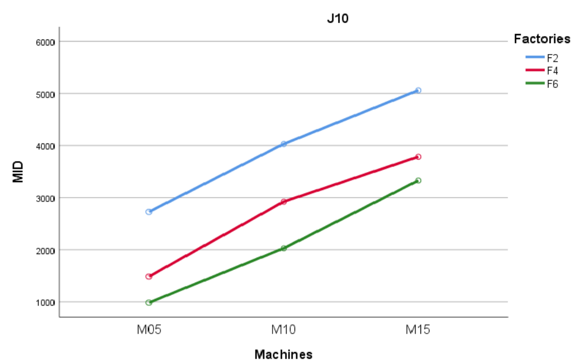

Figure 10. J30 Marginal Average

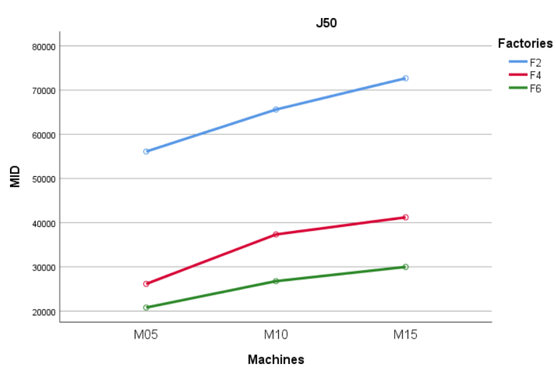

Figure 11. J50 Marginal Average

Based on a 95% confidence level ANOVA (Figure 8), results show that the MID is affected by all factors individually and all the double effects (combination of two factors).

Figure 9, 10 and 11 show that the MID behavior increases as the number of Jobs, Machines and Truck Capacity increase. This is to be expected since as instance size increases the problem complexity as well as the makespan and tardiness value increase. Results also show that, different from all other factors, when the number of Factories increases the MID decreases. This last relationship is explained knowing than when there are more factories the utilization of each individual factory decreases, as the TS tries to schedule jobs evenly between factories to minimize the makespan solution of the joint problem.

The MID is mostly affected by the size of the instance (number of jobs to process) having great effect in comparison to other factors. An LSD test was used on each factor and results show that C8 and C12 as truck capacity values have the same effect meaning that there is enough statistical evidence to suggest there is no difference in using either one of them. The adjusted correlation coefficient also shows that the statistic model has a good fit to our data, reporting a 97.7% value, this is ideal to make predictions of the MID behavior of lower or higher levels of the different factors not contemplated in the experiment.

7.4. Initial Solutions vs TS Solutions

Initial solutions for makespan and tardiness for each of the 324 instances were compared to the best makespan and tardiness solutions in each instances' non-dominated set of solutions. A Gap between this value was calculated for both makespan and tardiness showing an average of 8% improvement for the makespan and a 12% improvement for the tardiness when using the proposed TS.

