Computers & Operations Research 155 (2023) 106232

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor

The effect of different mathematical formulations on a matheuristic algorithm for the production routing problem

Mohamed Ben Ahmed [a,b,*], Lars Magnus Hvattum [a], Agostinho Agra [c]

[a] Faculty of Logistics, Molde University College, PO Box 2110, NO-6402 Molde, Norway
[b] Møreforsking Molde AS, Britvegen 4, 6410, Molde, Norway
[c] DMat and CIDMA, University of Aveiro, Aveiro, Portugal

ARTICLE INFO

Keywords:
Lot sizing
Inventory routing
Mixed integer formulation
Valid inequality
Proximity search
Experimental analysis

ABSTRACT

We perform an experimental study to evaluate the performance of a matheuristic for the production routing problem (PRP). First, we develop a basic matheuristic that prescribes starting from a partial initial solution, completing it using a sequence of constructive heuristics, and improving it using a general-purpose mixed-integer programming heuristic. Next, we investigate the effect of three state-of-the-art mathematical formulations on the proposed matheuristic convergence. The formulations are implemented and tested with and without the use of valid inequalities. In addition, by suggesting different techniques to generate a feasible starting solution for our matheuristic, we assess the contribution of an initial solution to the matheuristic's overall performance. We conduct extensive computational experiments on benchmark data instances for the PRP. The results show that a proper choice of an embedded mathematical formulation depends on the data instances' features, such as the number of customers and the length of the planning horizon. The comparisons undertaken in this study indicate that having a better initial solution does not necessarily lead to finding a better final solution.

1. Introduction

This study investigates the production routing problem (PRP) with a discrete and finite time horizon. The problem combines three famous subproblems from the literature: lot-sizing, vehicle routing, and inventory management problems. Contributions dealing with the PRP appeared in the mid-1990 when Chandra (1993) and Chandra and Fisher (1994) successfully combined production and distribution decisions. However, the problem remained widely unexplored until mid-2010s, when the interest in it gained momentum, perhaps in connection to the growing application of vendor-managed inventory (VMI) policies in the business. This interest is demonstrated by the surging number of its solution methods, which are often disparate. We differentiate between (*i*) exact algorithms, (*ii*) heuristic algorithms, and (*iii*) the hybridization of heuristic and exact algorithms, yielding the so-called matheuristic algorithms. For a comprehensive review on solution methods for the PRP, we refer the readers to Adulyasak et al. (2015) and Díaz-Madroñero et al. (2015). Hrabec et al. (2022) also reviewed some pertinent contributions to the PRP and showed that solving a PRP is better than solving sequential problems if the optimality gap of the integrated problem is below a certain threshold (11%).

Matheuristics, which are reputable for obtaining high-quality solutions, have recently come under the spotlight. For example, Russell (2017) developed a matheuristic that involves the solution of a relaxed version of a mixed-integer program (MIP) to derive lot-sizing decisions, followed by the use of a vehicle routing algorithm to determine final routes. Avci and Yildiz (2019) proposed a matheuristic that decomposes the problem into a sequence of subproblems. Distribution and routing subproblems are solved heuristically, while a mathematical model is used to derive the lot-sizing, inventory, and delivery quantity decisions. Chitsaz et al. (2019) proposed a three-phase matheuristic for the joint assembly, production, and inventory routing problem. The first phase determines a setup schedule, while the second phase optimizes production quantities, supplier visit schedules, and shipment quantities. The third phase solves a vehicle routing problem (VRP) for each period in the planning horizon. Recently, Manousakis et al. (2021a) designed an efficient matheuristic for solving the PRP. The crux is that exploring both feasible and infeasible solution spaces with a local search can lead to improved PRP solutions. The matheuristic employs mathematical models to restore feasibility and improve incumbent solutions. Vadseth et al. (2023) proposed a multi-start matheuristic that creates a set of

* Corresponding author at: Faculty of Logistics, Molde University College, PO Box 2110, NO-6402 Molde, Norway.
E-mail addresses: mohamed.ben.ahmed@himolde.no, ben.mohamed@moreforsking.no (M.B. Ahmed), Lars.M.Hvattum@himolde.no (L.M. Hvattum), aagra@ua.pt (A. Agra).

https://doi.org/10.1016/j.cor.2023.106232
Received 27 June 2022; Received in revised form 3 March 2023; Accepted 24 March 2023
Available online 10 April 2023

initial solutions to the PRP by first solving a production subproblem and then a routing subproblem. Differentiation techniques, such as relaxing or tightening the problem's constraints, are used to ensure that a distinct solution is obtained at each restart. Finally, a mathematical model that applies customer insertion and removal operations is repeatedly run on the pool of constructed solutions, leading to an improved solution. Mousavi et al. (2022) proposed a matheuristic for solving the stochastic PRP with perishable products. It breaks down the problem into five subproblems, where the solution to one subproblem is conveyed to the following one. Other approaches that go in line with the aforementioned contributions can be found in Miranda et al. (2018), Neves-Moreira et al. (2019), and Avci and Yildiz (2020).

The applications of matheuristics are not solely limited to the context of the PRP, and matheuristics have been implemented to solve other routing problems as documented by Doerner and Schmid (2010) and Archetti and Speranza (2014). Other surveys on matheuristics are presented by Blum et al. (2011), Hanafi and Todosijević (2017), and Boschetti and Maniezzo (2022). In the vast majority of these contributions, the design of matheuristics is involved in a manual, experimental approach guided by the algorithm designers' experience and the exploitation of problem-specific knowledge. The developed matheuristics are ad-hoc and do not necessarily provide an understanding of how a combination of MP-based exact methods and heuristic search techniques can best improve algorithmic performance (Blum et al., 2011). Besides, a deeper understanding of the performance of mathematical components is lacking, which would eventually provide, if it exists, insights into the optimal configuration of matheuristics for the PRP. Finally, one needs to further investigate the sensitivity of matheuristics to the initial solution. While hypothetically, an excellent initial solution can improve the performance of a matheuristic, it may be, in some cases, challenging to improve, resulting in higher execution times.

This paper's main contribution is to design a generic matheuristic for the PRP and evaluate its performance under different mathematical formulations. The matheuristic incorporates three phases, each one relying on a basic/general-purpose method. The first phase is a mixed-integer programming (MIP) relaxation of the entire problem where only the integrality of lot-sizing decisions is maintained. In the second phase, we develop a sequence of construction heuristics to determine the distribution and the routing decisions, thus yielding a complete initial solution. The last phase uses a general-purpose MIP heuristic that iteratively attempts to improve the given solution. The matheuristic's parameters do not require an extensive fine-tuning, and their values are adjusted using offline calibration techniques. Three mathematical formulations are developed and tested within the matheuristic framework, namely: (*1*) a four-index vehicle flow formulation, (*2*) a three-index vehicle flow formulation, and (*3*) a two-commodity flow formulation. The first formulation uses integer variables associated with each pair of nodes to count the number of times a vehicle travels between them. In the second formulation, these variables are aggregated over all vehicles. The third formulation defines two continuous flow variables for each pair of nodes representing the vehicle's total load and empty space while traversing these two nodes, respectively. These formulations are tested with and without using valid inequalities to assess the impact of these inequalities on the algorithm's overall performance. Furthermore, by allowing for different techniques to generate a feasible starting solution for our matheuristic, we aim to measure the contribution of an initial solution to the matheuristic's overall performance. Detailed experimental analyses follow our study, and we apply several metrics to evaluate the matheuristic components' performance. Additionally, we compare our matheuristic with other methods that solved the PRP. The obtained results show that our method competes well with exact algorithms, while it outperforms existing heuristic and matheuristic methods.

To the best of our knowledge, there are no studies in the existing literature that adopted a similar experimental methodology where the study of models is done while embedded in a matheuristics. A handful of similar contributions exist concerning the design of metaheuristics. For example, Prins (2004) investigated how features of a genetic algorithm influenced the quality of solutions to a VRP. Hemmati and Hvattum (2016) investigated the importance of randomization in adaptive large neighborhood search when solving maritime pickup and delivery problems, whereas Sousa et al. (2016) evaluated the importance of initial solution algorithms in simulated annealing when solving energy resource scheduling problems.

We are not aware of existing studies that compare PRP formulations with respect to their role in heuristic search. However, one can find insightful contributions in the literature about the quality of linear programming (LP) relaxations of different formulations for the inventory routing problem (Archetti and Ljubić, 2022) and the multi-depot routing problem (Bektaş et al., 2020), as well as the performance of flow formulations for the capacitated location-routing problem when used in a branch-and-cut framework (Contardo et al., 2013).

The remainder of this paper is organized as follows. In Section 2, we introduce the formal definition of the PRP and some useful notation used in the following sections. In Section 3 we introduce different formulations for the PRP. The details of the matheuristic framework are presented in Section 4. This is followed by a description of the experimental setup and the tests performed in Section 5 and some concluding remarks in Section 6.

2. Problem description

The PRP can be defined on a complete undirected graph $G = (N, E)$ where $N = N^C \cup \{0\}$ is the set of nodes, N^C is the subset of customers nodes, node 0 represents the plant facility, and $E = \{(i,j) : i,j \in N, i < j\}$ is the set of edges. We define the set $T = \{1,\dots,H\}$ of time periods. With each edge $(i,j) \in E$ is associated a routing cost c_{ij}. A fleet of m identical vehicles is made available in each period $t \in T$, and every vehicle has a transportation capacity of Q units. For each node $i \in N^C$ and time period $t \in T$, let d_{it} denote the demand of customer i in period t. For each node $i \in N$, a starting inventory level I_{i0} and a maximum inventory level L_i are given. We denote by h_i the inventory holding cost at node $i \in N$. The setup cost at the plant and the variable production cost are represented by s and u, respectively. The production capacity at the plant facility is given by C. Hence, the PRP aims at determining a production and a distribution plan such that:

- Production capacity at the plant facility is respected.
- Vehicle capacities and inventory capacities at both customers and the production plant are satisfied.
- Stock-out at customers are not allowed.
- Backlog of demand is not allowed.
- Each customer is visited at most once in each time period.
- Vehicle routes start and end at the plant in each time period.
- The total cost, given by the sum of the setup cost at the plant, the variable production cost, the inventory holding cost at the customers and the plant, plus the routing costs, is minimized.

We define the following additional notation. Let $\Delta(S) = \{(i,j) \in E : i \in S, j \notin S \text{ or } i \notin S, j \in S\}$ be the set of edges incident to a node set S (for simplicity, we write $\Delta(i)$ to represent the set of edges incident to $i \in N$). Let $E(S)$ the set of edges $(i,j) \in E$, such that $i,j \in S$, where $S \subseteq N$.

3. Multi-vehicle formulations for the PRP

This section introduces three state of the art mathematical formulations of the generic PRP version: a four-index vehicle flow formulation, a three-index vehicle flow formulation, and a two-commodity flow formulation.

3.1. A four-index vehicle flow formulation

In the four-index vehicle flow formulation, originally presented by Adulyasak et al. (2014a), we define for every time period $t \in T$ the binary decision variable x_{ijkt} that takes the value 1 if and only if a vehicle $k \in K$ travels directly between nodes $i, j \in N$. We also define z_{ikt} as a binary variable taking the value 1 if node i is visited by vehicle $k \in K$ in period $t \in T$. In addition, I_{it} is a continuous decision variable corresponding to the inventory level of node $i \in N$ at the end of time period $t \in T$, while q_{ikt} represents the quantity delivered to customer $i \in N^C$ by means of vehicle $k \in K$ in period $t \in T$. Finally, let y_t be the binary setup variable, and p_t the continuous production level variable, both defined for each time period $t \in T$. With these decision variables, the four-index vehicle flow formulation of the PRP, henceforth called FIVF, is given by:

$$\min f(x,y,z) = \sum_{t\in T}\left(up_t + sy_t + \sum_{i\in N} h_i I_{it} + \sum_{(i,j)\in E}\sum_{k\in K} c_{ij}x_{ijkt}\right) \tag{1.1}$$

subject to

$$I_{0,t-1} + p_t = \sum_{i\in N^C}\sum_{k\in K} q_{ikt} + I_{0t}, \quad \forall t \in T, \tag{1.2}$$

$$I_{i,t-1} + \sum_{k\in K} q_{ikt} = d_{it} + I_{it}, \quad i \in N^C, \quad \forall t \in T, \tag{1.3}$$

$$p_t \le \min\left\{ C, \sum_{i\in N^C}\sum_{l=t}^{H} d_{il}\right\} y_t, \quad \forall t \in T, \tag{1.4}$$

$$I_{0t} \le L_0, \quad \forall t \in T, \tag{1.5}$$

$$I_{i,t-1} + \sum_{k\in K} q_{ikt} \le L_i, \quad \forall i \in N^C, \quad t \in T, \tag{1.6}$$

$$\sum_{i\in N^C} q_{ikt} \le Q z_{0kt}, \quad \forall k \in K, \quad t \in T, \tag{1.7}$$

$$\sum_{k\in K} z_{ikt} \le 1, \quad \forall i \in N^C, \quad t \in T, \tag{1.8}$$

$$q_{ikt} \le \min\left\{ L_i, Q, \sum_{l=t}^{H} d_{il}\right\} z_{ikt}, \quad \forall i \in N^C, \quad k \in K, \quad t \in T, \tag{1.9}$$

$$\sum_{j:(i,j)\in E} x_{ijkt} = 2z_{ikt}, \quad \forall i \in N^C, \quad k \in K, \quad t \in T, \tag{1.10}$$

$$\sum_{(i,j)\in E(S)} x_{ijkt} \le \sum_{i\in S} z_{ikt} - z_{ekt}, \quad \forall S \subseteq N^C, \quad e \in S, \quad |S| \ge 2, \quad k \in K, \quad t \in T, \tag{1.11}$$

$$z_{0kt} \ge z_{0,k+1,t}, \quad 1 \le k \le m-1, \quad \forall t \in T, \tag{1.12}$$

$$\sum_{i=1}^{j} 2^{(j-i)} z_{ikt} \ge \sum_{i=1}^{j} 2^{(j-1)} z_{i,k+1,t}, \quad \forall i \in N^C, k \in K, \quad t \in T, \tag{1.13}$$

$$p_t, I_{it}, q_{ikt} \ge 0, \quad \forall i \in N, \quad k \in K, \quad t \in T, \tag{1.14}$$

$$y_t, z_{ikt} \in \{0,1\}, \quad \forall i \in N, \quad k \in K, \quad t \in T, \tag{1.15}$$

$$x_{ijkt} \in \{0,1\}, \quad \forall (i,j) \in E : i \in N^C, \quad k \in K, \quad t \in T, \tag{1.16}$$

$$x_{0jkt} \in \{0,1,2\}, \quad \forall j \in N^C, \quad k \in K, \quad t \in T. \tag{1.17}$$

The objective function (1.1) calls for the minimization of the total operational costs, that is, the sum of the setup costs, the unit production costs, the transportation costs, and the inventory holding costs at both the plant facility and the customers. Constraints (1.2)–(1.3) define the inventory level at the production plant and at the customers, respectively. If production occurs at a specific time period $t \in T$, then a setup cost is incurred at the production facility and is forced via constraints (1.4). The produced quantity cannot surpass the production capacity neither the total demand in the remaining time periods. Constraints (1.5)–(1.6) ensure that the inventory quantities at the plant and the customers, respectively, do not exceed their capacities at the end of each period. Constraints (1.7) are vehicle capacity constraints. Constraints (1.8) allow each customer to be visited at most once during each time period. If there is a positive delivery to a node, then a visit is forced by constraints (1.9). Constraints (1.10) are the degree constraints of customers; they require the number of edges incident to each node to be equal to 2 if it is visited or 0 otherwise. Constraints (1.11) are the subtour elimination constraints (SEC) for each vehicle route and each period. Constraints (1.12)–(1.13) are symmetry breaking constraints. Finally, constraints (1.14)–(1.17) define the domains of the variables.

The SECs are dynamically generated in each node of the branch-and-bound tree. Their separation amounts to the solution of the classical min-cut problem presented in Padberg and Rinaldi (1991). Adulyasak et al. (2014a) also strengthen the formulation (1.1)–(1.17) using several valid inequalities. They are valid for the multivehicle PRP with capacitated production, and they are described in Appendix A.1. The tighter formulation is, henceforth, referred to by FIVF-VI.

