Multiscale Production Routing in Multicommodity Supply Chains with Complex Production Facilities

Qi Zhang[a], Arul Sundaramoorthy[b], Ignacio E. Grossmann[a,*], Jose M. Pinto[c]

[a] *Center for Advanced Process Decision-making, Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA*
[b] *Business and Supply Chain Optimization R&D, Praxair, Inc., Tonawanda, NY 14150, USA*
[c] *Business and Supply Chain Optimization R&D, Praxair, Inc., Danbury, CT 06810, USA*

Abstract

In this work, we introduce the multiscale production routing problem (MPRP), which considers the coordination of production, inventory, distribution, and routing decisions in multicommodity supply chains with complex production facilities. We propose an MILP model involving two different time grids. While a detailed mode-based production scheduling model captures all critical operational constraints on the fine time grid, vehicle routing is considered in each time period of the coarse time grid. In order to solve large instances of the MPRP, we propose an iterative MILP-based heuristic approach that solves the MILP model with a restricted set of candidate routes at each iteration and dynamically updates the set of candidate routes for the next iteration. The results of an extensive computational study show that the proposed algorithm finds high-quality solutions in reasonable computation times, and in large instances, it significantly outperforms a standard two-phase heuristic approach and a solution strategy involving a one-time heuristic pre-generation of candidate routes. Similar results are achieved in an industrial case study, which considers a real-world industrial gas supply chain with 2 plants, approximately 240 customers, 20 vehicles, and a planning horizon of 4 weeks, resulting in 168 time periods on the fine grid and 56 time periods on the coarse grid.

Keywords: Production routing, supply chain management, multiscale optimization, MILP-based heuristic

*Corresponding author
Email address: `grossmann@cmu.edu` (Ignacio E. Grossmann)

1. Introduction

In today's competitive market environment, it is becoming increasingly important for companies in the process industry to improve the performance of their supply chains. One widely acknowledged approach for achieving a more efficient and reliable supply chain is the integrated planning of multiple supply chain operations such as production, inventory, and distribution (Thomas and Griffin, 1996; Erengüç et al., 1999). Typically, these operations are optimized in a sequential manner. For example, one may first forecast the demand for each production plant and set up production plans that minimize production and inventory costs at the plants. Then, using the production decisions as inputs, distribution planning is performed, which minimizes the distribution costs. However, since the distribution planning is restricted by the production planning decisions, the solution may be suboptimal due to the lack of coordination. In contrast, with an integrated supply chain planning approach, several major planning decisions are optimized simultaneously, which can result in significant cost savings, as shown in some recent successful industrial implementations (Brown et al., 2001; Çetinkaya et al., 2009).

Among the integrated supply chain planning problems, the so-called production routing problem (PRP), also sometimes referred to as the production inventory distribution routing problem (PIDRP), is the most comprehensive one as it considers production, inventory, distribution, and routing decisions simultaneously. The PRP in its classical form can be formulated as a mixed-integer linear program (MILP) and integrates two well-known problems, namely the lot-sizing problem (LSP) and the inventory routing problem (IRP), where the latter is an extension of the vehicle routing problem (VRP). For details on these more extensively studied subproblems of the PRP, we point the interested reader to the following references: Karimi et al. (2003) and Pochet and Wolsey (2006) for the LSP, Toth and Vigo (2002) and Laporte (2009) for the VRP, and Campbell et al. (1998) and Coelho et al. (2013) for the IRP. It is important to note that in the PRP, there is flexibility in the inventories not only at the production plants but also at the customer locations; hence, customer inventory levels are decision variables, and an IRP has to be solved as a subproblem. In many other integrated production and routing problems, fixed orders are assumed, which excludes the possibility of leveraging the customers' storage capacities.

Although the PRP has received increased attention in recent years, the literature on this subject remains scarce. In one of the first works on production routing, Chandra and Fisher (1994) consider a single plant producing multiple products for several customers. Two approaches are compared to each other, one in which the LSP and the IRP are solved sequentially, and another in which an integrated PRP is solved. Although the PRP is solved heuristically, the results show that cost reduction of 3–20 % can be achieved by applying the integrated approach. Fumero and Vercellis (1999) solve a similar problem using Lagrangean relaxation and obtain cost savings of the same order of magnitude.

Due to its high combinatorial complexity, the PRP is notoriously hard to solve. As a result, most existing solution approaches involve various heuristic procedures. Like several others, Lei et al. (2006) propose a two-phase approach for solving the PRP. In Phase I, the integrated problem is solved, allowing only direct shipments from plants to customers. The resulting inefficiencies are handled in Phase II by minimizing the transportation cost for each plant and each time period with the plant-to-customer allocation decisions from Phase I. Bard and Nananukul (2009a, 2010) apply a branch-and-price algorithm in which column generation is applied to solve the linear programming (LP) relaxation at each node of the branch-and-bound tree. Here, each generated column corresponds a feasible routing schedule in a particular time period. Since the pricing subproblems are extensions of the VRP, they are very difficult to solve exactly; hence, a heuristic two-step algorithm is applied, which determines delivery quantities for each customer in each time period in the first step and then finds actual routes using a VRP tabu search code in the second step. Archetti et al. (2011) first solve an IRP heuristically while assuming infinite production at the plant. Then an LSP is solved with the production quantities obtained from solving the IRP. Finally, the solution is improved iteratively by removing and inserting two customers at a time. Absi et al. (2013) propose an iterative two-phase approach. The first phase involves solving a PRP in which the routing part is simplified by direct assignment of vehicles to customers and time periods with fixed visiting costs. Routes are then constructed in the second phase, and based on the routing solution, the visiting costs are updated for the next iteration.

Besides the general heuristics reviewed above, metaheuristics have also been applied to the PRP. Boudia et al. (2007) propose a greedy randomized adaptive search procedure (GRASP) consisting of two main phases: construction and local search. In the construction phase, an initial solution is generated by developing a production and distribution plan sequentially for each time period without creating inventory at the plant. Then in the local search phase, the routing plan is improved by simple local moves in the same time period as well as across multiple time periods. Boudia and Prins (2009) introduce a memetic algorithm, which can be seen as a modified genetic algorithm that applies an improvement procedure to each generated solution. Bard and Nananukul (2009b) first solve an LSP with direct shipments and then apply tabu search to make routing decisions based on the solution of the LSP. Armentano et al. (2011) propose a similar tabu search algorithm and further incorporate a path relinking procedure, in which new solutions are generated by connecting high-quality solutions or solutions that exhibit contrasting features. Adulyasak et al. (2014b) apply an adaptive large neighborhood search (ALNS) algorithm, where at each iteration, a selection operator is applied to create a list of customer-time period combinations, and then a transformation operator is applied to remove or reinsert some of these customer-time period combinations to the current solution.

Only a handful of exact solution methods have been proposed for the PRP. Bard and Nananukul (2009a, 2010) propose a rigorous branch-and-price algorithm; however, it can only solve instances with up to 10 customers, 5 vehicles, and 2 time periods. Different branch-and-cut algorithms have been developed by Ruokokoski et al. (2010), Archetti et al. (2011), and Adulyasak et al. (2014a).

In all proposed branch-and-cut procedures, subtour elimination constraints are added as cuts when solving the LP relaxations at the nodes of the branch-and-bound tree. In addition, Ruokokoski et al. (2010) formulate so-called generalized comb and 2-matching inequalities that can be used as cuts in the algorithm. Adulyasak et al. (2014a) apply branch-and-cut to two different formulations of the PRP, one with and one without a vehicle index. Computational tests show that the vehicle index formulation is superior in finding optimal solutions, whereas the nonvehicle index formulation generally provides better bounds on larger instances that cannot be solved to optimality. Adulyasak et al. (2015a) further apply the proposed branch-and-cut algorithm combined with Benders decomposition to solve the two-stage and multistage stochastic PRP with demand uncertainty.

For more details on formulations and solution algorithms for the classical PRP, we refer to the excellent review of Adulyasak et al. (2015b), from which the following insights, among other ones, can be drawn: (1) The vast majority of existing works only consider problems with one plant and one product. (2) The state-of-the-art heuristic algorithms for the PRP can solve instances with up to 200 customers and 20 time periods. (3) The best existing exact algorithms can handle single-vehicle instances with up to 80 customers and 8 time periods.

Our work is concerned with an extension of the classical PRP and is motivated by the challenge of managing production and distribution operations in industrial gas supply chains. In the so-called merchant liquid business, industrial gas companies distribute liquid products (liquid oxygen, nitrogen, argon, hydrogen, etc.) in bulk to the customers using tractor-trailers. The products can be stored in tanks at the customer sites. Here, the concept of vendor-managed inventory (VMI) is applied such that the industrial gas companies have control over their customers' inventories. These are highly integrated supply chains with multiple products, multiple production plants, and typically hundreds of customers. Moreover, cryogenic air separation plants, which are used to produce high-quality industrial gases, are tightly integrated and highly power-intensive processes. Hence, when optimizing the operation of such plants, detailed scheduling models have to be applied that can capture all critical process features including time-sensitive prices, interdependent production rates, and constraints on transitions between operating points. This level of accuracy on the production side cannot be achieved by a lot-sizing model as used in the classical PRP formulation.

For the industrial gas supply chain case, Glankwamdee et al. (2008) formulate a simplified production-distribution LP model that does not consider routing decisions; instead, the distribution part is approximated by resource constraints on truck and driver hours required for the planned deliveries. Marchetti et al. (2014) propose a production routing framework in which a heuristic is applied to generate a number of routes a priori, where a route is defined as a set of customers that can be visited in one trip. These routes are then included in the integrated model such that the assignment of routes to available trucks can be optimized. A large-scale industrial test case with 2 products, 4 plants, 4 depots, 168 customers, and 14 time periods has been considered, for which

CPLEX finds a good feasible solution with an optimality gap of 3.6 % within 5 h. However, it should be mentioned that in this particular case, the delivery quantities are given, i.e. there is no inventory management at the customer sites involved. In their proposed frameworks, Glankwamdee et al. (2008) and Marchetti et al. (2014) apply rather simplistic models of the production processes, which can be a serious drawback as process dynamics are not accurately represented, and hence, solutions may be suboptimal or even infeasible when implemented in practice. Zamarripa et al. (2016) apply a rolling horizon heuristic to large-scale instances of the model proposed by Marchetti et al. (2014), obtaining near-optimal solutions in shorter computation times.

The goal of this work is to develop a production routing framework that can consider large-scale multicommodity multiplant supply chains with complex production facilities, such as industrial gas supply chains. The desired outputs are twofold: a production schedule that can be readily implemented, and plant-to-customer allocation decisions that can be used as input for a subsequent detailed inventory routing tool, such as the one developed by Dong et al. (2014). We propose a multiscale PRP (MPRP) model involving two time grids, a fine one for production scheduling and a coarse one for distribution planning. A detailed production scheduling model is applied to capture all critical operational constraints, and in order to obtain accurate distribution costs and guarantee feasible distribution decisions, routing is considered in each time period of the coarse time grid. Note that the MPRP is more involved than the classical PRP because of the added complexity on the production side. For solving large instances of the MPRP, we propose an MILP-based heuristic approach that relies on applying the integrated MILP model and a dynamic route generation procedure in an iterative fashion. The effectiveness of the proposed solution method is demonstrated in an extensive computational study as well as in a real-world industrial test case.

The remainder of this paper is organized as follows. After stating the problem in Section 2, we present the MPRP formulation in Section 3. Section 4 provides a description of the proposed solution method. In Section 5, the proposed approach is applied to various MPRP instances, including an illustrative example to show the main features of the model, an extensive computational study to demonstrate the effectiveness of the solution method, and a real-world industrial-scale test case with data provided by Praxair. Finally, in Section 6, we close with a summary of the results and concluding remarks.

2. Problem statement

We consider a multicommodity supply chain that is characterized by a set of products $i \in I$, a set of production plants $p \in P$, of which each can produce all or a subset of the products, and a number of product-specific customers, of which each customer $c \in C_i$ has a given demand and storage capacity for product i.

We assume that each production plant can operate in a set of discrete operating modes $m \in M_p$, where each mode is defined by its production capacity and cost function. The complexity in the production process arises from the fact that generally, the products cannot be produced independently from each other; hence, correlations in production rates have to be considered. Furthermore, the dynamic behavior of the plant is constrained by restrictions on the rate of change and transitions between operating modes. The plants have inventory capacities for storable products.

Product-specific vehicles, e.g. tanker trucks, are used to transport products from the plants to the customers. Each vehicle is assigned to one particular plant and is defined by its capacity, speed, and cost, which may include fuel and labor costs. For every trip, a vehicle leaves the plant, visits one or multiple customers, and returns to the plant at the end of the trip. The length of a trip is limited.

The goal of the MPRP is to optimize production and distribution operations at different levels of decision-making for a given planning horizon. On the production side, the solution should provide a detailed production schedule involving the following decisions for every time period: the operating mode, the production rate for each product, and the amounts of products stored. On the distribution side, we want to make tactical decisions regarding plant-to-customer allocation; hence, we determine the amounts of products distributed from each plant to each customer and the assignment of vehicles to trips. Since feasibility has to be guaranteed, more detailed routing decisions may be obtained as a byproduct of the solution method; however, these decisions are not required since detailed routing will be subject to reoptimization in a separate subsequent step in which plant-to-customer allocation is fixed.