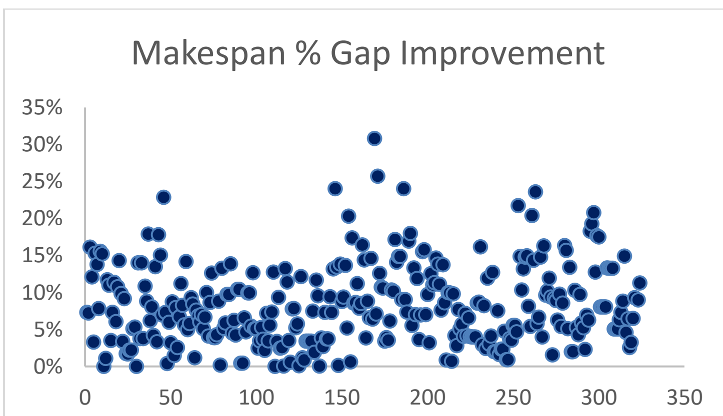

Figure 12. Makespan GAP plot graph

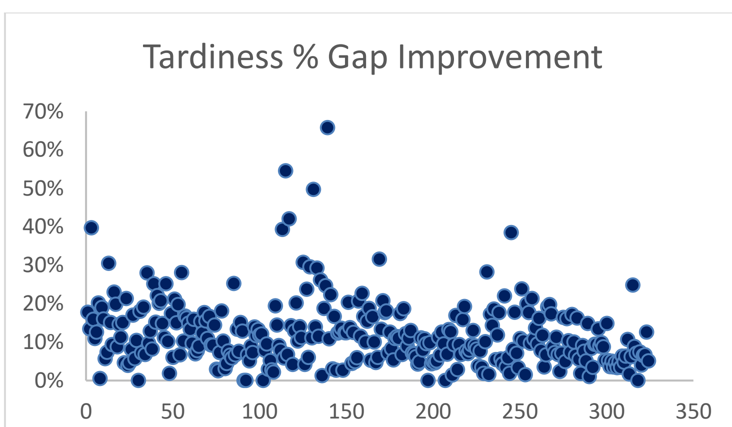

Figure 13. Tardiness GAP plot graph

Figures 12 and 13 show a visual representation of al the Gap values calculated for both makespan and tardiness for each instance. A confidence interval with a 95% confidence level was calculated for both sets of Gaps where it was found that the proposed TS reaches solutions with an 8% to 9% improvement of makespan and a 11% to 13% improvement when compared to the initial solution.

7.5. CPU Execution time

All instances were executed in a Windows 10 64-bit environment with 32GB DDR4 Ram memory and a 3.4GHz Intel Processor. CPU execution time varies greatly depending if the instance is small (10 jobs), medium (30 jobs) or large (50 jobs). It should also be noted that most large instances have a reported execution time of 9000 seconds because this was the time limit set for the experiment, therefore no comparison was made for this group.

