

## Article

# Matheuristics for the Design of a Multi-Step, Multi-Product Supply Chain with Multimodal Transport

David A. Ruvalcaba-Sandoval <sup>1</sup>, Elias Olivares-Benitez <sup>1</sup>, Omar Rojas <sup>2,3</sup> and Guillermo Sosa-Gómez <sup>2,\*</sup>

<sup>1</sup> Facultad de Ingeniería, Universidad Panamericana, Álvaro del Portillo 49, Zapopan, Jalisco 45010, Mexico; 0179018@up.edu.mx (D.A.R.-S.); eolivaresb@up.edu.mx (E.O.-B.)

<sup>2</sup> Facultad de Ciencias Económicas y Empresariales, Universidad Panamericana, Álvaro del Portillo 49, Zapopan, Jalisco 45010, Mexico; orojas@up.edu.mx

<sup>3</sup> Faculty of Economics and Business, Universitas Airlangga, Surabaya, East Java 60286, Indonesia

\* Correspondence: gsosag@up.edu.mx; Tel.: +52-33-1368-2200

**Abstract:** Supply-chain network design is a complex task because there are many decisions involved, and presently, global networks involve many actors and variables, for example, in the automotive, pharmaceutical, and electronics industries. This research addresses a supply-chain network design problem with four levels: suppliers, factories, warehouses, and customers. The problem considered decides on the number, locations, and capacities of factories and warehouses and the transportation between levels in the supply chain. The problem is modeled as a mixed-integer linear program. The main contribution of this work is the proposal of two matheuristic algorithms to solve the problem. Matheuristics are algorithms that combine exact methods and heuristics, attracting interest in the literature because of their fast execution and high-quality solutions. The matheuristics proposed to select the warehouses and their capacities following heuristic rules. Once the warehouses and their capacities are fixed, the algorithms solve reduced models using commercial optimization software. Medium and large instances were generated based on a procedure described in the literature. A comparison is made between the algorithms and the results obtained, solving the model with a time limit. The algorithms proposed are successful in obtaining better results for the largest instances in shorter execution times.

**Keywords:** matheuristic; multimodal transport; supply-chain network design; decomposition



**Citation:** Ruvalcaba Sandoval, D.; Olivares-Benitez, E.; Rojas, R.; Sosa-Gómez, G. Matheuristics for the Design of a Multi-Step, Multi-Product Supply Chain with Multimodal Transport. *Appl. Sci.* **2021**, *11*, 10251. <https://doi.org/10.3390/app112110251>

Academic Editors: Pal Varga, Daniel Kozma and Felix Larrinaga

Received: 10 September 2021

Accepted: 26 October 2021

Published: 1 November 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

An optimal supply chain is a fundamental part of any company's success; a good design and administration represent a competitive advantage or even a requirement for market participation. Supply chains represent a large part of a company's assets. Additionally, costs or savings are dependent on their design. Among the advantages obtained by a good design of the supply chain are reduced purchase costs, reduced production costs, increased company profits, reduced investment in fixed costs, and increased cash flow, among others.

In the end, the main objective of a supply chain is to provide an efficient way to supply the products to the client at the lowest possible cost (Council of Supply-Chain Management Professionals (CSCMP), 2019). However, within a supply chain, a large number of costs are incurred. Additionally, within large companies, supply chains are becoming increasingly complex. Knowing which is the best option among all the possible combinations becomes quite a complex task. The global production and distribution networks involve many actors and variables, exploding the combinatorial nature of the decisions involved, especially when more characteristics are considered, such as location, transportation, inventory, product architecture, or sustainability considerations. These networks exist in industries such as pharmaceutical, automotive, and electronic products, with suppliers, plants, and customers in different continents.

One way to address the decision-making problem in the design of supply chains has been to propose optimization models based on mathematical programming. As long as computers increase power, it has been possible to solve more complex models involving more elements. However, the mixed-integer linear programming models that have been widely used in the optimization of the supply chain are mostly NP-hard [1], making it impossible to obtain optimal solutions in reasonable times for instances of size similar to those found in real problems. Fortunately, the ability of computers to calculate results for different scenarios brought the opportunity to use heuristics and randomness to help in the construction of solutions for NP-hard combinatorial problems. The first results showed that although these solutions had variable and worse quality than those obtained with exact methods, the computational effort required shorter times. With the goal of improving the quality of these solutions, the research on metaheuristic algorithms was born. Many advances were achieved with algorithms such as simulated annealing, tabu search, ant colony optimization, genetic algorithms, and others. Recently, new algorithms are being proposed, combining heuristics and exact mathematical programming methods, known as matheuristics. The research in this field is new, and the goal is to combine the speed of processing using heuristic components with the quality obtained with exact methods. Thus, the main contribution of the work presented in this paper is to advance in the research of matheuristics, proposing two algorithms to solve a complex problem efficiently for supply-chain network design with good quality solutions in a reasonable time.

In this work, a supply-chain network design (SCND) problem will be presented, where a literature review of both the problem and the methods of solution are shown first; then, the problem is described in detail. A mixed-integer linear programming model is presented, and the matheuristics are used to solve the same problem. The instances presented range from 100 to 200 clients, and the results of the mixed-integer linear programming model will be analyzed and compared with those of the matheuristic algorithms proposed. Finally, the conclusions will be presented in the last section.

## 2. Literature Review

This section will describe some of the literature, first related to the problem of supply-chain network design, and later, related to matheuristics, commenting about the gap covered by the research presented in this paper.

Pirkul and Jayaraman [2] created a model called PLANWAR, which focused on optimizing the supply chain through a heuristic to decide on the opening of plants and warehouses and the flow between them. However, the model does not handle different levels of capacity between facilities nor multimodal transportation.

Wu et al. [3] solved the supply-chain planning problem where the same product can be produced in multiple facilities, but their work focuses more on analyzing different algorithms and the complexity of each one of them. In their problem of supply-chain design, Eskigun et al. [4] consider delivery times and transportation modes, but their work focuses mainly on outbound logistics and is not multi-tier.

Sadjady and Davoudpour [5] solved a multi-product supply-chain problem using a mixed-integer linear programming model in which the opening of facilities is decided, as well as the level of capacity and the mode of transport of flows between levels of the supply-chain. However, the work takes into account only finished products. Olivares-Benitez et al. [6] focus on optimizing the transportation of a two-tier supply chain, calculating the flow and time between facilities using bi-objective optimization. The problem addressed in this work is for a single product. Rahmaniani and Ghaderi [7] developed a mixed-integer linear programming model and a heuristic based on the evolutionary algorithm of the firefly, where transport and construction costs are taken into account, but their solution method focuses on problems in telecommunications or power distribution companies.