3. Model formulation

We propose an MILP model for the MPRP, for which the mathematical formulation is presented in the following. Note that all continuous variables in this model are constrained to be nonnegative. A list of indices, sets, parameters, and variables is given in the Nomenclature section.

3.1. Multiscale time representation

In the proposed model, a discrete time representation is applied. While short-term operational decisions are made on the production side, mid-term tactical decisions are made on the distribution side; hence, two different time scales have to be considered. We create two time grids, one with a fine and the other with a coarse time discretization, where the time horizon is divided into time periods of the lengths Δt^{f} and Δt^{c}, respectively, with Δt^{c} chosen to be a multiple of Δt^{f}. For the sake of clarity, we refer to a time period in the fine time grid as a level-1 time period and to a time period in the coarse time grid as a level-2 time period whenever this distinction is necessary. Also, the notation is such that time period t starts at time point $t-1$ and ends at time point t.

In the fine time grid, the planning horizon is defined by the set of time periods $\overline{T}^{\mathrm{f}} = \{1, 2, \ldots, \hat{t}^{\mathrm{f}}\}$, a subset of $T^{\mathrm{f}} = \{-\theta^{\max}+1, -\theta^{\max}+2, \ldots, 0, 1, \ldots, \hat{t}^{\mathrm{f}}\}$, which also includes time periods in the past that are used in some constraints involving mode transition variables. The coarse time discretization creates the following two sets of time periods: $\overline{T}^{c} = \{1, 2, \ldots, \hat{t}^{c}\}$ and $T^{c} = \{0, 1, \ldots, \hat{t}^{c}\}$. Furthermore, we define a set $\widetilde{T} = \{1, \Delta t^{c}/\Delta t^{f} + 1, 2\Delta t^{c}/\Delta t^{f} + 1, \ldots, (\hat{t}^{c} - 1)\Delta t^{c}/\Delta t^{f} + 1\}$, which is the set of level-1 time periods that begin at the same time points as the corresponding level-2 time periods.

We illustrate the notation with the example shown in Figure 1. Here, we have a planning horizon of 12 h, $\Delta t^{f} = 1$ h, $\Delta t^{c} = 4$ h; consequently, $\hat{t}^{f} = 12$, $\hat{t}^{c} = 3$, and $\widetilde{T} = \{1, 5, 9\}$. Also, we have $\theta^{\max} = 4$. Note that $\overline{T}^{f}$ and $\overline{T}^{c}$ only refer to the planning horizon starting with time period 1.

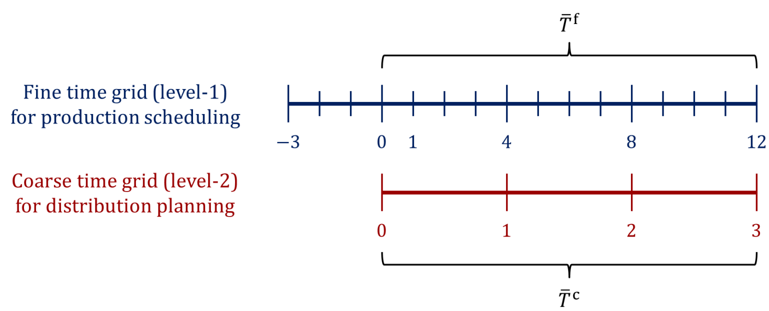


Figure 1: Fine and coarse time grids for a planning horizon of 12 h with $\Delta t^{f} = 1$ h and $\Delta t^{c} = 4$ h.

3.2. Production scheduling

The following production scheduling model is based on a mode-based formulation developed in previous works (Mitra et al., 2012; Zhang et al., 2016); hence, we only provide brief descriptions of the constraints.

3.2.1. Plant model

In this model, we assume that each plant can operate in different operating modes, which represent operating states such as “off”, “on”, and “startup”. To capture the complexity in the production process, the feasible region for each mode is defined by a union of convex subregions in the product space, and a linear cost function with respect to the production rates is given for each subregion. The key feature here is that every subregion has the form of a polytope. Such a model is generally referred to as a Convex Region Surrogate (CRS) model. For complex processes, CRS models can be constructed by either using a model-based (Sung and Maravelias, 2009) or a data-driven approach (Zhang et al., 2015).

At any point in time, a plant can only operate in one operating mode. For a given mode, the operating point has to lie in either one of the corresponding convex subregions. Any point in a subregion can be represented as a convex combination of the vertices of the polytope. These relationships can be described by the following constraints:

$$PD_{pit} = \sum_{m \in M_p} \sum_{r \in R_{pm}} \overline{PD}_{pmrit} \quad \forall\, p,\, i,\, t \in \overline{T}^{\mathrm{f}} \tag{1a}$$

$$\overline{PD}_{pmrit} = \sum_{j \in J_{pmr}} \lambda_{pmrjt}\, v_{pmrji} \quad \forall\, p,\, m \in M_p,\, r \in R_{pm},\, i,\, t \in \overline{T}^{\mathrm{f}} \tag{1b}$$

$$\sum_{j \in J_{pmr}} \lambda_{pmrjt} = \bar{y}_{pmrt} \quad \forall\, p,\, m \in M_p,\, r \in R_{pm},\, t \in \overline{T}^{\mathrm{f}} \tag{1c}$$

$$y_{pmt} = \sum_{r \in R_{pm}} \bar{y}_{pmrt} \quad \forall\, p,\, m \in M_p,\, t \in \overline{T}^{\mathrm{f}} \tag{1d}$$

$$\sum_{m \in M_p} y_{pmt} = 1 \quad \forall\, p,\, t \in \overline{T}^{\mathrm{f}} \tag{1e}$$

where M_p is the set of operating modes in which plant p can operate, R_{pm} is the set of subregions in mode $m \in M_p$, and J_{pmr} is the set of vertices of subregion $r \in R_{pm}$. The binary variable y_{pmt} equals 1 if mode $m \in M_p$ is selected in time period t, whereas the binary variable $\bar{y}_{pmrt}$ equals 1 if subregion $r \in R_{pm}$ is selected in time period t. The amount of product i produced at plant p in time period t is denoted by PD_{pit}. Associated with PD_{pit} is the disaggregated variable $\overline{PD}_{pmrit}$ for subregion $r \in R_{pm}$, which is expressed as a convex combination of the corresponding vertices, v_{pmrji}. Notice that Eqs. (1) are written for all $t \in \overline{T}^{\mathrm{f}}$, which refers to the fine time discretization.

3.2.2. Transition constraints

A transition occurs when the system changes from one operating point to another. For changes between operating points in the same operating mode, a bound on the rate of change, $\bar{\Delta}^{\max}_{pmi}$, can be set as follows:

$$-\bar{\Delta}^{\max}_{pmi} \le \sum_{r \in R_{pm}} \left(\overline{P}_{pmrit} - \overline{P}_{pmri,t-1} \right) \le \bar{\Delta}^{\max}_{pmi} \quad \forall\, p,\, m \in M_p,\, i,\, t \in \overline{T}^{\mathrm{f}}. \tag{2}$$

Additional constraints have to be imposed on transitions between different operating modes, which is achieved by enforcing constraints (3)–(5). The binary variable $z_{pmm't}$ equals 1 if and only if plant p switches from mode $m \in M_p$ to mode $m' \in M_p$ at time t, which is stated in the following constraint:

$$\sum_{m' \in \overline{TR}_{pm}} z_{pm'm,t-1} - \sum_{m' \in \widehat{TR}_{pm}} z_{pmm',t-1} = y_{pmt} - y_{pm,t-1} \quad \forall\, p,\, m \in M_p,\, t \in \overline{T}^{\mathrm{f}} \tag{3}$$

where $\overline{TR}_{pm} = \{m' : (m', m) \in TR_p\}$ and $\widehat{TR}_{pm} = \{m' : (m, m') \in TR_p\}$ with TR_p being the set of all possible mode-to-mode transitions at plant p.

The restriction that a plant has to remain in a certain mode for a minimum amount of time after a transition is stated as follows:

$$y_{pm't} \ge \sum_{k=1}^{\theta_{pmm'}} z_{pmm',t-k} \quad \forall\, p,\, (m, m') \in TR_p,\, t \in \overline{T}^{\mathrm{f}} \tag{4}$$

with $\theta_{pmm'}$ being the minimum stay time in mode $m' \in M_p$ after switching to it from mode $m \in M_p$.

For predefined sequences, each defined as a fixed chain of transitions from mode m to mode m' to mode m'', we can specify a fixed stay time in mode m' by imposing the following constraint:

$$z_{pmm',t-\bar{\theta}_{pmm'm''}} = z_{pm'm''t} \quad \forall\, p,\, (m,m',m'') \in SQ_p,\, t \in \overline{T}^{\mathrm{f}} \tag{5}$$

where SQ_p is the set of predefined sequences for plant p and $\bar{\theta}_{pmm'm''}$ is the fixed stay time in mode m' in the corresponding sequence.

The following equations fix the initial mode of each plant according to the parameters y_{pm}^{ini} and include the required information on the mode switching history in the form of the parameters $z_{pmm't}^{\mathrm{ini}}$:

$$y_{pm,0} = y_{pm}^{\mathrm{ini}} \quad \forall\, p,\, m \in M_p \tag{6a}$$

$$z_{pmm't} = z_{pmm't}^{\mathrm{ini}} \quad \forall\, p,\, (m,m') \in TR_p,\, t \in \overline{T}^{\mathrm{f}},\, -\tilde{\theta}_p^{\max} + 1 \le t \le -1 \tag{6b}$$

with $\tilde{\theta}_p^{\max} = \max\left(\max\limits_{(m,m')\in TR_p} \{\theta_{pmm'}\}, \max\limits_{(m,m',m'')\in SQ_p} \{\bar{\theta}_{pmm'm''}\} \right)$, which defines for how far back in the past the mode switching information has to be provided. Note that the fine time discretization can then be established by using $\theta^{\max} = \max\limits_p \{\tilde{\theta}_p^{\max}\}$.

3.2.3. Inventory constraints

First, we distinguish between storable and nonstorable products by creating the two disjoint product sets $\bar{I}$ and $\hat{I}$, respectively. While in general, storable products have to be transported to the customer locations, demands for non-storable products are assumed to occur at the production plants. Therefore, for nonstorable products, it suffices to simply constrain the production to be higher than the demand:

$$PD_{pit} \ge \widehat{D}_{pit} \quad \forall\, p,\, i \in \hat{I},\, t \in \overline{T}^{\mathrm{f}} \tag{7}$$

where $\widehat{D}_{pit}$ denotes the demand for product i at plant p in time period t.

Formulating the inventory constraints for the storable and therefore transportable products requires the following assumption: The products distributed to the customers in each level-2 time period are loaded into the vehicles within the first Δt^{f} of the same level-2 time period. This restriction is necessary due to the multiple time scales and it ensures that we always have sufficient inventory such that vehicles can leave the plants close to the beginning of the time period; otherwise, the vehicles may not be able to complete their trips within the same time period. With this assumption, we arrive at the following inventory constraints:

$$IV_{pit} = IV_{pi,t-1} + PD_{pit} - LD_{ipt'} \quad \forall\, p,\, i \in \bar{I},\, t \in \widetilde{T},\, t' = \pi_t \tag{8a}$$

$$IV_{pit} = IV_{pi,t-1} + PD_{pit} \quad \forall\, p,\, i \in \bar{I},\, t \in \overline{T}^{\mathrm{f}} \setminus \widetilde{T} \tag{8b}$$

$$IV_{pit}^{\min} \leq IV_{pit} \leq IV_{pit}^{\max} \quad \forall\, p,\, i \in \bar{I},\, t \in \overline{T}^{\mathrm{f}} \tag{8c}$$

$$IV_{pi,0} = IV_{pi}^{\mathrm{ini}} \quad \forall\, p,\, i \in \bar{I} \tag{8d}$$

where IV_{pit} is the inventory level of product i at plant p at level-1 time point t, and $LD_{ipt'}$ is the amount of product i loaded into vehicles at plant p in level-2 time period t'. Since IV_{pit} and $LD_{ipt'}$ refer to time periods in different time grids, they need to be matched, which is achieved by introducing the parameter π_t, which denotes the level-2 time period that begins at the same time point as level-1 time period t. Eq. (8a) states that the inventory level at time point t is the inventory level at the previous time point plus the amount of product produced in time period t minus the amount loaded into vehicles in the same time period. Eq. (8b) tracks the inventory in time periods in which no product is drawn from the storage to be loaded into vehicles. Eq. (8c) sets lower and upper bounds on the inventory levels, denoted by $IV_{pit}^{\min}$ and $IV_{pit}^{\max}$, respectively. Eq. (8d) fixes the initial inventory level to the value of the parameter IV_{pi}^{ini}.

3.3. Distribution planning

For the modeling of the distribution planning part of the problem, we make the following assumptions:

- Vehicles that transport the same product have the same capacity, speed, and route-specific costs.
- At the end of every trip, a vehicle returns to its assigned production plant.
- Every trip is completed within a level-2 time period.
- A vehicle cannot make more than one trip in each level-2 time period.
- In each level-2 time period, a particular route can only be used by one vehicle.
- A customer can be visited by multiple vehicles in the same level-2 time period.