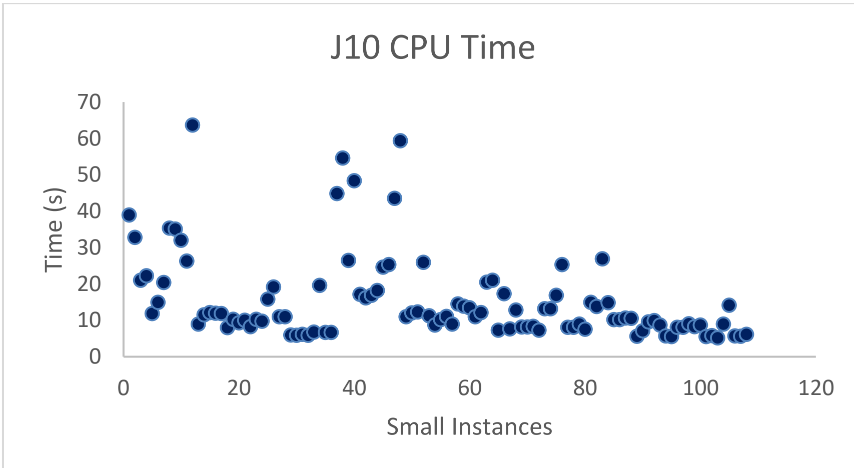

Figure 14. J10 CPU time plot graph

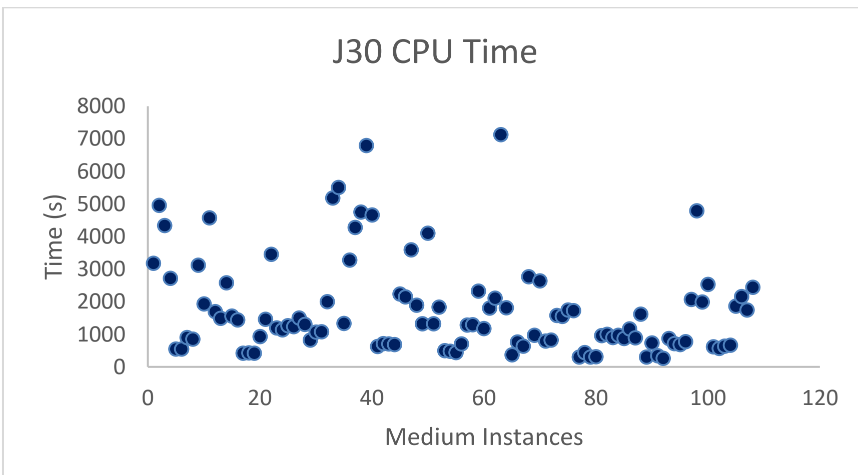

Figure 15. J30 CPU time plot graph

Figures 14 and 15 show a visual representation of execution time for small and medium instances. The average execution time of small instances is 15 seconds while the average for medium instances is 1727 seconds. A confidence interval with a 95% confidence level was calculated for both types mentioned were it was found that small instances range from 13 to 18 seconds of execution time while medium instances range from 1454 to 1999.

8. Conclusions and Future work

Scheduling and distribution are two very important tasks in modern day Supply Chain Systems**.** DPFSP environments and CVRP environments both prove to have important applications in the real world. In this paper we present the combination of both NP-hard problems, which makes this thesis to distinguish itself from other studies made on the same individual subjects. Since modern day operations often require the optimization of more than a single objective, our study is focused around a multi-objective minimization of both makespan and total tardiness.

A multi-objective TS metaheuristic model was constructed to solve such problem and after comparing the proposed metaheuristic solution to benchmark results for DPFSP (Naderi & Ruiz, 2010) and benchmark results for CVRP (Augerat, 1995) individually, these stated that the model produces solutions close to the optimal values reported in the literature (10% or less deviation for DPFSP and 20% deviation for CVRP) . In addition, the joint-problem study showed that the model reaches better solutions in smaller computational time than a MILP model when using NEH as the starting solution for each instance.

Even though tardiness function could not be tested on the MILP, complete makespan for the TS shows small gap against the MILP optimum. Another important effect of the PAES - TS solution and the joint problem is to allow the user of the metaheuristic to choose which objective function to prioritize or if the focus of the problem is to have a balance of both functions and therefore which scheduling and order of jobs on each truck to choose.

Finally, the results and the instances created for the model (since the joint problem has not been studied before) will serve other authors to compare themselves and their future studies in this type of problem combinations, as well as encouraging students or professionals to pursue solutions for joint problems that come close to modern day SC environments. It should be noted that future works include the use of diverse metaheuristics or simulation models, as well as different combinations among factors that could extend the knowledge of this interesting problem. Introducing different objectives such as transport or total chain costs and additional constraints such as setup time could further increase the problem's real-world applications.

9. Annexes table.

Table 8. Annexes Table

Annex	Name	Type of file	Link
1	Excel (VBA) used for the problem resolution	Excel	https://drive.google.com/open?id=1eqvmHx6YOwFhYAIoeH0xfYawM4phTjab
2	324 Instances	ZIP archive	https://drive.google.com/open?id=1vTS9PkaELKk21HaH-3Tu8aJV_BAV8oiy
3	Parametrization results	Excel	https://drive.google.com/open?id=1RyjYGMR1B6zIuje0TEf5a5nu5JBBigYE
4	Pareto Frontier VBA all instances Solutions	Excel	https://drive.google.com/open?id=1FkQUwk9TS0F4MwPdr5m9Dq637xEeuoVB
5	MID all instances	Excel	https://drive.google.com/open?id=1HougaTLBUSqdw1TG9kX1h4CtkCpixzhc
6	Starting Solutions	Excel	https://drive.google.com/open?id=18umtqowr3CAZkLIL2DZWOxWWVIjYypc8

References

Bargaoui, H., Belkahla Driss, O., & Ghédira, K. (2017). A novel chemical reaction optimization for the distributed permutation flowshop scheduling problem with makespan criterion. *Computers and Industrial Engineering*, *111*, 239–250. https://doi.org/10.1016/j.cie.2017.07.020