3.2. A three-index vehicle flow formulation

The previous formulation has the drawback that the number of variables grows in proportion to the number of vehicles. Alternatively, one can express the routing constraints with variables that do not comprise a vehicle index, in a similar manner to Adulyasak et al. (2014a). The formulation is written using the variables q, z, and x with the same notation, but the vehicle index k is dropped. The only exception is that variable z_{0t}, which is changed to be an integer variable representing the number of vehicles leaving the plant in period $t \in T$. Below, the three-index vehicle flow formulation is provided, henceforward named TIVF:

$$\min f(x,y,z) = \sum_{t\in T}\left(up_t + sy_t + \sum_{i\in N} h_i I_{it} + \sum_{(i,j)\in E} c_{ij}x_{ijt}\right) \tag{2.1}$$

subject to (1.5)–(1.6) and

$$I_{0,t-1} + p_t = \sum_{i\in N^C} q_{it} + I_{0t}, \quad \forall t \in T, \tag{2.2}$$

$$I_{i,t-1} + q_{it} = d_{it} + I_{it}, \quad \forall i \in N^C, \quad t \in T, \tag{2.3}$$

$$p_t \le \min\left\{ C, \sum_{i\in N^C}\sum_{l=t}^{H} d_{il}\right\} y_t, \quad \forall t \in T, \tag{2.4}$$

$$I_{0t} \le L_0, \quad \forall t \in T, \tag{2.5}$$

$$I_{i,t-1} + q_{it} \le L_i, \quad \forall i \in N^C, \quad t \in T, \tag{2.6}$$

$$q_{it} \le \min\left\{ L_i, Q, \sum_{l=t}^{H} d_{il}\right\} z_{it}, \quad \forall i \in N^C, \quad t \in T, \tag{2.7}$$

$$\sum_{j:(i,j)\in E} x_{ijt} = 2z_{it}, \quad \forall i \in N^C, \quad t \in T, \tag{2.8}$$

$$z_{0t} \le m, \quad \forall t \in T, \tag{2.9}$$

$$Q\sum_{(i,j)\in E(S)} x_{ijt} \le \sum_{i\in S}\left(Qz_{it} - q_{it}\right), \quad \forall S \subseteq N^C, \quad |S| \ge 2, \quad t \in T, \tag{2.10}$$

$$p_t, I_{it}, q_{it} \ge 0, \quad \forall i \in N, \quad t \in T. \tag{2.11}$$

$$y_t, z_{it} \in \{0,1\}, \quad \forall i \in N^C, \quad t \in T, \tag{2.12}$$

$$z_{0t} \in \mathbb{Z}^+, \quad \forall t \in T, \tag{2.13}$$

$$x_{ijt} \in \{0,1\}, \quad \forall (i,j) \in E : i \in N^C, \quad t \in T, \tag{2.14}$$

$$x_{0jt} \in \{0,1,2\}, \quad \forall j \in N^C, \quad t \in T. \tag{2.15}$$

Constraints (2.2)–(2.8) are similar to (1.2)–(1.3), (1.6), and (1.9)–(1.10), respectively. Constraints (2.9) limit the number of vehicles leaving the production facility to the number of available vehicles in each period. Constraints (2.10) are the subtour elimination and vehicle capacity constraints. They have a form similar to the generalized fractional subtour elimination constraints (GFSEC) for the VRP, and they are separated using the four heuristic separation algorithm presented by Lysgaard et al. (2004). Besides, the model can be further strengthened by adding valid inequalities. We describe in Appendix A.2 the valid inequalities associated with the TIVF formulation. The tighter formulation is, henceforward, called TIVF-VI.

3.3. A two-commodity flow formulation

The third formulation for solving the basic variant of the PRP is based on the two-commodity flow formulation introduced by Manousakis et al. (2021a). The formulation requires an extended graph $G' = (N', E')$ obtained from G by adding the artificial node $n+1$, duplicating the production facility. Vehicle paths now start from vertex 0 and end at vertex $n+1$. The formulation is written using the variables q, z, and x with the same notation, but the vehicle index k is dropped. In addition, two continuous flow variables f_{ijt} and f_{jit} are introduced for each edge $(i,j) \in E'$ and they represent the load and the residual capacity of the vehicle traveling from i to j at time $t \in T$, respectively. More specifically, for any route of a feasible solution, the flow variables define two directed paths, one from node 0 to $n+1$, whose variables represent the vehicle load, and another from node $n+1$ to 0, whose variables represent the residual capacity of the vehicle. The two-commodity flow formulation is as follows, henceforward referred to as TCF:

$$\min f(x,y,z) = \sum_{t\in T}\left(up_t + sy_t + \sum_{i\in N'} h_i I_{it} + \sum_{(i,j)\in E'} c_{ij} x_{ijt} \right) \tag{3.1}$$

subject to (2.2)–(2.7) and

$$p_t \le \sum_{i\in N^C}\left(\sum_{l=t}^{H} d_{il} - I_{i,t-1}\right) - I_{0,t-1}, \quad \forall t \in T, \tag{3.2}$$

$$\sum_{j:(i,j)\in E} x_{ijt} = 2z_{it}, \quad \forall i \in N^C, \quad t \in T, \tag{3.3}$$

$$\sum_{i\in N^C} x_{0it} \le m, \quad \forall t \in T, \tag{3.4}$$

$$\sum_{j\in N^C} x_{0jt} = \sum_{i\in N^C} x_{i,n+1,t}, \quad \forall t \in T, \tag{3.5}$$

$$f_{ijt} + f_{jit} = Qx_{ijt}, \quad \forall i,j \in N', \quad i<j, \quad t \in T, \tag{3.6}$$

$$\sum_{j\in N, i\neq i} f_{ijt} = Qz_{it} - q_{it}, \quad \forall i \in N^C, \quad t \in T, \tag{3.7}$$

$$\sum_{j\in N^C} f_{0jt} = \sum_{i\in N^C} q_{it}, \quad \forall t \in T, \tag{3.8}$$

$$\sum_{i\in N^C} f_{i,n+1,t} = 0, \quad \forall t \in T, \tag{3.9}$$

$$p_t, I_{it}, q_{it} \ge 0, \quad \forall i \in N', \quad t \in T. \tag{3.10}$$

$$y_t, z_{it} \in \{0,1\}, \quad \forall i \in N^C, \quad t \in T, \tag{3.11}$$

$$f_{ijt} \ge 0, \quad \forall i,j \in N', \quad i \neq j, \quad t \in T, \tag{3.12}$$

$$x_{ijt} \in \{0,1\}, \quad \forall (i,j) \in E', \quad t \in T. \tag{3.13}$$

The objective function (3.1) minimizes the total production, setup, inventory, and routing costs. Constraints (3.2) ensure that the produced quantities together with the remaining plant facility inventory at any time period $t \in T$ cannot exceed the actual customers' stocks. These constraints serve to further tighten the formulation, and they were presented in the previous formulations. Constraints (3.3) are the degree constraints for the customers. Constraints (3.4) ensure that the number of vehicles leaving the plant facility are equal to those returning to it, and their number should not exceed the fleet size as defined by constraints (3.5). Constraints (3.6) and (3.7) define the relationship between vehicle flow and commodity flow variables. Constraints (3.8) and (3.9) impose the correct values for the commodity flow variables incident to the plant facility vertices. We emphasize that the TCF formulation directly implies the subtour elimination constraints from the flow reformulated constraints (3.6)–(3.9). Finally, constraints (3.10)–(3.13) set the domains and the integrality of the decision variables. For completeness, we describe the valid inequalities associated with the TCF formulation in Appendix A.3, and which were originally presented by Manousakis et al. (2021a). The tighter formulation is, henceforth, referred to by TCF-VI.

4. Solution method

To solve the PRP, we propose a matheuristic that is composed of three phases: an initialization phase, a completion phase, and an improvement phase. In Phase I, we run a MIP relaxation of the PRP. Phase II takes as an input the partial solution obtained from the previous stage and completes it using a sequence of heuristics. Once a complete solution is obtained, a weighted proximity search algorithm is invoked in Phase III to improve the incumbent solution. In the following, we describe each phase in detail. The outline of the overall matheuristic is displayed in Fig. 1.

4.1. Initialization phase

In Phase I, we decide when to produce and how much to produce by solving a relaxation of the PRP model. In the relaxed form, the integrality conditions on the customer visit variables z and the vehicle routing variables x are relaxed. Only the integrality of the production variables y is retained. Subtour elimination constraints are removed from the formulations, but for the FIVF and the TIVF formulations they are dynamically inserted when violated. Note that maintaining the routing and distribution requirements in a relaxed form ensures that there can exist a feasible compatible solution to the y–variables (in terms of vehicle capacities and stock-out at customers); however, it does not ensure that deliveries do not split. This follows from the facts that the q-variables satisfy the vehicle capacity, the maximum level inventory policy requirements, and the constraints that disallow stock-outs. The optimal value of the relaxed model yields a lower bound on the original model's optimal value.

4.2. Completion phase

At this stage, decisions about visit times, customers' assignment to vehicles, and vehicle routes must be determined to obtain a complete solution to the PRP. These decisions are made by solving a sequence of heuristics. The obtained solution may be infeasible in terms of vehicle capacity and stock-out constraints at the production plant. However, it can be used to initialize Phase III even if it is infeasible. We hence proceed as follows:

1. Assign customer to time periods and set the quantities to be delivered to each customer in each time period using a construction heuristic, that we refer to as the customer urgency heuristic.
2. Assign customers to vehicles by applying a modified version of the first fit decreasing bin packing algorithm (Johnson et al., 1974).
3. Determine the route of each vehicle on each day by solving a TSP on the subset of customers visited by each vehicle and using the Lin–Kernighan algorithm (Lin and Kernighan, 1973).

A detailed description of the customer urgency heuristic, and the modified fit decreasing bin packing algorithm is provided in Appendix B.

4.3. Improvement phase

The improvement phase includes a weighted proximity search (WPS) algorithm and a feasibility recovery procedure. The latter is invoked only if the starting solution is infeasible, and it works by eliminating violations of vehicle capacities and stock-outs at the production plant. Both components are described below.

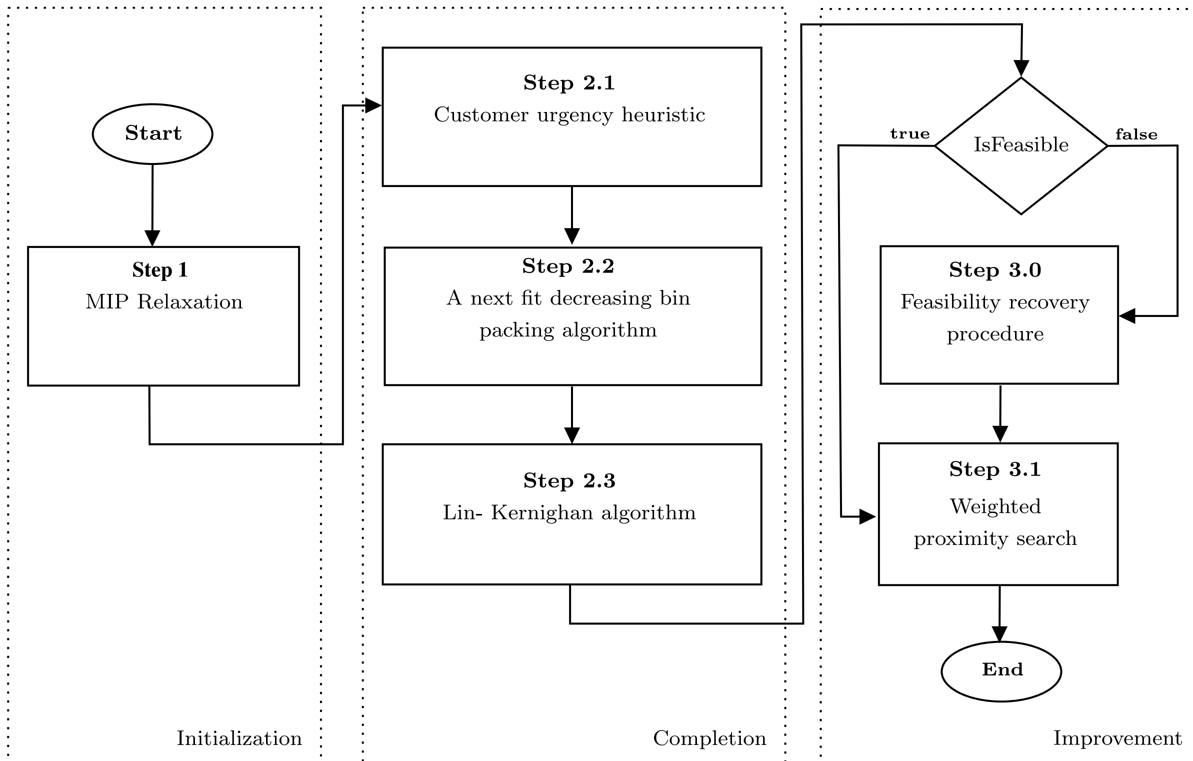


Fig. 1. Matheuristic for the production routing problem.

4.3.1. *Weighted proximity search*

The improvement phase is based on the WPS algorithm (Rodrigues et al., 2021), which is adapted from the classic proximity search proposed by Fischetti and Monaci (2014). The WPS aims to improve a given solution to the PRP, referred to as $(\bar{x}, \bar{y}, \bar{z})$, by looking within the search space for an improved solution with a fixed minimum amount. This can be done by replacing the problem's original objective function with a Hamming distance function centered on the given solution. The new objective function, henceforth called the proximity function, is defined by:

$$\Delta_w^x(x, \bar{x}) := \sum_{t \in T} \sum_{(i,j) \in E | \bar{x}_{ijt}=0} w_{ijt}^x x_{ijt} + \sum_{t \in T} \sum_{(i,j) \in E | \bar{x}_{ijt}=1} w_{ijt}^x (1 - x_{ijt}), \quad (4.1)$$

$$\Delta_w^y(y, \bar{y}) := \sum_{t \in T | \bar{y}_t=0} w_t^y y_t + \sum_{t \in T | \bar{y}_t=1} w_t^y (1 - y_t), \quad (4.2)$$

$$\Delta_w^z(z, \bar{z}) := \sum_{t \in T} \sum_{i \in N^C | \bar{z}_{it}=0} w_{it}^z z_{it} + \sum_{t \in T} \sum_{i \in N^C | \bar{z}_{it}=1} w_{it}^z (1 - z_{it}), \quad (4.3)$$

$$\min \Delta_w := \Delta_w^x(x, \bar{x}) + \Delta_w^y(y, \bar{y}) + \Delta_w^z(z, \bar{z}). \quad (4.4)$$

where $w_{ijt}^x \in \mathbb{R}$ represent the weights of variables $x_{ijt}, (i,j) \in E, t \in T$, $w_t^y \in \mathbb{R}$ the weights of variables $y_t, t \in T$, and $w_i^z \in \mathbb{R}$ the weights of variables $z_{it}, i \in N^C, t \in T$, respectively. These weights intend to capture information about which variables are more promising to change to find an improved solution. Therefore, given a feasible solution, higher weights should be assigned to the less promising variables to change, while lower weights should be associated with the more promising variables to change.

Hence, by using the weighted Hamming distance function (4.4), the WPS can be described as follows. We start with the initial solution, obtained from Phase II. Then, at each iteration, the cutoff constraint:

$$f(x, y, z) \leq f(\bar{x}, \bar{y}, \bar{z}) - \theta \quad (4.5)$$

(depending on a given cutoff tolerance value $\theta > 0$) is added to the MIP problem. The cut-off constraint (4.5) limits the search to solutions improving the incumbent solution's cost by the specified tolerance value. The problem's original objective is replaced by the Hamming distance function (4.4) to promote finding nearby better solutions, and the resulting model is solved using a MIP solver. The obtained solution is then used to recenter the Hamming distance function and define the new cutoff constraint, and the process is repeated until a given stopping criterion is reached. Algorithm 1 describes the steps of the WPS.