With these assumptions, we essentially have to solve in each level-2 time period a distance-constrained capacitated VRP (DCVRP) where each customer can be visited by multiple vehicles from multiple plants. However, we also manage the customers' inventories. As a result, we do not have fixed orders; instead, the amounts of products distributed to the customers are variables and therefore subject to optimization. Therefore, the level-2 time periods are coupled by the customer inventories, leading to an IRP over the entire planning horizon. Note that all following distribution planning constraints are formulated with respect to the coarse level-2 time discretization.

3.3.1. Flow conservation constraints

For each product i and time period t, flow conservation has to be satisfied at every node in the network representation of the distribution model, as depicted in Figure 2.

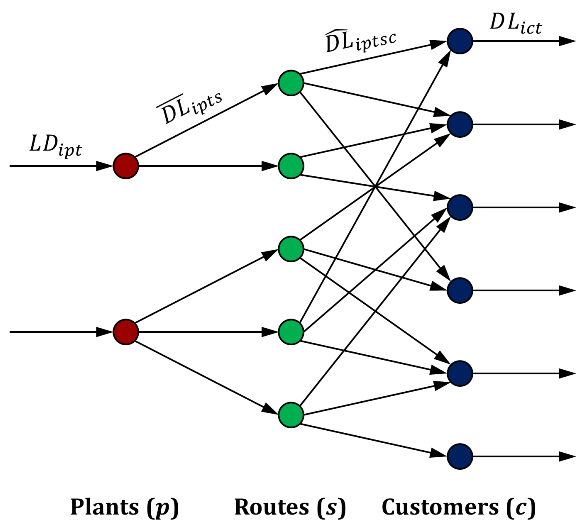


Figure 2: Network representation of the distribution model. Each node corresponds to a plant p, route s, or customer c. The notation of the flow variables is shown on the top arcs.

We apply a set-partitioning formulation (Balinski and Quandt, 1964) in which a set of feasible transportation routes is used where each route is defined as the set of customers that can be visited on the route. The resulting flow conservation constraints are as follows:

$$LD_{ipt} = \sum_{s \in \overline{S}_{ipt}} \overline{DL}_{ipts} \quad \forall\, i \in \bar{I},\, p,\, t \in \overline{T}^{c} \tag{9a}$$

$$\overline{DL}_{ipts} = \sum_{c \in \overline{C}_{ips}} \widehat{DL}_{iptsc} \quad \forall\, i \in \bar{I},\, p,\, t \in \overline{T}^{c},\, s \in \overline{S}_{ipt} \tag{9b}$$

$$DL_{ict} = \sum_{p} \sum_{s \in \overline{S}_{ipt}} \widehat{DL}_{iptsc} \quad \forall\, i \in \bar{I},\, c \in C_i,\, t \in \overline{T}^{c} \tag{9c}$$

where for time period t, $\overline{DL}_{ipts}$ denotes the amount of product i delivered on route s by a vehicle from plant p, $\widehat{DL}_{iptsc}$ is the amount delivered to customer c on route s, and DL_{ict} is the total amount of product i delivered to customer c. While S_{ip} denotes the set of routes that can be used by vehicles assigned to plant p, $\overline{S}_{ipt}$ is the subset of S_{ip} that can be used in time period t. The set of customers that can be visited on route $s \in S_{ip}$ is denoted by $\overline{C}_{ips}$.

3.3.2. Capacity constraints

The distribution resource constraints in terms of the vehicle capacity and the available number of vehicles are stated as follows:

$$\overline{DL}_{ipts} \le V_i\, x_{ipts} \quad \forall\, i \in \bar{I},\, p,\, t \in \overline{T}^{\mathrm{c}},\, s \in \overline{S}_{ipt} \tag{10a}$$

$$\widehat{DL}_{iptsc} \le \widehat{DL}^{\max}_{ict}\, x_{ipts} \quad \forall\, i \in \bar{I},\, p,\, t \in \overline{T}^{\mathrm{c}},\, s \in \overline{S}_{ipt},\, c \in \overline{C}_{ips} \tag{10b}$$

$$\sum_{s \in \overline{S}_{ipt}} x_{ipts} \le L_{ipt} \quad \forall\, i \in \bar{I},\, p,\, t \in \overline{T}^{\mathrm{c}} \tag{10c}$$

where V_i is the capacity of a vehicle transporting product i, and $\widehat{DL}^{\max}_{ict}$ can be set to $\min\left\{V_i,\, \overline{IV}^{\max}_{ict} - \overline{IV}^{\mathrm{low}}_{ic,t-1} + \overline{D}_{ict},\, \sum_{t'=t}^{\hat{t}^{\mathrm{c}}} \overline{D}_{ict'} + \overline{IV}^{\min}_{ic,\hat{t}^{\mathrm{c}}} - \overline{IV}^{\mathrm{low}}_{ic,t-1}\right\}$ with $\overline{IV}^{\mathrm{low}}_{ic,t-1} = \max\left\{\overline{IV}^{\min}_{ic,t-1},\, \overline{IV}_{ic,0} - \sum_{t'=1}^{t-1} \overline{D}_{ict'}\right\}$, which is the lowest possible inventory level at time $t-1$. The binary variable x_{ipts} equals 1 if a vehicle from plant p transporting product i takes route s in time period t. Eqs. (10a)–(10b) set upper bounds on the distribution variables and force them to zero if the corresponding routes are not selected. In Eq. (10c), L_{ipt} denotes the number of vehicles that transport product i and are available at plant p in time period t.

3.3.3. Inventory constraints

The constraints on the inventories at the customer sites are formulated as follows:

$$\overline{IV}_{ict} = \overline{IV}_{ic,t-1} + DL_{ict} + PC_{ict} - \overline{D}_{ict} \quad \forall\, i \in \bar{I},\, c \in C_i,\, t \in \overline{T}^{\mathrm{c}} \tag{11a}$$

$$\overline{IV}^{\min}_{ict} \le \overline{IV}_{ict} \le \overline{IV}^{\max}_{ict} \quad \forall\, i \in \bar{I},\, c \in C_i,\, t \in \overline{T}^{\mathrm{c}} \tag{11b}$$

$$\overline{IV}_{ic,0} = \overline{IV}^{\mathrm{ini}}_{ic} \quad \forall\, i \in \bar{I},\, c \in C_i \tag{11c}$$

where $\overline{IV}_{ict}$ denotes the inventory level for product i at customer c at time point t, and PC_{ict} is the amount of product purchased externally in case the demand, denoted by $\overline{D}_{ict}$, cannot be satisfied by drawing from the own inventory. Lower and upper bounds on $\overline{IV}_{ict}$ are denoted by $\overline{IV}^{\min}_{ict}$ and $\overline{IV}^{\max}_{ict}$, respectively, and $\overline{IV}^{\mathrm{ini}}_{ic}$ is the initial inventory.

3.4. Objective function

The objective is to minimize the total operating cost, TC, consisting of production costs, purchasing costs, distribution costs, and inventory costs; hence, the objective function is:

$$\begin{aligned} TC = &\sum_{p} \sum_{m \in M_p} \sum_{r \in R_{pm}} \sum_{t \in \overline{T}^{\mathrm{f}}} \left(\delta_{pmrt}\, \bar{y}_{pmrt} + \sum_{i} \gamma_{pmrit}\, \overline{PD}_{pmrit} \right) \\ &+ \sum_{i \in \bar{I}} \sum_{c \in C_i} \sum_{t \in \overline{T}^{\mathrm{c}}} \alpha_{ict}\, PC_{ict} + \sum_{i \in \bar{I}} \sum_{p} \sum_{t \in \overline{T}^{\mathrm{c}}} \sum_{s \in \overline{S}_{ipt}} \beta_{ips}\, x_{ipts} \\ &+ \sum_{p} \sum_{i \in \bar{I}} \sum_{t \in \overline{T}^{\mathrm{f}}} \rho_{pit} IV_{pit} + \sum_{i \in \bar{I}} \sum_{c \in C_i} \sum_{t \in \overline{T}^{\mathrm{c}}} \bar{\rho}_{ict} \overline{IV}_{ict} \end{aligned} \tag{12}$$

where δ_{pmrt} and γ_{pmrit} are the fixed and unit production costs, respectively, in operating subregion $r \in R_{pm}$ in level-1 time period t. The unit cost for purchasing product i to satisfy demand at customer c in level-2 time period t is α_{ict}. The fixed distribution cost for using route $s \in S_{ip}$ is β_{ips}. The unit inventory costs for storing product i in time period t at plant p and customer c are denoted by ρ_{ict} and $\bar{\rho}_{ict}$, respectively. With this objective function, the MILP for the MPRP then becomes:

$$\begin{aligned} \min \quad & TC \\ \text{s.t.} \quad & \text{Eqs. (1)–(12).} \end{aligned} \tag{MPRP}$$

4. Solution method

The difficulty in solving (MPRP) is mainly due to the integration of two very complex problems: a detailed MILP production scheduling problem and an IRP with high combinatorial complexity. Especially in such multiplant multicommodity supply chains, the interdependencies are very strong and have to be taken into account in order to obtain good solutions. In the following, we propose an MILP-based heuristic solution method involving dynamic route generation, which is designed to solve MPRPs of industrially relevant sizes.

In the distribution part of the proposed MPRP model, a set-partitioning formulation is applied where routing decisions are made by selecting a set of feasible routes. Note that a route is considered feasible if the trip time does not exceed $\bar{\tau}^{\max}$, which is typically set to Δt^{c}. This kind of formulation is known to exhibit a relatively tight LP relaxation, but it can require an exponential number of routes to fully describe the problem. However, at a feasible solution, only a very small fraction of all possible routes are selected. Hence, instead of working with the full route set, we propose to only consider a small subset of routes when solving (MPRP) and dynamically update the route set such that only good candidate routes are included. An outline of the proposed algorithm is as follows:

Step 1 For each product i and plant p, create an initial set of routes, S_{ip}. Each route $s \in S_{ip}$ is defined by the set of customers that can be reached on this route, C_{ips}, and the fixed distribution cost, β_{ips}. Furthermore, for each level-2 time period t, create $\overline{S}_{ipt}$, which is the subset of S_{ip} that is considered in time period t.

Step 2 Solve (MPRP) with the current set of possible routes.

Step 3 Based on the solution of (MPRP), add new routes to or remove existing routes from the current route set, i.e. update all S_{ip}, C_{ips}, β_{ips}, and $\overline{S}_{ipt}$.

Step 4 If a stopping criterion is satisfied, stop; otherwise, go to Step 2.

Since only a subset of all possible routes is considered when solving (MPRP) in Step 2, the computational complexity is reduced, but we are likely to only obtain a suboptimal solution. Inefficiencies on the distribution side are treated in Step 3 by updating the route set such that it includes candidate routes that can potentially improve the solution. The selection of new candidate routes is based on a local analysis of the current solution, i.e. it does not consider all relationships that exist in the integrated problem. Therefore, instead of directly applying a new route to improve the current solution, we decide whether the proposed route should be selected by solving (MPRP) in the next iteration.

The proposed solution algorithm is inspired by the concept of column generation, with the main difference being that here, new columns are generated by using a heuristic rather than by solving a rigorous pricing problem. In the following subsections, the major steps of the algorithm are described in more detail. As we will show, the solution is guaranteed to improve or at least remain the same at each iteration if (MPRP) is always solved to optimality. However, convergence to the optimal solution cannot be guaranteed, which is the main limitation of the proposed algorithm.

4.1. Initialization

For the initial set of routes, we may consider all single-stop routes, i.e. only one customer can be visited in each trip; however, in large-scale instances, even this route set can be prohibitively large. We realize that in most practical applications, the vast majority of the customers are only visited in a few time periods over the planning horizon. Hence, in order to reduce the number of single-stop routes considered in the initial iteration, we determine for each customer $c \in C_i$ the time periods in which it will likely be receiving delivery, and denote this set of time periods by T_{ic}^{del}. We then only consider feasible single-stop routes to customer c in these time periods as well as in the ω previous and ω following time periods, i.e. in time periods t such that $t' - \omega \leq t \leq t' + \omega$ where $t' \in T_{ic}^{\text{del}}$. By changing the parameter ω, we can adjust the number of routes included in the initial route set. We propose to determine T_{ic}^{del} as follows: Apply an inventory policy in which a customer's inventory is refilled to its maximum level, $\overline{IV}_{ict}^{\max}$, in time period t if otherwise the inventory level falls below $\overline{IV}_{ict}^{\min}$ at the end of time period t. Choose T_{ic}^{del} to be the set of replenishment points.

4.2. Updating set of candidate routes

Algorithm 1 shows the general scheme for generating routes based on the current solution of (MPRP), which may not be optimal or near-optimal (especially in the first iteration), but provides a good estimate of the amount of product that needs to be delivered to each customer in each time period. Using this information, the algorithm identifies inefficiencies in the current selection of routes and proposes new candidate routes that may improve the solution.