Cassettari, L., Demartini, M., Mosca, R., Revetria, R., & Tonelli, F. (2018). A multi-stage algorithm for a capacitated vehicle routing problem with time constraints. *Algorithms*, *11*(5). https://doi.org/10.3390/a11050069

Chen, L., Yang, J., & Guo, Z. (2015). A heuristic algorithm for integrated production and transportation scheduling with multiple transportation modes. *2015 12th International Conference on Service Systems and Service Management, ICSSSM 2015*, *2014*. https://doi.org/10.1109/ICSSSM.2015.7170221

Chen, Z.-L. (2010). Integrated Production and Outbound Distribution Scheduling: Review and Extensions. *Operations Research*, *58*(1), 130–148. https://doi.org/10.1287/opre.1080.0688

Chen, Z.-L., & Vairaktarakis, G. L. (2005). Integrated Scheduling of Production and Distribution Operations. *Management Science*, *51*(4), 614–628. https://doi.org/10.1287/mnsc.1040.0325

Dantzig, G. B., & Ramser, J. H. (1959). The Truck Dispatching Problem. *Management Science*, *6*(1), 80–91. https://doi.org/10.1287/mnsc.6.1.80

Ebrahimi, M., Fatemi Ghomi, S. M. T., & Karimi, B. (2014). Hybrid flow shop scheduling with sequence dependent family setup time and uncertain due dates. *Applied Mathematical Modelling*, *38*(9–10), 2490–2504. https://doi.org/10.1016/j.apm.2013.10.061

Eren, T. (2007). A multicriteria flowshop scheduling problem with setup times. *Journal of Materials Processing Technology*, *186*(1–3), 60–65. https://doi.org/10.1016/j.jmatprotec.2006.12.022

Gao, J., Chen, R., & Deng, W. (2013). An efficient tabu search algorithm for the distributed permutation flowshop scheduling problem. *International Journal of Production Research*, *51*(3), 641–651. https://doi.org/10.1080/00207543.2011.644819

Glover, F. (1989). Tabu Search - Part I. *ORSA Journal on Computing*, *2 1*(3), 4–32. https://doi.org/10.1287/ijoc.2.1.4

Gupta, J. N. D., Palanimuthu, N., & Chen, C. L. (1999). Designing a tabu search algorithm for the two-stage flow shop problem with secondary criterion). *Production Planning and Control*, *10*(3), 251–265. https://doi.org/10.1080/095372899233217

Ismail, Z., Teknologi, U., Rohaizad, W., Ibrahim, W., Teknologi, U., Study, A., Metaheuristics, H. L., Hard, N. P., Models, R., Index, S., National, M., Area, K., & Ismail, Z. (2008). Traveling Salesman Approach for Solving Petrol Distribution Using Simulated Annealing. *American Journal of Applied Sciences*, *5*(November), 1543–1546. https://doi.org/10.3844/ajassp.2008.1543.1546

Iswari, T., & Asih, A. M. S. (2018a). Comparing genetic algorithm and particle swarm optimization for solving capacitated vehicle routing problem. *IOP Conference Series: Materials Science and Engineering*, *337*, 012004. https://doi.org/10.1088/1757-899X/337/1/012004

Iswari, T., & Asih, A. M. S. (2018b). Comparing genetic algorithm and particle swarm optimization for solving capacitated vehicle routing problem. *IOP Conference Series: Materials Science and Engineering*, *337*, 012004. https://doi.org/10.1088/1757-899X/337/1/012004

Kim, Y.-D., Lim, H.-G., & Park, M.-W. (1996). Search heuristics for a flowshop scheduling problem in a printed circuit board assembly process. *European Journal of Operational Research*, *91*(1), 124–143. https://doi.org/10.1016/0377-2217(95)00119-0

Knowles, J. D., & Corne, D. W. (2000). Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy. *Evolutionary Computation*, *8*(2), 149–172. https://doi.org/10.1162/106365600568167