Algorithm 1: The weighted proximity search algorithm

```
Let (x^0, y^0, z^0) be an initial feasible solution.
Set (x̄) := (x^0), (ȳ) := (y^0), and (z̄) := (z^0),
repeat
    add the cutoff constraint (4.5) to the MIP problem
    replace the original objective function by the Hamming
      distance function (4.4), and add equations (4.1)–(4.3) to
      the model
    run the MIP solver on the modified problem
    if a new feasible solution (x̄, ȳ, z̄) is found then
        recenter Δ_w at the new solution (x̄, ȳ, z̄)
        update weights' calculations
    else
        go to 15
    end
    update θ (facultative)
until until a termination criteria is reached;
return (x̄, ȳ, z̄)
```

4.3.2. *Feasibility recovery procedure*

The feasibility recovery procedure is based on the idea that the WPS (see Section 4.3.1) can be extended to eliminate violations of vehicle capacities and stock-outs constraints at the plant. This is done by identifying in the incumbent solution customers and vehicles yielding an infeasibility and adding a cut-off constraint to the WPS model to reduce the amount of these infeasibilities. Prior to describing the recovery procedure, we introduce the following notation. Given an infeasible solution $(\bar{x}, \bar{y}, \bar{z})$, let $\mathcal{R}_t$ denote the set of routes of $(\bar{x}, \bar{y}, \bar{z})$ that yield an excess vehicle load over the vehicle capacity Q in each period $t \in T$. We introduce a_{itr} a parameter that takes value 1 if customer $i \in N^C$ is visited by route $r \in \mathcal{R}_t, t \in T$, and 0 otherwise. We define $\xi \in \mathbb{R}$, a variable that represents the amount exceeding the vehicle load, computed as follows:

$$\xi = \sum_{r \in R} \left(\sum_{i \in N^C} a_{itr} q_{it} \right) - Q.$$

Also, let $\mathcal{T} \subseteq T$ denote the set of time periods when a stock-out occurs at the plant node 0. Similarly, let $\psi \in \mathbb{R}$, be a variable that defines the amount for stock-outs at the plant, obtained by:

$$\psi = \sum_{t \in \mathcal{T}} -I_{0t}.$$

Hence, a new cut-off constraint (4.6) that replaces the original one (4.5) is appended to the MIP model for the sake of reducing the infeasibility:

$$\xi + \psi \leq \bar{\xi} + \bar{\psi} - \theta_f. \tag{4.6}$$

where $\theta_f > 0$ is a given cutoff tolerance, $\bar{\xi}$ is the constant amount associated with vehicle excess in $(\bar{x}, \bar{y}, \bar{z})$, and $\bar{\psi}$ is the constant amount associated with stock-outs at the plant in $(\bar{x}, \bar{y}, \bar{z})$, respectively. The resulting model is solved using a MIP solver. One needs to run this process only once since the vehicle capacity and the inventory constraints will enforce a feasible solution. The cutoff constraint (4.6) helps, in this regard, the MIP solver to repair infeasibilities within short computing times by increasing the *relaxation grip* of the formulation (Fischetti and Fischetti, 2018). Finally, to ensure that the recovery procedure does not deteriorate the solution quality, an additional cut-off constraint (4.7), defined by:

$$f(x, y, z) \leq f(\bar{x}, \bar{y}, \bar{z}) + \theta_q. \tag{4.7}$$

is appended to the MIP model, and where $\theta_q > 0$ is a given quality deviation value.

4.3.3. Weight calculations

Here we explain how the weights associated with the proximity search binary variables are computed. We use a combination of the *Linear Relaxation Proximity (LRP)* approach and the *Three-Value System (3-VS)* approach, both introduced by Rodrigues et al. (2021). By applying the LPR, the weight w_j associated with any generic variable x_j can be expressed at each iteration by:

$$w_j = 1 - |\bar{x}_j - x_j^{LP}|.$$

where x_j^{LP} is the optimal value taken by x_j in the model's LP-relaxation, and $\bar{x}_j$ its integer value in the incumbent solution. Then, a variable x_j whose value in the current solution is closer to its value in the LP-relaxation yields a higher weight. Hence, the weights associated with the x−variables, the y−variables, and the z−variables in Eq. (4.4) are given by:

$$w_{ijt}^x = 1 - |\bar{x}_{ijt} - x_{ijt}^{LP}|, \quad \forall (i,j) \in E, \quad t \in T,$$
$$w_t^y = 1 - |\bar{y}_t - y_t^{LP}|, \quad \forall t \in T,$$
$$w_{it}^z = 1 - |\bar{z}_{it} - z_{it}^{LP}|, \quad \forall i \in N, \quad t \in T.$$

The LRP approach performs well for optimization problems with an optimal solution close to the solution of the LP-relaxation (Rodrigues et al., 2021). It is not computationally intensive, since the required computational time corresponds only to the solution time of the LP-relaxation.

Remark 1. Given that: (*i*) we are solving a relaxation of the PRP model in Phase I of our matheuristic, (*ii*) the integrality of the production variables is maintained in this relaxation, and (*iii*) the production costs are usually the major cost component in a PRP setting, it would be natural to assume that the objective function of the solution obtained in Phase I is relatively close to the optimal one; thus it could be effectively used to compute the proximity search weights. In doing so, the weights associated with the y−variables will turn to zero.

Next, we apply the 3-VS discretization approach to transform the problem's weights into discrete values. This is done in two steps. First, the weights are discretized into R different values $\{1, \ldots, R\}$, where R is an integer number greater than one defined a priori. The weights associated with the x−variables can be transformed as follows:

$$w_{ijt}^x = (R_x + 1) - \left\lceil R_x - \frac{(w_{ijt}^x - w^{\min,x})(R_x - 1)}{w^{\max,x} - w^{\min,x}} \right\rceil, \quad \forall (i,j) \in E, \quad t \in T.$$

where $w^{\min,x} = \min\{w_{ijt}^x : \forall (i,j) \in E, t \in T\}$ and $w^{\max,x} = \max\{w_{ijt}^x : \forall (i,j) \in E, t \in T\}$. The resulting weights are next converted into a three-value system as follows:

$$w_{ijt}^x = \begin{cases} 1, & \text{if } 1 \leq w_{ijt}^x < t_2^x, \\ R_x/2, & \text{if } t_2^x \leq w_{ijt}^x < t_3^x, \\ R_x, & \text{if } t_3^x \leq w_{ijt}^x < R_x, \end{cases} \quad \forall (i,j) \in E, \quad t \in T.$$

where t_2^x and t_3^x are integer threshold values between 1 and R defined a priori such that $t_2^x \leq t_3^x$. Similarly, we proceed for the z−variables to obtain:

$$w_{it}^z = \begin{cases} 1, & \text{if } 1 \leq w_{it}^z < t_2^z, \\ R_z/2, & \text{if } t_2^z \leq w_{it}^z < t_3^z, \\ R_z, & \text{if } t_3^z \leq w_{it}^z < R_z, \end{cases} \quad \forall i \in N, \quad t \in T.$$

where t_2^z and t_3^z are integer threshold values between 1 and R and $t_2^x \leq t_3^x$. With respect to the y−variables whose initial weights are equal to 0, we set their corresponding weights as follows:

$$w_{it}^y = \begin{cases} 1, & \text{if } \bar{y}_t \neq y_t^{LP}, \\ R_y, & \text{otherwise,} \end{cases} \quad \forall t \in T.$$

Remark 2. Variables' weights under their initial continuous representation would induce a computationally hard model (Rodrigues et al., 2021) and thus, must be mapped into discrete values using the 3-VS discretization approach. To avoid any loss of quality due to the approximation, the choice of the parameters t_2 and t_3 is extensively studied in the computational section.

5. Computational experiments

Computational experiments were carried out using a Lenovo NextScale nx360 M5 machine, with a dual 2.3 GHz Intel Xeon E5-2670v3 processor with 64 GB RAM. The matheuristic was implemented in C++, compiled with the Linux compiler g++ in release mode. The mathematical components were solved using CPLEX 12.10 with the default settings, except when solving the proximity search subproblems, where we emphasize the search toward finding early feasible solutions.

5.1. Experimental setup

Our computation experiments follow three main goals. First, we evaluate the impact of mathematical formulations within a matheuristic framework by comparing three state-of-the-art formulations for the PRP. Second, we expand our analysis by assessing the effect of valid inequalities using the considered formulations. Finally, we evaluate the matheuristic's convergence behavior with different starting solutions. For these analyses, we create a set of experiments by selecting, in each trial, a distinct mathematical formulation to be used by the mathematical components (namely, Phase I & Phase III) of the matheuristic described in Section 4. Hence, the matheuristic is run once using each of the $3^2 = 9$ combinations of formulations. By further considering the use of valid inequalities, the number of experiments is doubled,

Table 1
Comparison of the matheuristic's performance for different schemes of $3-VS$ discretization.

Scheme	Average gap (in %)	Average time (in s.)
PS	0.63	4980
$3-VS_{(20,40)}$	0.52	5184
$3-VS_{(30-60)}$	0.37	5509
$3-VS_{(40-80)}$	0.5	4594

yielding a total of $9\times 2 = 18$ variations of the matheuristic. The tests are conducted on benchmark data instances for the PRP (Adulyasak et al., 2014a). The data set contains 168 instances with N^C from 10 to 50 and with time horizons of $H = 3$, 6, and 9 periods, respectively. The number of vehicles is set to either $m = 2$ or $m = 3$ for the instances with $N \leq 25$ and to $m = 3$ or $m = 4$ for the rest of instances.

5.2. Choice of parameters and calibration

This section explains the choice of the parameters used in our computational experiments. First, the matheuristic method has a computational time limit of 4 h, of which 600 s is the time limit imposed on Phase I. No limit was imposed on the total number of iterations in Phase III; however, each iteration stops when the first feasible solution is found (Fischetti and Monaci, 2014). Besides, the duration of every single iteration was restricted to one hour at maximum. We consider cut-off tolerances of $\theta = 1$ and $\theta_f = 1$, respectively. Rodrigues et al. (2021) demonstrated that a unitary value of the cut-off tolerance performs better than higher values. For the quality deviation, using $\theta_q = 5\%$ of the starting solution's objective value led to subproblems that were relatively quickly solved, while leading to the first feasible solution being found having a comparable objective function value as the starting solution.

The discretization scheme described in Section 4.3.3 requires defining a maximum possible value for the weights R. We set $R = R_x = R_z = 100$ for both x and z variables, whereas we assign a higher value $R_y = 1000$ for the y−variables to restrict the change in their values during the WPS. Next, we choose the parameters t_2 and t_3 of the $3-VS$ discretization approach. Three combinations of values are tested: (20,40), (30,60), and (40,80). Preliminary experiments for each setting of values were carried out on a training set of 27 instances. Moreover, we tested the classic proximity search (PS) heuristic where all weights are equal to one. Table 1 displays the average obtained results. The first column defines the discretization scheme, while the following columns present the average gap $\frac{Z^M - Z^{best}}{Z^{best}}$ to the best-known upper bound and the average running time used by the matheuristic, respectively.

The results indicate that the WPS approach is instrumental to improve the matheuristic performance. Under all discretization schemes, it provides better results on the set of selected instances than the standard PS without a significant increase in the computational time. According to Table 1, the lowest average gap corresponds to (t_2, t_3)= (30,60), which will be selected in our study.

5.3. Main results

In this section, we provide the main results derived from the set of tests performed on Adulyasak et al. (2014a) instances. Table 2 describes, in columns 1–3, the configuration of each conducted test, that is to say, the formulation implemented in each mathematical component of the matheuristic. Then, for each test, we report the average percentage deviation of the MIP-relaxation bound (MIP gap) with respect to the value of the best-known upper bound and the average computing time in seconds. Columns 4 and 5 display the average percentage deviation of the constructive solution objective to the best-known upper bound (Cnst. gap) and the MIP-relaxation bound (LB gap), respectively. The computing time of Phase II is always negligible (a few milliseconds) and thus not reported. Columns 6–8 relate to the results derived from Phase III of our matheuristic. They display the average percentage deviation of the final solution's objective to the best-known upper bound (Final gap), the average percentage improvement (% impr.) with respect to the starting solution, and the average computing time in seconds, respectively. In all columns of Table 2, we put boldface letters on the shortest average computing time and average deviation to the best-known upper bounds; otherwise, boldface letters are used for the best average improvement to the starting solution.

Table 2 abounds with interesting observations. A first observation is that the TCF formulation produces, on average, tighter MIP relaxations than the other formulations, which are at most within 5.75% of the best-known integer bound. Besides, the FIVF formulation outperforms the TIVF formulations in this regard. This observation extends the findings of Adulyasak et al. (2014a) and Archetti et al. (2014), which showed that the FIVF formulation is superior to the TIVF one in terms of root node lower bounds and optimality gaps. The relaxed form of the TIVF formulation, in general, converges faster than the alternative formulations. A further observation is that the relaxed representation does not produce the same results across all runs (for example tests four and five). This issue is related to floating point arithmetic, where the rounding of continuous values is unpredictable and can cause the same program to produce different results on the same operating machine/system (Klotz, 2014).

The matheuristic builds initial solutions always within at most 10% of the best primal bound. We see that the initial solutions produced by the TCF formulation yield a better optimality gap than those derived from alternative formulations, which is, on average, 8.84% far from the best integer bound. Additionally, the gap between the starting solution and the MIP relaxation, given by *Cnst. gap*, is relatively high. It increases to up to 22.28% for solutions derived from the TIVF formulation, while it is lower for those obtained by the FIVF formulation, averaging up to 20.88%. However, the starting solutions originating from the TCF formulation deviate less from their MIP relaxation by up to 14.41% on average. On one side, these observations explain that the quality of the produced initial solution is highly contingent on the mathematical formulation used. On the other side, they show that the TCF formulation fits better with the constructive heuristic associated with our matheuristic.

Next, we are interested in evaluating the quality of the final solutions derived from Phase III. Surprisingly, the FIVF formulation, when used to guide the WPS, provides the best performance on the benchmark PRP instances where the deviations from the optimal solutions or best upper bounds range between 1.15% and 1.61%. Better performance is achieved when valid inequalities are considered. The percentage improvement with respect to the starting solutions is significant and increases to 7.16% on average. The results obtained from the alternative formulations are slightly worse, where gaps of the best upper bound for the TIVF and TCF formulations are at most 2.29% and 2.21%, respectively; the corresponding average percentage improvements are 5.73% and 5.89%, respectively. On average, the execution times are lower when the TIVF formulation is used to drive the search. However, this behavior is a direct consequence of the non-improvement termination criterion imposed during the WPS. Clearly, after a few iterations, it becomes difficult for the TIVF formulation to improve the incumbent solution, thus, forcing the matheuristic to stop. The matheuristic does not necessarily perform better with good initial solutions, although their choice may affect the search quality. We further investigate this aspect in the following sections. Moreover, our experiments showed that Phase III often starts with an infeasible solution in terms of vehicle capacities and stock-outs at the production plant. The feasibility recovery procedure was capable of quickly restoring feasibility (in a few seconds or less) without significantly increasing the objective function value.

Table 2
Average results obtained from the tests performed on Adulyasak et al. (2014a) instances.

Test #	Configuration		Phase I		Phase II		Phase III		
	Phase I	Phase III	MIP gap (%)	CPU (s)	Cnst. gap (%)	LB gap (%)	Final gap (%)	% impr.	CPU (s)
1	FIVF-VI	FIVF-VI	7.16	2.51	9.77	19.41	1.26	**7.16**	8,396
2		TCF-VI	7.16	2.51	9.77	19.41	1.49	6.98	8,798
3		TIVF-VI	7.16	2.55	9.79	19.43	1.91	6.13	**4,724**
4	FIVF	FIVF	8.35	0.46	9.50	20.83	1.50	6.81	8,138
5		TCF	8.35	0.45	9.50	20.83	2.12	6.31	8,262
6		TIVF	8.33	0.45	9.57	20.88	2.13	6.64	5,606
7	TIVF-VI	TIVF-VI	7.22	0.24	9.30	18.94	1.95	6.23	5,168
8		TCF-VI	7.16	0.26	9.35	18.91	1.33	6.85	8,580
9		FIVF-VI	7.16	0.27	9.35	18.91	**1.15**	6.99	8,482
10	TIVF	TIVF	9.54	0.07	9.03	22.23	2.29	5.73	5,494
11		TCF	9.56	**0.06**	9.06	22.28	2.21	5.89	7,949
12		FIVF	9.56	**0.06**	9.06	22.28	1.42	6.52	8,425
13	TCF-VI	TCF-VI	**4.48**	3.01	8.89	14.41	1.60	6.33	8,669
14		FIVF-VI	**4.48**	3.00	8.89	14.41	1.37	6.50	8,358
15		TIVF-VI	**4.48**	3.04	**8.84**	**14.35**	1.27	6.48	5,274
16	TCF	TCF	5.76	0.36	9.48	16.83	2.07	6.38	8,058
17		FIVF	5.76	0.36	9.48	16.83	1.61	6.74	7,867
18		TIVF	5.74	0.38	9.48	16.79	1.59	6.73	5,477

* Indicates the number of new best bounds found, or matched out of 168 instances.