At each iteration, the algorithm is applied to every product $i \in \bar{I}$, plant p, and time period $t \in \overline{T}^{\text{c}}$. First, the procedure RemoveRoutes(i, p, t, Ω) removes routes that have not been selected for Ω consecutive iterations from the set S_{ipt}. Next, the distribution inefficiency due to underutilized vehicles is considered. We examine every selected route s for which the delivery quantity is less than the vehicle capacity, i.e. $\overline{DL}_{ipts} < V_i$. The procedure CreateRoutesA(i, p, t, s)

Algorithm 1 General scheme for route generation based on current solution.

for all $i \in \bar{I}$, p, $t \in \overline{T}^{\mathrm{c}}$ **do**
    RemoveRoutes(i, p, t, Ω)
    for all s for which $x_{ipts} = 1$ and $\overline{DL}_{ipts} < V_i$ **do**
        CreateRoutesA(i, p, t, s)
    end for
    for all $c \in \widehat{C}_{ip}$ for which $PC_{ict} > 0$ **do**
        CreateRoutesB(i, p, t, c)
    end for
end for

generates new routes, if possible, by inserting additional customers into the current route s. A selection of these new routes are added to the route set S_{ipt} based on a ranking of the potential savings. Besides underutilized vehicles, another indicator for distribution inefficiency is the purchase of products at high costs, which usually occurs due to the lack of efficient multistop routes. Hence, in the next step, we consider customers whose demands are met by purchasing additional products, i.e. all $c \in \widehat{C}_{ip}$ for which $PC_{ict} > 0$, where $\widehat{C}_{ip}$ is a subset of C_i and denotes the the set of customers that can be reached from plant p. Similar to CreateRoutesA(i, p, t, s), the procedure CreateRoutesB(i, p, t, c) generates multistop routes involving customer c and adds them to S_{ipt} based on a ranking of the potential savings.

In Algorithm 2, we describe the procedure CreateRoutesA(i, p, t, s) in more detail. As stated in lines 1–2, we first choose C^{del}, which is the set of customers to which delivery on a new route is considered. A customer is included in C^{del} if it can be reached from plant p, is not already part of route s, and is expected to receive delivery in time period t or any of the ξ subsequent time periods. We consider the latter condition to be satisfied if $PC_{ict'} > 0$ or $DL_{ict'} > 0$ for any t' between t and $t + \xi$. The parameter ξ can be adjusted to control the number of customers considered. The size of C^{del} increases with increasing ξ, and hence the computational effort increases; however, it has the benefit of making the search for better routes less localized. In lines 4–6, S^{pot} and S^{check} are initialized with the current route s. While S^{pot} is the set of potential new routes, S^{check} is the subset of S^{pot} that need to be further examined because more customers may be included in these routes. In general, the procedure AddRoute$(\widetilde{C}, \tilde{\beta}, \widetilde{S}, \tilde{n})$ adds the route characterized by customer set $\widetilde{C}$ and distribution cost $\tilde{\beta}$ to the route set $\widetilde{S}$, where the new route is indexed by $\tilde{n}$.

For each $s' \in S^{\mathrm{check}}$, $c \in C^{\mathrm{del}}$, we check whether by inserting customer c into route s' results in a new feasible route. By executing the procedure ComputeTSP$(i, p, \widetilde{C})$, the traveling salesman problem (TSP) is solved, which provides the minimum travel time, τ^{travel}, for a vehicle to transport product i from plant p to all customers in $\widetilde{C}$ and returning to the same plant at the end of the trip. In addition to the travel time, the time spent on a trip also includes the time that the vehicle stays at each location for the purpose of loading and

Algorithm 2 Create new routes based on route selected in the current solution.

1: **procedure** CreateRoutesA(i, p, t, s)
2: $C^{\text{del}} \leftarrow \{c : \sum_{t'=t}^{t+\xi} PC_{ict'} > 0, c \in \widehat{C}_{ip}, c \notin \overline{C}_{ips}\}$
3: $C^{\text{del}} \leftarrow C^{\text{del}} \cup \{c : \sum_{t'=t}^{t+\xi} DL_{ict'} > 0, c \in \widehat{C}_{ip}, c \notin \overline{C}_{ips}\}$
4: $S^{\text{pot}} \leftarrow \varnothing$, $n^{\text{pot}} \leftarrow 1$
5: AddRoute$(C_{ips}, \beta_{ips}, S^{\text{pot}}, n^{\text{pot}})$
6: $S^{\text{check}} \leftarrow \{n^{\text{pot}}\}$
7: **for all** $s' \in S^{\text{check}}, c \in C^{\text{del}}$ **do**
8: $\widetilde{C} \leftarrow C^{\text{pot}}_{s'} \cup \{c\}$
9: $\tau^{\text{travel}} \leftarrow$ ComputeTSP$(i, p, \widetilde{C})$
10: **if** $\tau^{\text{travel}} + \tau^{\text{stay}}_i(|\widetilde{C}| + 1) \le \bar{\tau}^{\max}$ **then**
11: $n^{\text{pot}} \leftarrow n^{\text{pot}} + 1$
12: $\tilde{\beta} \leftarrow \beta^{\text{travel}}_i \tau^{\text{travel}} + \beta^{\text{stay}}_i(|\overline{C}_{ips}| + 2)$
13: AddRoute$(\widetilde{C}, \tilde{\beta}, S^{\text{pot}}, n^{\text{pot}})$
14: $\overline{DL}^{\text{pot}}_{n^{\text{pot}}} \leftarrow$ ComputeLoad(n^{pot}, s')
15: $SAV^{\text{pot}}_{n^{\text{pot}}} \leftarrow$ ComputeSavings(n^{pot}, s')
16: **if** $\overline{DL}^{\text{pot}}_{n^{\text{pot}}} < V_i$ and $|C^{\text{pot}}_{n^{\text{pot}}}| < N^{\text{cmax}}$ **then**
17: $S^{\text{check}} \leftarrow S^{\text{check}} \cup \{n^{\text{pot}}\}$
18: **end if**
19: **end if**
20: **end for**
21: RankAndAdd$(S^{\text{pot}}, S_{ipt}, N^{\text{smax}})$
22: **end procedure**

unloading; hence, the total trip time is $\bar{\tau} = \tau^{\text{travel}} + \tau^{\text{stay}}_i(|\widetilde{C}| + 1)$, where τ^{stay}_i is the average time that a vehicle transporting product i spends at each location. A route is feasible if $\bar{\tau} \le \bar{\tau}^{\max}$.

If the potential new route is feasible, the distribution cost is computed, and the route is added to the route set S^{pot} (see lines 11–13). Here, β^{travel}_i and β^{stay}_i denote the unit travel cost and the fixed cost for loading and unloading, respectively. Then we apply the procedure ComputeLoad(n^{pot}, s') to compute $\overline{DL}^{\text{pot}}_{n^{\text{pot}}}$, which is an estimate of the vehicle load if route n^{pot} is used. This estimate is obtained by filling the vehicle used on route s' in a greedy fashion; for instance, if $\sum_{t'=t}^{t+\xi} PC_{ict'} > 0$, then $\overline{DL}^{\text{pot}}_{n^{\text{pot}}} = \min\left\{V_i, \overline{DL}^{\text{pot}}_{s'} + \sum_{t'=t}^{t+\xi} PC_{ict'}\right\}$. Under this vehicle load assumption, the savings of taking route n^{pot} instead of s', $SAV^{\text{pot}}_{n^{\text{pot}}}$, can be computed by using the procedure ComputeSavings(n^{pot}, s'). If there is still remaining capacity in the vehicle, i.e. $\overline{DL}^{\text{pot}}_{n^{\text{pot}}} < V_i$, and the number of customers on route n^{pot} has not reached the set maximum, N^{cmax}, n^{pot} is added to S^{check} such that it can be further examined and extended to another new route if possible. Finally, after the set of potential new routes, S^{pot}, is generated, CreateRoutesA(i, p, t, s) is completed by the procedure RankAndAdd$(S^{\text{pot}}, S_{ipt}, N^{\text{smax}})$, which ranks all routes in S^{pot} according to their potential savings and adds the top N^{smax} routes to S_{ipt}.

The route generation algorithm in CreateRoutesB(i,p,t,c) is very similar to the one in CreateRoutesA(i,p,t,s). The main difference is that CreateRoutesB(i,p,t,c) considers the single-top route from p to c as the initial route in S^{pot}, while CreateRoutesA(i,p,t,s) initializes S^{pot} with route s.

4.3. Stopping criteria

Let TC^k be the total cost value obtained by solving (MPRP) in iteration k. If (MPRP) is solved to optimality in every iteration, then $TC^k \geq TC^{k+1}$, i.e. the objective function value is guaranteed to improve or remain the same at each iteration. This statement holds since the routes selected in iteration k remain in the route set considered in iteration $k+1$, i.e. the optimal solution of (MPRP) in iteration k is a feasible solution of (MPRP) in iteration $k+1$.

In the form as it is presented here, the proposed algorithm does not guarantee convergence to the optimal solution. It can be modified such that at some point, all possible routes are included in the model. In that case, the algorithm would converge to the optimal solution; however, such an implementation has little practical value since industrial-scale instances of (MPRP) with all possible routes cannot be solved in a reasonable time. Our goal is to obtain good solutions in short computation times; hence, besides setting a time limit, we propose to terminate the algorithm when one of the following two stopping criteria applies:

1. The relative improvement in the objective function from one iteration to the next, defined as $(TC^k - TC^{k+1})/TC^k$, has been less than ϵ for Φ consecutive iterations.
2. Less than Ψ new routes have been generated in the current iteration.

4.4. Algorithmic parameters

In our computational experiments, the proposed algorithm has proven to be very robust with regard to the algorithmic parameters. In the following, we list all required parameters and provide guidelines for their settings:

- ω - number of time periods preceding and following time periods in T_{ic}^{del} that are considered in the alternative initial single-stop route generation procedure; we recommend setting ω to an integer between 0 and 3.

- Ω - number of consecutive iterations in which a route has not been used before it is deleted from the route set, typically set to 1 for the first iteration and 2 for all remaining iterations.

- N^{cmax} - maximum number of customers considered on a new route; we recommend setting N^{cmax} such that it increases with each iteration until it reaches the maximum number of customers at which efficient routes can still be expected, such a gradual increase in N^{cmax} prevents the algorithm from getting trapped in a local solution too quickly.

- N^{smax} - maximum number of new routes added to the route set after one run of CreateRoutesA(i,p,t,s) or CreateRoutesB(i,p,t,c), should be as large as the computational budget allows.

- ξ - number of subsequent time periods from which customers with deliveries or product purchases can be considered in a new route for the current time period, typically set to 1 or 2.

- ϵ - relative change between the costs from two consecutive iterations below which the improvement is considered insignificant; we recommend setting ϵ to a number between 0.001 and 0.01.

- Φ - number of consecutive iterations with no significant improvement after which the algorithm terminates, typically set to 2.

- Ψ - number of new routes below which the algorithm terminates, can be conservatively set to 5 or 10 in most applications.

5. Numerical results

In the following, we use an illustrative example to demonstrate the main features of the proposed framework, test the algorithm's performance in an extensive computational study, and apply it to a real-world industrial gas supply chain. Except for the industrial test case, the data for all problem instances are provided in the supplementary material. All models were implemented in GAMS 24.4.6 (GAMS Development Corporation, 2015), and the commercial solver CPLEX 12.6.2 (IBM ILOG, 2015) was applied to solve the MILPs on an Intel® Core™ i7-4770 machine at 3.40 GHz with 8 processors and 16 GB RAM running Windows 7 Enterprise.

In all instances, we set $\Omega = \min\{k, 2\}$ with k being the iteration counter, $\epsilon = 0.001$, $\Phi = 2$, $\Psi = 5$ when applying the proposed algorithm. The choice of the other algorithmic parameters varies slightly across the different instances.

5.1. Illustrative example

In the illustrative example, we consider a supply chain with two products, I1 and I2, two production plants, P1 and P1, and 50 customers, among which 20 require Product I1 and 30 require Product I2. The corresponding supply chain network is shown in Figure 3. Plant P1 has a fleet with 3 vehicles for Product I1 and 3 vehicles for Product I2; Plant P2 has 3 vehicles for Product I1 and 4 vehicles for Product I2.

The feasible regions of the given production modes are shown in Figure 4. Plant P1 can only operate in one mode, P1-M1, whereas Plant P2 can operate in three different modes, P2-M1, P2-M2, and P2-M3. Plant P2 cannot directly switch from Mode P2-M1 to Mode P2-M2; instead, it has to transition through the intermediate mode P2-M3. Note that Mode P2-M3 is described by a single operating point.

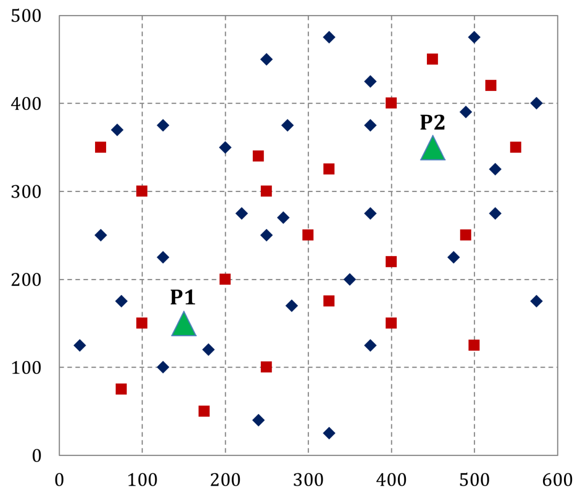


Figure 3: Supply chain network for the illustrative example, with two plants and 50 customers.

