Pesquisa Operacional (2023) 43: e269080 p.1-20
doi: 10.1590/0101-7438.2023.043.00269080

Printed version ISSN 0101-7438 / Online version ISSN 1678-5142
www.scielo.br/pope
ARTICLES

LARGE SCALE SUPPLY CHAIN NETWORK DESIGN: AN EFFECTIVE HEURISTIC APPROACH

Juan Leandro Andres Parra Galvez[1*] and Denis Borenstein[2]

Received October 28, 2022 / Accepted January 9, 2023

ABSTRACT. This work considers the strategic Supply Chain Network Design (SCND) problem, which is to define the number and location of facilities, and the flow of products among them to fulfilling a long-term deterministic demand. A two-phase heuristic approach was specially developed to solve large scale problems in reasonable time, extending a previous algorithm introduced in Farias et al. (2017). In the construction phase, a multi-start approach was developed to generate diversified initial solutions from each new iteration of a layered-based rounding heuristic. In the second phase, a local search heuristic improves the solution provided by the rounding method. The solution method is evaluated using randomly generated instances, and a evaluated strategic of marketing in a real case study applied to a company to redesigning the supply chain to two lines of products.The obtained results evidence the effectiveness and flexibility of the developed approach for handling very large instances.

Keywords: strategic supply chain design, multi-start, layered-based.

1 INTRODUCTION

The supply chain (SC) management has attracted considerable attention in the last decades (der Vaat & Donk, 2008). An effective and efficient management requires many decisions that can be grouped in three levels, namely: strategic level (e.g., selection of suppliers, location, and capacities of factories and warehouses, assignment of customers, raw materials and products flows), tactical level (e.g., production, and distribution planning), and operational level (e.g., cargo sizes, cargo allocation to vessels, and vessel scheduling and routing (Farahani et al., 2014)). Given the diversity and contexts of the decisions within the SC management, several models were formulated to support the decision-making process, considering multi-period (Melo et al., 2014), stochastic demand (Govindan et al., 2015), uncertain demand (Fattahi et al., 2018), sustainable SCs (Eskandarpour et al., 2015), global logistic systems (Goetschalkx et al., 2002), and reverse logistics (Keyvanshokooh et al., 2013).

*Corresponding author
[1] Facultad de Ingenieria, Arquitectura y Diseño, Universidad San Sebastian, Santiago, Chile – E-mail: juan.parra@uss.cl
https://orcid.org/0000-0001-9244-0019
[2] Escola de Administração, Universidade Federal do Rio Grande do Sul, RS, Brazil – E-mail: denis.borenstein@ufrgs.br
https://orcid.org/0000-0002-8276-7197

In this paper, we focus on the strategic supply chain network design (SCND). In certain sense, the tactical and operational decisions are highly dependent on the strategic SCND. The problem consists in the following decisions: (i) the selection and location of suppliers, factories, and distribution centers (DCs); (ii) the volume of each raw material from each selected supplier to each open factory; (iii) the volume of each product from each factory to each DC; and (iv) the assignment of the demand of each customer zone (CZ) to an open DC, considering deterministic demand, multi-product, and a long single-period.

The strategic SCND was mainly formulated as a mixed integer linear programming (MILP) (Jayaraman & Pirkul, 2001; Lee & Kwon, 2010), where the strategic aspects are usually represented by integer variable decisions, while the material and product flows along the network are represented by continuous decision variables (Thanh et al., 2012). Due to the combinatorial nature of the problem, several solution methods were developed. Some researchers used traditional methods to solve integer programming, such as Lagrangean relaxation (Wu & Golbasi, 2004; Kumar & Tiwari, 2013), Benders decomposition (Keyvanshokooh et al., 2016), and column generation (Romeijn et al., 2007). Metaheuristic techniques were also widely used, such as tabu search (Lee & Kwon, 2010; Armentano et al., 2011; Melo et al., 2012), genetic algorithms (Altiparmak et al., 2006), particle swarm optimization (Khalifehzadeh et al., 2015), fuzzy logic (Pham & Yenradee, 2017), and hybrid approaches (Soleimani & Kannan, 2015). Some approaches solved the problem considering multi objectives (Ghasemian Zarini & Javadian, 2020; Gholami et al., 2019).

However, due to the complexity of the problem, the previously mentioned solution methods were only able to solve instances with very limited number of elements in the network, using artifices such as product aggregation to solve real-world problems. Farias et al. (2017) pointed out this issue and developed a multi-start based heuristic to solve large instances of the problem. Although the results were quite impressive in terms of the dimensions of the solved instances, the developed method requires intensive experiments to set up important parameters to obtain good solutions for each different configurations of the problem instances, in terms of the number of products, CZs, and DCs. Further, additional experiments demonstrated that the heuristics presents some instabilities related to halt conditions.

The solution approach is an extension of the heuristic introduced in Farias et al. (2017), by integrating a local search in the solution process, we present a two-stage heuristic framework for efficiently solving the deterministic, four-echelon, multi-product, single-source strategic SCND problem for very large-scale problems. The first stage is a constructive phase in which a multi-start layered-based rounding algorithm proposal in Farias et al. (2017) to find several possible feasible solutions. The second phase is based on a local search heuristic to improve the solutions delivered by the previous phase. The two phases interact until a near optimal solution is obtained. The effectiveness and efficacy of the developed approach is demonstrated in a computational study, using randomly generated instances, with up to 25 suppliers, 25 raw materials, 20 factories, 50 DCs, 170 products, and 300 CZs. Further, the heuristic is evaluated in a real-world case, the redesign of a large supply chain, considering a new distribution strategy for two product lines.

In the real case, the flexibility of the solution approach to handle different design contexts is also demonstrated.

The paper is organized as follows. Section 2 presents the mathematical formulation of the SCND. In Section 3, the heuristic approach is explained with details. Section 4 presents the computational study carried out, comparing the numerical results with previous solution methods in the literature. Section 5 described the heuristic application for solving a real world problem faced by a Brazilian company. In Section 6, a summary of the results is presented, and areas of future research are discussed.

2 FORMULATION

We consider a supply chain composed of four layers: suppliers, manufacturing centers (factories), DCs, and customers zones. Further, it is proposed a supply chain based on the following assumptions: (i) the different customers are geographically grouped in customer zones (CZs); (ii) the demand of the CZs for each product must be completely fulfilled in the planning period; (iii) a single-source requirement is used, e.g., CZs are served from one and just one DC; (iv) the suppliers capacities for each raw material are known; and (v) suppliers, factories and DCs capacities must be respected.