Table 3
Benchmark algorithms for Adulyasak et al. (2014a) instances.

Reference	Abbreviation	Sol	CPU	Threads	Solver	Runtime (in s.)
Schenekemberg et al. (2021)	SSPG-BC	E	Intel (R) Xeon (R) 2.60 GHz	6	Gurobi 8.1	5,700.8
Adulyasak et al. (2014a)	ACJ-BC	E	AMD Opteron 2.40 GHZ	8	Cplex 12.3	15,785
Adulyasak et al. (2014b)	ACJ-ALNS	H	2.10 CPU PC Duo	Default	Cplex 12.2	30.4
Vadseth et al. (2023)	VACS-M	M	Intel Xeon Gold 6144 3.5 GHz	1	Gurobi 9.1	21
This paper	BHA-M	M	Intel Xeon E5-2670 2.3 GHz	1	Cplex 12.10	25,236

Note: E = Exact method, H = Heuristic, M = Matheuristic.

5.4. Comparison to the state-of-the-art approaches

In this section, we compare our aggregated results, obtained from tests scenarios {1,9,14}, to previous studies that have tackled the same instances. These studies include the branch and cut algorithms (B&C) of Adulyasak et al. (2014a) and Schenekemberg et al. (2021), the adaptive large neighborhood (ALNS) algorithm of Adulyasak et al. (2014b), and the multi-start matheuristic of Vadseth et al. (2023), which are referred to as ACJ-BC, SSPG-BC, ACJ-ALNS, and VACS-M, respectively. Our matheuristic is named BHA-M. In Table 3, we summarize the essential features of these solution methods. The table displays, for each method, the abbreviation, the solution type, the CPU machine's characteristic, the number of used threads, the solver, as well as the average runtime. However, an entirely precise comparison of these methods' performances is never possible due to different operating systems, programming languages, and runtimes.

Table 4 provides a comparison of the performance of these methods. It indicates the number of instances where each method produces an equal or better objective than the others. Instances with a strictly better objective are shown in parentheses. The bottom two rows indicate the number of best known solutions (BKS), and the average gap to the BKSs obtained by each method. A more detailed comparison can be found in Appendix C.

We see from Table 4 that our matheuristic largely outperforms the VACS-M matheuristic in terms of the number of best known solutions (BKS) found or improved. Surprisingly, VACS-M yields a slightly better optimality gap than ours. Taking a closer look at the detailed results provided in Appendix C, we observe that VACS-M, over most test instances, yields higher objective function values, except on a few large instances (particularly with $N \geq 45, 50$). The disparity in performance between VACS-M and our method over these instances is significant. However, the average gap is skewed towards VACS-M,

Table 4
Performance comparison between different solution methods on Adulyasak et al. (2014a) instances.

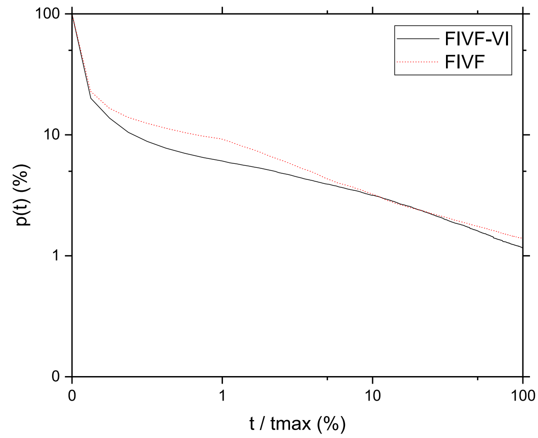
	ACJ-BC	SSPG-BC	ACJ-ALNS	VACS-M	BHA-M
ACJ-BC	*	130(6)	168(155)	148(113)	157(61)
SSPG-BC	162(38)	*	167(160)	162(126)	165(68)
ACJ-ALNS	13(0)	8(1)	*	48(28)	48(17)
VACS-M	55(20)	42(6)	154(144)	*	75(42)
BHA-M	107(11)	90(3)	151(144)	126(93)	*
#BKS	128	158	8	41	95
Gap (%)	0.13	0.005	1.39	0.39	0.43

which is expected considering that our matheuristic, which relies on a complete mathematical formulation of the PRP, struggles with an increased number of customers and time periods. VACS-M proposed a decomposition-based matheuristic that overcomes this issue, however, it behaves poorly on the small and medium instances. Our matheuristic also largely outperforms the ACJ-ALNS metaheuristic.

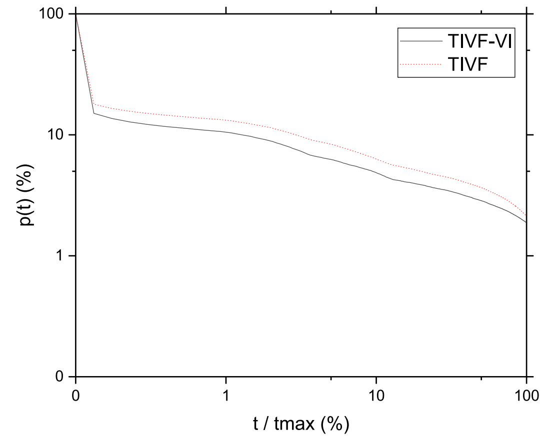
The branch-and-cut algorithms ACJ-BC and SSPG-BC, successfully tackled these instances, but they had to rely on larger CPU times and use parallel runs with several threads (8 and 6 threads). Yet, our matheuristic was able to improve upon them in a few instances and match their performance oi many other instances.

5.5. Experimental analysis

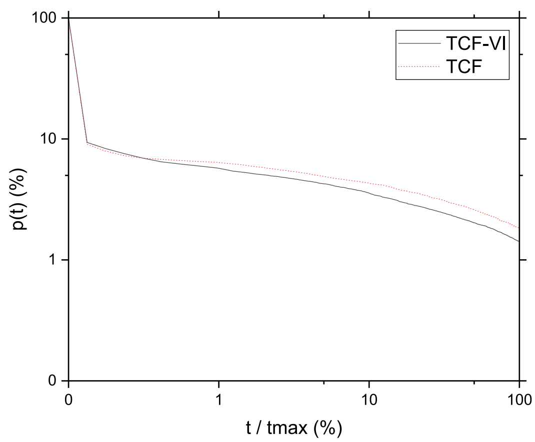
In this section, we analyze the factors influencing the matheuristic's performance. Toward this aim, we use the primal gap of Berthold (2013) as a performance measure. Given an incumbent solution (x, y, z) with the objective function value Z, and assume that an optimal (or best-known solution) is given with the objective function value Z^{best},



(a) FIVF formulation.



(b) TIVF formulation.



(c) TCF formulation.

Fig. 2. Course of the primal gap when running the matheuristic with and without valid inequalities.

the primal gap of (x, y, z) is defined as

$$\gamma(x,y,z) = \begin{cases} 0, & \text{if } Z = Z^{best}, \\ 1, & \text{if } (x,y,z) \text{ is infeasible}, \\ |Z^{best} - Z| / \max\{|Z^{best}|, |Z|\}, & \text{otherwise.} \end{cases}$$

Then, assuming that we are given the objective function values of intermediate incumbent solutions and the points in time when they have been found, we define the primal gap function $p(t)$ as

$$p(t) = \begin{cases} 1, & \text{if no feasible solution has been found} \\ & \quad \text{until point } t, \text{and otherwise} \\ \gamma(x_t, y_t, z_t), & \text{with } (x_t, y_t, z_t) \text{ being the incumbent} \\ & \quad \text{solution at point } t. \end{cases}$$

The function $p(t)$ is, by definition, a step function that measures the quality of the solution found over time, and it changes whenever a new incumbent solution is found. It monotonically decreases and converges to 0 if an optimal solution is found.

5.5.1. Effect of valid inequalities

As a first analysis, we compare the performance of the matheuristic using the primal gaps when running with and without valid inequalities. Comparisons are carried out for different options of mathematical formulations, thus contrasting the results of test scenarios (described in Table 2) 1 to 4, 7 to 10, and 13 to 16, respectively. We show in Fig. 2 the evolution of the primal gap over time. The figure is based on the average result of all instances. The running time of each instance is expressed as a percentage of the maximum permitted running time (t^{max} = 4 hours). Fig. 2 has a logarithmic representation, which helps to highlight the scale of performance variations between the tested scenarios.

Fig. 2 demonstrates the importance of valid inequalities within a matheuristic search. The convergence of the primal gap when using valid inequalities (represented by the solid black line in the figure) is faster overall than without using valid inequalities (represented by the dotted red line). However, a limited variability is observed in the case of the FIVF-embedded matheuristic. The plots' trajectories are indistinguishable for minimal values of (t/t^{max}), which indicates that the effect of valid inequalities is marginal for: *(i)* small-sized instances (those that do not require a significant computational time), and *(ii)* in the very early stages of heuristic search. Hence, one may conclude that using valid inequalities within a matheuristic setting is overall superior in terms of the primal gap. For this reason, only experiments with valid inequalities are considered subsequently.

5.5.2. Effect of mathematical formulations

The second experiment examines whether the choice of mathematical formulations affects algorithmic convergence. Hence, contributions

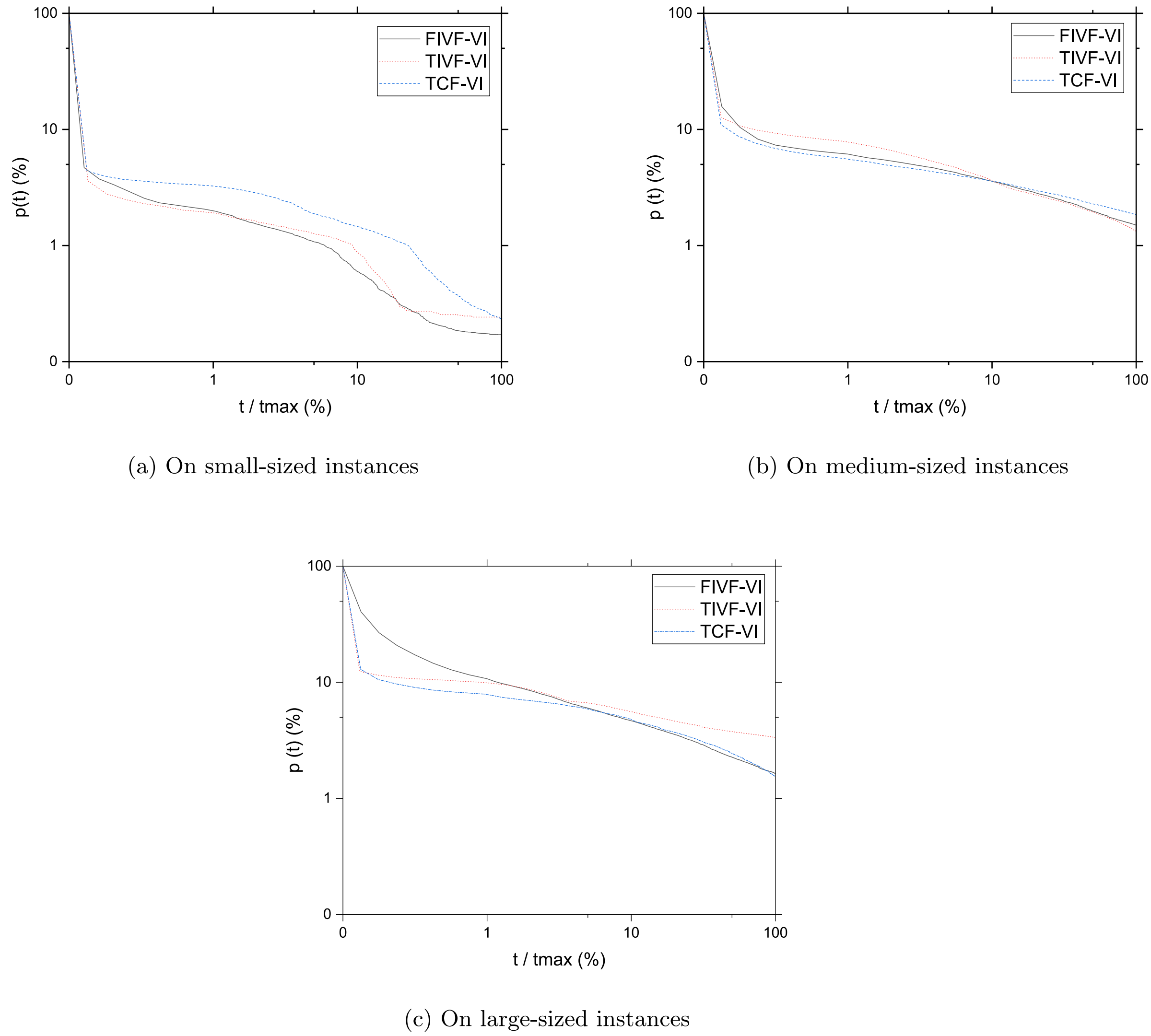


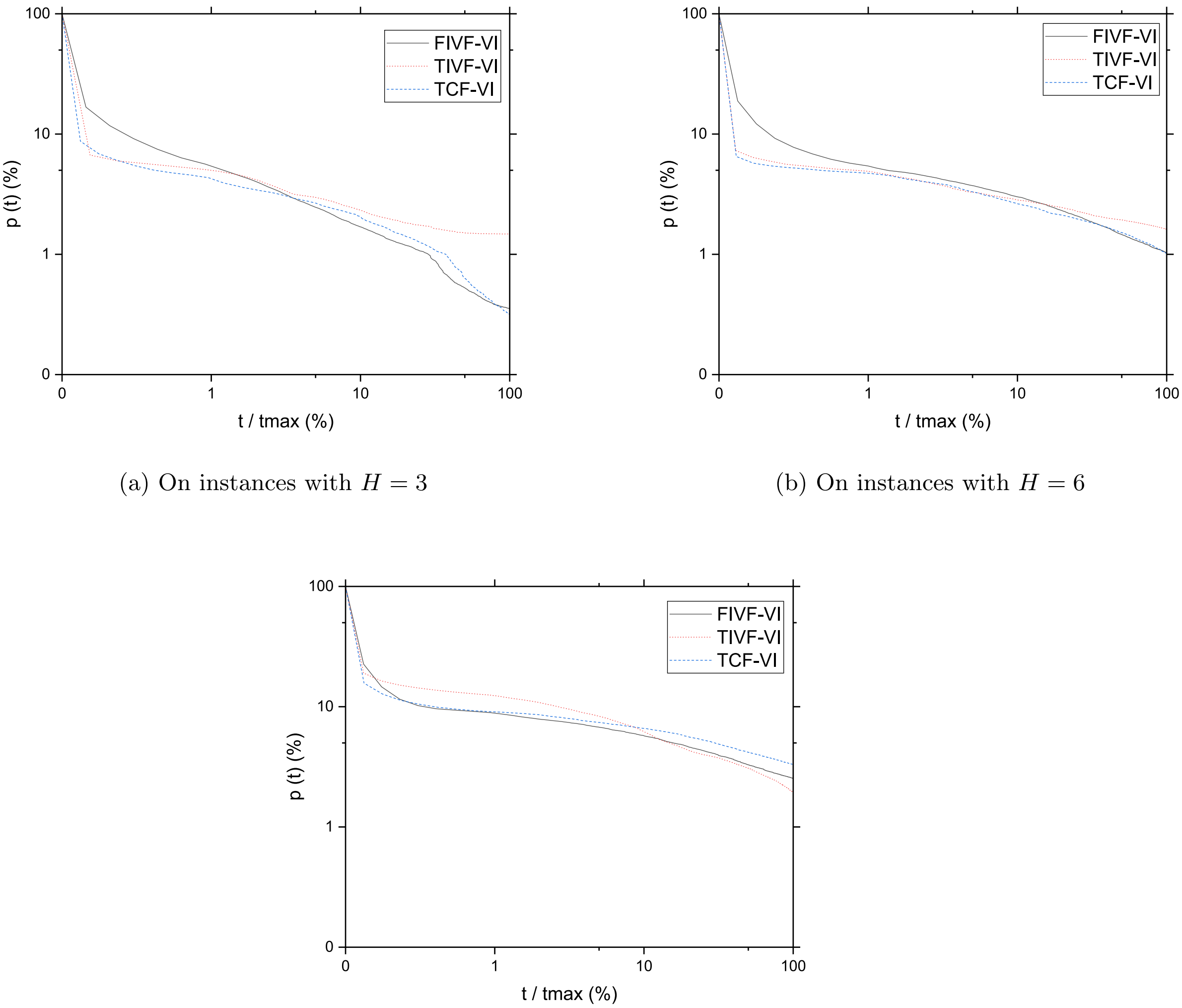
Fig. 3. Course of the primal gap when running the matheuristic with different mathematical formulations–Comparison by instance size.

gained from the mathematical formulations embedded within the WPS (Phase III) are weighted against each other. Precisely, we are contrasting the average results obtained from tests scenarios {1,9,14} against those obtained from {2,8,13}, and {3,7,14}, respectively. For a comprehensive evaluation, analyses are carried out separately on data instances with comparable numbers of customers and planning horizon lengths. First, instances are arranged into the following subsets:

- Small: First 48 instances including 10 to 15 customers.
- Medium: Next 72 instances including 20 to 30 customers.
- Large: Last 48 instances including 35 to 50 customers.