Bertazzi et al. [8] developed min-max methods and a heuristic to solve the multi-tier inventory problem taking into account purchasing, manufacturing, and transportation

costs, but they do not address the issue of opening of facilities. Additionally, they do not take into account the bill of materials.

Many more recent models address the problem of supply-chain network design with different considerations, for example, ref. [9] describe a problem with environmental and financial considerations, or [10] analyze a model considering inventory decisions besides the classic decisions for location and transportation. However, the research in the literature is far from a general model for supply-chain network design that includes all the situations, variables, and decisions. Our purpose was to analyze a problem with a high level of complexity to be a challenge for new methods of solution. In particular, very few models consider the product architecture represented in the bill of materials. One of the reasons is that introducing this element creates an explosion in the number of variables and interactions, making it very hard to solve even small instances. As can be observed in the literature presented, the most used solution methods are based on metaheuristics because of the computational complexity of the models used. We are proposing new methods of solution into the field of matheuristics, which rarely has been used in SCND.

The term matheuristic is relatively new; the term began to be used between 2008 and 2009. Maniezzo et al. [11] talk about the hybridization that can exist between mathematical programming and metaheuristics. Fischetti and Fischetti [12] explain that matheuristics exploit heuristics and metaheuristics to improve and facilitate the mixed-integer programming (MIP) model. They show three applications: optimization of layout, packaging, and routing. Moreover, there is a rising interest in these hybrid methods, demonstrated by a growing literature and specialized tutorials in operations research [13]. Matheuristics have been applied to different decision-making problems, for example in routing [14], production planning [15], lot sizing [16], and risk management [17], among others. However, we found in the literature few applications of matheuristics to supply-chain network design.

Boschetti et al. [18] solve a Single-Source Capacitated Facility Location problem (SCFLP) using different matheuristics. Raa et al. [19] use a matheuristic to solve their aggregate production-distribution problem, but the work is focused on mould-sharing between factories. Tautenhain et al. [20] use the combination of a heuristic called MathFix and a matheuristic called AugMathFix to solve a bi-objective model for sustainable supply-chain design. However, the model has only two tiers: suppliers-plant and plant-clients. Cantú et al. [21] propose a matheuristic for the design of sustainable hydrogen supply chains using a multi-objective perspective, but the model is focused on a single product and transportation mode. Souto et al. [22] propose a matheuristic algorithm to solve the problem of supply-chain network design only with two levels, a single transportation mode, and a single product.

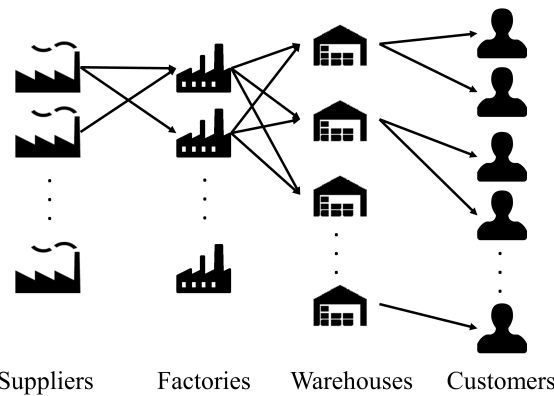
Table 1 describe the features for the literature presented in this section. It can be noted that the problem we are solving in this work has a higher level of complexity than other problems solved in the literature using matheuristics. The elements that add this complexity beyond the classic decisions on location and transportation are: a hierarchical product architecture as described by a bill of materials, different capacities in the facilities, and different available transportation modes. The complexity of the problem represents a challenge for any solution method, and is a good choice to demonstrate the efficiency of the matheuristic algorithms proposed.

**Table 1.** Description the features for the literature presented.

	Facilities Opening	Transportation Cost	Multi-Mode Transportation	Multi-Product	BOM	4-Levels	Facility Capacity Levels	Matheuristics
[2]	X	X						
[3]	X	X		X				
[4]		X	X					
[5]	X	X	X	X			X	
[6]	X	X	X					
[7]	X	X	X			X		
[18]	X	X						X
[19]		X						X
[20]	X	X	X				X	X
[21]	X	X	X			X	X	X
[22]	X	X						X
Our proposal, based on [23]	X	X	X	X	X	X	X	X

### 3. Problem Description

This paper addresses the problem of designing a four-tier multi-product supply-chain network (SCND): suppliers, factories, warehouses, and customers (Figure 1). The number and locations of plants and warehouses must be chosen from a set of potential plants and warehouses, respectively, and the capacity level of each factory and each warehouse must be chosen from a set of predetermined capacity levels for each location. Each location with each of its capacity levels has a fixed opening cost.



**Figure 1.** Representation of the supply chain posed in the problem.

Each of the materials can be supplied by a set of suppliers that have this material. Each material has a different purchase price for each supplier. Materials are shipped to factories, where they are converted into finished products. Each finished product has its bill of materials (BOM), which states what materials are needed for each finished product and how many units, so all the materials needed to produce a finished product will have to be taken to the factory where it is produced. Figure 2, shows the representation of the BOM; it exemplifies that to produce product  $p_1$ , 4 units of material  $r_1$  and 3 units of material  $r_2$  are needed. The generic description is for any product  $p$  the necessary units of material  $r$  are given by the parameter  $A^r p$ .

Each product can be manufactured in a pre-established set of factories. Each of the products has a manufacturing cost in each plant. Once the products are manufactured, they can be taken to a set of pre-established warehouses. Finished products must be supplied

to each customer in a single delivery; that is, products cannot be shipped to the customer from two different warehouses.

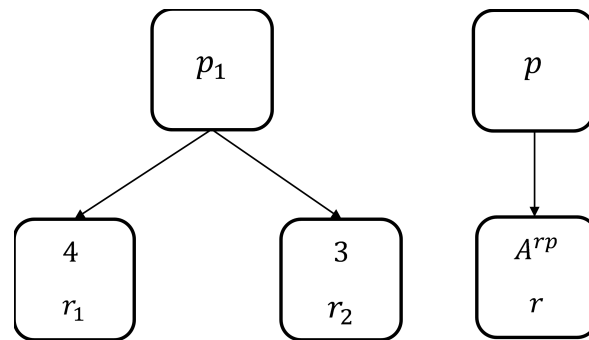


Figure 2. Representation of the BOM of the products.