We use the arc-flow MILP model from Farias et al. (2017), which is presented here for the sake of completeness. Constraints are imposed on the raw materials supply, production capacity, DC capacity, and demand of CZs. The objective is to minimize the fixed and variable costs of the supply chain network. Based on the mathematical notation presented in Table 1, the deterministic, single-period, multi-product strategic SCND can be formally defined through the following MILP model:

Table 1 – Mathematical notation.

Symbol	Definition
Sets	
C	the set of customer zones, indexed by c
W	the set of DCs, indexed by w
F	the set of factories, indexed by f
R	the set of raw materials, indexed by r
V	the set of suppliers, indexed by v
P	the set of products, indexed by p

Table 1 (cont.)

Symbol	Definition
Parameters	
d_{cp}	the demand for product $p \in P$ at CZ $c \in C$
U_w	the maximum number of DCs that are opened
U_f	the maximum number of factories that are opened
u_{rp}	the utilization rate of raw material $r \in R$ per unit of finished product $p \in P$
u_p	capacity utilization rate per unit of product $p \in P$
CAP_w	the annual throughput at DC $w \in W$
CAP_{vr}	the supply capacity of raw material $r \in R$ at vendor $v \in V$
CAP_f	the capacity of factory $f \in F$
CT_w^o	the annual fixed cost of operating DC $w \in W$
CT_f^o	the annual fixed cost of operating factory $f \in F$
CT_{wp}^g	the unit cost of throughput of product $p \in P$ at DC $w \in W$
CT_{fp}^p	the unit production cost of product $p \in P$ at factory $f \in F$
CT_{fvr}^t	the unit transportation cost of raw material $r \in R$ from supplier $v \in V$ to factory $f \in F$
CT_{fwp}^t	the unit transportation cost of product $p \in P$ from factory $f \in F$ to DC $w \in W$
CT_{wcp}^t	the unit transportation cost of product $p \in P$ from DC $w \in W$ to CZ $c \in C$
Variables	
a_w	a binary variable, and it is 1 if DC $w \in W$ is selected, and 0 otherwise
b_f	a binary variable, and it is 1 if factory $f \in F$ is selected, and 0 otherwise
g_{wc}	a binary variable, and it is 1 if customer $c \in C$ is assigned to DC $w \in W$, and 0 otherwise
z_{fw}^p	the amount of product $p \in P$ shipped from factory F to DC $w \in W$
y_{vf}^r	the amount of raw materials $r \in R$ shipped from supplier $v \in V$ to factory $f \in F$

$$\begin{aligned}\min \sum_{w\in W} CT_w^o a_w + \sum_{f\in F} CT_f^o b_f + \sum_{w\in W}\sum_{c\in C}\sum_{p\in P} CT_{wp}^g d_{cp} g_{wc} + \\ \sum_{f\in F}\sum_{w\in W}\sum_{p\in P} CT_{fp}^p z_{fw}^p + \sum_{f\in F}\sum_{r\in R}\sum_{v\in V} CT_{fvr}^t y_{vf}^r + \\ \sum_{f\in F}\sum_{w\in W}\sum_{p\in P} CT_{fwp}^t z_{fw}^p + \sum_{w\in W}\sum_{c\in C}\sum_{p\in P} CT_{wcp}^t d_{cp} g_{wc}\end{aligned} \tag{1}$$

st

$$\sum_{w\in W} g_{wc} = 1 \qquad \forall c \tag{2}$$

$$\sum_{c\in C}\sum_{p\in P} d_{cp} g_{wc} \le CAP_w a_w \qquad \forall w \tag{3}$$

$$\sum_{c\in C} d_{cp} g_{wc} \le \sum_{f\in F} z_{fw}^p \qquad \forall p, w \tag{4}$$

$$\sum_{f\in F} y_{vf}^r \le CAP_{vr} \qquad \forall r, v \tag{5}$$

$$\sum_{w \in W} \sum_{p \in P} u_{rp} z_{fw}^{p} \leq \sum_{v \in V} y_{vf}^{r} \qquad \forall r, f \tag{6}$$

$$\sum_{w \in W} \sum_{p \in P} u_{p} z_{fw}^{p} \leq CAP_f b_f \qquad \forall f \tag{7}$$

$$\sum_{w \in W} a_w \leq U_w \tag{8}$$

$$\sum_{f \in F} b_f \leq U_f \tag{9}$$

$$z_{fw}^{p}, y_{vf}^{r}, \geq 0 \qquad \forall f, r, p, v, w \tag{10}$$

$$a_w, g_{wc}, b_f \in \{0,1\} \qquad \forall c, f, w \tag{11}$$

Objective (1) consists in minimizing the sum of the annual cost of DCs, the throughput costs of DCs, the production costs of factories and transportation costs of materials to factories and transportation costs of products from factories to CZs through the DCs. Constraints (2) ensure that each CZ is assigned to one DC. Constraints (3) guarantee that the capacity of each DC is not violated. Constraints (4) force that each DC has sufficient products for its associated customers. Constraints (5) ensure the capacity of any raw material at any vendor is satisfied. Constraints (6) limit the quantity of manufactured products to the amount of acquired raw materials. Constraints (7) guarantee that the capacity of any factory is satisfied. Constraints (8) and (9) impose an upper bound on the number of DCs and factories, respectively. Constraints (10) and (11) impose valid values for the decision variables of the model.

3 SOLUTION METHOD

In order to overcome the computational difficulties associated with producing a good solution for very large instances, we implemented an extension of the multi-start heuristic framework introduced in Farias et al. (2017). A multi-start mechanism is conceived as a way to better exploit new solution spaces by applying multiple random solutions, avoiding the stagnation in a local optimum (García-López et al., 2010).

The overall framework is outlined in Algorithm 1. The first stage is a constructive phase, which applies the multi-start layered-based rounding algorithm proposed in Farias et al. (2017) to find several possible feasible solutions. The main objective of the constructive phase is to find a lower bound (LB) for the problem, and a set of feasible solutions with good upper bounds (UBs), e.g., with low gaps in relation to the defined LB. In order to generate solution diversity in the constructive phase, the multi-start and the rounding algorithms generate several solutions, with different configurations of factory, DCs, and different assignments of CZs to DCs. The perturbation procedure disables these decision elements in a previous solution to start a new solution in each iteration. This strategy allows to explore different solution spaces for feasible and good solutions found in the previous iterations. Next, a Local search is employed to find improvement solutions, based on a solution obtained in the constructive phase. The local search employs two methods:

distribution centers exchange and arc exchange at the CZ level, using a tabu list like strategy to allow diversification, avoiding stagnation in a local optimum. If the best solution obtained by the local search is not acceptable, this solution is send to the multi-start mechanism in the constructive phase for the generation of new possible models, and a new iteration begins. The process is repeated either until an optimal gap or a time limit is obtained.