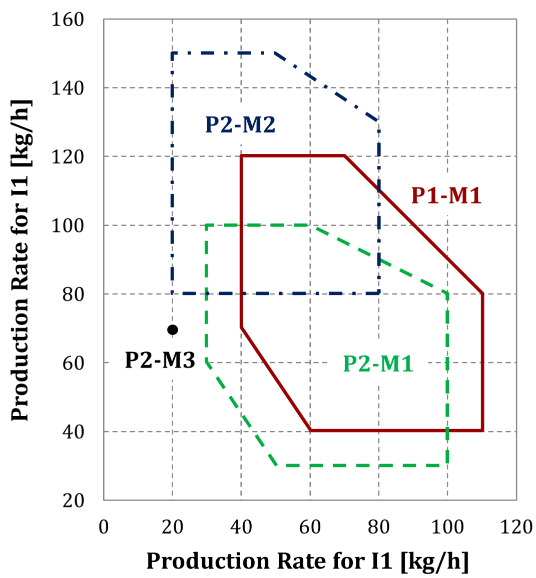


Figure 4: Feasible operating regions of the production modes given for the two plants in the illustrative example.

A scheduling horizon of 36 h is considered. We set $\Delta t^{\mathrm{f}} = 1$ and $\Delta t^{\mathrm{c}} = 12$, resulting in 36 level-1 and 3 level-2 time periods. The resulting MPRP has 9192 continuous variables, 2486 binary variables, and 11,197 constraints, and is solved to optimality in about 10 min. Figures 5 and 6 show the product flows and inventory profiles for both products at Plants P1 and P2, respectively. Note that in the figures, the y-axes for the inventory levels and the product flows are shown on the left and right hand sides, respectively. Positive columns (production) indicate accumulation of products in the inventory, while negative columns (shipments) indicate depletion of products. In overall, Plant P1 produces more than Plant P2 because of its lower unit production cost.

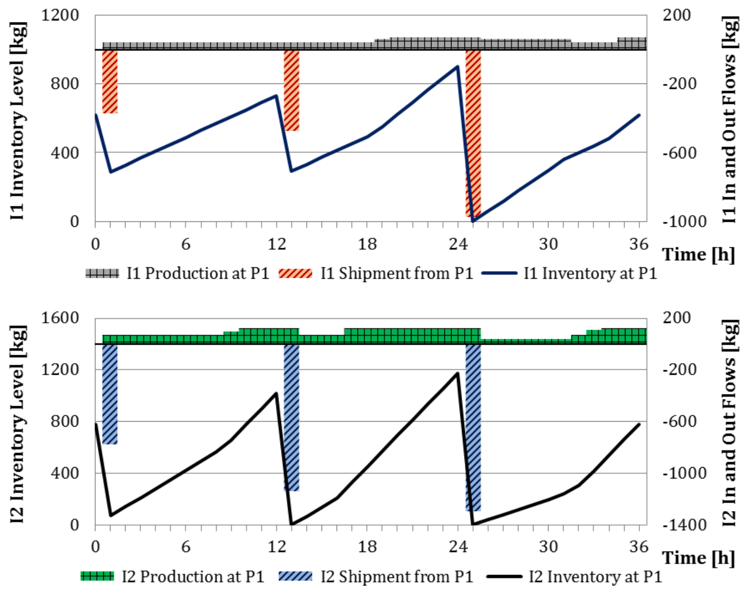


Figure 5: Production quantities, shipments, and inventory levels of products I1 and I2 at Plant P1 in the illustrative example.

The optimal routing decisions are shown in Figure 7, where each subfigure refers to one of the 3 level-2 time periods and shows the selected routes including the corresponding delivery quantities. The load capacity of each vehicle is 500 kg. As one can see, the solution suggests loading the vehicles as close to full capacity as possible. Also, not all vehicles are used in every time period. The total number of selected routes is 22.

We now apply the proposed heuristic solution algorithm to this illustrative example. The algorithmic parameters are set as follows: $\xi = 1$, $N^{\text{cmax}} = \min\{k + 1, 4\}$ where k is the iteration counter, and $N^{\text{smax}} = 4$. Furthermore, all possible single-stop routes are considered in the first iteration.

To illustrate the evolution of the solution from one iteration to the next, we show the routing decisions from the first three iterations in Figure 8. At Iteration 1 (Figure 8a), only single-stop routes are considered, resulting in the dispatch of a large number of vehicles, most of which only carry a fraction of the maximum possible load. A total number of 35 routes are selected in this initial solution. At

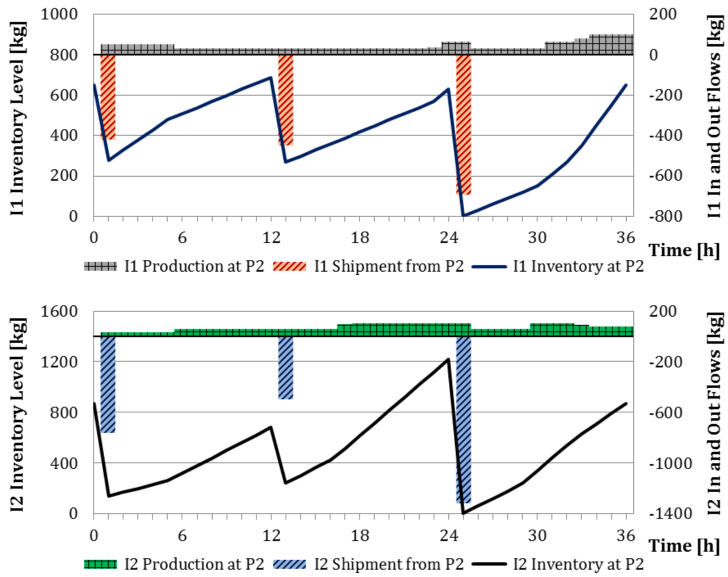


Figure 6: Production quantities, shipments, and inventory levels of products I1 and I2 at Plant P2 in the illustrative example.

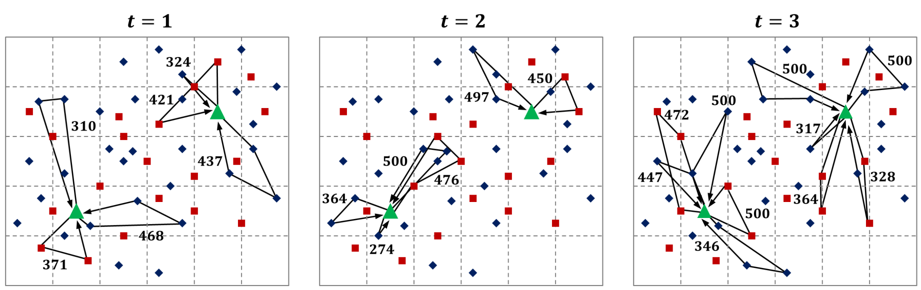
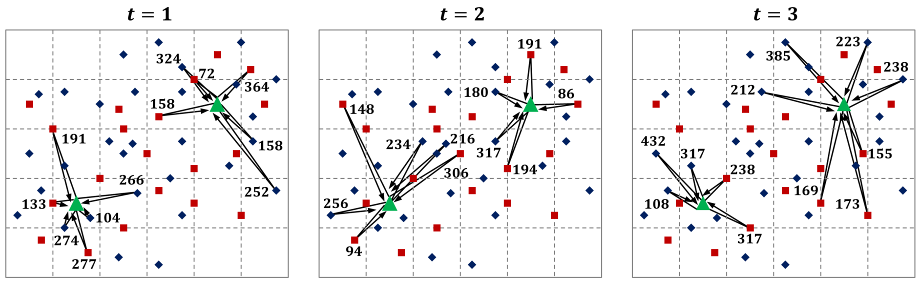


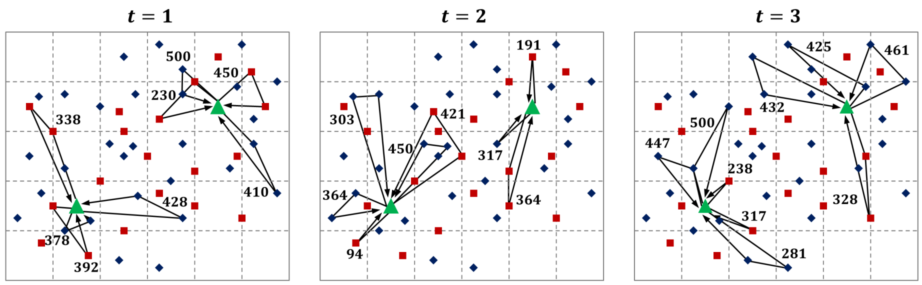
Figure 7: Optimal routing solution for the illustrative example.

Iteration 2 (Figure 8b), new candidate routes are considered, which also include routes with two customers. The solution obtained at Iteration 2 is significantly more efficient in terms of distribution, as indicated by larger amounts of products delivered with fewer vehicles. The change in the routing decisions is smaller from Iteration 2 to Iteration 3 than from Iteration 1 to Iteration 2. However, one can see that the distribution plan has been further improved by considering also routes with three customers. In the solution obtained at Iteration 3, a total number of 23 routes are selected; recall that 22 routes are selected in the optimal solution.

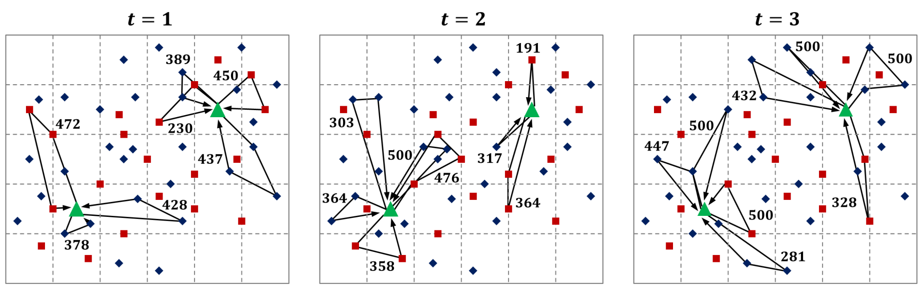
In this case, the algorithm terminates after five iterations; however, the same solution is obtained at Iterations 3 to 5, although different candidate routes are



(a) Solution from Iteration 1, considering only single-stop routes.



(b) Solution from Iteration 2, considering routes with up to two customers.



(c) Solution from Iteration 3, considering routes with up to three customers.

Figure 8: Evolution of the routing solution obtained from applying the proposed heuristic algorithm to the illustrative example.

considered. Note that no routes with four customers are selected although such routes are considered at Iterations 4 and 5. Table 1 compares the optimal solution with the solutions obtained at each iteration of the heuristic algorithm. The table shows the breakdown of the total costs (TC) into production costs (CPD), purchasing costs (CPC), distribution costs (CDI), inventory costs at the plants (CIP), and inventory costs at the customer sites (CIC). From the results of the heuristic algorithm, one can clearly see that CPD increases while CPC and CDI decrease from one iteration to the next, which indicates that improved routing decisions are made at each iteration such that more products can be delivered from the plants and less has to be purchased from external sources. The algorithm terminates after no improvement is seen at Iterations 4 and 5. The final heuristic solution exhibits a total cost of $57,345, which is 1 % higher than the total cost at the optimal solution ($56,760).

Table 1: Comparison of costs and number of candidate routes in the full MPRP formulation and the restricted MPRPs solved in the heuristic algorithm.

	Optimal	Heuristic				
		Iteration 1	Iteration 2	Iteration 3	Iteration 4	Iteration 5
TC [$]	56,760	67,475	58,543	57,345	57,345	57,345
CPD [$]	27,975	25,282	27,767	27,940	27,940	27,940
CPC [$]	1703	13,140	2586	1882	1882	1882
CDI [$]	24,050	25,928	25,130	24,456	24,456	24,456
CIP [$]	615	518	445	498	498	498
CIC [$]	2418	2606	2616	2569	2569	2569
NR	2076	198	352	459	249	229

Table 1 further shows the number of candidate routes, NR, considered in each problem. This number is obtained by counting the routes for all products, plants, and time periods, i.e. $NR = \sum_{i\in\bar{I}} \sum_p \sum_{t\in\overline{T}^c} |S_{ipt}|$. One can see that in each iteration of the heuristic algorithm, a route set of significantly smaller size is considered compared with the full MPRP formulation. Because of this reduction in problem size through the dynamic route generation procedure, the near-optimal heuristic solution was found in less than 20 s, which is a significant reduction in computation time compared to the 10 min required to solve the full MPRP model.

5.2. Computational study

In the following, we test the computational performance of the proposed algorithm on a set of MPRP instances of different sizes.

5.2.1. Data generation

For the computational study, we generate five sets of MPRP instances, Sets A to E, each containing ten instances of the same size. Table 2 lists for the instances in each data set the number of products, $|\bar{I}|$ (only storable products are considered), number of plants, $|P|$, number of customers for each product i, $|C_i|$, number of vehicles across all plants for each product i, $\sum_p \overline{L}_{ip}$ (with $\overline{L}_{ip}$ being the number of vehicles that can transport product i from plant p), number of level-1 time periods, $|\overline{T}^{\mathrm{f}}|$, and number of level-2 time periods, $|\overline{T}^{\mathrm{c}}|$. Note that the ratio between $|\overline{T}^{\mathrm{f}}|$ and $|\overline{T}^{\mathrm{c}}|$ is 12 in all instances.