All materials and finished products can be transported by different transport methods for each destination-origin pair, which can be the following: supplier-factory, factory-warehouse, warehouse-customer. Each product or material has a transport cost for each means of transport for each possible pair. Each means of transportation for each origin-destination pair has a minimum quantity to be transported and a maximum transport capacity.

Each supplier has a maximum capacity to supply each material. Each product occupies a different manufacturing or storage capacity. Each factory has a maximum production capacity for each product, a total maximum production capacity, and a minimum production capacity so that the factory can be opened for each of its capacity levels. Each warehouse has a maximum storage capacity and a minimum of storage to be opened.

The objective of the supply-chain design is to minimize the sum of the fixed cost by opening factories and warehouses, the total purchasing costs, the total manufacturing costs, and the total transportation costs.

This supply-chain design can be applied to global networks of manufacturing and distribution in industries such as pharmaceuticals, automotive, and electronics. For example, in the production of drugs, many ingredients can be blended, purchased from suppliers in different countries, and distributed to customers in other continents using alternative transportation modes. A potential example is to have suppliers in Asia, shipping materials to plants in Europe to manufacture products, to be transported to markets in the US, Canada, and Brazil. Similar cases can be devised to manufacture vehicles, tablets, cell phones, computers, or vaccines.

Some assumptions of the model that may represent limitations to its applicability are:

1. It is a single-period problem that usually is applied to long-term planning.
2. The model is deterministic and is not considering variability, although being fast to solve is useful for what-if analysis.
3. The model is not incorporating other operational and strategic elements such as routing, sustainability, inventory, service level, etc.

#### 4. MILP Model and Matheuristics

##### 4.1. Mixed-Integer Linear Programming Model (MILP)

Following is the notation used for the MILP, based on the model of [23].

##### 4.1.1. Decision Variables

###### Binary Variables

$$z_{fq} = \begin{cases} 1 & \text{if it is decided to open a factory } w \text{ with a capacity level } q, (f \in F; q \in Q) \\ 0 & \text{other case} \end{cases}$$

$$z_{wq} = \begin{cases} 1 & \text{if it is decided to open a warehouse } w \text{ with a capacity level } q, (w \in W; q \in Q) \\ 0 & \text{other case} \end{cases}$$

$$v_{sf}^r = \begin{cases} 1 & \text{if material } r \text{ is supplied to factory } f \text{ by a supplier } s, (s \in S; f \in F; r \in R) \\ 0 & \text{other case} \end{cases}$$

$$u_{od}^t = \begin{cases} 1 & \text{if origin } o \text{ supplies destination } d \text{ by mode of transport } t, \\ & ((o, d) \in \{(S, F) \cup (F, W) \cup (W, C)\}; t \in T) \\ 0 & \text{other case} \end{cases}$$

*Real Variables*

$x_{sf}^{rt}$ : Amount of material  $r$  that is supplied by supplier  $s$  to factory  $f$  by mode of transport  $t$ ,  
 ( $s \in S; f \in F; r \in R; t \in T$ )

$x_{fw}^{pt}$ : Amount of product  $p$  that is sent from factory  $f$  to warehouse  $w$  by mode of transport  $t$ ,  
 ( $f \in F; w \in W; p \in P; t \in T$ )

4.1.2. Objective Function

The objective function (1) is to minimize the sum of all the costs that we are taking into account in the model. The first term adds the fixed costs of opening a factory, the second, the fixed costs of opening a warehouse, the third the purchase and transportation costs from the supplier to the factories of the materials, the fourth, the cost of manufacturing and transportation from the factories to the warehouses and the fifth the transportation from the warehouses to the customers.

$$\begin{aligned} \text{Min} \sum_{f \in F} \sum_{q \in Q} FC_{fq} z_{fq} + \sum_{w \in W} \sum_{q \in Q} FC_{wq} z_{wq} + \sum_{r \in R} \sum_{s \in S^r} \sum_{f \in F} \sum_{t \in T_{sf}} (PC_s^r + TC_{sf}^{rt}) x_{sf}^{rt} + \\ \sum_{p \in P} \sum_{f \in F^p} \sum_{w \in W^p} \sum_{t \in T_{fw}} (MC_f^p + TC_{fw}^{pt}) x_{fw}^{pt} + \\ \sum_{p \in P} \sum_{c \in C^p} \sum_{w \in W^p} \sum_{t \in T_{wc}} DEM_c^p TC_{wc}^{pt} u_{wc}^t \end{aligned} \quad (1)$$

4.1.3. Constraints

*Supplier constraints*

Constraint (2) limits the order quantity to a material supplier's maximum capacity, constraint (3) prevents a factory ordering a specific material from more than one supplier and constraint (4) ensures that not only can the material be transported from a supplier to a factory if we select that supplier.

$$\sum_{f \in F} \sum_{t \in T_{sf}} x_{sf}^{rt} \leq SQR_s^r \quad \mathbf{r} \in \mathbf{R}, \mathbf{s} \in \mathbf{S}^r \quad (2)$$

$$\sum_{s \in S^r} v_{sf}^r \leq 1 \quad \mathbf{r} \in \mathbf{R}, \mathbf{f} \in \mathbf{F} \quad (3)$$

$$\sum_{t \in T_{sf}} x_{sf}^{rt} \leq SQR_s^r v_{sf}^r \quad \mathbf{r} \in \mathbf{R}, \mathbf{s} \in \mathbf{S}^r, \mathbf{f} \in \mathbf{F} \quad (4)$$

*Factory and warehouse constraints*

Constraint (5) ensures that each potential location can only be opened with one level of capacity. Constraint (6) ensures that the amount of material ordered from the supplier is exactly what is needed to produce the finished products. Constraints (7) and (8) limit

production to the minimum and maximum quantities in general of each factory and also limit the maximum capacity per product, respectively. Finally, constraint (9) imposes the lower and upper limits of storage in each warehouse.