The developed algorithm enhances the capabilities of the solution approach introduced in Farias et al. (2017), by integrating a local search to the multi-start mechanism. Moreover, while in Farias et al. (2017), the rounding algorithm was just responsible for finding good initial solutions, the local search improves the solution obtained by the rounding algorithm at each iteration towards improving a feasible solution using reasonable computing time in this new version. Both integration have not only allowed to define an objective halting criterion, but also has improved the quality of the solution process.

One of the most important issues to contribute to the good performance of a multi-start framework is the definition of a good set of perturbation parameters, since they allow to obtain a greater variability during the exploration of different solution spaces. Our objective is to generate feasible solutions for each iteration with different sets of factory, DCs, and assignments DCs to customers. As typical in heuristic development, tuning these parameters is a critical issue. Based on a number of computational experiments, in Section 4 the parameter settings that best contributed to a good performance of our multi-start procedure are presented.

3.1 Layered-based rounding heuristic

The main idea of our layered-based rounding heuristic is to solve the linear relaxation of the problem, rounding the fractional variables to recover integer feasible solutions Melo et al. (2012). We used a slightly modified version of the algorithm presented in Farias et al. (2017), by integrating a local search strategy after a rounding heuristic. The algorithm here is presented by completeness. In our MILP formulation, it is possible to find three types of integer variables in the model: the first type corresponds to determining factories and DC locations, while the three type refers to assigning CZs to DCs. In general, the number of factories location candidates is much smaller than the remaining variables. As a consequence, a natural layered structure arise in this problem: we have to use the rounding to fix the DC locations first and then fix the assignment of CZs to DCs. If we fix the CZ-DC assignment first, the corresponding DC is also fixed at the same time, which may result in poor solutions. The developed layered-based rounding heuristic is outlined in Algorithm 2.

The first steps of algorithm 2, specifically steps 2 and 3 apply certain criteria for obtained rounding values for decision variables a_w, g_{wc}. Once fixed the decision variables related with DCs and CZs, which correspond to the third and fourth layers of model SCND, we solve the first two layers with model RTM, presented below. RTM model is used to found values for the following decision variables values of the first two layers, characterising a complete solution of model SCND. Model RTM can be solved very fast using a contemporary optimization solver, since

Algorithm 1 Multi-start heuristic framework

Step 1 (Construction phase): Find a good feasible solution

Step 1.1 (Initialization) Define the parameters for the multi-start mechanism, namely number of factories, DCs, and assignment arcs to be disabled.

Step 1.2 (Initial Solution) Obtain a lower bound for the problem by solving model (1)-(11), relaxing the binary condition on the decision variables in constraints (11).

Step 1.3 (Rounding Initial Solution) Run the rounding heuristic (Algorithm 2) for the current solution for a feasible good solution. If no feasible solution is obtained, terminate. Otherwise, go to Step 2.

Step 1.4 (Multi-start mechanism) Randomly disable factory, DCs and a percentage of assigned arcs of CZs that appear in the current solution, obtaining a new model.

Step 1.5 (Rounding Solutions) Run the rounding heuristic (Algorithm 2) for the several instances generated in the previous step. If no feasible solution is obtained, go to Step 1.4. Otherwise, go to Step 2.

Step 2 (Improvement phase): Perform a Local Search, as follows:

Step 2.1 Apply the DC Exchange routine (Algorithm 3) in the current solution from Step 1.

Step 2.2 Apply the Arc Exchange routine (Algorithm 4) in the solution from Step 2.1.

Step 2.3 Update the overall upper bound if a better solution is found. Store this as the best overall solution.

Step 3 (Termination checking): If the given time limit is reached or an acceptable gap is reached, output the overall best solution and stop. Otherwise, go to Step 1.4.

only variables $g_{wc} = 1, \forall w, c$, previously computed by Algorithm 2, are considered in constraints (4), significantly decreasing its complexity. Note that the demand in each DC is updated to consider the current assignment, so the model RTM considers (i) sending materials from vendors to plants; (ii) sending products from plants to DCs; and (iii) production costs at the plants. Throughput costs at DCs, transportation costs from DCs to customer zones, and DC fixed costs are known and determined by the rounding algorithm.

Model RTM:

$$\min \sum_{f\in F} CT^o_f b_f + \sum_{f\in F}\sum_{w\in W}\sum_{p\in P} CT^p_{fp} z^p_{fw} + \sum_{f\in F}\sum_{r\in R}\sum_{v\in V} CT^t_{fvr} y^r_{vf} + \sum_{f\in F}\sum_{w\in W}\sum_{p\in P} CT^t_{fwp} z^p_{fw} \tag{12}$$

st

$$(4)-(7),(9)-(10) \tag{13}$$

$$b_f \in \{0,1\} \qquad \forall f \tag{14}$$

Algorithm 2 Layered-based rounding

Step 1 **(Initialization)**: Consider the linear relaxation solution of the model.

Step 2 **(Locating the DCs)**: Let dc be the number of DC location variables. Set $dc \leftarrow 0$.

- **Step 2.1:** Sort binary variables a_w in a non-increasing order.
- **Step 2.2:** For all a_w do: If a_w is almost 1 (more than 0.95), fix it to 1, and set $dc = dc + 1$. If $dc > U_w$, go to step 3. If $a_w < 0.95$ then exit the loop.
- **Step 2.3:** If no a_w was fixed in Step 2.2, set a_w with the biggest value to 1.
- **Step 2.4:** With fixed a_w, solve the updated linear programming model again.

Step 3 **(Assign customers to DCs)**: Let cz be the number of CZs that have been assigned. Set $cz \leftarrow 0$.

- **Step 3.1:** Sort binary variables g_{wc} in a non-increasing order.
- **Step 3.2:** For all g_{wc} do: If g_{wc} is almost 1 (more than 0.95), fix it to 1, and set $cz = cz + 1$. If all CZs are fixed, go to step 4. If $g_{wc} < 0.95$, exit the loop.
- **Step 3.3:** If no g_{wc} was fixed in Step 3.2, choose the CZ c' with the largest total demand. Choose then the DC w' with the largest remaining capacity. Fix $g_{w'c'} = 1$.
- **Step 3.4:** With fixed g_{wc} in Steps 3.2 or 3.3, solve the updated linear programming model again.