Fig. 3 compares the course of the primal gap obtained by the matheuristic for each subset of instances and where different mathematical formulations are implemented within the WPS. The solid black line corresponds to the course of the average primal gap function (taken over 168 instances) when running the matheuristic with the FIVF-VI formulation. Additionally, the dotted red line and the dashed blue line correspond to the course of the average primal gap function when running the matheuristic with the TIVF-VI and the TCF-VI formulations, respectively. On the subset of small-sized instances, we observe that the three implementations find good-quality solutions (the average final primal gap is less than 1%) and that the FIVF-implementation acts more effectively almost at any point of the time (for $t/t^{max} \geq 1\%$). The TCF implementation performs poorly in general, converging toward good-quality solutions only later in the run. Regarding the subset of medium-sized instances, it is difficult to establish a clear dominance among the three formulations, although the TIVF-implementation led to the best improvement in the primal gap on average. The TIVF-implementation is inferior to the values reported on the subset of small-sized instances, which was expected considering the added complexity caused by the increased number of customers. Regarding the subset of large instances, the TIVF-embedded matheuristic quickly finds a good-quality solution but fails later to achieve significant improvements. The TIVF formulation does not appear to be competitive in large-sized instances as widely claimed in the literature. The TCF-embedded matheuristic is slightly better, particularly in the early and the late stages of the search, which could explain its ability to improve the best-known upper bounds (Table 2). Looking also at Table 6 in Appendix B, we see that many of the improved best-known upper bounds belonged to the subset of large-sized instances and were obtained using a TCF formulation.

Next, our comparisons are conducted on new subsets of instances, arranged following the length of their planning horizon. The subsets include 72 instances with $H = 3$, 56 instances with $H = 6$, and 40 instances with $H = 9$, respectively. Fig. 4 shows the course of the primal gap recorded by the matheuristic on each subset of instances and with different implemented mathematical formulations in Phase III.



(a) On instances with $H = 3$

(b) On instances with $H = 6$

(c) On instances with $H = 9$

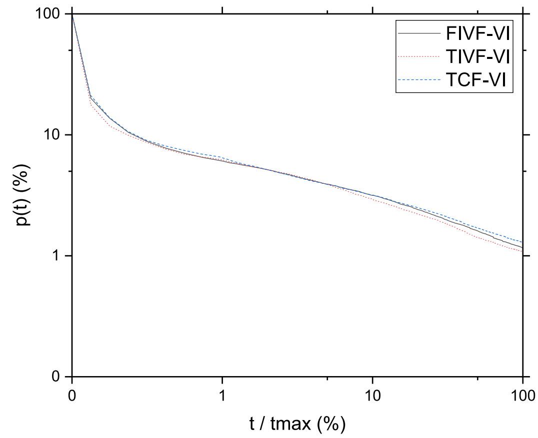
Fig. 4. Course of the primal gap when running the matheuristic with different mathematical formulations – Comparison by the planning horizon length.

As visible in Fig. 4, the TIVF-implementation yields the worst performance among the three implementations in instances with short planning horizons. The implementation's performance is worsened for higher values of (t/t^{max}) due to its deficiency on large-sized instances (as shown previously). Recall that higher values of (t/t^{max}) correspond primarily to instances with many customers. Such interference is further corroborated when looking at the plot associated with the TCF implementation, which records the lowest primal gap for high values of (t/t^{max}). The FIVF-implementation has a consistent behavior regardless of (t/t^{max}) values (by inference, of customers' number), which might suggest its usefulness on this subset of instances. On the subset of instances with $H = 6$, the same assessment as before for the TIVF-implementation still holds, while both the FIVF and the TCF implementations exhibit better behaviors, intriguingly proportionate. Larger planning horizons have a more pronounced effect on the FIVF and the TIVF implementations whose performances degraded, while the TIVF-implementation is seemingly more efficacious on instances with $H = 9$.

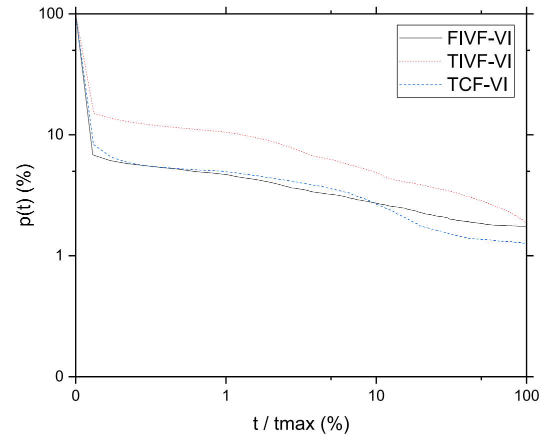
5.5.3. Effect of initial solutions

As a final experiment, we investigate whether the starting solution's quality significantly affects the performance of the matheuristic. We test the matheuristic with three starting solutions, each derived from the MIP relaxation of a different PRP formulation. In all cases, the mathematical formulation associated with the WPS is varied to rule out any bias that might be induced by the performance inequality of the three formulations. That is, we are contrasting the average results from test case one against those resulting from 9 and 14, from test case two against those from 8 and 13, and finally, the results obtained from test case three against the results of tests 7 and 15 (described in Table 2), respectively. Similar to the previous experiments, we map in each run the course of the matheuristic's primal gap over time, and it is shown in Fig. 5. The solid black line corresponds to the course of the average primal gap function (taken over 168 instances) when initiating the matheuristic with a FIVF-based solution. In contrast, the dotted red line and the dashed blue line correspond to the course of the average primal gap function when initiating the matheuristic with TIVF-based and TCF-based solutions, respectively. Each subplot within the figure corresponds to an alternative implementation of the matheuristic obtained using a different formulation during Phase III, and where valid inequalities are appended.

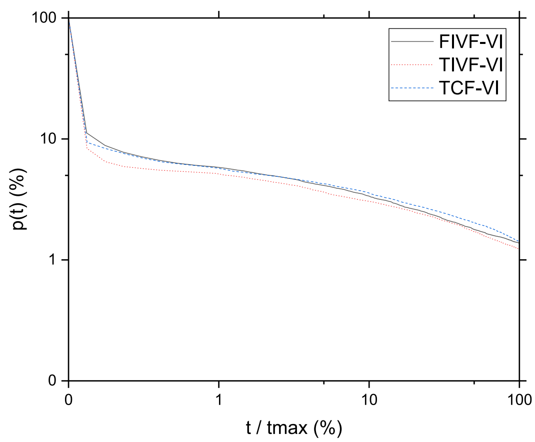
Fig. 5 puts in evidence almost similar behaviors of the matheuristic for different options of starting solutions. In the first and the third implementations, initiating the matheuristic with a TIVF-based solution leads to a tiny improvement in the primal gap. However, when the TIVF formulation simultaneously initiates and guides the algorithmic search, the matheuristic becomes less efficient. The same observation is also valid for the other two formulations. Another indication provided by



(a) A FIVF-based Phase III.



(b) A TIVF-based Phase III.



(c) A TCF-based Phase III.

Fig. 5. Course of the primal gap when running the matheuristic under different initial solutions options.

Fig. 5 is that the best initial solution, in this case, obtained through the TCF formulation, does not necessarily lead to finding the best final solution.

Finally, to further consolidate our findings on the effect of the initial solution quality on the matheuristic performance, we conduct additional analyses using Spearman's rank correlation test (Pirie, 2006). This type of testing was preferred over other non-parametric tests because it does not assume data normality. Here, two hypotheses, the null hypothesis H_0 and the alternative hypothesis H_a, are defined. The null hypothesis is that the quality of the initial solutions does not affect the quality of the final solutions of the matheuristic, whereas the alternative hypothesis is that a better initial solution leads to a better final solution. For all pairs of matheuristic runs (M_1, M_2) with distinct initial solutions but identical implementation in Phase III, we compute their primal gap's deviations at the beginning and the end of both runs, given by $\delta(0) = p^{M_1}(0) - p^{M_2}(0)$, and $\delta(t^{max}) = p^{M_1}(t^{max}) - p^{M_2}(t^{max})$ respectively.

Hence, we run Spearman's test to measure the strength and direction of correlation between the two primal gap deviations. The output of the test is the p-value and the correlation coefficient r_s. The p-value is a measure of how likely an observed correlation is due to chance. A p-value within the significance level $[0, 0.05]$ allows us to reject the null hypothesis (Pirie, 2006). The correlation coefficient r_s can take a range of values from +1 to −1. A value of 0 indicates no association between the two variables. A value greater than 0 indicates a positive association, while a value less than 0 indicates a negative association. Table 5 shows the test results grouped by the number of customers N. We report, for each group, the number of instances, the sample size (number of pairwise comparisons between matheuristic runs), the correlation coefficient r_s, and the p- value. In the final row of Table 5, we report the obtained results aggregated over all data instances.

Table 5
Spearman's test results.

N	Nb. of instances	Spearman test		
		Sample size	r_s	p-value
10	24	216	0.04	0.30
15	24	216	−0.01	0.44
20	24	216	0.06	0.19
25	24	216	0.00	0.49
30	24	216	0.00	0.48
35	16	144	**−0.16**	**0.03**
40	16	144	**0.20**	**0.01**
45	8	72	0.07	0.28
50	8	72	−0.02	0.35
All	168	1512	0.01	0.35

Table 5 indicates that for all data instances, except those with 35 and 40 customers, no apparent correlation exists between the quality of the initial and final solutions. All of these tests returned non-significant p-values and almost zero correlation values. For instances with 35 customers, it was shown that good starting solutions have a detrimental effect on matheuristic quality, while they exhibit a positive impact when run on instances with 40 customers. Nevertheless, the correlation is too weak to be generalized.

6. Concluding remarks

This paper presents an experimental study of matheuristic performance for the PRP considering different mathematical formulations. For the sake of this study, we developed a novel matheuristic consisting of three phases. First, initial lot-sizing decisions are obtained by a master problem relaxation. Next, a sequence of heuristic algorithms completes the associated partial solution in terms of distribution and routing decisions. Finally, the incumbent solution is iteratively improved using a general-purpose MIP heuristic. Vehicle capacity violations and inventory stock-outs at the plant facility are permitted in the construction phase; however, they are later repaired using a recovery procedure. Three mathematical formulations have been implemented and tested within the matheuristic framework: a four-index vehicle flow formulation, a three-index vehicle flow formulation, and a two-commodity flow formulation. The proposed matheuristic turned out to be very competitive with state-of-the-art algorithms by matching or improving the best-known upper bounds, although this was not the primary aim of this effort.

The experimental study assesses the impact of various design alternatives and choices of matheuristics for the PRP. More specifically, we investigated the computational effects stemming from the choice of mathematical formulations, using valid inequalities, and alternating the starting solutions within a matheuristic framework. We perform extensive computational tests on benchmark PRP instances and use consistent performance measures to compare the performance of the matheuristic under the proposed configurations. The results give us a better view of the models that work better for a matheuristic in practice. In particular, they showed that the contribution of a mathematical formulation to the matheuristic convergence is instance-specific and mainly attributable to the number of customers and the length of the planning horizon considered. It is also clear that integrating valid inequalities is highly beneficial, regardless of the considered mathematical formulation. Additionally, a matheuristic with a strong initial solution (in terms of optimality gap) is not necessarily superior to one with a weaker initial solution. Several reasons might explain this behavior, and they relate to the data instances' inherent features, the definition of the search neighborhood, and the particular features of solvers, all of which are details often not fully known to the practitioner.

CRediT authorship contribution statement

Mohamed Ben Ahmed: Methodology, Software, Writing – original draft. **Lars Magnus Hvattum:** Conceptualization, Methodology, Supervision, Writing – review & editing. **Agostinho Agra:** Conceptualization, Methodology, Writing – review & editing.

Data availability

Data will be made available on request.

Acknowledgments

The authors thank the two anonymous referees for their valuable comments and suggestions that helped improve the paper. This research was carried out with financial support from the AXIOM project, partly funded by the Research Council of Norway.

Appendix A. Valid inequalities

Several valid inequalities have been added to strengthen the proposed PRP formulations, and all of them originate from the existing literature. In what follows, we present the valid inequalities associated with each mathematical formulation.

A.1. Valid inequalities for the FIVF formulation

We present here the inequalities that are valid for the FIVF formulation, and which were originally proposed by Adulyasak et al. (2014a). Denote by t' and t'' the earliest period when the plant must produce and the earliest period when at least one customer must be replenished to prevent a stock-out, respectively; i.e., $t' = argmin_{1\leq t\leq H}\left\{\sum_{i\in N^C}\max\{0,\sum_{j=1}^{t} d_{ij} - I_{i0}\} - I_{00} > 0\right\}$, and $t'' = \min_{i\in N^C} t_i\varepsilon$, where $t_i\varepsilon = argmin_{1\leq t\leq H}\left\{\sum_{j=1}^{t} d_{ij} - I_{i0} > 0\right\}$. Let M be the minimum shipping quantity in t''; i.e, $M = \sum_{i\in N^C}\max\left\{0,\sum_{j=1}^{t''} d_{ij} - I_{i0}\right\}$. The first two inequalities are intended to prevent stock-outs and are given by:

$$\sum_{t=1}^{t'} y_t \geq 1, \tag{A.1.1}$$

$$\sum_{k\in K}\sum_{t=1}^{t''} z_{0kt} \geq \left\lceil \frac{M}{Q} \right\rceil. \tag{A.1.2}$$

Constraints (A.1.1) ensure that a production process will be launched no later than t', while constraints (A.1.2) compute a lower bound on the number of used vehicles. The following inequalities further strengthen customer replenishment:

$$I_{i,t-l-1} \geq \left(\sum_{j=0}^{l} d_{i,t-j}\right)\left(1-\sum_{k\in K}\sum_{j=0}^{l} z_{ik,t-j}\right), \quad \forall i\in N^C, \quad t\in T, \quad l=0,1,\ldots,t-1. \tag{A.1.3}$$

Constraints (A.1.3) imply that if a customer $i \in N^C$ is not served at time period $t \in T$, then the available stock at inventory I_{it} should cover the customer's demand in the corresponding period. Finally, the following inequalities are concerned with the routing decisions:

$$z_{ikt} \leq z_{0kt}, \quad \forall i\in N^C, \quad k\in K, \quad t\in T, \tag{A.1.4}$$

$$x_{ijkt} \leq z_{jkt} \text{ and } x_{ijkt} \leq z_{ikt}, \quad \forall (i,j)\in E(N^C), \quad k\in K, \quad t\in T. \tag{A.1.5}$$

Constraints (A.1.4) force the plant node and the customer node to be included in the same route traveled at time $t \in T$, if customer $i \in N^C$ is visited, that is $z_{it} = 1$. Constraints (A.1.5) are referred to as logical inequalities, and they say that if a customer i is the successor, respectively the predecessor, of customer j in the route traveled at period $t \in T$, then j has to be visited too.