$$\sum_{q \in Q} z_{lq} \leq 1 \quad l \in F \cup W \quad (5)$$

$$\sum_{s \in S^r} \sum_{t \in T_{sf}} x_{sf}^{rt} = \sum_{p \in P^r} \sum_{w \in W^p} \sum_{t \in T_{fw}} A^{rp} x_{fw}^{pt} \quad (6)$$

$$\sum_{q \in Q} \underline{MQ}_{fq} z_{fq} \leq \sum_{p \in P} \sum_{w \in W^p} \sum_{t \in T_{fw}} MQU^p x_{fw}^{pt} \leq \sum_{q \in Q} \overline{MQ}_{fq} z_{fq} \quad f \in F \quad (7)$$

$$\sum_{w \in W^p} \sum_{t \in T_{fw}} MQU^p x_{fw}^{pt} \leq \sum_{q \in Q} MQP^p_{fq} z_{fq} \quad p \in P, f \in F \quad (8)$$

$$\sum_{q \in Q} \underline{SQ}_{wq} z_{wq} \leq \sum_{p \in P} \sum_{c \in C^p} \sum_{t \in T_{wc}} SQU^p DEM_c^p u_{wc}^t \leq \sum_{q \in Q} \overline{SQ}_{wq} z_{wq} \quad w \in W \quad (9)$$

*Customer and flow retention constraints*

Constraint (10) ensures that a customer is only dispatched from a warehouse and constraint (11) ensures that the quantity of product that reaches customers is the same as that which reaches the warehouses.

$$\sum_{w \in W} \sum_{t \in T_{wc}} u_{wc}^t = 1 \quad c \in C \quad (10)$$

$$\sum_{c \in C^p} \sum_{t \in T_{wc}} DEM_c^p u_{wc}^t = \sum_{f \in F^p} \sum_{t \in T_{fw}} x_{fw}^{pt} \quad p \in P, w \in W \quad (11)$$

*Transport related constraints*

Constraints (12)–(14) limit the minimum and maximum quantities to transport between origins and destinations.

$$\underline{TQ}_{sf}^t u_{sf}^t \leq \sum_{r \in R} TQU^{rt} x_{sf}^{rt} \leq \overline{TQ}_{sf}^t u_{sf}^t \quad s \in S, f \in F, t \in T_{sf} \quad (12)$$

$$\underline{TQ}_{fw}^t u_{fw}^t \leq \sum_{p \in P} TQU^{pt} x_{fw}^{pt} \leq \overline{TQ}_{fw}^t u_{fw}^t \quad f \in F, w \in W, t \in T_{fw} \quad (13)$$

$$\underline{TQ}_{wc}^t u_{wc}^t \leq \sum_{p \in P} TQU^{pt} DEM_c^p u_{wc}^t \leq \overline{TQ}_{wc}^t u_{wc}^t \quad w \in W, c \in C, t \in T_{wc} \quad (14)$$

*Binary and non-negativity constraints*

$$z_{fq} \in \{0, 1\} \quad f \in F, q \in Q \quad (15)$$

$$z_{wq} \in \{0, 1\} \quad w \in W, q \in Q \quad (16)$$

$$v_{sf}^r \in \{0, 1\} \quad s \in S, f \in F, r \in R \quad (17)$$

$$x_{fw}^{pt} \in \mathbb{R}_{\geq 0} \quad f \in F, w \in W, p \in P, t \in T \quad (18)$$

$$x_{sf}^{rt} \in \mathbb{R}_{\geq 0} \quad s \in S, f \in F, r \in R, t \in T \quad (19)$$

$$u_{od}^t \in \{0, 1\} \quad (o, d) \in \{(S, F) \cup (F, W) \cup (W, C)\}, t \in T \quad (20)$$

*4.2. Matheuristics*

The model described above can be reduced to the single-source capacitated facility location problem [23] and therefore it belongs to the class of NP-hard problems. Thus, a heuristic method is justified, especially to solve large instances. The matheuristics consist of the steps described in Algorithms 1–3. The algorithm described in Algorithm 1 is the general one. The problem is divided in two parts. In the first part the warehouses are opened

following one of two methods: simple or selective (with probability). Once the decision for the warehouses was fixed, an allocation sub-problem is solved using commercial optimization software. The sub-problem is described in Section 4.2.1. This sub-problem helps to determine the flows between the warehouses and the customers. Finally, a second sub-problem is solved considering the decisions fixed previously. This sub-problem is described in Section 4.2.2 as a reduced supply-chain network design (SCND) problem. Here, the remaining non-fixed variables are solved.

The algorithm described in Algorithm 2 helps to open warehouses with a certain capacity using a simple, random selection. The process finishes when the aggregated capacity is greater than the total demand. This method is reported as “Simple Heuristic”.

The other method for warehouse selection is described in Algorithm 3. In this algorithm, an estimated cost is calculated for each warehouse. This cost is a combination of the fixed cost divided by the capacity of the facility, and an estimation of a potential transportation cost per unit. A probability is calculated such that the more expensive facility has a lower probability, and the cheapest facility has a higher probability. These probabilities are used with a random number to select warehouses with a certain capacity. The process finishes when the aggregated capacity is greater than the total demand. This selective method is reported as “Heuristic with Probability”.

The complexity of both matheuristics is dominated by the complexity of the solution of the Allocation Model which can be reduced to the Linear Assignment Problem, which can be solved in  $O(n^3)$  time [24].

---

#### Algorithm 1 General algorithm.

---

- 1: Perform one of the two heuristics to decide the opening of warehouses  $w$ .
  - 2: Solve the allocation model with the parameter  $z_{wk}$  obtained in step 1.
  - 3: Solve the reduced SCND model with the parameters  $z_{wk}$  and  $x_{wc}^{pt}$  obtained in the previous steps.
- 

---

#### Algorithm 2 Simple Constructive Heuristics.

---

- 1: All the closed warehouses  $z_{w,q} = \{\{0,0,0\}, \{0,0,0\}, \dots, \{0,0,0\}\}$
  - 2: Demanded storage capacity =  $\sum_{p \in P} \sum_{c \in C^p} \sum_{t \in T_{wc}} SQU^p DEM_c^p u_{wc}^t$
  - 3: Available capacity = 0
  - 4:  $Open_w = \{0,0,0, \dots, 0\}$
  - 5: **repeat**
  - 6:     **repeat**
  - 7:         Generate random  $ws$
  - 8:         Generate random  $qs$
  - 9:     **while**  $Open_{ws} = 1$
  - 10:          $Open_{ws} = 1$
  - 11:          $z_{ws,qs} = 1$
  - 12:         Available capacity = Available capacity +  $\overline{SQ}_{ws,qs}$
  - 13: **while** Available capacity < Demanded storage capacity
-