Step 4 **(Locate plants)**: If $(b_f = 0, 1, \forall f \in F)$, go to Step 5. Otherwise, solve model RTM, fixing the plants, DCs locations, and DCs–CZs assignments of the current solution. Define the total costs.

Step 5 : Output the best solution.

3.2 Local Search

Local search is used in attempt to improve the solution obtained in the constructive phase, after the multi-start mechanism is used and an integer solution is obtained by the rounding algorithm. Local search attempts to find new better solutions using two neighborhood methods, distribution centers exchange and arc exchange. In order to guide the neighborhood selection, two indexes were used to prioritize the DCs and arcs to be exchanged, RDC_w and RAR_a, respectively. Both indexes use relevant costs in the selection of DCs or arcs to be exchanged following Lee & Kwon (2010). After applying each method and if a better feasible solution is found, model RMT is again employed to obtain a complete solution to the problem. Next both local search routines are described.

3.2.1 DC exchange

The basic idea of routine DC exchange is to replace an open distribution center by a closed one, in an attempt to improve the current solution. The open DCs are sorted in descending order of their indexes *RDC*, while the closed ones are sorted in ascending order. The routine attempts to replace an open DC with a larger value of *RDC* by a closed one with a smaller value of *RDC*. Index RDC_w for each distribution center w is computed as follows:

$$RDC_w = \frac{\sum_{f \in F} \sum_{p \in P} CT^t_{fwp}}{NF_w} + \frac{\sum_{c \in C} \sum_{p \in P} CT^t_{wcp}}{NC_w} + \frac{CT^o_w}{Q_w} \tag{15}$$

where NF_w denotes the number of plants from where the products are transported to DC w, NC_w denotes the number of CZs served by DC w, and Q_w denotes the amount of products that uses DC w. The first term of Equation (15) refers to the average transportation cost per unit of products from plants to DC w, the second term determines the average transportation cost per unit of products from DC w to all served CZs, and the last term defines the fixed cost of DC w per unit of product.

The DC exchange procedure is outlined in Algorithm 3. Observe the RDC_w for a closed DC w is computed using values of NF_o, NC_o, and Q_o, associated with the open DC o that is a candidate to be exchanged with w.

Algorithm 3 DC Exchange

Step 1 Considering the current solution, compute *RDC* for open DCs.

Step 2 Set O as the sorted set of open DCs in decreasing order of RDC_w.

Step 3 For all $ow \in O$ do

- **Step 3.1** Compute RDC_w for all closed DCs, using NF_{ow}, NC_{ow}, and Q_{ow} in Equation (15).
- **Step 3.2** Set C as the sorted set of closed DCs in increasing order of RDC_w.
- **Step 3.3** For all $cw \in C$ do
  - **Step 3.3.1** If $(CAP_{cw} > CAP_{ow})$ then exchange the DCs in the current model, by closing ow and opening cw, creating a temporary model. Adjust the flow of products to the new open DC in the temporary model. Otherwise, go to Step 3.3.
  - **Step 3.3.2** Run RTM for the temporary model. If the total cost of the new model is smaller than the current solution then update the current solution and the current model. Go to Step 3.

Step 4 Return the current solution and model.

3.2.2 Arc exchange

The arc exchange procedure is based on an intuitive idea of replacing an active arc with a high value of index *RA* by another active arc with a smaller value of *RA*. Index RA_{wc} is computed by an arc $a = (w,c) | w \in W, c \in C$ as follows:

$$RAR_{wc} = \frac{\sum_{p \in P} CT^t_{wcp}}{NP_{wc}} + \frac{\sum_{p \in P} CT^g_{wp}}{QP_{wc}} + \frac{CT^o_w}{NC_w QP_{wc}} \tag{16}$$

where NC_w is the number of customer zones served by distribution center w, QP_{wc} is the amount of products transported from w to c, and NP_{wc} is the number of products send from DC w to CZ c. The first term of Equation 16 refers to the average unit cost transportation of arc (w,c).

The second term refers to the product handling costs divided by the number of used arcs serving product p through w. The last term considers the fixed cost of DC w divided by the number of used out-going arcs from w. Algorithm 4 outlines the arc exchange procedure.

Algorithm 4 Arc Exchange

Step 1 Considering the current solution and model, for all used arcs $(w,c), \forall w,c$ in the current solution compute RAR.

Step 2 Set U as the sorted set of used arcs in decreasing order of RAR.

Step 3 For all $u = (w,c) \in U$ do

Step 3.1 Set $n = (w',c') \leftarrow$ the last element of set U

Step 3.1.1 If $(CAP_{w'} > CAP_w)$ then exchange the arcs in the current model, creating a temporary model. Adjust the flow of products, transferring all product flows from arc (w,c) to arc (w',c'), and vice-versa in the temporary model. Otherwise, set $u \leftarrow Next(u)$ and go to Step 4.

Step 3.1.2 Run Model RTM with the new temporary model. If the total cost of the temporary model is smaller than the current solution then update the current solution and current model.

Step 3.1.3 Set $u \leftarrow Next(u)$ and delete n from set U.

Step 4 Return the current solution and current model.

4 COMPUTATIONAL EXPERIMENTS

In this section, we report the results of the computational experience obtained over some randomly generated instances. The algorithms were coded using C++, and the optimization engine COIN-OR (Computational Infrastructure for Operations Research) to solve the MILPs (Gassmann et al., 2016). The computational experiments were carried out on a Dell Precision T3600 Server using Xeon CPU ES5-1603 with 2.80 GHz and 16 GB RAM in the LINUX UBUNTU 14.04 LTS operational system. In the next paragraph, the computational tests and the analysis of the results are introduced.

Several instances representing different number of suppliers (V), raw materials (R), plants (F), DCs, CZs, and products (P) were generated for evaluating the developed solution method, a totaling of 25 instances. They are shown in Table 2. The instances were associated with a four-echelon network similar or larger than instances researched in the literature about strategic SCND. The cost structure of the instances has fixed costs, production, throughput costs and transportation costs associated with the capacity of plants and DCs. Every instances has been checked as being feasibles and solved independently in order to compare the performance of the proposed approach to solve it. Upper limits in the number of plants and DCs were not taken into account in the experiments.