A.2. Valid inequalities for the TIVF formulation

The valid inequalities proposed above can be adapted for the TIVF formulation as follows:

$$\sum_{t=1}^{t'} y_t \geq 1, \tag{A.2.1}$$

$$\sum_{t=1}^{t''} z_{0t} \geq \left\lceil \frac{M}{Q} \right\rceil, \tag{A.2.2}$$

$$I_{i,t-l-1} \geq \left(\sum_{j=0}^{l} d_{i,t-j}\right)\left(1-\sum_{j=0}^{l} z_{i,t-j}\right), \quad \forall i\in N^C, \quad t\in T, \quad l=0,1,\ldots,t-1, \tag{A.2.3}$$

$$z_{it} \leq z_{0t}, \quad \forall i\in N^C, \quad t\in T, \tag{A.2.4}$$

$$x_{ijt} \leq z_{it} \text{ and } x_{ijt} \leq z_{jt}, \quad \forall (i,j)\in E(N^C), \quad t\in T. \tag{A.2.5}$$

Besides, we add a priori the following inequalities:

$$Qz_{0t} \geq \sum_{i \in N^C} q_{it}, \quad \forall t \in T. \tag{A.2.6}$$

Constraints (A.2.6) ensures that the number of routed vehicles is sufficient to carry the total delivered quantities to all customers at each time period. Finally, because the GFSECs (2.10) are weak, we can further strengthen the TIVF formulation by considering the following subtour elimination constraints (SECs):

$$\sum_{(i,j) \in E(S)} x_{ijt} \leq \sum_{i \in S} z_{it} - z_{et}, \quad \forall S \subseteq N^C, \quad |S| \geq 2, \quad e \in S, \quad t \in T. \tag{A.2.7}$$

These inequalities are inserted to prevent subtours in each time period, however, they do not prevent violations at vehicle capacities, and they must necessarily be used together with the GFSECs (2.10). These cuts are separated dynamically using the min-cut algorithm.

A.3. Valid inequalities for the TCF formulation

In their paper, Manousakis et al. (2021a) described and implemented three sets of valid inequalities for the TCF formulation. The first set of inequalities is adopted from a similar formulation for the inventory routing problem, which was described in Manousakis et al. (2021b). These inequalities are given by:

$$x_{0it} \leq z_{it}, \quad \forall i \in N^C, \quad t \in T, \tag{A.3.1}$$

$$x_{ijt} \leq z_{it}, \quad \forall (i,j) \in E(N^C), \quad t \in T, \tag{A.3.2}$$

$$x_{ijt} \leq z_{jt}, \quad \forall (i,j) \in E(N^C), \quad t \in T, \tag{A.3.3}$$

$$f_{ijt} \geq d_{jt} x_{ijt} - I_{j,t-1}, \quad \forall (i,j) \in E(N^C), \quad t \in T, \tag{A.3.4}$$

$$f_{jit} \geq d_{it} x_{ijt} - I_{i,t-1}, \quad \forall (i,j) \in E(N^C), \quad t \in T, \tag{A.3.5}$$

$$\sum_{l=1}^{t} z_{il} \geq \left\lceil \frac{\sum_{l=1}^{t} d_{il} - I_{i0}}{\min(Q, L_i)} \right\rceil, \quad \forall i \in N^C, \quad t \in T, \tag{A.3.6}$$

$$\sum_{l=t_1}^{t_2} z_{il} \geq \left\lceil \frac{\sum_{l=t_1}^{t_2} d_{il} - L_i}{\min(Q, L_i)} \right\rceil, \quad \forall i \in N^C, \quad t_1, t_2 \in T, \quad t_1 < t_2, \tag{A.3.7}$$

$$\sum_{l=t_1}^{t_2} z_{il} \geq \frac{\sum_{l=t_1}^{t_2} d_{il} - I_{i,t_1-1}}{\min(Q, L_i)}, \quad \forall i \in N^C, \quad t_1, t_2 \in T, \quad t_1 < t_2, \tag{A.3.8}$$

$$\sum_{l=t_1}^{t_2} z_{il} \geq \frac{\sum_{l=t_1}^{t_2} d_{il} - I_{i,t_1-1}}{\sum_{l=t_1}^{t_2} d_{il}}, \quad \forall i \in N^C, \quad t_1, t_2 \in T, \quad t_1 < t_2. \tag{A.3.9}$$

Inequalities (A.3.1)–(A.3.3) are the logical inequalities, and they are similar to those described within the two previous formulations. Inequalities (A.3.4) and (A.3.5) define lower bounds on the flow variables. It ensures that the flow (which translates to the load carried by a vehicle before visiting a customer $i \in N^C$, or the residual capacity after visiting customer $i \in N^C$) is higher or equal to the stock-out at the customer. Moreover, inequalities (A.3.6)–(A.3.9) impose a lower bound on the number of performed visits to a customer $i \in N^C$ during any time interval within the planning horizon. An additional set of inequalities was proposed by Manousakis et al. (2021a), and they describe as follow:

$$f_{ijt} \leq Q x_{ijt}, \quad \forall (i,j) \in E(N^C), \quad t \in T, \tag{A.3.10}$$

$$f_{jit} \leq Q x_{ijt}, \quad \forall (i,j) \in E(N^C), \quad t \in T, \tag{A.3.11}$$

$$q_{it} \leq \min\left\{ L_i + d_{it}, Q, \sum_{l=t}^{H} d_{il} \right\} z_{it}, \quad \forall i \in N^C, \quad t \in T, \tag{A.3.12}$$

$$q_{it} \leq \min\left\{ L_i + d_{it} - I_{i,t-1}, \sum_{l=t}^{H} d_{il} - I_{i,t-1} \right\} z_{it}, \quad \forall i \in N^C, \quad t \in T. \tag{A.3.13}$$

Inequalities (A.3.10) and (A.3.11) are the logical constraints related to the flow variables. These variables can be positive only when the edge (i,j) is traversed. Furthermore, inequalities (A.3.12) and (A.3.13) impose an upper bound on the delivered quantities. Finally, Manousakis et al. (2021a) presented new valid inequalities for the production setup variables, and which were inspired from the work of Coelho and Laporte (2014). They can be described as follows:

$$y_t C \geq \sum_{i \in N^C} (d_{it} - I_{i,t-1}) - I_{0,t-1}, \quad \forall t \in T, \tag{A.3.14}$$

$$\sum_{l=1}^{t} y_l \geq \left\lceil \frac{\sum_{i \in N^C} \sum_{l=1}^{t} d_{il} - I_{i0}}{C} \right\rceil, \quad \forall t \in T, \quad t > 1, \tag{A.3.15}$$

$$\sum_{l=t_1}^{t_2} y_l \geq \left\lceil \frac{\sum_{i \in N^C} \left(\sum_{l=t_1}^{t_2} d_{il} \right) - L_i - L_0}{C} \right\rceil, \quad \forall t_1, t_2 \in T, \quad t_1 < t_2, \tag{A.3.16}$$

$$\sum_{l=t_1}^{t_2} y_l \geq \frac{\sum_{i \in N^C} \left(\sum_{l=t_1}^{t_2} d_{il} \right) - I_{i,t_1-1} - I_{0,t_1-1}}{C}, \quad \forall t_1, t_2 \in T, \quad t_1 < t_2. \tag{A.3.17}$$

Inequalities (A.3.14) ensure that a production process will be launched when the stock at both the customer and the plant is not enough to satisfy the demands. Inequalities (A.3.15)–(A.3.17) impose lower bounds on the minimum number of setups during any time interval within the planning horizon.

Appendix B. Constructive heuristics

We present here a description of the customer urgency heuristic, and the modified next fit decreasing algorithm, both used in the second phase of our matheuristic. The customer urgency heuristic is a merge-and-round heuristic that iterates over the set of fractional z−variables. Any *fractional* visit (given by a fractional value of z) is shifted to the next one in time, and both deliveries are merged. If this operation induces stock-outs at customer inventories, then the rounding is performed in the opposite direction. If a merge cannot be performed, or no other fractional value of z exists, the heuristic will round up the fractional visit to 1. Algorithm 2 describes the main steps of the customer urgency heuristic. In Algorithm 2, we use $\mathcal{N}_t^{UC}$, to refer to the set of urgent customers which shall incur a stock-out at time period $t \in T$, while $\mathcal{N}_t^{NUC}$, called the set of non-urgent customers, contains the remaining ones. The modified next fit decreasing algorithm assigns customer deliveries, ordered from the largest to the smallest, to vehicles. If a delivery fits none of the available vehicles, then it is assigned to the one with the largest residual capacity. Algorithm 3 describes the bin-packing algorithm. Since the values of the z−variables in Phase I can be fractional, we cannot ensure deliveries do not split; thus, violations may occur concerning vehicle capacities and stock-outs at the production plant. We allow for an infeasible starting solution; however, we propose later a procedure to recover vehicles and the plant infeasibilities.

Remark 3. The feasibility condition introduced in lines 9 and 19 in Algorithm 2 is given by:

$$I_{it} + q_{i,t_i^*} \leq L_i, \quad \forall t \in T, \quad i \in \mathcal{N}_t^{UC}, \quad t_i^* \in T, \quad t_i^* > t,$$

$$I_{is} - q_{it} \geq 0, \quad \forall t \in T, \quad i \in \mathcal{N}_t^{NUC}, \quad t_i^* \in T, \quad t_i^* > t, \quad s = t, \dots, t_i^* - 1.$$

Algorithm 2: Customer urgency heuristic.

for $t \leftarrow 1$ **to** H **do**
  for $i \in N^C$ **do**
    Define $r_{it} = I_{i,t-1}/d_{it}$, the urgency ratio
  end
  Let $\mathcal{N}_t^{UC} := \{i \in N^C : r_{it} < 1$ **and** $0 < z_{it} < 1\}$
  Sort $\mathcal{N}_t^{UC}$ by $\downarrow r_{it}$
  for $i \in \mathcal{N}_t^{UC}$ **do**
    Let $t_i^* \leftarrow$ next fractional visit of customer i
    if *<Feasibility condition>* **then**
      $q_{it} \leftarrow q_{it} + q_{i,t_i^*}$
      $I_{st} \leftarrow I_{st} + q_{i,t_i^*}, \quad s = t, \dots, t_i^* - 1$
      $z_{it} \leftarrow 1, \quad z_{i,t_i^*} \leftarrow 0, \quad q_{i,t_i^*} \leftarrow 0$
    end
  end
  Let $\mathcal{N}_t^{NUC} := \{i \in N^C : r_{it} \geq 1$ **and** $0 < z_{it} < 1\}$
  Sort $\mathcal{N}_t^{NUC}$ by $\uparrow r_{it}$
  for $i \in \mathcal{N}_t^{NUC}$ **do**
    Let $t_i^* \leftarrow$ next fractional visit for customer i
    if *<Feasibility condition>* **then**
      $q_{i,t_i^*} \leftarrow q_{i,t_i^*} + q_{it}$
      $I_{st} \leftarrow I_{st} - q_{it}, \quad s = t, \dots, t_i^* - 1$
      $z_{it} \leftarrow 0, \quad z_{i,t_i^*} \leftarrow 1, \quad q_{it} \leftarrow 0$
    end
  end
end
Round to 1 all remaining fractional z-values (if exist).

It ensures that customers' inventory levels remain within the preset limits.

Algorithm 3: A modified next fit decreasing bin packing algorithm.

for $t \leftarrow 1$ **to** H **do**
  Let N_t the set of customer to be served on time period t
  Sort elements of N_t by $\downarrow q_{it}$
  Let $\mathcal{K} = \emptyset$ the set of bins used so far
  for $i \in N_t$ **do**
    for $k \leftarrow 1$ **to** $|\mathcal{K}|$ **do**
      Let σ_k the residual capacity of bin k
      if $\sigma_k \geq q_{it}$ **then**
        Assign customer i to bin k
        $\sigma_k \leftarrow \sigma_k - q_{it}$
        break
      end
    end
    if $k = |\mathcal{K}|$ **and** $|\mathcal{K}| < m$ **then**
      $|\mathcal{K}| \leftarrow |\mathcal{K}| + 1$
      Assign customer i to bin $|\mathcal{K}|$
      $\sigma_{|\mathcal{K}|} \leftarrow Q - q_{it}$
    else
      Let k^* be the bin with highest residual capacity
      Assign customer i to bin k^* even it violates the capacity
      $\sigma_{k^*} \leftarrow \sigma_{k^*} - q_{it}$
    end
  end
end

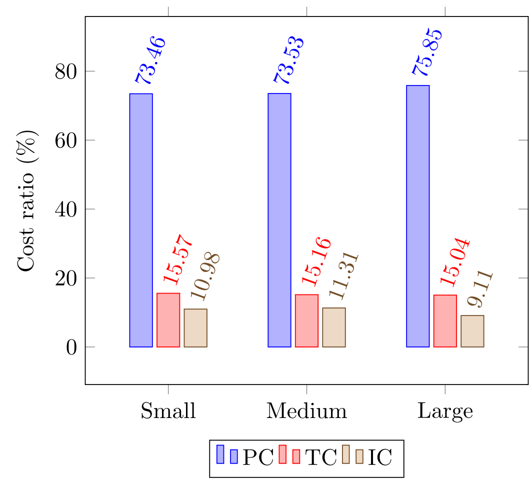


Fig. 6. Cost structure of PRP solutions.

Appendix C. Detailed numerical results

Comparison with state-of-the-art methods

In Table 6, we report the results obtained by the proposed matheuristic, aggregated over all the tested configurations, versus the ones reported by other papers that have solved the same instances. For each instance, we report the value of the objective function obtained by each method, the lower bound (LB), the best-known solution (UB), and the percentage optimality gap (Gap) given by $\frac{|LB-UB|}{|LB|}$. Next, we display the percentage deviation of objective functions obtained by each method from best know values. In the final row of Table 6, we report the average calculations over all instances. We use boldface letters if our matheuristic finds the best know solution, while an underline indicates that the best solution was obtained only by our matheuristic. Since all methods were run on machines with different capabilities and parameter settings, comparing CPU times becomes inconsistent. It is worth noting that our matheuristic was run with a time limit of four hours.

Solution analysis

Additional analyses are performed on the obtained results from our matheuristics to gain insights into the characteristics of PRP solutions. We compute the percentage ratios PC, TC, and IC corresponding to the production, transportation, and inventory costs, divided by the total costs. Fig. 6 displays the average ratio calculations for each class of data instances (small, medium, large). It is apparent that production yields the highest costs, while transportation and inventory costs are much lower.

Additionally, Fig. 7 compares the normalized cost values of different intermediate and final solutions generated by the matheuristic. In more detail, the outlined values show: (i) the average total, production, transportation, and inventory costs of the relaxed solution (highlighted in green); (ii) the average total, production, transportation, and inventory costs of the starting solution (highlighted in blue); and (iii) the average total, production, transportation, and inventory costs after improvement (highlighted in red). A couple of interesting conclusions can be made from this figure. First, the starting solution yields, on average, the highest transportation cost, which is significantly reduced after the improvement phase. It can also be observed that there are no significant differences in production and inventory costs before and after refinement. One can assume that obtaining a solution with the lowest total costs after the improvement phase is due to the improvement in routing decisions.