**Algorithm 3** Constructive Heuristics with Probability.

- 1: All the closed warehouses  $z_{w,q} = \{\{0,0,0\}, \{0,0,0\}, \dots, \{0,0,0\}\}$
- 2: Demanded storage capacity =  $\sum_{p \in P} \sum_{c \in C^p} \sum_{t \in T_{wc}} SQU^p DEM_c^p u_{wc}^t$
- 3: Total warehouse unit cost $_{wq}$  = Unit fixed cost $_{wq}$  + Unit transportation cost $_{wq}$
- 4: Unit fixed cost $_{wq} = \frac{FC_{wq}}{\overline{SQ}_{wq} \sum_{p \in P} SQU^p}$
- 5: Unit transportation cost $_{wq} = \frac{\sum_{w \in W} \sum_{p \in P} \sum_{c \in C^p} DEM_c^p}{|T_{wc}| \cdot \sum_{w \in W} \sum_{p \in P} \sum_{c \in C^p} \sum_{t \in T_{wc}} TC_{wc}^{pt} \cdot DEM_c^p}$
- 6: Probability $_{wq} = \frac{1000}{(\text{Total warehouse unit cost}_{wq})^4}$
- 7: A probability distribution is made with Probability $_{wq}, \forall w \in W, q \in Q$
- 8: Available capacity = 0
- 9:  $Open_w = \{0,0,0, \dots, 0\}$
- 10: {ns: random number}
- 11: **repeat**
- 12:     **repeat**
- 13:         Generate random  $ns$
- 14:         Select  $w$  and  $q$  corresponding to the random number  $ns$
- 15:         **while**  $Open_{ws} = 1$
- 16:              $Open_w = 1$
- 17:              $z_{wq} = 1$
- 18:             Available capacity = Available capacity +  $\overline{SQ}_{wq}$
- 19:         **while** Available capacity < Demanded storage capacity

4.2.1. Allocation Model

Allocation model 2.1 defines the decision variables  $u_{wc}^{pt}$  which due to the nature of the problem is the largest set of variables since the number of these variables is defined by the expression  $|W| \cdot |C| \cdot |P| \cdot |T|$  assuming that you always have more customers than warehouses or factories.

In the case of the instances used for this work, the smallest number of variables  $u_{wc}^{pt}$  used in one instance was 6000, so defining those variables alternately greatly reduces the number of branches explored by the branching and cutting algorithm. This is used in its default configuration in the IBM ILOG program CPLEX to solve mixed-integer linear programming models. The allocation model is presented below.

Variables

$$u_{od}^t = \begin{cases} 1 & \text{if origin } o \text{ supplies destination } d \text{ by mode of transport } t, \\ & ((o, d) \in \{(S, F) \cup (F, W) \cup (W, C)\}; t \in T) \\ 0 & \text{other case} \end{cases}$$

The constraints used in the original model are constraints (9), (10), (14) and (21)

$$u_{wc}^t \in \{0,1\} \quad w \in W, c \in C, t \in T \tag{21}$$

And the objective function is the following (22):

$$\text{Min} \sum_{p \in P} \sum_{c \in C^p} \sum_{w \in W^p} \sum_{t \in T_{wc}} DEM_c^p TC_{wc}^{pt} u_{wc}^t \tag{22}$$

4.2.2. Model of Reduced SCND

The variables of the model that become parameters are the following:

Parameters

$$z_{wq} = \begin{cases} 1 & \text{if it is decided to open a warehouse } w \text{ with a capacity level } q, \\ & (w \in W; q \in Q) \\ 0 & \text{other case} \end{cases}$$

$$u_{wc}^t = \begin{cases} 1 & \text{if it is decided to open a warehouse } w \text{ with a client } c \text{ by mode of transport } t, \\ & (w \in W; c \in C; t \in T) \\ 0 & \text{other case} \end{cases}$$

The parameters that will no longer be used in model 2.2 with respect to the original are the following:

Parameters not used	
$SQU^p$	Storage capacity required to store a unit of $p$ , ( $p \in P$ )
$\overline{SQ}_{wq}$	Maximum storage capacity of warehouse $f$ with level of capacity $q$ , ( $w \in W; q \in Q$ )
$\underline{SQ}_{wq}$	Minimu production capacity used by factory $f$ with level of capacity $q$ , ( $w \in W; q \in Q$ )

The SCND model defines the remaining decision variables of the original model, taking as parameters, the decision variables resolved by both the heuristic and the 2.1 allocation model, removing the constraints that are not necessary.

The model uses constraints (2) to (4), (6) to (8), (11) to (13), (15) to (19), and (23) to (24).

$$\sum_{q \in Q} z_{fq} \leq 1 \quad f \in F \tag{23}$$

$$u_{od}^t \in 0, 1 \quad (o, d) \in \{(S, F) \cup (F, W)\}, t \in T \tag{24}$$

The objective function is used the same as the original model, although some terms do not contain decision variables but parameters, with the aim of comparing the matheuristic solutions with those of the MILP.

### 5. Instances

The instances are generated by means of a methodology described by [25] that simulate reality. The size of the generated instances is shown in Table 2 and the size of the corresponding model is shown in Table 3.

**Table 2.** Size of the sets in each instance.

$ C $	$ S $	$ F $	$ W $	$ Q $	$ T $	$ R $	$ P $
100	10	10	20	3	3	20	20
125	13	13	25	3	3	25	25
150	15	15	30	3	3	30	30
175	18	18	35	3	3	35	35
200	20	20	40	3	3	40	40

Transport mode 1 represents the train, which is not available for all locations and the minimum amount of transport is very high. Modes 2 and 3 represent land vehicles, mode 2 represents a small vehicle such as delivery vans, and mode 3 represents a larger cargo vehicle; these two are assumed to be hired through an external logistics provider

so they do not have a maximum amount of transport. For mode 3 there is a minimum quantity to transport to justify the use of the larger vehicle. For each product-customer pair a random demand  $DEM_{pc} = X \sim UNIFD[1, 10]$ , is generated, where  $X$  is a random number that follows a discrete uniform distribution  $UNIFD[min, max]$  that goes from  $min$  to  $max$ . The fixed cost of opening a factory and a warehouse are defined by the following expression:

$$FC_{fq} = 1000 \sqrt{\frac{\overline{MQ}_{fq}}{\frac{\sum_{p \in P} MQU_p}{|P|}}} \tag{25}$$

$$FC_{wq} = 1000 \sqrt{\frac{\overline{SQ}_{wq}}{\frac{\sum_{p \in P} SQU_p}{|P|}}} \tag{26}$$

The acquisition cost of a material is generated by the following expression  $PC_s^r = X \sim UNIFC[0.075, 0.625]$  where  $UNIFC[min, max]$  represents a continuous uniform distribution. The manufacturing cost of a product is generated by the following expression  $MC_f^p = X \sim UNIFC[0.375, 1.25]$ .