The settings of the required parameters of the constructive phase of the multi-start heuristic was defined according to the size of each experimental instance, regarding following ranges. We disabled 1 to 2 plants. The limited number of plants did not recommend making this parameter very large. In contract, the number of DCs that should be disabled in the initial solution was defined as between 2 and 3. The percentage arcs disabled in the experimental tests were from

20% to 35%. Also, these percentages cannot be too large to preserve the main choices of the initial solutions.

Table 2 compares the results of the developed method with the heuristic presented in Farias et al. (2017), labeled Farias' Algorithm, for 25 instances. The relaxed version of each instance solved by COIN-OR was used as benchmark (column "B&B") for computing the solution gap presented in columns "Gap". The values of time limit (TL) presented in the table refers to the CPU time required to solve with gaps lower than 1% using this developed heuristic. This CPU limit was defined as the stopping condition for the heuristic developed by Farias et al. (2017), given the difficulties of defining a halting condition for this algorithm. The same CPU limit of 2 hours was used as the halting criteria for the new algorithm. Note that this is not the best criterion for the developed heuristic, but it was used to allow a fair comparison between the two algorithms.

Table 2 – Experiment results.

									Developed Algorithm		Farias' Algorithm	
I	V	R	F	DC	P	CZ	TL(s)	B&B	Solution	Gap	Solution	Gap
1	5	5	3	10	5	150	50	425108	427181	0.48%	435332	2.40%
2	5	5	3	20	5	150	70	158261	159772	0.95%	160637	1.50%
3	5	5	3	30	5	150	200	129608	130849	0.96%	133333	2.87%
4	5	5	3	40	5	150	300	19012505	19190346	0.93%	19226313	1.12%
5	5	5	3	20	10	170	150	1487115	1495372	0.56%	1495372	0.56%
6	5	5	3	10	10	150	100	292283	293185	0.30%	296266	1.36%
7	5	5	3	10	40	150	200	301464	298570	0.96%	297243	1.40%
8	5	5	3	10	100	150	200	25358151	25433384	0.29%	25476402	0.47%
9	5	5	3	20	10	150	100	319322	321423	0.65%	321476	0.67%
10	5	5	3	30	10	150	300	227527	229360	0.80%	236082	3.75%
11	5	5	3	40	10	150	100	297083	297740	0.22%	298715	0.54%
12	5	5	3	50	10	150	200	141258	141668	0.29%	142030	0.55%
13	5	5	3	20	10	150	200	249768	252119	0.94%	255173	2.16%
14	5	5	3	20	50	150	200	1306830	1312622	0.44%	1327892	1.61%
15	10	10	5	10	40	150	200	132466656	133347168	0.66%	134070925	1.21%
16	10	10	5	10	150	250	500	10954463	11063021	0.99%	11110108	1.42%
17	10	10	5	20	150	250	500	64622125	64754133	0.20%	64796606	0.27%
18	10	10	5	10	5	150	200	132769	133759	0.74%	134277	1.13%
19	5	5	3	5	150	250	250	8382428	8448909	0.79%	8512811	1.56%
20	25	25	20	15	20	270	200	1585938	1597115	0.70%	1617470	1.98%
21	20	20	20	10	15	200	100	20129199	20214454	0.42%	20214454	0.42%
22	15	15	10	5	10	150	300	8685207	8761766	0.88%	8761766	0.88%
23	20	20	5	5	170	300	400	14036308	14174991	0.98%	14206526	1.21%
24	5	3	1	5	3	250	200	180509	181017	0.28%	181878	0.75%
25	5	5	3	30	5	100	60	262573	263224	0.24%	269622	2.63%

Given the same CPU time to solve the instances for the two algorithms, the developed method outperformed Farias et al. (2017)'s algorithm in the quality of the solution. The developed approach obtained smaller or equal gaps, considering the optimal solution of the relaxed problem as benchmark, for all tested instances. The developed approach and Farias et al. (2017) heuristic obtained average gaps of 0.63% and 1.36%, respectively. Thus, the developed heuristic reduced on 115% the gap in comparison with the latter algorithm. In general, the CPU time increased following the dimensions of the instances. The developed approach required an average CPU time of 221.1s to solve the 25 instances with gaps smaller than 1%. The maximum required CPU time was 500s (instances 16 and 17). Although instance 20 and 21 present large dimensions in the number of suppliers, plants, and raw materials, they were solved very quickly. It seems that the number of products, DCs, and CZs are the most influential parameters in terms of the efficiency of the developed algorithm, see instances 15 and 16; and instances 2, 3, and 4. We could not identify any pattern concerning the dimensions of the instances with the obtained gap values. Additional experiments are required.

Table 3 compares the developed method with benchmark methods in the literature. It should be noted that the performance values cited in the table were those reported by the authors in their works, since the instances are not available for experimentation. In this table, it is possible to identify the largest instance tested in each research study. The heuristic approach had an average gap of 0.63% and a gap ranged between 0.20% and 0.99% for the 25 experimental instances. Thus, considering the tests shown in Table 3, the heuristic approach obtained an average gap less than the average gap of the literature review (0.63% versus 2.1%). When the results are individually compared, some small disadvantage were found in terms of minimum gaps to Vidal & Goetschalckx (1997) and Lee & Kwon (2010). However, it stands out that, except for Lee & Kwon (2010), all authors used computational test instances with smaller dimensions, mainly in the number of products. Moreover, some authors present experimental instances with three echelons. Both issues imply in less complexity for solving the considered instances.

In summary, our heuristic shows satisfactory results with smallest average and maximum gaps than previous developed methods.In particular, the new heuristic provides solutions within an acceptable optimal range for large-scale instances in reasonable computational time.

Table 3 – Comparison with previous models in the literature.

Author	Maximum Dimensions of the Instances						#Inst	Average Gap	Min Gap	Max Gap
	V	R	F	DC	P	CZ				
Vidal and Goetschalckx (2001)	50	35	8	10	12	80	5	1.53	0.02	3.78
Jayaraman and Pirkul (2001)	3	2	10	30	5	150	5	2.71	1.56	3.78
Lee and Kwon (2010)	-	-	10	20	80	30	12	3.95	0.00	15.06
Thanh et al. (2012)	27	27	22	13	18	270	15	0.96	0.20	3.00
Farias et al. (2017)	20	20	20	50	170	300	25	1.36	0.27	3.75
Developed Approach	20	20	20	50	170	300	25	0.63	0.20	0.99

5 CASE STUDY

Company X is one of the largest producer and distributor of tires in South America, but with a small presence in the Brazilian market. To increase the penetration of its products in Brazil, the largest tire market in South America, the company has built a new plant in Brazil, capable of producing not only the current line of products, but a new line of products customized to the diverse Brazilian market. Currently, the company is responsible for producing and distributing over 200 products. The set of products involved in this application consisted of 102 different kinds of tires and divided into the following two categories: (i) *Buses and Trucks (BT)* - 53 cross-ply or radial tires for trucks and buses; and (ii) *Agricultural and OTR (AOTR)* - 49 different tires for agricultural tractors and Off-The-Road (OTR) vehicles such as road machinery and earth movers.