Li, Z., Duan, W., Ji, M., Yang, Y., Wang, S., & Liu, B. (2016). The distributed permutation flowshop scheduling problem with different transport timetables and loading capacities. *2016 IEEE Congress on Evolutionary Computation, CEC 2016*, *71101139*, 2433–2437. https://doi.org/10.1109/CEC.2016.7744090

Lin, S. W., Ying, K. C., & Huang, C. Y. (2013). Minimizing makespan in distributed permutation flowshops using a modified iterated greedy algorithm. *International Journal of Production Research*, *51*(16), 5029–5038. https://doi.org/10.1080/00207543.2013.790571

Mohammadi, S., Cheraghalikhani, A., & Ramezanian, R. (2018). A joint scheduling of production and distribution operations in a flow shop manufacturing system. *Scientia Iranica*, *25*(2E), 911–930. https://doi.org/10.24200/sci.2017.4437

Naderi, B., & Ruiz, R. (2010). The distributed permutation flowshop scheduling problem. *Computers & Operations Research*, *37*(4), 754–768. https://doi.org/10.1016/j.cor.2009.06.019

Nawaz, M., Enscore, E. E., & Ham, I. (1983). A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *Omega*, *11*(1), 91–95. https://doi.org/10.1016/0305-0483(83)90088-9

Pinto, T., Alves, C., & Valério de Carvalho, J. (2018). Column generation based primal heuristics for routing and loading problems. *Electronic Notes in Discrete Mathematics*, *64*, 135–144. https://doi.org/10.1016/j.endm.2018.01.015

Pundoor, G., & Chen, Z. L. (2005). Scheduling a production-distribution system to optimize the tradeoff between delivery tardiness and distribution cost. *Naval Research Logistics*, *52*(6), 571–589. https://doi.org/10.1002/nav.20100

Soto, M., Sevaux, M., Rossi, A., & Reinholz, A. (2017). Multiple neighborhood search, tabu search and ejection chains for the multi-depot open vehicle routing problem. *Computers and Industrial Engineering*, *107*, 211–222. https://doi.org/10.1016/j.cie.2017.03.022

Ta, Q. C., Billaut, J. C., & Bouquard, J. L. (2016). Heuristic algorithms to minimize the total tardiness in a flow shop production and outbound distribution scheduling problem. *Proceedings of 2015 International Conference on Industrial Engineering and Systems Management, IEEE IESM 2015*, *October*, 128–134. https://doi.org/10.1109/IESM.2015.7380147

Wang, K., Luo, H., Liu, F., & Yue, X. (2017). Permutation Flow Shop Scheduling With Batch Delivery to Multiple Customers in Supply Chains. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 1–12. https://doi.org/10.1109/TSMC.2017.2720178

Xia, Y., Fu, Z., Lijun, P., & Duan, F. (2018). Tabu search algorithm for the distance-constrained vehicle routing problem with split deliveries by order. *PLoS ONE*, *11*(5). https://doi.org/10.1371/journal

Yang, W., & Ke, L. (2018). *An improved fi reworks algorithm for the capacitated vehicle*.

Zhu, W., Qin, H., Lim, A., & Wang, L. (2012). A two-stage tabu search algorithm with enhanced packing heuristics for the 3L-CVRP and M3L-CVRP. *Computers & Operations Research*, *39*(9), 2178–2195. https://doi.org/10.1016/j.cor.2011.11.001

Zou, X., Liu, L., Li, K., & Li, W. (2017a). A coordinated algorithm for integrated production scheduling and vehicle routing problem. *International Journal of Production Research*, *7543*, 1–20. https://doi.org/10.1080/00207543.2017.1378955

Zou, X., Liu, L., Li, K., & Li, W. (2017b). A coordinated algorithm for integrated production scheduling and vehicle routing problem. *International Journal of Production Research*, *7543*, 1–20. https://doi.org/10.1080/00207543.2017.1378955