The other parameters are calculated using a more sophisticated methodology explained in Appendix A of the work by [25]. A Windows console program developed in C++ was carried out to generate an instance automatically to generate these instances.

**Table 3.** Statistics of the instances and their values for MILP.

Name of the Instance	C	Number of Continuous Variables	Number of Binary Variables	Number of Constraints
*-100	100	18,000	8990	17,190
*-125	125	37,050	15,196	28,103
*-150	150	60,750	22,410	40,785
*-175	175	100,170	32,736	57,893
*-200	200	144,000	43,780	76,380

### 6. Results

The generated instances were resolved with the mixed-integer linear programming model programmed in OPL in the IBM ILOG CPLEX 12.8.0 IDE, and the case of the matheuristics, an application in C++ Concert Technology was programmed with the use of CPLEX. Both were run on a Lenovo ThinkPad T580 computer with an Intel Core i7-8550U @ 1.80GHz processor and 16 GB of RAM.

The results obtained are shown in Table 4, indicating the total cost obtained for each instance with the different methods compared.

The results show that as the instances increase in size, the computation time to obtain optimal solutions increases exponentially. In all the cases, the MILP reached the time limit of 4 h. Only for instances with 100 clients, the matheuristic algorithms achieved times below 25 min. However, for instances larger than 100 clients, the matheuristic algorithms reached the time limit of 2 h. Hence, obtaining results through the use of mathematical modeling becomes impractical for large instances.

It can be observed in Table 4 that when the size of the instance grows, i.e., it has a higher number of clients, the total cost increases also. Likewise, it can be seen that as the instance size increase, the difference between the quality of solutions provided by CPLEX and those provided by the matheuristics decreases. For instances of 100, 125 and 150 clients, the results of the matheuristic algorithm with the “simple heuristic” were just 14.20% on average above those obtained with CPLEX, and the results with the matheuristic algorithm

with the “heuristics with probability” were only 6.06% on average above those obtained with CPLEX.

In the instances of both 175 and 200 clients, at least one of the two implemented matheuristics obtained a better solution than that of CPLEX in most of the cases. For the instances with 175 customers, the average improvement was 1.41% comparing the result of CPLEX and the result of the matheuristic algorithm with the “simple heuristic”. For the same instances, the average improvement was 5.96% comparing the result of CPLEX and the result of the matheuristic algorithm with the “heuristics with probability”. For the instance 2–200, the improvement was 3.93% comparing the result of CPLEX and the result of the matheuristic algorithm with the “simple heuristic”. For the same instance, the improvement was 9.22% comparing the result of CPLEX and the result of the matheuristic algorithm with the “heuristics with probability”. In six cases out of 15, with 175 and 200 clients, the matheuristic algorithms were able to find a feasible solution while CPLEX could not.

Comparing the two matheuristics, it can be observed that the heuristic method used to decide the opening of warehouses has a great impact on the objective functions for the type of instances used because the fixed cost of opening a warehouse is the cost with the greatest impact. Both the first and second heuristics diversify the results at each iteration so that in small instances, the algorithms may be used on more than one occasion to increase the quality of the solutions. Regarding the use of the probability distribution used for constructive heuristics with probability, although some randomness is applied, the selected warehouses will tend to be those with the lowest cost of opening divided by their capacity, so the quality of the solutions tends to be better. In average, for all the instances tested, the matheuristic algorithm with the “heuristics with probability” was 7.15% better than the algorithm with the “simple heuristic”.

**Table 4.** Table of results.

Instance	MILP *	Matheuristic ** Simple Heuristic	Matheuristic ** Heuristic with Probability	No. of Clients
1-100	\$457,969	\$533,871	\$494,606	100
2-100	\$396,124	\$469,641	\$437,017	100
3-100	\$454,499	\$528,758	\$484,789	100
4-100	\$454,535	\$499,858	\$451,079	100
5-100	\$421,027	\$438,128	\$464,642	100
1-125	\$603,112	\$692,654	\$643,217	125
2-125	\$524,209	\$624,083	\$565,659	125
3-125	\$569,020	\$602,907	\$617,547	125
4-125	\$566,296	\$646,428	\$604,259	125
5-125	\$521,739	\$622,127	\$577,284	125
1-150	\$713,102	\$796,459	\$753,579	150
2-150	\$709,303	\$861,745	\$770,492	150
3-150	\$695,731	\$789,834	\$668,408	150
4-150	\$713,282	\$816,635	\$717,607	150
5-150	\$679,902	\$768,746	\$713,473	150
1-175	\$912,030	\$945,357	\$904,579	175

Table 4. Cont.

Instance	MILP *	Matheuristic ** Simple Heuristic	Matheuristic ** Heuristic with Probability	No. of Clients
2-175	\$1,100,417	\$970,647	\$898,271	175
3-175	***	\$930,802	\$857,758	175
4-175	\$1,099,033	\$994,140	\$939,693	175
5-175	\$915,431	\$865,468	\$929,748	175
6-175	\$882,227	\$925,900	\$798,514	175
7-175	\$856,623	\$912,108	\$881,164	175
8-175	\$940,480	\$955,951	\$912,570	175
9-175	***	\$970,484	\$854,358	175
10-175	\$977,692	\$965,596	\$919,445	175
1-200	***	\$1,153,589	\$1,115,358	200
2-200	\$1,061,059	\$1,019,363	\$963,204	200
3-200	***	\$1,138,016	\$1,015,605	200
4-200	***	\$1,129,623	\$1,019,119	200
5-200	***	\$1,130,032	\$994,430	200

\* Target function values of incumbent solution running the solver with a 4-h limit; \*\* Target function values of incumbent solution running executable with a 2-h limit; \*\*\* No integer solution found.

For the interested reader, a further discussion about the SCND configurations obtained and the distribution of costs, according to the design of instances proposed in [25], can be found at [23].

## 7. Conclusions

In this work, a problem for supply-chain network design is addressed. This problem was presented in the literature by Corthinal et al. [23]. We selected this complex NP-Hard problem to prove the efficiency of novel matheuristic algorithms. Matheuristics are algorithms that combine heuristic rules with exact optimization methods based on mathematical programming. There is a growing interest in these methods in the search for improving the quality of solutions with a fast computation. Solving hard combinatorial optimization problems, Although heuristics and metaheuristics demonstrated efficiency in obtaining feasible but poor-quality solutions in short computation times, exact methods implemented in commercial and open-source software deliver optimal solutions at the cost of long execution times. In this way, the research in matheuristics looks for the benefits of hybridization.