A marketing research carried out by an independent consultancy revealed that inventory centralization in large warehouses was not working well for the replacement market of these two lines. The customers were complaining about the excessive replacement lead-time, making the vehicles either being out of operation or in dangerous operational conditions, waiting for a spare tire. Brazil is a huge country with some infra-structure problems in the countryside, where these two line of products are in high demand, mainly during the different harvest periods. Based on successive complaints, the marketing department of the company suggested a redesigning of the supply chain for these two lines, based both on a decentralization of the inventories and creating a space for customers to better interact with the company. The marketing department proposed a service center (SC) like concept. The idea is to have a space that can simultaneously act as a store, a warehouse, and a repair center for buses, farm tractors, and heavy construction vehicles. The resources of the SCs will be shared by selected partners to expand the services offered. However, the SCs will be located, constructed, and managed by the company. In the SCs, the customer will find several advantages either to repair or receive guidance on using the company and partners' branded products, as follows:

- Guaranteed to use original or certified parts by the company and partners when repairing their vehicle, which can increase the life of the vehicle after repair.
- Quality assurance of the repair carried out during some months, dependent on the product. offering more peace of mind about the quality of the technical assistance service received.
- Greater comfort: they have air-conditioned space, kids area, free use of tablet and free wi-fi while waiting to be attended.
- Experts at service, getting access to the best professionals, all trained and certified by on the domain when repairing products.

The idea is to have standardized SCs in several locations, close to high demand areas with easy access for customers of BT and AOTR products. The company wants to use an incremental strategy, initially implementing a limited number of SCs. If the concept is well accepted by customers, the company will slowly increase the number of SCs. Our problem is to determine the number, location, ans size of the first SCs to be build. A specialized construction company was hired to design the SCs, offering all required facilities of sales and storage in the same spot. The company has defined 35 possible SC locations, based on current operations and in the areas pointed out by the logistics department. Four options were presented to the logistics department of the company, as follows: tiny (with storage capacity of around 3600 tyres/year, small (with storage capacity of around 6000 tyres/year), medium (with storage capacity of around 9000 tyres/year), and large (with storage capacity of around 12000 tyres/year). The capacity is defined based on an weighted average size tyre, where the weights are the estimated demand of each tyre in the portfolio. The storage and fixed costs for the SCs were estimated by the logistic department based on historical data. Note that due to confidentiality issues, several data and information were intentionally disclosed.

The transportation costs from factories to possible SC locations, and from them to CZs were computed using geographical information systems. The fixed and variable costs of the warehouses and SCs were based on historical data from the company. In order to prioritize the most profitable markets, CZs have been established in a way that a major consumer city incorporated the demand of surrounding smaller cities, towns, or rural zones. Note that some small cities, in terms of population, have a large demand for a specific type of tire for being located in very important agricultural or mining zones. In addition, it was decided that only CZs with demand exceeding 1,000 units per year will be considered in the analysis. Considering the location criteria, 270 customer zones were identified. The specific problem to be solved has involved three levels, as follows: (i) one factory (already constructed); (ii) 4 large warehouses (already constructed) + 35 possible SC locations (to be defined); and (iii) and 270 CZs. Figure 1 illustrates the specific network to be designed.

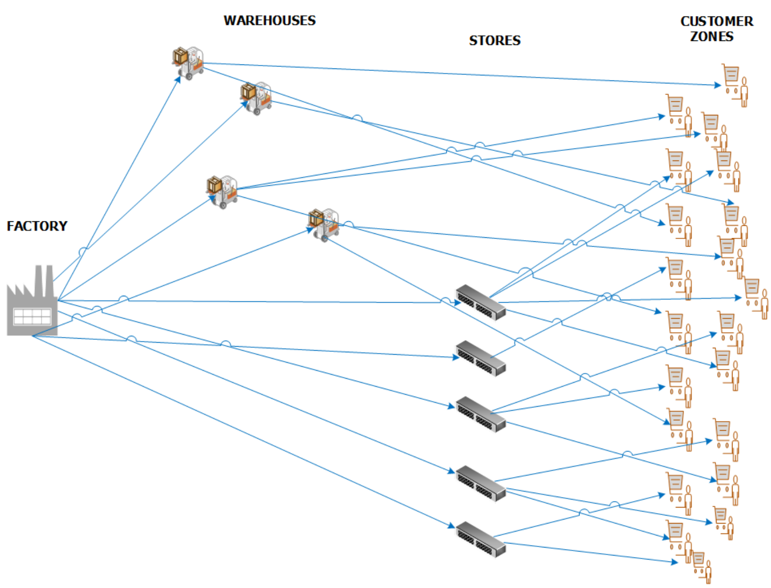

Figure 1 – Supply chain network lay-out for the case study.

The following parameters of the algorithm were used in the case study. A gap of less than 1% was defined as the halting condition. In general, the algorithm has obtained such gap in less than 200s of CPU time. In the multi-start mechanism, two warehouses and 25% of the assigned arcs between warehouses and CZs were disabled at each iteration. A planning horizon of one year was chosen.

Table 4 presents the main results concerning the investigated scenarios, combining the number of initial SCs to be implemented (column '#SCs') and the estimated annual capacity designed by the construction company (column 'Capacity'). These scenarios were defined by the company managers. The next columns show the models results in terms of the objective function (column 'Solution'), and service centers' performance measures, namely: the percentage of total CZs assigned to the SCs (column 'CZ(%)'), the percentage of the total demand that will pass through SCs (column 'Flow(%)'), instead of the large warehouse, and the average percentage of the capacity utilization of all SCs (column 'AU(%)'). The first line of the table refers to a scenario without SCs.

Table 4 – Service center results of the initially analyzed scenarios.