The instances in each set differ in the customer locations, which are randomly generated on a 600×500 Euclidean grid, inventory capacities, initial inventory levels, and demands. While the customer demands are constant in the first five instances of each set, demands in the latter five instances vary over time.

Table 2: Overview of generated MPRP instances, grouped into five data sets.

Set	$\lvert\bar{I}\rvert$	$\lvert P\rvert$	$\lvert C_i\rvert$			$\sum_p \overline{L}_{ip}$			$\lvert\overline{T}^{\mathrm{f}}\rvert$	$\lvert\overline{T}^{\mathrm{c}}\rvert$
			I1	I2	I3	I1	I2	I3		
A	2	2	20	30	-	4	5	-	36	3
B	2	2	20	30	-	4	5	-	120	10
C	3	2	50	60	40	9	13	7	120	10
D	3	2	50	60	40	9	13	7	360	30
E	3	3	100	100	100	14	18	12	360	30

5.2.2. Solution methods

In the computational study, we compare the following four solution methods:

Exact method Solve (MPRP) considering all possible routes. Due to the computational limitations, the exact method is only applied to the instances in Sets A and B.

Heuristic H1 This is a typical two-phase heuristic. In Phase 1, we solve the MPRP with a simplified distribution model only considering direct shipments. Here, we only consider all feasible single-stop routes, relax the integrality constraints on the variables x_{ipts}, and solve (MPRP) where we replace Eq. (10c) by

$$\sum_{s\in\overline{S}_{ipt}} x_{ipts} \leq \eta\, L_{ipt} \quad \forall\, i \in \bar{I},\, p,\, t \in \overline{T}^{\mathrm{c}} \tag{13}$$

and add

$$\sum_{s\in\overline{S}_{ipt}} \tau_{ips}\, x_{ipts} \leq \bar{\eta}\, L_{ipt}\, \bar{\tau}^{\max} \quad \forall\, i \in \bar{I},\, p,\, t \in \overline{T}^{\mathrm{c}}. \tag{14}$$

Eqs. (13) and (14) are resource constraints on the total delivery quantity and travel time, respectively. Parameter η is typically set to a value between 0.8 and 1.2, while $\bar{\eta}$ can be set to a value between 1 and 1.5. Setting $\bar{\eta} > 0$ accounts for the overestimation of the travel time in the model caused by only considering single-stop routes. For the instances considered in this computational study, setting both η and $\bar{\eta}$ to 1 has proven to be a good choice.

The delivery quantities obtained from solving the simplified MPRP in Phase 1 are used as fixed orders in Phase 2, where routing decisions are made. Since the orders are fixed, the routing problem decomposes into independent subproblems, one for each product i, plant p, and time period $t \in \overline{T}^{\mathrm{c}}$. Each subproblem is a DCVRP with the additional option of purchasing products if the orders cannot be met by delivering from the plant. We solve the DCVRPs with the MILP formulation shown in Appendix A. After solving the routing problem, the total cost is updated by replacing the distribution cost from Phase 1 with the routing cost obtained in Phase 2, adding the purchasing cost from Phase 2, and discounting the production cost associated with products that should be delivered according to Phase 1 but could not in Phase 2.

Although the vast majority of the DCVRPs considered in Phase 2 are very small in size (only a few customers) and can be solved within seconds, we set a time limit of 60 s to avoid stalling of the algorithm.

Heuristic H2 Create a set of routes a priori and solve (MPRP) considering these candidate routes. The effectiveness of this solution strategy strongly depends on the number and quality of the generated routes. Here, we use the heuristic route generation procedure proposed by Marchetti et al. (2014), who have successfully applied this approach to industrial gas supply chain cases.

Marchetti et al. (2014) introduce four parameters: $cmax$, $smax$, $vmin$, and $vmax$. The route generation procedure first generates all feasible routes with up to $cmax$ customers, and computes for each route a so-called logistics ratio, which is the ratio between the distribution cost and the maximum quantity that can be delivered on this route. The routes with the lowest logistics ratios are selected to be considered in (MPRP) such that, if possible, each customer can be visited on at least $vmin$ and not more than $vmax$ routes, and the number of routes for each product and plant is not larger than $smax$. The resulting route set for each product and plant, S_{ip}, is considered in every time period, i.e. $\overline{S}_{ipt} = S_{ip} \; \forall \, t \in \overline{T}^{\mathrm{c}}$.

Table 3 shows the parameter settings chosen in this computational study. For Sets A–D, two runs of Heuristic H2, denoted by H2a and H2b, are performed, where H2b considers more routes.

Heuristic H3 Apply the MILP-based heuristic algorithm with dynamic route generation proposed in Section 4. The parameter settings for the different sets of instances are shown in Table 4. For solving the instances in Sets A–D, the algorithm is initialized with all possible single-stop routes, while for Set E, we create a smaller number of single-stop routes using the alternative procedure described in Section 4.1. For Sets B–E, we solve (MPRP) to 0.5 % optimality gap if possible, and further specify a time limit of 600 s for each MILP.

It should be mentioned that Heuristics H1 and H2 are solution approaches that are commonly used in practice for solving large-scale integrated supply chain problems like the MPRP; hence, we choose to compare the proposed algorithm, Heuristic H3, with these two solution strategies.

5.2.3. Results and discussion

In the following, we present and discuss the results from the computational study, which are are shown in Tables 5–9. For all instances and solution methods, we set a limit of 3600 s on the solution time. Note that the solution time

Table 3: Parameter settings for Heuristic H2.

Set	Heuristic H2a/H2				Heuristic H2b			
	cmax	*smax*	*vmin*	*vmax*	*cmax*	*smax*	*vmin*	*vmax*
A–D	4	200	2	5	4	200	5	10
E	3	200	2	5				

Table 4: Parameter settings for Heuristic H3, with k being the iteration counter.

Set	ξ	N^{cmax}	N^{smax}	ω
A–D	1	$\min\{k+2,4\}$	4	-
E	1	$\min\{k+1,3\}$	3	2

does not include the time required for pre-generating the candidate routes in the exact method and in Heuristic H2 because route generation in these two methods is considered to be an offline step that only is performed once. It should be mentioned that for the larger instances, this route generation procedure takes several hours. In contrast, dynamic route generation is performed online in Heuristic H3; hence, the required time is included in the reported solution time.

The tables list the following statistics:

- TC - total cost in \$.
- ST - solution time in s; ST is not reported if the limit of 3600 s is reached.
- OG - optimality gap in %, which is reported for the exact method and for Heuristic H2 if the MILP cannot be solved to zero optimality gap within the time limit; note that OG is the optimality gap output by the MILP solver, it is not the gap to the true optimal solution.
- NR - number of routes considered, reported for the exact method and Heuristic H2; recall that $NR = \sum_{i\in\bar{I}} \sum_p \sum_{t\in\overline{T}^c} |S_{ipt}|$.
- NR^* - maximum number of routes considered in an iteration of Heuristic H3.
- NI - number of iterations used in Heuristic H3.
- RD - relative difference to optimal (or near-optimal) solution in %, i.e. $RD = (TC - \overline{TC})/\overline{TC}$ with $\overline{TC}$ being the total cost obtained from solving the MPRP exactly.
- RI - relative improvement to Heuristic H1 in %, i.e. $RI = (\overline{TC} - TC)/\overline{TC}$ with $\overline{TC}$ being the total cost obtained from Heuristic H1.

Table 5: Comparison of solutions for all instances in Set A.

	Exact			Heuristic H1			Heuristic H2a				Heuristic H2b				Heuristic H3				
	TC	*ST*	*NR*	*TC*	*RD*	*ST*	*TC*	*RD*	*ST*	*NR*	*TC*	*RD*	*ST*	*NR*	*TC*	*RD*	*ST*	*NI*	*NR**
A1	51,960	31	1797	61,781	18.9	3	53,318	2.6	1	585	**52,026**	0.1	1	852	52,292	0.6	13	6	369
A2	52,923	16	1227	64,026	21.0	3	54,288	2.6	2	531	**52,923**	0.0	2	777	55,179	4.3	13	5	387
A3	49,534	25	2157	60,033	21.2	3	51,424	3.8	7	540	**50,446**	1.8	16	840	51,042	3.0	18	5	452
A4	48,156	12	2139	57,412	19.2	3	49,307	2.4	1	570	**48,442**	0.6	1	918	48,906	1.6	17	5	423
A5	50,405	9	2892	61,078	21.2	3	51,864	2.9	1	582	**51,427**	0.2	3	921	52,634	4.4	17	5	405
A6	53,045	5	1983	63,555	19.8	3	55,253	4.2	1	537	**53,483**	0.8	1	834	54,712	3.1	13	4	366
A7	53,126	65	1386	67,586	27.2	3	53,978	1.6	1	570	**53,126**	0.0	11	858	54,731	1.1	14	4	392
A8	49,862	32	1863	59,722	19.8	3	51,176	2.6	1	585	**50,053**	0.4	6	975	50,418	1.1	14	5	456
A9	52,000	51	1470	64,014	23.1	3	52,648	1.2	1	510	**52,367**	0.7	2	777	53,005	1.9	11	5	363
A10	50,341	30	1575	59,468	18.1	3	50,995	1.3	1	585	**50,373**	0.1	1	909	50,993	1.3	12	5	432
Avg.	51,135	28	1849	61,868	21.0	3	52,425	2.5	2	560	51,467	0.7	4	866	52,391	2.4	14	5	405

Table 6: Comparison of solutions for all instances in Set B.

	Exact			Heuristic H1			Heuristic H2a				Heuristic H2b				Heuristic H3				
	TC	*OG*	*NR*	*TC*	*RD*	*ST*	*TC*	*RD*	*OG*	*NR*	*TC*	*RD*	*OG*	*NR*	*TC*	*RD*	*ST*	*NI*	*NR**
B1	151,685	0.8	5990	170,619	12.5	10	152,252	0.4	0.9	1950	**151,640**	0.0	0.7	2840	153,548	1.2	424	6	600
B2	155,942	1.5	4090	180,944	16.0	11	157,082	0.7	1.5	1770	**156,012**	0.0	1.5	2590	159,846	2.5	740	4	580
B3	149,268	1.5	7190	174,001	16.6	11	149,739	0.3	1.5	1800	**149,208**	0.0	1.3	2800	152,801	2.4	62	5	738
B4	143,719	1.7	7130	163,509	13.8	11	**143,818**	0.1	1.3	1900	143,854	0.1	1.6	3060	147,309	2.5	531	6	709
B5	149,232	1.1	9640	173,951	16.6	11	149,762	0.4	0.8	1940	**149,249**	0.0	0.8	3070	152,315	2.1	624	5	630
B6	157,077	0.7	6610	183,503	16.8	12	157,888	0.5	0.3	1790	**157,010**	0.0	0.6	2780	160,328	2.1	52	5	601
B7	153,999	1.6	4620	179,971	16.9	11	154,190	0.1	1.6	1900	**153,900**	-0.1	1.5	2860	157,276	2.1	948	5	620
B8	143,700	0.9	6210	165,127	14.9	12	144,678	0.7	0.6	1950	**143,648**	0.0	0.8	3250	147,808	2.9	565	5	721
B9	151,021	1.2	4900	172,087	13.9	13	150,836	-0.1	1.0	1700	**150,820**	-0.1	1.0	2590	152,591	1.0	649	5	575
B10	146,996	1.4	5250	171,743	16.8	11	**146,668**	-0.2	1.0	1950	146,984	0.0	1.3	3030	149,617	1.8	508	5	697
Avg.	150,264	1.3	6163	173,546	15.5	11	150,691	0.3	1.0	1865	150,233	0.0	1.1	2887	153,344	2.1	510	5	647

Table 7: Comparison of solutions for all instances in Set C.

	Heuristic H1		Heuristic H2a				Heuristic H2b				Heuristic H3				
	TC	*ST*	*TC*	*RI*	*OG*	*NR*	*TC*	*RI*	*OG*	*NR*	*TC*	*RI*	*ST*	*NI*	*NR**
C1	265,473	26	228,752	13.8	0.9	5800	**227,257**	14.4	1.0	9470	233,491	12.0	85	5	1940
C2	255,050	28	232,683	8.8	1.0	5650	230,355	9.7	1.0	9430	**226,333**	11.3	86	5	2010
C3	259,630	17	233,913	9.9	1.0	5430	230,447	11.2	1.2	9040	**229,088**	11.8	91	7	1980
C4	273,655	20	249,064	9.0	0.7	5190	246,713	9.8	0.5	8720	**245,731**	10.2	187	6	1980
C5	253,007	22	230,963	8.7	0.9	5210	228,874	9.5	1.0	8380	**222,141**	12.2	147	6	2050
C6	257,728	18	231,888	10.0	0.7	5860	229,373	11.0	1.0	9510	**227,032**	11.9	91	6	2050
C7	260,528	56	228,780	12.2	0.7	5990	**225,787**	13.3	1.1	10,030	228,270	12.4	119	5	1900
C8	268,916	23	233,155	13.3	0.9	5900	**231,582**	13.9	1.2	10,010	235,908	12.3	109	11	1920
C9	257,249	19	239,763	6.8	0.7	5240	237,943	7.5	0.8	9170	**232,596**	9.6	85	7	2020
C10	249,358	78	234,768	5.9	0.5	5480	233,398	6.4	0.8	9220	**220,856**	11.4	94	6	2020
Avg.	260,059	31	234,373	9.8	0.8	5575	232,173	10.7	1.0	9298	230,145	11.5	109	6	1987

Moreover, for every instance, the lowest total cost obtained from a heuristic method is shown in bold.