The problem addressed has characteristics that increase the level of complexity compared to other problems of supply-chain network design solved with matheuristics. In addition to the classic decisions on the location of facilities and transportation, the problem involves determining capacities for the facilities, choosing between different transportation modes, and considering a hierarchical product architecture as described by a bill of materials. Hence, this problem is a good challenge to demonstrate the efficiency of the methods proposed. The structure of the network and decisions presented can be used for the design of global manufacturing and distribution networks in high technology industries such as pharmaceuticals, automotive, or electronics.

The main contribution of this paper is the proposal of novel matheuristic algorithms to solve a challenging combinatorial optimization problem for supply chain network design.

The algorithms proposed were efficient in obtaining solutions of high quality in reasonable computation times for large instances.

To study larger instances or longer supply chains, the problem can be divided into more sub-problems with heuristic decisions in between. As we observed, it is possible to obtain better solutions with a well-thought heuristic in less time as data sets grow. Additionally, more complex decisions can be added to the model to integrate routing, inventory management, sustainability issues, and variability through stochastic modeling. The key to designing efficient metaheuristics is the adequate decomposition of the problem to take advantage of the quality of exact methods using heuristics and randomness to boost the execution speed.

**Author Contributions:** Conceptualization, D.A.R.-S. and E.O.-B.; Data curation, D.A.R.-S.; Formal analysis, D.A.R.-S., E.O.-B. and O.R.; Investigation, D.A.R.-S. and E.O.-B.; Methodology, D.A.R.-S. and E.O.-B.; Project administration, E.O.-B.; Resources, D.A.R.-S.; Software, D.A.R.-S.; Supervision, E.O.-B. and O.R.; Validation, D.A.R.-S., E.O.-B. and O.R.; Visualization, D.A.R.-S. and G.S.-G.; Writing—original draft, D.A.R.-S., E.O.-B. and O.R.; Writing—review and editing, E.O.-B., O.R. and G.S.-G. All authors have read and agreed to the published version of the manuscript.

**Funding:** E.O.-B. acknowledges support from Universidad Panamericana, Mexico [grant number: UP-CI-2021-GDL-04-ING].

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

Notation	Description
<b>Sets</b>	
$S$	Set of suppliers
$C$	Set of clients
$F$	Set of potential factories
$T$	Set of modes of transport
$Q$	Set of capacity levels for factory $F$ or warehouse $W$
<b>Subsets</b>	
$R$	Subset of raw material
$P$	Subset of finished products
$W$	Subset of potential warehouses
$P^r$	Subset of finished products that require raw material $r$ , ( $P^r \subset P; r \in R$ )
$S^r$	Subset of suppliers who supply raw material $r$ , ( $S^r \subset S; r \in R$ )
$F^p$	Subset of factories that can produce finished product $p$ , ( $F^p \subset F; p \in P$ )
$W^p$	Subset of warehouses that can store finished product $p$ , ( $W^p \subset W; p \in P$ )
$C^p$	Subset of clients with demand of finished product $p$ , ( $C^p \subset C; p \in P$ )
$T_{od}$	Subset of modes of transport available from origin to destination $d$ , ( $T_{od} \subset T$ )
<b>Parameters</b>	
<i>Parameters of cost</i>	
$FC_{fq}$	Fixed cost of opening a factory $f$ with capacity level $q$ , ( $f \in F; q \in Q$ )
$FC_{wq}$	Fixed cost of opening a warehouse $w$ with capacity level $q$ , ( $w \in W; q \in Q$ )
$PC_s^r$	Cost of buying raw material $r$ with supplier $s$ , ( $s \in S; r \in R$ )
$MC_f^p$	Cost of manufacturing product $p$ in factory $f$ , ( $f \in F; p \in P$ )
$TC_{od}^{rt}$	Cost of transporting material $r$ from origin $o$ or to destination $d$ with transport mode $t$ , ( $(o, d) \in (S, F); r \in R; t \in T$ )

$TC_{od}^{pt}$	Cost of transporting product $p$ from origin $o$ or to destination $d$ with transport mode $t$ , $((o, d) \in (F, W) \cup (W, C); p \in P; t \in T)$
<i>Parameters of the product</i>	
$A^{rp}$	Number of units of material $r$ necessary to produce product $p$ , $(r \in R; p \in P)$
$DEM_c^p$	Demand of client $c$ for product $p$ , $(c \in C; p \in P)$
<i>Parameters of capacity</i>	
$MQU^p$	Production capacity required to produce a unit of $p$ , $(p \in P)$
$SQU^p$	Storage capacity required to store a unit of $p$ , $(p \in P)$
$TQU^{pt}$	Transport capacity used by a unit of a product $p$ and by a mode of transport $t$ , $(i \in R \cup P; t \in T)$
$SQR_s^r$	Capacity of supplier $s$ to supply material $r$ , $(s \in S; r \in R)$
$MQP_{fq}^p$	Capacity of factory $f$ to produce product $p$ with capacity level $q$ , $(f \in F; q \in Q; p \in P)$
$\overline{MQ}_{fq}$	Maximum production capacity of factory $f$ with capacity level $q$ , $(f \in F; q \in Q)$
$\underline{MQ}_{fq}$	Minimum used production capacity of factory $f$ with capacity level $q$ , $(f \in F; q \in Q)$
$\overline{SQ}_{wq}$	Maximum storage capacity of warehouse $w$ with capacity level $q$ , $(w \in W; q \in Q)$
$\underline{SQ}_{wq}$	Minimum used storage capacity of warehouse $w$ with capacity level $q$ , $(w \in W; q \in Q)$
$\overline{TQ}_{od}^t$	Maximum transport capacity of mode $t$ from origin $o$ to destination $d$ , $((o, d) \in \{(S, F) \cup (F, W) \cup (W, C)\}; t \in T)$
$\underline{TQ}_{od}^t$	Minimum quantity of goods transported by mode $t$ from origin $o$ to destination $d$ , $((o, d) \in \{(S, F) \cup (F, W) \cup (W, C)\}; t \in T)$
<b>Notation</b>	<b>Description</b>
<b>Parameters</b>	
$TC_{wc}^{pt}$	Cost of transporting product $p$ from warehouse $w$ to customer $c$ with the mode of transport $t$ , $(w \in W; c \in C; p \in P; t \in T)$