Scenario	#SCs	Capacity	Solution	CZ (%)	Flow(%)	AU (%)
1	–	–	296597	–	–	–
2	5	3600	287563	8.15	8.14	100.00
3	5	6000	273536	15.93	14.3	95.34
4	5	9000	287681	10.37	16.02	74.78
5	5	12000	289169	22.22	27.73	95.67
6	10	3600	442257	18.52	15.81	91.17
7	10	6000	413694	27.41	27.82	97.42
8	10	9000	416767	21.48	23.71	54.07
9	10	12000	1319856	55.18	56.29	97.59
10	15	3600	438304	25.55	21.80	84.25
11	15	6000	413952	31.11	30.89	70.71
12	15	9000	470563	49.26	50.55	78.67
13	15	12000	1784654	84.44	84.82	97.48

The results in Table 4 indicate that the use of a small number of SCs can lead to an effective process of decentralization, including with reduction in the value of the total cost of the network. With 5 SCs, and for all SCs' capacities, the total costs have been slightly decreased. The variable costs has compensated the increase in the fixed costs of implementing SCs. The transportation costs are quite high for the tyres in these two lines. As the number of SCs increase, the fixed costs overcame the variable costs, significantly increasing costs for 10 and 15 SCs. The company managers expect that this increase of costs might be compensated by an increase in the sales, justifying the expansion of the pilot project. Unfortunately, it is difficult to compute profit due to the lack of reliable data on sales.

Clearly, scenarios 9 and 13, with capacities above 9000 tyre/year, significantly increased the costs and were immediately discarded. As the implementation of SCs was mainly motivated by the ATOR line, the company managers have found average utilization of SCs above 90% very risky in terms of properly answering the peak demands caused by harvesting different crops seasonality, additionally discarding scenarios 2, 3, 5, 6, and 7. Also, the low average utilization of scenario 8 was considered inappropriate in terms of an effective decentralization. Finally, the high costs associated with scenarios 10, 11, and 12 led the decision to scenario 4, which combines an acceptable cost, a reasonable average utilization, and a service to CZs consistent with the incremental strategy of the SCs implementation.

The analysts have proposed the evaluation of additional scenarios to the company managers, taking scenario 4 as basis. Several new scenarios were interactively analyzed, since the running time was around 4 minutes, considering different number of SCs and capacities. However, there were a consensus that an increase in the number of SCs was problematic due to the effects in the total costs of the supply chain. For economy's sake, we present a limited number of additional scenarios. Table 5 presents the results of some interesting elected additional scenarios by the company managers towards making an effective decision making process.

Table 5 – Service center results of some extra analyzed scenarios.

Scenario	#SCs	Capacity	Solution	CZ (%)	Flow(%)	AU (%)
14	4	9000	290177	7.04	10.56	61.62
15	5	7000	287866	4.67	6.29	87.06
16	5	8000	287960	13.70	15.87	97.93
17	5	10000	297507	17.78	17.08	71.76
18	5	11000	299829	20.37	20.38	77.84

On the one hand, scenario 14 presents a very low average utilization of the SCs, indicating that 4 SCs might be a low number in terms of the objectives to the company. On the other hand, scenario 16 has a very high average utilization. Scenario 15 presents a low percentage of both served CZs and storage products, having a little impact in the decentralization strategy. Scenarios 18 dominates scenario 17, presenting better average percentages of served CZs, manipulated products by the SCs, and a similar average utilization, but with a lower total costs. Scenarios 4 and 18 present similar results. The former presents a slightly smaller cost, while the percentage of CZs and products using the SCs is higher. However, in case of future expansion, the high capacity of scenario 18 can lead to high costs and excessive use of SCs, as indicated by the results of scenarios 9 and 13. Based on the analysis process, the managers have defined scenario 4 as the most appropriate one to be implemented following the incremental strategy. Figure 2 illustrates the case study solution.

The company managers have positively responded to the modeling approach. The main advantage of applying such approach is the quick generation and solution of possible several scenarios. Particularly, the analysis and evaluation of possible SCND different configurations, have pro-

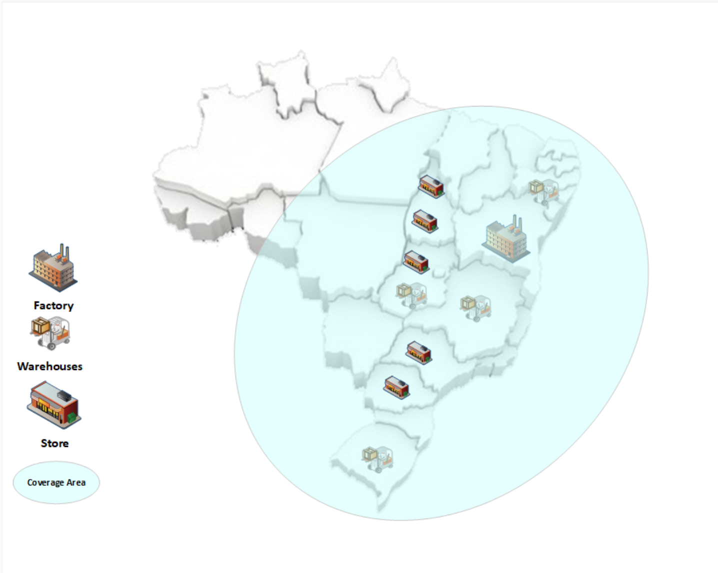

Figure 2 – Scenario 4 solution.

vided a means to perform a complete study, resulting in a better and quicker decision making process. The flexibility of the optimization approach, allowing the incorporation of peculiarities of a specific problem, and the efficiency and efficacy of the optimization approach were highly praised by the managers.

6 CONCLUSIONS

This paper presents a heuristic approach to solve very large instances of the strategic SCND, consisting of a four-echelon system. The main objective is to minimize the total costs in the network. The heuristic is a two-step iterative algorithm based on a multi-start/layered-based rounding constructive approach, and a local search for solution improvement. The heuristic was first evaluated using 25 randomly generated instances of different dimensions, some of them quite large, in terms of the number of products, DCs, and CZs. The heuristic has obtained very good optimal gaps for all tested instances, within acceptable CPU times. Particularly, the heuristic was able to solve very large instances with similar gaps than the ones found for smaller ones. Next, we describe the application of the developed algorithm for a case study, illustrating the flexibility of the optimization approach in helping managers to solve large real world problems concerning the supply chain network design.

Considering the contributions of the heuristic approach developed in this study for the SCND, future studies could include the following: (i) extension of the model and solution approach to consider a multi-period horizon; (ii) integration of strategic inventory decisions and stochastic data (e.g. demand uncertainty); and (iii) extension of the model to consider multimodal transportation.