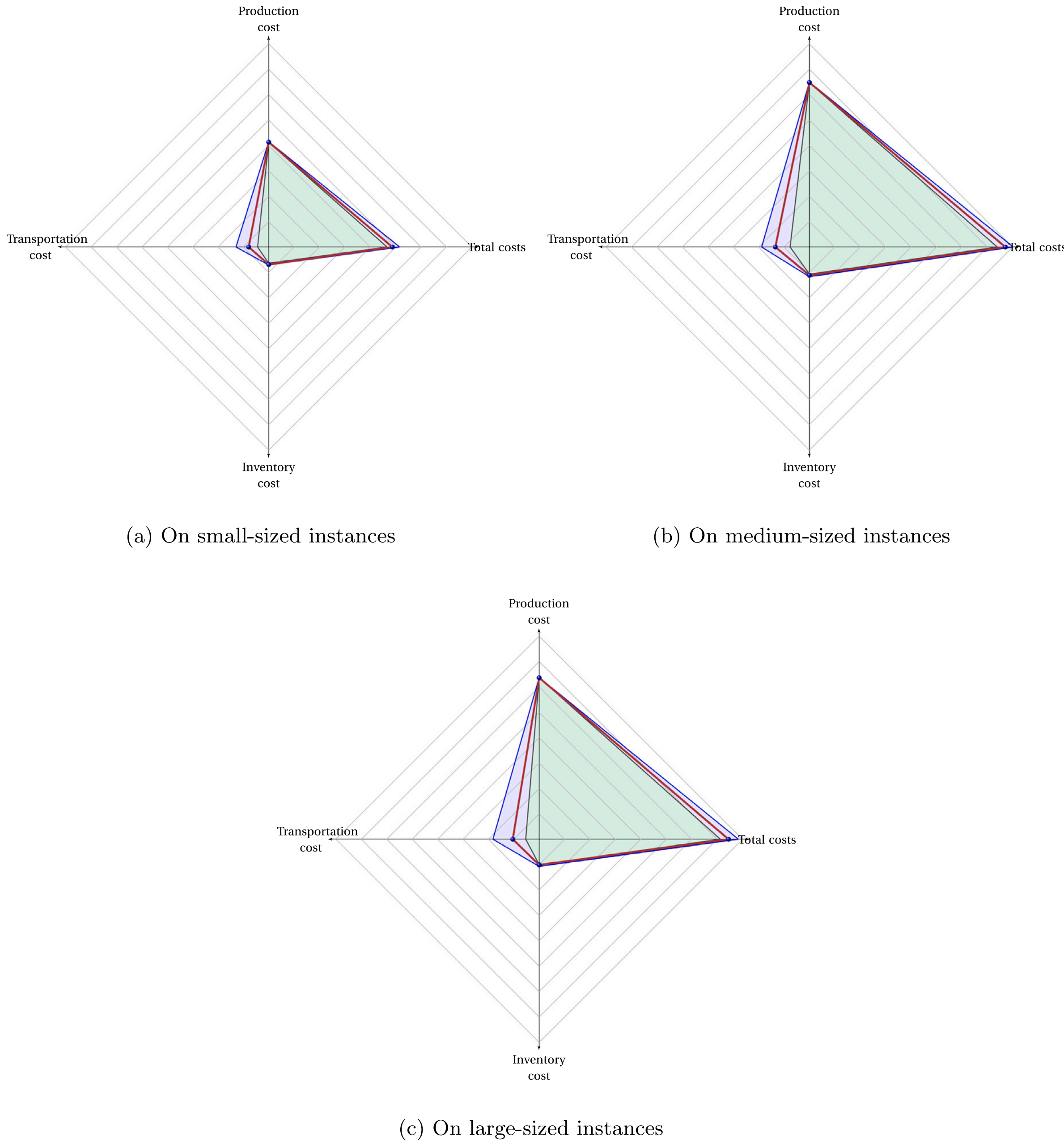


Fig. 7. Spider-chart comparing cost evolution throughout the matheuristic.

Table 6
Detailed results obtained on Adulyasak et al. (2014a) benchmark data instances.

Instance	Total cost					Bounds			% dev. from current best solution				
	ACJ-BC	SSPG-BC	ACJ-ALNS	VACS-M	BHA-M	LB	UB	Gap (%)	ACJ-BC	SSPG-BC	ACJ-ALNS	VACS-M	BHA-M
1	13,924	13,924	14,102	14,102	**13,924**	13,924	13,924	0.00	0.00	0.00	1.28	1.28	0.00
2	98,434	98,434	98,612	98,612	**98,434**	98,434	98,434	0.00	0.00	0.00	0.18	0.18	0.00
3	19,751	19,751	19,751	19,799	**19,751**	19,751	19,751	0.00	0.00	0.00	0.00	0.24	0.00
4	10,792	10,792	10,792	10,792	**10,792**	10,792	10,792	0.00	0.00	0.00	0.00	0.00	0.00
5	14,002	14,002	14,102	14,102	**14,002**	14,002	14,002	0.00	0.00	0.00	0.71	0.71	0.00
6	98,512	98,512	98,648	98,612	**98,512**	98,512	98,512	0.00	0.00	0.00	0.14	0.10	0.00
7	20,550	20,550	21,235	20,712	**20,550**	20,550	20,550	0.00	0.00	0.00	3.33	0.79	0.00
8	10,974	10,974	11,004	10,974	**10,974**	10,974	10,974	0.00	0.00	0.00	0.27	0.00	0.00
9	29,559	29,559	29,659	29,559	**29,559**	29,559	29,559	0.00	0.00	0.00	0.34	0.00	0.00

(continued on next page)

Table 6 (*continued*).

10	197,769	197,769	198,287	197,769	**197,769**	197,769	197,769	0.00	0.00	0.00	0.26	0.00	0.00
11	40,249	40,249	40,784	40,249	**40,249**	40,249	40,249	0.00	0.00	0.00	1.33	0.00	0.00
12	21,248	21,248	21,318	21,313	**21,248**	21,248	21,248	0.00	0.00	0.00	0.33	0.31	0.00
13	29,897	29,897	29,942	30,124	**29,897**	29,897	29,897	0.00	0.00	0.00	0.15	0.76	0.00
14	198,107	198,107	198,674	198,334	**198,107**	198,107	198,107	0.00	0.00	0.00	0.29	0.11	0.00
15	41,668	41,668	42,064	41,898	**41,668**	41,668	41,668	0.00	0.00	0.00	0.95	0.55	0.00
16	21,559	21,559	21,573	21,559	**21,559**	21,559	21,559	0.00	0.00	0.00	0.06	0.00	0.00
17	50,929	50,929	51,563	50,929	**50,929**	50,929	50,929	0.00	0.00	0.00	1.24	0.00	0.00
18	369,259	369,259	370,176	369,259	**369,259**	369,223	369,259	0.01	0.00	0.00	0.25	0.00	0.00
19	69,464	69,464	72,900	69,464	**69,464**	69,464	69,464	0.00	0.00	0.00	4.95	0.00	0.00
20	39,631	39,631	40,004	39,638	**39,631**	39,631	39,631	0.00	0.00	0.00	0.94	0.02	0.00
21	51,739	51,739	52,200	51,761	**51,739**	51,739	51,739	0.00	0.00	0.00	0.89	0.04	0.00
22	370,069	370,069	370,624	370,091	**370,069**	370,055	370,069	0.00	0.00	0.00	0.15	0.01	0.00
23	72,148	72,148	76,376	73,933	**72,148**	72,148	72,148	0.00	0.00	0.00	5.86	2.47	0.00
24	40,276	40,276	40,564	40,331	**40,276**	40,276	40,276	0.00	0.00	0.00	0.72	0.14	0.00
25	19,488	19,488	19,610	19,504	**19,488**	19,488	19,488	0.00	0.00	0.00	0.63	0.08	0.00
26	136,128	136,128	136,144	136,144	**136,128**	136,119	136,128	0.01	0.00	0.00	0.01	0.01	0.00
27	28,687	28,687	29,599	29,206	**28,687**	28,687	28,687	0.00	0.00	0.00	3.18	1.81	0.00
28	15,163	15,163	15,163	15,163	**15,163**	15,163	15,163	0.00	0.00	0.00	0.00	0.00	0.00
29	19,680	19,680	19,689	19,881	**19,680**	19,680	19,680	0.00	0.00	0.00	0.05	1.02	0.00
30	136,320	136,320	136,839	136,521	**136,320**	136,320	136,320	0.00	0.00	0.00	0.38	0.15	0.00
31	30,376	30,376	31,523	30,734	**30,376**	30,376	30,376	0.00	0.00	0.00	3.78	1.18	0.00
32	15,469	15,469	15,469	15,469	**15,469**	15,469	15,469	0.00	0.00	0.00	0.00	0.00	0.00
33	41,627	41,627	41,628	41,870	**41,627**	41,623	41,627	0.01	0.00	0.00	0.00	0.58	0.00
34	273,827	273,827	274,028	274,070	**273,827**	273,827	273,827	0.00	0.00	0.00	0.07	0.09	0.00
35	57,294	57,294	59,158	57,294	**57,294**	57,294	57,294	0.00	0.00	0.00	3.25	0.00	0.00
36	29,556	29,556	29,770	29,556	**29,556**	29,556	29,556	0.00	0.00	0.00	0.72	0.00	0.00
37	42,191	42,191	42,543	42,361	**42,191**	42,191	42,191	0.00	0.00	0.00	0.83	0.40	0.00
38	274,391	274,391	275,057	274,561	**274,391**	274,391	274,391	0.00	0.00	0.00	0.24	0.06	0.00
39	60,087	60,087	63,331	60,941	**60,087**	60,087	60,087	0.00	0.00	0.00	5.40	1.42	0.00
40	30,090	30,090	30,196	30,206	**30,090**	30,090	30,090	0.00	0.00	0.00	0.35	0.39	0.00
41	73,204	73,204	74,908	73,368	**73,204**	73,204	73,204	0.00	0.00	0.00	2.33	0.22	0.00
42	522,754	522,754	524,310	522,918	522,896	522,713	522,754	0.01	0.00	0.00	0.30	0.03	0.03
43	102,002	102,002	104,339	105,334	103,138	102,002	102,002	0.00	0.00	0.00	2.29	3.27	1.11
44	56,644	56,644	57,669	56,944	56,680	56,644	56,644	0.00	0.00	0.00	1.81	0.53	0.06
45	74,854	74,764	75,253	75,045	74,787	74,349	74,764	0.56	0.12	0.00	0.65	0.38	0.03
46	524,376	524,314	524,875	524,441	524,387	523,630	524,314	0.13	0.01	0.00	0.11	0.02	0.01
47	108,055	108,046	113,184	110,036	111,351	104,317	108,046	3.57	0.01	0.00	4.76	1.84	3.06
48	57,914	57,893	59,486	58,293	57,919	57,092	57,893	1.40	0.04	0.00	2.75	0.69	0.04
49	22,209	22,209	22,394	22,388	**22,209**	22,209	22,209	0.00	0.00	0.00	0.83	0.81	0.00
50	157,479	157,479	157,767	157,658	**157,479**	157,479	157,479	0.00	0.00	0.00	0.18	0.11	0.00
51	31,708	31,708	32,697	31,708	**31,708**	31,708	31,708	0.00	0.00	0.00	3.12	0.00	0.00
52	17,171	17,171	17,171	17,171	**17,171**	17,171	17,171	0.00	0.00	0.00	0.00	0.00	0.00
53	22,375	22,375	22,552	22,463	**22,375**	22,375	22,375	0.00	0.00	0.00	0.79	0.39	0.00
54	157,645	157,645	157,822	157,733	**157,645**	157,645	157,645	0.00	0.00	0.00	0.11	0.06	0.00
55	32,851	32,851	34,596	32,851	**32,851**	32,851	32,851	0.00	0.00	0.00	5.31	0.00	0.00
56	17,400	17,400	17,550	17,400	**17,400**	17,400	17,400	0.00	0.00	0.00	0.86	0.00	0.00
57	47,466	47,466	47,466	47,564	**47,466**	47,463	47,466	0.01	0.00	0.00	0.00	0.21	0.00
58	315,846	315,846	316,446	315,944	**315,846**	315,820	315,846	0.01	0.00	0.00	0.19	0.03	0.00
59	63,746	63,746	66,848	64,017	**63,746**	63,746	63,746	0.00	0.00	0.00	4.87	0.43	0.00
60	33,488	33,488	33,663	33,497	**33,488**	33,488	33,488	0.00	0.00	0.00	0.52	0.03	0.00
61	47,681	47,681	48,107	48,005	**47,681**	47,681	47,681	0.00	0.00	0.00	0.89	0.68	0.00
62	316,061	316,061	316,572	316,385	**316,061**	316,035	316,061	0.01	0.00	0.00	0.16	0.10	0.00
63	65,789	65,789	68,730	65,869	**65,789**	65,789	65,789	0.00	0.00	0.00	4.47	0.12	0.12
64	33,859	33,859	34,136	33,859	**33,859**	33,859	33,859	0.00	0.00	0.00	0.82	0.00	0.00
65	82,150	82,150	82,926	82,150	**82,150**	82,150	82,150	0.00	0.00	0.00	0.94	0.00	0.00
66	596,230	596,230	597,370	596,230	**596,230**	596,187	596,230	0.01	0.00	0.00	0.19	0.00	0.00
67	110,925	110,925	114,032	110,925	112,565	110,925	110,925	0.00	0.00	0.00	2.80	0.00	1.48
68	62,920	62,920	63,008	62,928	**62,920**	62,920	62,920	0.00	0.00	0.00	0.14	0.01	0.00
69	82,685	82,685	83,005	83,077	82,741	82,685	82,685	0.00	0.00	0.00	0.39	0.47	0.07
70	596,765	596,765	597,052	596,874	**596,765**	596,708	596,765	0.01	0.00	0.00	0.05	0.02	0.00
71	115,499	115,182	120,371	116,688	117,771	115,182	115,182	0.00	0.28	0.00	4.51	1.31	2.25
72	63,919	63,919	64,394	63,919	**63,919**	63,914	63,919	0.01	0.00	0.00	0.74	0.00	0.00
73	26,055	26,055	26,177	26,177	**26,055**	26,055	26,055	0.00	0.00	0.00	0.47	0.47	0.00
74	182,115	182,115	182,241	182,237	**182,115**	182,112	182,115	0.00	0.00	0.00	0.07	0.07	0.00
75	35,939	35,939	37,990	35,939	**35,939**	35,939	35,939	0.00	0.00	0.00	5.71	0.00	0.00
76	19,528	19,528	19,528	19,528	**19,528**	19,528	19,528	0.00	0.00	0.00	0.00	0.00	0.00
77	26,269	26,269	26,413	26,386	**26,269**	26,269	26,269	0.00	0.00	0.00	0.55	0.45	0.00
78	182,329	182,329	182,616	182,392	**182,329**	182,329	182,329	0.00	0.00	0.00	0.16	0.03	0.00
79	37,754	37,754	39,293	37,754	**37,754**	37,754	37,754	0.00	0.00	0.00	4.08	0.00	0.00
80	19,891	19,891	19,956	19,891	**19,891**	19,891	19,891	0.00	0.00	0.00	0.33	0.00	0.00
81	56,869	56,869	57,063	57,062	**56,869**	56,869	56,869	0.00	0.00	0.00	0.34	0.34	0.00
82	366,289	366,289	366,671	366,487	**366,289**	366,289	366,289	0.00	0.00	0.00	0.10	0.05	0.00
83	73,339	73,339	78,374	73,339	**73,339**	73,339	73,339	0.00	0.00	0.00	6.87	0.00	0.00
84	38,231	38,231	38,409	38,320	**38,231**	38,231	38,231	0.00	0.00	0.00	0.47	0.23	0.00

(*continued on next page*)

Table 6 (*continued*).