All instances in Set A (see Table 5) are solved to optimality, most of them within one minute due to the moderate number of feasible routes (on average 1849). On average, the total cost obtained with Heuristic H1 is 21 % higher

Table 8: Comparison of solutions for all instances in Set D.

	Heuristic H1		Heuristic H2a				Heuristic H2b				Heuristic H3			
	TC	ST	TC	RI	OG	NR	TC	RI	OG	NR	TC	RI	NI	NR^*
D1	1,257,807	788	970,109	22.9	2.9	17,400	972,943	22.6	3.6	28410	**952,830**	24.2	5	9058
D2	1,300,298	1421	1,035,147	20.4	3.9	16,950	1,041,365	19.9	5.0	28,290	**991,955**	23.7	4	9505
D3	1,308,509	1285	1,030,173	21.3	3.9	16,290	1,037,219	20.7	5.7	27,120	**993,238**	24.1	4	9295
D4	1,321,906	1512	1,070,040	19.1	2.8	15,570	1,070,243	19.0	3.0	26,160	**1,027,871**	22.2	4	8194
D5	1,256,515	838	1,001,010	20.3	3.4	15,630	1,004,142	20.1	4.0	25,140	**949,683**	24.4	4	9596
D6	1,211,123	497	995,681	17.8	2.7	17,580	991,704	18.1	2.9	28,530	**976,056**	19.4	5	9415
D7	1,210,258	683	1,004,152	17.0	3.5	17,970	982,698	18.8	3.5	30,090	**963,653**	20.4	5	8459
D8	1,275,422	379	1,042,101	18.3	2.6	17,700	1,042,662	18.2	3.3	30,030	**1,031,971**	19.1	5	8194
D9	1,240,704	578	1,035,173	16.6	3.3	15,720	1,026,566	17.3	3.4	27,510	**983,521**	20.7	5	8730
D10	1,149,234	427	989,109	13.9	3.0	16,440	985,583	14.2	3.5	27,660	**924,796**	19.5	4	8993
Avg.	1,253,177	841	1,017,269	18.8	3.2	16725	1,015,513	18.9	3.8	27,894	979,557	21.8	5	8944

Table 9: Comparison of solutions for all instances in Set E. The average values for Heuristic H2 are computed over the available numbers.

	Heuristic H1		Heuristic H2				Heuristic H3			
	TC	ST	TC	RI	OG	NR	TC	RI	NI	NR^*
E1	1,246,916	1006	1,004,541	19.4	12.5	56,070	**938,875**	24.7	5	11,425
E2	1,231,859	908	929,386	24.6	7.2	56,520	**915,591**	25.7	5	11,401
E3	1,287,454	1183	1,735,108	-34.8	49.4	58,260	**942,903**	26.8	5	10,983
E4	1,260,251	1231	971,264	22.9	11.0	59,130	**933,140**	26.0	4	13,036
E5	1,274,681	1045	n/a	n/a	n/a	61,380	**934,511**	26.7	5	12,608
E6	1,222,821	854	1,619,761	-32.5	46.2	57,210	**932,725**	23.7	5	11,525
E7	1,192,426	722	1,668,140	-39.9	48.6	60,660	**918,307**	23.0	4	12,378
E8	1,231,443	980	n/a	n/a	n/a	59,310	**939,735**	23.7	5	12,182
E9	1,198,309	986	n/a	n/a	n/a	58,200	**922,481**	23.0	4	12,260
E10	1,209,579	653	927,813	23.3	6.8	58,200	**917,877**	24.1	5	11,111
Avg.	1,235,574	957	1,265,145	-2.4	26.0	58,494	929,615	24.7	5	11,891

than the optimal total cost. Compared with Heuristic H1, Heuristics H2a, H2b, and H3 achieve significantly improved solutions. Heuristics H2a and H3 provide solutions of similar quality, on average within 2.5 % to optimality. As expected, Heuristic H2b outperforms Heuristic H2a since it considers additional routes; in fact, for all 10 instances, the best heuristic solutions are obtained with Heuristic H2b.

Unlike in Set A, the instances in Set B are not solved to optimality within the given time limit; however, near-optimal solutions are obtained, where the optimality gap is on average 1.3 %. Also the MPRPs used in Heuristics H2a and H2b are solved with nonzero optimality gaps; however, the obtained solutions are close to optimal, some even better than the ones obtained with the exact method (indicated by a negative RD). Here, one can observe that a solution obtained with Heuristic H2b may not be as good than the one obtained with Heuristic H2a because the MILPs are not solved to optimality. Heuristic H3 again achieves high-quality solutions, but does not perform as well as Heuristics H2a and H2b.

Solving the MPRP exactly becomes computationally intractable for instances in Sets C–E; hence, we only show results from the heuristic algorithms in Tables 7–9. Note that here we compare the results with the solutions obtained with Heuristic H1, and RI is defined such that the larger RI, the better the solution. With increasing problem size, the MILPs considered in Heuristic H2 become more difficult to solve, resulting in reduced solution quality. This effect is less pronounced in Heuristic H3 because of its dynamic route generation procedure that keeps the route set sufficiently small. As a result, in Set C, the best solutions to 7 of the 10 instances are obtained with Heuristic H3. In Sets D and E, Heuristic H3 consistently achieves the best solution.

From the results for Set E (see Table 9), one can see that the performance of Heuristic H2 deteriorates in these large instances. Due to the large number of candidate routes, solving the MILPs in Heuristic H2 becomes intractable. In three instances, the optimality gaps obtained after one hour are still close to 50 %; in three other instances, where no numerical results are reported (n/a), the solver was not able to find any feasible solutions within the time limit. Heuristic H3, however, still achieves good feasible solutions with significantly lower costs than the ones obtained by Heuristics H1 and H2.

In terms of solution quality, Heuristic H1 exhibits the worst performance among all solutions methods due to the inaccurate representation of the distribution constraints, which results in inefficient routing decisions and large additional product purchases in Phase 2. Heuristic H2 performs well in small instances, where one can afford generating a sufficiently large number of routes to obtain good solutions; however, the performance deteriorates in larger instances. In contrast, the proposed solution method, Heuristic H3, consistently obtains high-quality solutions in a few iterations and significantly outperforms the other solution methods in the larger instances.

5.3. Industrial case study

We now apply the proposed MPRP framework to a real-world industrial test case provided by Praxair. Here, we consider an industrial gas business that produces and sells liquid oxygen (LO2), liquid nitrogen (LN2), gaseous oxygen (GO2), and gaseous nitrogen (GN2). While LO2 and LN2 can be stored and transported to customer sites using tractor-trailers, GO2 and GN2 are nonstorable and have to be distributed via pipelines immediately after their production; hence, routing decisions only involve liquid product customers. We consider a supply chain consisting of 2 plants, P1 and P2, and approximately 240 customers. The two plants have a combined fleet of 10 LO2 and 10 LN2 tractor-trailers. While Plant P1 has to satisfy demand for both liquid and gaseous products, Plant P2 only serves liquid product customers.

The production process, namely cryogenic air separation, is highly power-intensive such that the vast majority of the variable production cost is the cost of electricity. Electricity prices can vary significantly across different locations. In this case, Plant P1 participates in the day-ahead market in which the price varies over time, whereas Plant P2 purchases power at a constant unit price. A forecast of the day-ahead prices is available for the given planning horizon.

The MPRP is solved for a planning horizon of 4 weeks, where we choose Δt^{f} and Δt^{c} to be 4 h and 12 h, respectively, resulting in 168 level-1 and 56 level-2 time periods. We apply the proposed algorithm to this large-scale MPRP and present the solution obtained after one hour runtime. Note that due to confidentiality reasons, we cannot disclose detailed information about the supply chain network, plant specifications, and actual product demands. Therefore, all results are given as dimensionless quantities, and numerical values are normalized if necessary.

Figure 9 shows the electricity consumption and price profiles for both plants over the entire planning horizon. One can see that the electricity price at Plant P2 is significantly higher than the average electricity price at Plant P1. As a result, in order to reduce energy cost, Plant P2 is shut down three times for extensive periods of time and also at the end of the planning horizon. One can further see that the solution suggests load shifting at Plant P1 in order to take advantage of low-price hours.

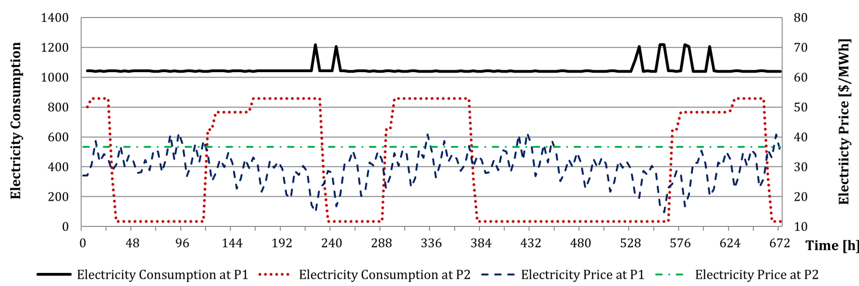


Figure 9: Electricity consumption and electricity price profiles for each plant.

There is a trade-off between production and distribution costs that is not apparent from Figure 9. Although the electricity price is almost always lower at Plant P1, it does not utilize its full production capacity, i.e. more production could be shifted from Plant P2 to Plant P1. However, the higher production cost is offset by the reduction in distribution cost because more customers are located closer to Plant P2 than to Plant P1.

Figures 10 and 11 show the product flows and inventory profiles for the liquid products at Plants P1 and P2, respectively. In Figure 10, one can clearly see the effect of load shifting at Plant P1. At Plant P2, inventory is accumulated during hours of production such that products can be drawn from the inventory and distributed to the customers when the plant is shut down, as depicted in Figure 11.

Now we compare our solution with the ones obtained from two alternative solution methods. The first method applies a similar approach as Heuristic H1, however, with more sophisticated and tailored constraints on the distribution resources. In the following, we refer to this approach as Heuristic PH1. The second approach is an extension of Heuristic PH1, referred to as Heuristic PH2, which further incorporates fixed costs for customer visits. The fixed distribution

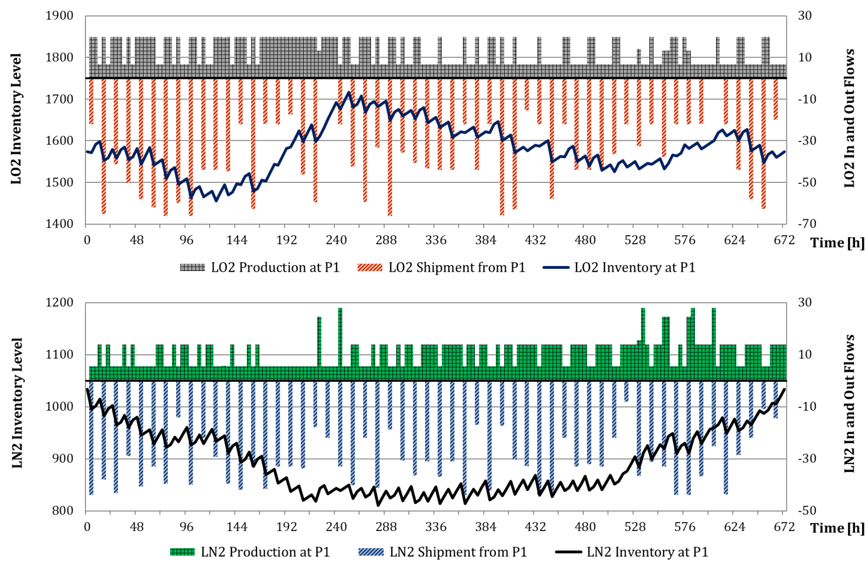


Figure 10: Production quantities, shipments, and inventory levels of LO2 and LN2 at Plant P1.