References

ALTIPARMAK F, GEN M & NAD T PARKSAY LL. 2006. A genetic algorithm approach for multi-objective optimization of supply chain networks. *Computers and Industrial Engineering*, **51**: 196–215.

ARMENTANO VA, SHIGUEMOTO A & LØKKETANGEN A. 2011. Tabu search with path relinking for an integrated production–distribution problem. *Computers & Operations Research*, **38**(8): 1199–1209.

DER VAAT TV & DONK DPV. 2008. A critical review of survey-based research in supply chain integration. *International Journal of Production Economics*, **111**: 42–55.

ESKANDARPOUR M, DEJAX P, MIEMCZYK J & PÉTON O. 2015. Sustainable supply chain network design: An optimization-oriented review. *Omega*, **54**: 11–32.

FARAHANI RZ, REZAPOUR S, DREZNER T & FALLAH S. 2014. Competitive supply chain network design: An overview of classifications, models, solution techniques and applications. *Omega*, **45**: 92–118.

FARIAS E, LI JQ, GALVEZ JP & BORENSTEIN D. 2017. Simple heuristic for the strategic supply chain design of large-scale networks: A Brazilian case study. *Computers & Industrial Engineering*, **113**: 746–756.

FATTAHI M, GOVINDAN K & KEYVANSHOKOOH E. 2018. A multi-stage stochastic program for supply chain network redesign problem with price-dependent uncertain demands. *Computers & Operations Research*, **100**: 314–332.

GARCÍA-LÓPEZ F, MELIÁN-BATISTA B, MORENO-PÉREZ JA & MORENO-VEGA JM. 2010. The Parallel Variable Neighborhood Search for the p-Median Problem. *Journal of Heuristics*, **8**(3): 375–388. Available at: http://dx.doi.org/10.1023/A:1015013919497.

GASSMANN H, MA J & MARTIN K. 2016. Communication protocols for options and results in a distributed optimization environment. *Mathematical Programming Computation*, **8**(2): 161–189.

GHASEMIAN ZARINI F & JAVADIAN N. 2020. A multi objective mixed integer programming model for design of a sustainable meat supply chain network. *Journal of Industrial and Systems Engineering*, **13**: 78–92.

GHOLAMI F, PAYDAR MM, HAJIAGHAEI-KESHTELI M & CHERAGHALIPOUR A. 2019. A multi-objective robust supply chain design considering reliability. *Journal of Industrial and Production Engineering*, **36**(6): 385–400.

GOETSCHALKX M, VIDAL C & DOGAN K. 2002. Modeling and design of global logistics systems: a review of integrated strategic and tactical models and design algorithms. *European Journal of Operational Research*, **143**(1): 1–18.

GOVINDAN K, JAFARIAN A & NOURBAKHSH V. 2015. Bi-objective integrating sustainable order allocation and sustainable supply chain network strategic design with stochastic demand using a novel robust hybrid multi-objective metaheuristic. *Computers & Operations Research*, **62**: 112 – 130. Available at: http://www.sciencedirect.com/science/article/pii/S030505481500009X.

JAYARAMAN V & PIRKUL H. 2001. Planning and coordination of production and distribution facilities for multiple commodities. *European Journal of Operational Research*, **133**: 394–408.

KEYVANSHOKOOH E, FATTAHI M, SEYED-HOSSEINI S & TAVAKKOLI-MOGHADDAM R. 2013. A dynamic pricing approach for returned products in integrated forward/reverse logistics network design. *Applied Mathematical Modelling*, **37**(24): 10182–10202.

KEYVANSHOKOOH E, RYAN SM & KABIR E. 2016. Hybrid robust and stochastic optimization for closed-loop supply chain network design using accelerated Benders decomposition. *European Journal of Operational Research*, **249**(1): 76 – 92. Available at: http://www.sciencedirect.com/science/article/pii/S0377221715007833.

KHALIFEHZADEH S, SEIFBARGHY M & NADERI B. 2015. A four-echelon supply chain network design with shortage: Mathematical modeling and solution methods. *Journal of Manufacturing Systems*, **35**: 164–175.

KUMAR SK & TIWARI M. 2013. Supply chain system design integrated with risk pooling. *Computers & Industrial Engineering*, **64**(2): 580 – 588. Available at: http://www.sciencedirect.com/science/article/pii/S036083521200294X.

LEE YH & KWON SG. 2010. The hybrid planning algorithm for the distribution center operation using tabu search and decomposed optimization. *Expert Systems with Applications*, **37**: 3093–3103.

MELO M, NICKEL S & DA GAMA FS. 2014. An efficient heuristic approach for a multi-period logistics network. *TOP*, **22**(1): 80–108.

MELO M, NICKEL S & SALDANHA-DA GAMA F. 2012. A tabu search heuristic for redesigning a multi-echelon supply chain network over a planning horizon. *International Journal of Production Economics*, **136**(1): 218–230.

PHAM T & YENRADEE P. 2017. Optimal supply chain network design with process network and BOM under uncertainties: A case study in toothbrush industry. *Computers & Industrial Engineering*, **108**: 177 – 191. Available at: http://www.sciencedirect.com/science/article/pii/S0360835217301602.

ROMEIJN HE, SHU J & TEO CP. 2007. Designing two-echelon supply networks. *European Journal of Operational Research*, **178**(2): 449–462.

SOLEIMANI H & KANNAN G. 2015. A hybrid particle swarm optimization and genetic algorithm for closed-loop supply chain network design in large-scale networks. *Applied Mathematical Modelling*, **39**(14): 3990–4012.

THANH PN, BOSTEL N & PÉTON O. 2012. A DC programming heuristic applied to the logistics network design problem. *International Journal of Production Economics*, **135**(1): 94 – 105.

VIDAL CJ & GOETSCHALCKX M. 1997. Strategic production-distribution models: A critical review with emphasis on global chain models. *European Journal of Operational Research*, **98**: 1–18.

WU SD & GOLBASI H. 2004. Multi-item, multi-facility supply chain planning: Models, complexities, and algorithms. *Computational Optimization and Applications*, **28**(3): 325–356.

How to cite
GALVEZ JLAP AND BORENSTEIN D. 2023. Large scale supply chain network design: An effective heuristic approach. *Pesquisa Operacional*, **43**: e269080. doi: 10.1590/0101-7438.2023.043.00269080.