85	57,077	57,077	57,684	57,472	**57,077**	57,077	57,077	0.00	0.00	0.00	1.06	0.69	0.00
86	366,497	366,497	367,089	366,892	**366,497**	366,497	366,497	0.00	0.00	0.00	0.16	0.11	0.00
87	76,112	76,112	79,814	76,428	**76,112**	76,112	76,112	0.00	0.00	0.00	4.86	0.42	0.00
88	38,726	38,726	39,093	38,726	38,798	38,726	38,726	0.00	0.00	0.00	0.95	0.00	0.19
89	99,868	99,868	100,654	100,068	99,879	99,868	99,868	0.00	0.00	0.00	0.79	0.20	0.01
90	719,788	719,809	720,803	719,852	719,892	719,738	719,788	0.01	0.00	0.00	0.14	0.01	0.01
91	133,290	133,260	133,290	134,591	136,388	133,260	133,260	0.00	0.02	0.00	0.02	1.00	2.35
92	76,197	76,197	77,482	76,819	76,632	76,190	76,197	0.01	0.00	0.00	1.69	0.82	0.57
93	100,690	100,689	101,165	101,215	101,371	100,680	100,689	0.01	0.00	0.00	0.47	0.52	0.68
94	720,609	720,609	721,543	720,873	721,370	720,537	720,609	0.01	0.00	0.00	0.13	0.04	0.11
95	139,695	138,295	141,218	140,296	145,528	134,383	138,295	2.91	1.01	0.00	2.11	1.45	5.23
96	77,725	77,457	78,889	77,471	77,882	76,383	77,457	1.41	0.35	0.00	1.85	0.02	0.55
97	28,872	28,872	29,186	29,081	**28,872**	28,872	28,872	0.00	0.00	0.00	1.09	0.72	0.00
98	205,992	205,992	206,434	206,215	**205,992**	205,975	205,992	0.01	0.00	0.00	0.21	0.11	0.00
99	41,164	41,164	43,657	42,223	**41,164**	41,164	41,164	0.00	0.00	0.00	6.06	2.57	0.00
100	22,589	22,589	22,648	22,589	**22,589**	22,589	22,589	0.00	0.00	0.00	0.26	0.00	0.00
101	29,157	29,157	29,536	29,483	**29,157**	29,157	29,157	0.00	0.00	0.00	1.30	1.12	0.00
102	206,277	206,277	206,592	206,603	**206,277**	206,265	206,277	0.01	0.00	0.00	0.15	0.16	0.00
103	42,543	42,543	44,323	43,574	**42,543**	42,543	42,543	0.00	0.00	0.00	4.18	2.42	0.00
104	22,860	22,860	22,869	22,860	**22,860**	22,860	22,860	0.00	0.00	0.00	0.04	0.00	0.00
105	61,287	61,287	61,747	61,450	61,350	61,287	61,287	0.00	0.00	0.00	0.75	0.27	0.10
106	413,637	413,637	414,332	413,674	413,764	413,609	413,637	0.01	0.00	0.00	0.17	0.01	0.03
107	82,249	82,249	86,464	82,893	82,897	82,249	82,249	0.00	0.00	0.00	5.12	0.78	0.79
108	44,007	44,007	44,263	44,024	**44,007**	44,007	44,007	0.00	0.00	0.00	0.58	0.04	0.00
109	61,741	61,741	62,083	62,152	61,784	61,741	61,741	0.00	0.00	0.00	0.55	0.67	0.07
110	414,091	414,091	415,132	414,440	414,222	414,057	414,091	0.01	0.00	0.00	0.25	0.08	0.03
111	85,895	85,072	89,079	85,586	85,931	85,072	85,072	0.00	0.97	0.00	4.71	0.60	1.01
112	44,593	44,588	44,909	44,644	**44,588**	44,271	44,588	0.72	0.01	0.00	0.72	0.13	0.00
113	110,070	109,977	110,557	110,373	111,786	109,745	109,977	0.21	0.08	0.00	0.53	0.36	1.64
114	800,629	800,387	802,301	801,084	802,413	800,308	800,387	0.01	0.03	0.00	0.24	0.09	0.25
115	151,878	148,252	152,780	151,002	164,341	144,501	148,252	2.60	2.45	0.00	3.05	1.85	10.85
116	86,292	85,530	88,253	85,974	86,138	84,465	85,530	1.26	0.89	0.00	3.18	0.52	0.71
117	111,263	111,142	111,500	111,384	116,151	109,852	111,142	1.17	0.11	0.00	0.32	0.22	4.51
118	802,111	801,634	803,361	802,120	805,823	800,599	801,634	0.13	0.06	0.00	0.22	0.06	0.52
119	158,605	155,142	158,932	156,604	164,033	145,376	155,142	6.72	2.23	0.00	2.44	0.94	5.73
120	88,578	86,882	89,020	86,939	87,370	84,691	86,882	2.59	1.95	0.00	2.46	0.07	0.56
121	35,503	35,503	35,980	35,616	**35,503**	35,503	35,503	0.00	0.00	0.00	1.34	0.32	0.00
122	256,903	256,903	257,471	257,016	256,916	256,887	256,903	0.01	0.00	0.00	0.22	0.04	0.01
123	49,660	49,660	53,959	50,453	**49,660**	49,660	49,660	0.00	0.00	0.00	8.66	1.60	0.00
124	27,774	27,774	27,829	27,774	**27,774**	27,774	27,774	0.00	0.00	0.00	0.20	0.00	0.00
125	35,854	35,854	36,340	36,001	**35,854**	35,854	35,854	0.00	0.00	0.00	1.36	0.41	0.00
126	257,254	257,254	257,724	257,401	**257,254**	257,251	257,254	0.00	0.00	0.00	0.18	0.06	0.00
127	51,475	51,475	55,641	52,375	51,749	51,475	51,475	0.00	0.00	0.00	8.09	1.75	0.53
128	28,160	28,160	28,227	28,160	**28,160**	28,160	28,160	0.00	0.00	0.00	0.24	0.00	0.00
129	74,939	74,939	75,286	75,189	75,147	74,939	74,939	0.00	0.00	0.00	0.46	0.33	0.28
130	515,039	515,039	515,683	515,327	515,334	514,989	515,039	0.01	0.00	0.00	0.13	0.06	0.06
131	99,846	99,282	103,930	99,930	101,235	99,282	99,282	0.00	0.57	0.00	4.68	0.65	1.97
132	54,410	54,410	54,705	54,501	54,483	54,187	54,410	0.41	0.00	0.00	0.54	0.17	0.13
133	75,487	75,472	76,054	75,817	75,938	75,467	75,472	0.01	0.02	0.00	0.77	0.46	0.62
134	515,594	515,572	516,096	515,878	516,115	515,522	515,572	0.01	0.00	0.00	0.10	0.06	0.11
135	104,045	102,869	106,455	103,034	109,106	99,822	102,869	3.05	1.14	0.00	3.49	0.16	6.06
136	55,439	54,975	55,578	55,040	55,094	54,339	54,975	1.17	0.84	0.00	1.10	0.12	0.22
137	44,422	44,422	44,640	44,463	**44,422**	44,422	44,422	0.00	0.00	0.00	0.49	0.09	0.00
138	320,512	320,512	320,832	320,647	320,533	320,489	320,512	0.01	0.00	0.00	0.10	0.04	0.01
139	61,530	61,530	64,253	61,952	61,705	61,530	61,530	0.00	0.00	0.00	4.43	0.69	0.28
140	34,162	34,162	34,350	34,162	**34,162**	34,162	34,162	0.00	0.00	0.00	0.55	0.00	0.00
141	44,461	44,461	44,767	44,468	44,468	44,461	44,461	0.00	0.00	0.00	0.69	0.02	0.02
142	320,816	320,818	321,225	320,914	320,855	320,786	320,816	0.01	0.00	0.00	0.13	0.03	0.01
143	63,823	63,596	65,326	64,363	63,611	61,489	63,596	3.43	0.36	0.00	2.72	1.21	0.02
144	34,688	34,644	34,691	34,644	34,728	34,461	34,644	0.53	0.13	0.00	0.14	0.00	0.24
145	96,393	96,393	97,071	96,615	96,926	96,210	96,393	0.19	0.00	0.00	0.70	0.23	0.55
146	646,929	646,926	647,543	647,145	647,463	646,784	646,926	0.02	0.00	0.00	0.10	0.03	0.08
147	123,837	123,149	126,750	123,495	127,922	120,868	123,149	1.89	0.56	0.00	2.92	0.28	3.88
148	67,180	66,854	67,180	66,895	67,140	66,510	66,854	0.52	0.49	0.00	0.49	0.06	0.43
149	97,104	96,757	97,235	96,701	98,855	96,076	96,701	0.65	0.42	0.06	0.55	0.00	2.23
150	647,331	647,287	647,955	647,235	648,690	646,600	647,235	0.10	0.01	0.01	0.11	0.1	0.32
151	129,933	127,335	130,286	127,117	135,637	121,524	127,117	4.60	2.22	0.17	2.49	0.00	6.70
152	68,115	67,815	68,115	67,873	67,944	66,632	67,815	1.77	0.44	0.00	0.44	0.09	0.19
153	50,446	50,446	50,592	50,446	**50,446**	50,446	50,446	0.00	0.00	0.00	0.29	0.00	0.00
154	376,125	376,125	376,804	376,283	376,166	376,061	376,125	0.02	0.00	0.00	0.18	0.04	0.01
155	69,629	69,662	72,914	72,250	70,072	69,292	69,629	0.49	0.00	0.05	4.72	3.76	1.13
156	39,929	39,965	39,929	39,980	39,959	39,788	39,929	0.36	0.00	0.09	0.00	0.13	0.08
157	50,872	50,905	51,425	50,902	50,915	50,543	50,872	0.65	0.00	0.06	1.09	0.06	0.08
158	376,654	376,693	377,454	376,638	**376,604**	376,068	376,604	0.14	0.01	0.02	0.23	0.01	0.00
159	73,658	72,182	73,658	72,408	72,570	69,432	72,182	3.96	2.04	0.00	2.04	0.31	0.54
160	40,862	40,587	40,862	40,762	**40,500**	40,009	40,500	1.23	0.89	0.21	0.89	0.65	0.00

(*continued on next page*)

Table 6 (*continued*).

161	46,726	46,726	47,005	46,812	**46,726**	46,726	46,726	0.00	0.00	0.00	0.60	0.18	0.01
162	340,782	340,782	341,118	340,996	340,928	340,751	340,782	0.01	0.00	0.00	0.10	0.06	0.04
163	65,260	65,260	68,045	66,387	65,449	65,260	65,260	0.00	0.00	0.00	4.27	1.73	0.29
164	36,347	36,347	36,695	36,347	**36,347**	36,347	36,347	0.00	0.00	0.00	0.96	0.00	0.00
165	47,138	47,138	47,577	47,297	47,149	47,138	47,138	0.00	0.00	0.00	0.93	0.34	0.02
166	341,426	341,378	341,938	341,379	341,573	341,312	341,378	0.02	0.01	0.00	0.16	0.00	0.06
167	68,125	67,900	71,560	69,266	69,352	66,691	67,900	1.81	0.33	0.00	5.39	2.01	2.14
168	36,969	36,967	37,123	36,923	36,971	36,701	36,923	0.61	0.12	0.12	0.54	0.00	0.13
Average								0.32	0.13	0.005	1.39	0.39	0.43

References

Adulyasak, Y., Cordeau, J.-F., Jans, R., 2014a. Formulations and branch-and-cut algorithms for multivehicle production and inventory routing problems. INFORMS J. Comput. 26, 103–120.

Adulyasak, Y., Cordeau, J.-F., Jans, R., 2014b. Optimization-based adaptive large neighborhood search for the production routing problem. Transp. Sci. 48, 20–45.

Adulyasak, Y., Cordeau, J.-F., Jans, R., 2015. The production routing problem: a review of formulations and solution algorithms. Comput. Oper. Res. 55, 141–152.

Archetti, C., Bianchessi, N., Irnich, S., Speranza, M.G., 2014. Formulations for an inventory routing problem. Int. Trans. Oper. Res. 21, 353–374.

Archetti, C., Ljubić, I., 2022. Comparison of formulations for the inventory routing problem. European J. Oper. Res.

Archetti, C., Speranza, M.G., 2014. A survey on matheuristics for routing problems. EURO J. Comput. Optim. 2, 223–246.

Avci, M., Yildiz, S.T., 2019. A matheuristic solution approach for the production routing problem with visit spacing policy. European J. Oper. Res. 279, 572–588.

Avci, M., Yildiz, S.T., 2020. A mathematical programming-based heuristic for the production routing problem with transshipments. Comput. Oper. Res. 123, 105042.

Bektaş, T., Gouveia, L., Santos, D., 2020. Compact formulations for multi-depot routing problems: Theoretical and computational comparisons. Comput. Oper. Res. 124, 105084.

Berthold, T., 2013. Measuring the impact of primal heuristics. Oper. Res. Lett. 611–614.

Blum, C., Puchinger, J., Raidl, G.R., Roli, A., 2011. Hybrid metaheuristics in combinatorial optimization: A survey. Appl. Soft Comput. 11, 4135–4151.

Boschetti, M.A., Maniezzo, V., 2022. Matheuristics: using mathematics for heuristic design. 4OR.

Chandra, P., 1993. A dynamic distribution model with warehouse and customer replenishment requirements. J. Oper. Res. Soc. 44, 681–692.

Chandra, P., Fisher, M.L., 1994. Coordination of production and distribution planning. European J. Oper. Res. 72, 503–517.

Chitsaz, M., Cordeau, J.-F., Jans, R., 2019. A unified decomposition matheuristic for assembly, production, and inventory routing. INFORMS J. Comput. 31, 134–152.

Coelho, L.C., Laporte, G., 2014. Improved solutions for inventory-routing problems through valid inequalities and input ordering. Int. J. Prod. Econ. 155, 391–397.

Contardo, C., Cordeau, J.-F., Gendron, B., 2013. A computational comparison of flow formulations for the capacitated location-routing problem. Discrete Optim. 10, 263–295.

Díaz-Madroñero, M., Peidro, D., Mula, J., 2015. A review of tactical optimization models for integrated production and transport routing planning decisions. Comput. Ind. Eng. 88, 518–535.

Doerner, K.F., Schmid, V., 2010. Survey: Matheuristics for rich vehicle routing problems. In: Hybrid Metaheuristics. Springer Berlin Heidelberg, pp. 206–221.

Fischetti, M., Fischetti, M., 2018. Matheuristics. In: Handbook of Heuristics. Springer International Publishing, pp. 121–153.

Fischetti, M., Monaci, M., 2014. Proximity search for 0-1 mixed-integer convex programming. J. Heuristics 20, 709–731.

Hanafi, S., Todosijević, R., 2017. Mathematical programming based heuristics for the 0–1 MIP: a survey. J. Heuristics 23, 165–206.

Hemmati, A., Hvattum, L.M., 2016. Evaluating the importance of randomization in adaptive large neighborhood search. Int. Trans. Oper. Res. 24, 929–942.

Hrabec, D., Hvattum, L.M., Hoff, A., 2022. The value of integrated planning for production, inventory, and routing decisions: A systematic review and meta-analysis. Int. J. Prod. Econ. 248, 108468.

Johnson, D.S., Demers, A., Ullman, J.D., Garey, M.R., Graham, R.L., 1974. Worst-case performance bounds for simple one-dimensional packing algorithms. SIAM J. Comput. 3, 299–325.

Klotz, E., 2014. Identification, assessment, and correction of ill-conditioning and numerical instability in linear and integer programs. In: Bridging Data and Decisions. INFORMS, pp. 54–108.

Lin, S., Kernighan, B.W., 1973. An effective heuristic algorithm for the traveling-salesman problem. Oper. Res. 21, 498–516.

Lysgaard, J., Letchford, A.N., Eglese, R.W., 2004. A new branch-and-cut algorithm for the capacitated vehicle routing problem. Math. Program. 100, 423–445.

Manousakis, E.G., Kasapidis, G.A., Kiranoudis, C.T., Zachariadis, E.E., 2021a. An infeasible space exploring matheuristic for the production routing problem. European J. Oper. Res.

Manousakis, E., Repoussis, P., Zachariadis, E., Tarantilis, C., 2021b. Improved branch-and-cut for the inventory routing problem based on a two-commodity flow formulation. European J. Oper. Res. 290, 870–885.

Miranda, P.L., Cordeau, J.-F., Ferreira, D., Jans, R., Morabito, R., 2018. A decomposition heuristic for a rich production routing problem. Comput. Oper. Res. 98, 211–230.

Mousavi, R., Bashiri, M., Nikzad, E., 2022. Stochastic production routing problem for perishable products: Modeling and a solution algorithm. Comput. Oper. Res. 142, 105725.

Neves-Moreira, F., Almada-Lobo, B., Cordeau, J.-F., Guimarães, L., Jans, R., 2019. Solving a large multi-product production-routing problem with delivery time windows. Omega 86, 154–172.

Padberg, M., Rinaldi, G., 1991. A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. SIAM Rev. 33, 60–100.

Pirie, W., 2006. Spearman rank correlation coefficient. In: Encyclopedia of Statistical Sciences. John Wiley & Sons, Ltd.

Prins, C., 2004. A simple and effective evolutionary algorithm for the vehicle routing problem. Comput. Oper. Res. 31, 1985–2002.

Rodrigues, F., Agra, A., Hvattum, L.M., Requejo, C., 2021. Weighted proximity search. J. Heuristics 27, 459–496.

Russell, R.A., 2017. Mathematical programming heuristics for the production routing problem. Int. J. Prod. Econ. 193, 40–49.

Schenekemberg, C.M., Scarpin, C.T., Pecora, Jr., J.E., Guimarães, T.A., Coelho, L.C., 2021. The two-echelon production-routing problem. European J. Oper. Res. 288, 436–449.

Sousa, T., Morais, H., Castro, R., Vale, Z., 2016. Evaluation of different initial solution algorithms to be used in the heuristics optimization to solve the energy resource scheduling in smart grids. Appl. Soft Comput. 48, 491–506.

Vadseth, S.T., Andersson, H., Stålhane, M., Chitsaz, M., 2023. A multi-start route improving matheuristic for the production routeing problem. Int. J. Prod. Res. 1–22.