## References

- Shen, Z.J.M.; Zhan, R.L.; Zhang, J. The reliable facility location problem: Formulations, heuristics, and approximation algorithms. *INFORMS J. Comput.* **2011**, *23*, 470–482. [\[CrossRef\]](#)
- Pirkul, H.; Jayaraman, V. A multi-commodity, multi-plant, capacitated facility location problem: Formulation and efficient heuristic solution. *Comput. Oper. Res.* **1998**, *25*, 869–878. [\[CrossRef\]](#)
- Wu, S.D.; Golbasi, H. Multi-item, multi-facility supply chain planning: Models, complexities, and algorithms. *Comput. Optim. Appl.* **2004**, *28*, 325–356. COAP.0000033967.18695.9d. [\[CrossRef\]](#)
- Eskigun, E.; Uzsoy, R.; Preckel, P.V.; Beaujon, G.; Krishnan, S.; Tew, J.D. Outbound supply chain network design with mode selection, lead times and capacitated vehicle distribution centers. *Eur. J. Oper. Res.* **2005**, *165*, 182–206. [\[CrossRef\]](#)
- Sadjady, H.; Davoudpour, H. Two-echelon, multi-commodity supply chain network design with mode selection, lead-times and inventory costs. *Comput. Oper. Res.* **2012**, *39*, 1345–1354. [\[CrossRef\]](#)
- Olivares-Benitez, E.; Ríos-Mercado, R.Z.; González-Velarde, J.L. A metaheuristic algorithm to solve the selection of transportation channels in supply chain design. *Int. J. Prod. Econ.* **2013**, *145*, 161–172. [\[CrossRef\]](#)
- Rahmaniani, R.; Ghaderi, A. A combined facility location and network design problem with multi-type of capacitated links. *Appl. Math. Model.* **2013**, *37*, 6400–6414. [\[CrossRef\]](#)
- Bertazzi, L.; Bosco, A.; Laganà, D. Min-Max exact and heuristic policies for a two-echelon supply chain with inventory and transportation procurement decisions. *Transp. Res. Part E Logist. Transp. Rev.* **2016**, *93*, 57–70. [\[CrossRef\]](#)
- Tsao, Y.C.; Nugraha Ridhwan Amir, E.; Thanh, V.V.; Dachyar, M. Designing an eco-efficient supply chain network considering carbon trade and trade-credit: A robust fuzzy optimization approach. *Comput. Ind. Eng.* **2021**, *160*, 107595. [\[CrossRef\]](#)
- Fathi, M.; Khakifirooz, M.; Diabat, A.; Chen, H. An integrated queuing-stochastic optimization hybrid Genetic Algorithm for a location-inventory supply chain network. *Int. J. Prod. Econ.* **2021**, *237*, 108139. [\[CrossRef\]](#)
- Vittorio, M.; Thomas, S.; Stephan, V. *Matheuristics*; Springer: New York, NY, USA, 2010.
- Fischetti, M.; Fischetti, M. *Matheuristics*. In *Handbook of heuristics*; Springer: New York, NY, USA, 2018; pp. 121–153.
- Maniezzo, V.; Boschetti, M.A.; Stützle, T. *Matheuristics*; Springer International Publishing: New York, NY, USA, 2021. [\[CrossRef\]](#)

14. Queiroga, E.; Sadykov, R.; Uchoa, E. A POPMUSIC matheuristic for the capacitated vehicle routing problem. *Comput. Oper. Res.* **2021**, *136*, 105475. [[CrossRef](#)]
15. Guzman, E.; Andres, B.; Poler, R. Matheuristic Algorithms for Production Planning in Manufacturing Enterprises. In *Doctoral Conference on Computing, Electrical and Industrial Systems*; Springer: New York, NY, USA, 2021; pp. 115–122.
16. Goerler, A.; Lalla-Ruiz, E.; Voß, S. Late acceptance hill-climbing matheuristic for the general lot sizing and scheduling problem with rich constraints. *Algorithms* **2020**, *13*, 138. [[CrossRef](#)]
17. Bayliss, C.; Serra, M.; Nieto, A.; Juan, A.A. Combining a Matheuristic with Simulation for Risk Management of Stochastic Assets and Liabilities. *Risks* **2020**, *8*, 131. [[CrossRef](#)]
18. Boschetti, M.; Maniezzo, V.; Roffilli, M. Decomposition Techniques as Metaheuristic Frameworks. In *Matheuristics*; Springer: New York, NY, USA, 2009; pp. 135–158. [[CrossRef](#)]
19. Raa, B.; Dullaert, W.; Aghezzaf, E.H. A matheuristic for aggregate production-distribution planning with mould sharing. *Int. J. Prod. Econ.* **2013**, *145*, 29–37. [[CrossRef](#)]
20. Tautenhain, C.P.; Barbosa-Povoa, A.P.; Nascimento, M.C. A multi-objective matheuristic for designing and planning sustainable supply chains. *Comput. Ind. Eng.* **2019**, *135*, 1203–1223. [[CrossRef](#)]
21. Cantú, V.H.; Azzaro-Pantel, C.; Ponsich, A. A Novel Matheuristic based on bi-level optimization for the multi-Objective design of hydrogen supply chains. *Comput. Chem. Eng.* **2021**, *152*, 107370. [[CrossRef](#)]
22. Souto, G.; Morais, I.; Mauri, G.R.; Ribeiro, G.M.; González, P.H. A hybrid matheuristic for the Two-Stage Capacitated Facility Location problem. *Expert Syst. Appl.* **2021**, *185*, 115501. [[CrossRef](#)]
23. Cortinhal, M.J.; Lopes, M.J.; Melo, M.T. A multi-stage supply chain network design problem with in-house production and partial product outsourcing. *Appl. Math. Model.* **2019**, *70*, 572–594. [[CrossRef](#)]
24. Papadimitriou, C.H.; Steiglitz, K. *Combinatorial Optimization: Algorithms and Complexity*; Dover Publications: Mineola, NY, USA, 1998.
25. Cortinhal, M.; Lopes, M.; Melo, M. Impact of Partial Product Outsourcing, Transportation Mode Selection, and Single-Assignment Requirements on the Design of a Multi-Stage Supply Chain Network. Hochschule für Technik und Wirtschaft des Saarlandes, Fakultät für Wirtschaftswissenschaften. Technical Report. Saarland University of Applied Sciences (htw saar), Saarland Business School, 2018. Available online: <http://hdl.handle.net/10419/181867> (accessed on 10 September 2021).