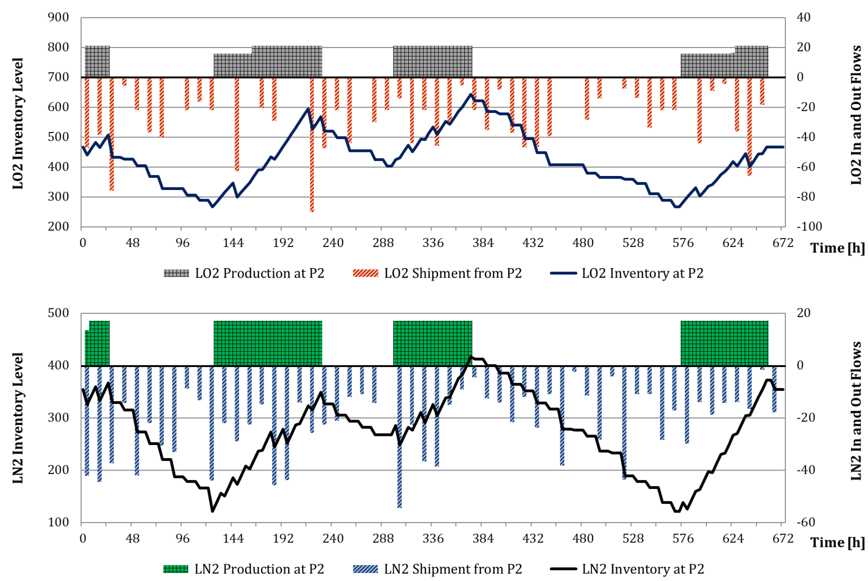


Figure 11: Production quantities, shipments, and inventory levels of LO2 and LN2 at Plant P2.

costs in Heuristic PH2 prevent the model from suggesting a large number of deliveries with small quantities; however, they also introduce additional binary variables that considerably increase the computational complexity.

Heuristics PH1, PH2, and H3, with the latter being our proposed algorithm with dynamic route generation, apply equivalent representations of the production side; however, the distribution side is modeled with different levels of accuracy. For this comparative study, we first apply Heuristics PH1, PH2, and H3 to obtain the production plan and the plant-to-customer allocation decisions for each of the three solution approaches. Then, the same routing tool is applied to the three sets of plant-to-customer allocation decisions to determine optimal (or near-optimal) routes and accurate routing costs.

Table 10 compares the solutions obtained from Heuristics PH1, PH2, and H3. The table shows the breakdown of the total costs (TC) into the production costs (CPD) and distribution costs (CDI) for each plant. In this test case, no additional product purchase is required, and inventory costs are negligible; hence, these costs are omitted. Furthermore, the table shows the computation time for each solution method. In terms of total cost, Heuristic H3 outperforms both Heuristics PH1 and PH2, with relative cost savings of 8.7 and 2.4 %, respectively, which can be attributed to the more rigorous modeling of routing decisions. One can see that compared to Heuristics PH1 and PH2, Heuristic H3 suggests producing less at Plant P1 and more at Plant P2. This production plan results in higher total production cost, but in overall proves to be the better choice since the routing cost can be significantly reduced by distributing more from Plant P2.

Table 10: Comparison of costs and solution times for the industrial test case.

	Heuristic PH1	Heuristic PH2	Heuristic H3
TC	100.00	93.46	91.26
CPD_{P1}	32.67	32.66	31.88
CPD_{P2}	13.05	13.12	15.01
CDI_{P1}	42.53	36.61	32.40
CDI_{P2}	11.75	11.07	11.97
ST [s]	218	900	3600

Figure 12 shows for each day of the planning horizon the number of customers to visit as suggested by each of the three solutions. While Heuristic PH1 proposes to visit on average 66 customers per day, the average numbers of visited customers per day are 30 and 25 for Heuristics PH2 and H3, respectively. Heuristic PH1 creates many deliveries with small quantities, which leads to inefficient routes. This effect is mitigated in Heuristic PH2 by introducing fixed distribution costs, ultimately resulting in lower routing costs. However, the improved solution quality comes at the cost of higher computational expense. While Heuristic PH1 solves in 218 s, the solution from Heuristic PH2 is obtained after 900 s. Among the three solution approaches, Heuristic H3 obtains the best solution, but only after 3600 s.

Under normal circumstances, the plant-to-customer allocation is fixed, i.e.

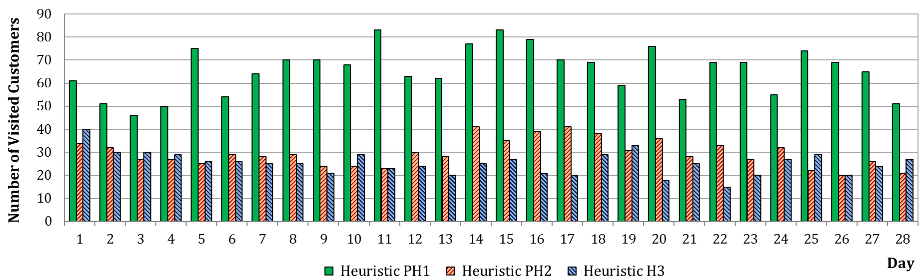


Figure 12: Comparison of the numbers of customers to be visited on each day of the planning horizon as suggested by Heuristics PH1, PH2, and H3.

each customer is assigned to a particular plant and only receives delivery from this plant, which may limit the flexibility in the supply chain operations. To compare the differences between the proposed solutions and the current practice, we show in Figure 13 for each of the three solutions the changes in plant-to-customer allocation compared to the current plant-to-customer allocation. Here, an allocation change is defined as one customer that is to be visited in the corresponding solution from a plant different from the one to which it is currently assigned. The number of allocation changes can be interpreted as a measure for the amount of disruption in the default assignment required to obtain the suggested solution. In practice, small changes are desired; a large number of allocation changes may suggest that the current plant-to-customer allocation or the current assignment of vehicles to plants is inadequate. In this case, significantly fewer allocation changes, on average 7 per day, are required for Heuristic H3 than for Heuristics PH1 and PH2, which require on average 24 and 11 allocation changes per day, respectively.

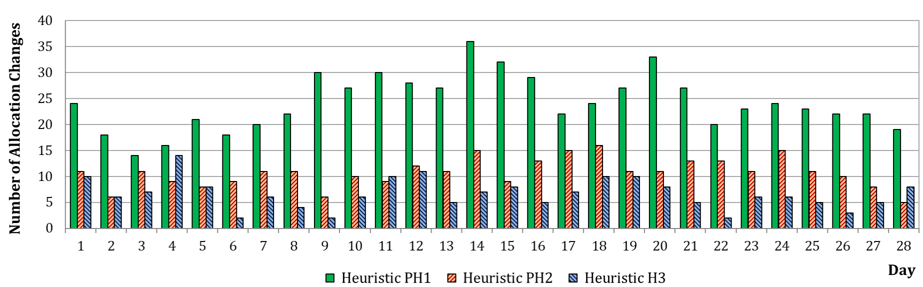


Figure 13: Comparison of the numbers of plant-to-customer allocation changes from the current assignment required for Heuristics PH1, PH2, and H3.

Another advantage of Heuristic H3 is that it only considers feasible routes; hence, the proposed deliveries are guaranteed to be feasible. In contrast, Heuristics PH1 and PH2 may make plant-to-customer allocation decisions that are infeasible in the subsequent routing step, in the sense that not all proposed deliveries can actually be made. In this particular test case, routing infeasibility does not occur because the customers are located relatively close to each other such that the limit on the travel distance is not an issue. However, in other supply chain networks with longer inter-customer distances, the situation of routing infeasibility may very well arise when Heuristics PH1 and PH2 are applied.

6. Conclusions

In this work, we have introduced the multiscale production routing problem, which considers the integrated optimization of production, inventory, distribution, and routing decisions in multicommodity supply chains with complex production facilities. In the MPRP, the objective is to make decisions at two different levels: operational scheduling decisions on the production side and tactical plant-to-customer allocation decisions on the distribution side.

The proposed MILP model incorporates two different time scales. For production scheduling, a mode-based formulation is applied to the fine time grid such that all critical operational features, including interdependent production rates, limitations on transitions between operating points, and time-sensitive production costs, can be captured. In addition, for distribution planning, vehicle routing is considered in each time period of the coarse time grid. An iterative heuristic solution method has been developed in order to solve large instances of the MPRP. At each iteration of the proposed algorithm, a restricted MPRP considering a subset of all possible routes is solved, and the set of candidate routes is updated based on the solutions obtained in previous iterations.

The proposed MPRP framework has been applied to an illustrative example, in a computational study with 50 instances of various sizes, as well as to an industrial test case with real-world data provided by Praxair. In the computational study, where the largest instances consider supply chains with 3 products, 3 plants, and 300 customers, the proposed algorithm is compared with a standard two-phase heuristic and a solution strategy involving a one-time heuristic pre-generation of candidate routes. The results show that the proposed algorithm finds high-quality solutions in reasonable computation times and significantly outperforms the other two solution approaches in large instances. In the industrial case study, which considers a real-world industrial gas supply chain with 2 plants and approximately 240 customers and a planning horizon of 4 weeks, the proposed algorithm outperforms available alternative solution approaches in terms of solution quality, although longer computation times are required.

Nomenclature for (MPRP)

Indices

c customers

i	products
j	vertices
m, m', m''	operating modes
p	production plants
r	operating subregions
s	routes
t	time periods

Sets

C_i	customers requiring product i
$\overline{C}_{ips}$	customers that can be visited on route $s \in S_{ip}$
I	products
$\bar{I}$	storable products, $\bar{I} \subseteq I$
$\hat{I}$	nonstorable products, $\hat{I} \subseteq I$
J_{pmr}	vertices of the polytope describing subregion $r \in R_{pm}$
M_p	operating modes of plant p
P	production plants
R_{pm}	subregions of mode $m \in M_p$
SQ_p	predefined sequences of mode transitions at plant p
T^{c}	level-2 time periods, $T^{\mathrm{c}} = \{0, 1, \ldots, \hat{t}^{\mathrm{c}}\}$
T^{f}	level-1 time periods, $T^{\mathrm{f}} = \{-\theta^{\max} + 1, -\theta^{\max} + 2, \ldots, 0, 1, \ldots, \hat{t}^{\mathrm{f}}\}$
$\overline{T}^{\mathrm{c}}$	level-2 time periods in the planning horizon, $\overline{T}^{\mathrm{c}} = \{1, 2, \ldots, \hat{t}^{\mathrm{c}}\}$
$\overline{T}^{\mathrm{f}}$	level-1 time periods in the planning horizon, $\overline{T}^{\mathrm{f}} = \{1, 2, \ldots, \hat{t}^{\mathrm{f}}\}$
$\widetilde{T}$	level-1 time periods that begin at the same time points as the corresponding level-2 time periods, $\widetilde{T} = \{1, \Delta t^{\mathrm{c}}/\Delta t^{\mathrm{f}} + 1, 2\Delta t^{\mathrm{c}}/\Delta t^{\mathrm{f}} + 1, \ldots, (\hat{t}^{\mathrm{c}} - 1)\Delta t^{\mathrm{c}}/\Delta t^{\mathrm{f}} + 1\}$
TR_p	possible mode transitions at plant p
$\overline{TR}_{pm}$	modes from which mode $m \in M_p$ can be directly reached
$\widehat{TR}_{pm}$	modes which can be directly reached from mode $m \in M_p$

Parameters

$\widehat{D}_{pit}$	demand for nonstorable product i at plant p in time period t [kg]
$\overline{D}_{ict}$	demand for storable product i at customer c in time period t [kg]
$\widehat{DL}_{ict}^{\max}$	maximum amount of product i that can be delivered to customer c in time period t [kg]
IV_{pi}^{ini}	initial inventory of product i at plant p [kg]
$IV_{pit}^{\min}$	minimum inventory of product i at plant p at time t [kg]
$IV_{pit}^{\max}$	maximum inventory of product i at plant p at time t [kg]
$\overline{IV}_{ic}^{\mathrm{ini}}$	initial inventory of product i at customer c [kg]
$\overline{IV}_{ict}^{\mathrm{low}}$	lowest possible inventory of product i at customer c at time t [kg]
$\overline{IV}_{ict}^{\min}$	minimum inventory of product i at customer c at time t [kg]

$\overline{IV}_{ict}^{\max}$	maximum inventory of product i at customer c at time t [kg]
L_{ipt}	number of vehicles that can transport product i from plant p in time period t
$\hat{t}^{\mathrm{c}}$	final level-2 time period in the planning horizon
$\hat{t}^{\mathrm{f}}$	final level-1 time period in the planning horizon
v_{pmrji}	amount of product i produced in one time period at vertex $j \in J_{pmr}$ [kg]
V_i	load capacity of a vehicle transporting product i [kg]
y_{pm}^{ini}	1 if plant p was operating in mode m in the time period before the start of the planning horizon
$z_{pmm't}^{\mathrm{ini}}$	1 if operation at plant p switched from mode m to mode m' at time t before the start of the planning horizon
α_{ict}	unit cost for purchasing product i for customer c in time period t [\$/kg]
β_{ips}	fixed distribution cost for using route $s \in S_{ip}$ [\$]
γ_{pmrit}	unit cost for producing product i in subregion $r \in R_{pmr}$ in time period t [\$/kg]
δ_{pmrt}	fixed cost for operating in subregion $r \in R_{pm}$ in time period t [\$]
$\overline{\Delta}_{pmi}^{\max}$	maximum rate of change in the amount of product i produced in mode $m \in M_p$ [kg]
Δt^{c}	length of each level-2 time period [h]
Δt^{f}	length of each level-1 time period [h]
$\theta_{pmm'}$	minimum stay time in mode m' after switching from mode m to m' at plant p [Δt^{f}]
$\bar{\theta}_{pmm'm''}$	fixed stay time in mode m' of the predefined sequence (m, m', m'') at plant p [Δt^{f}]
$\theta^{\max}$	maximum minimum or predefined stay time in a mode [Δt^{f}]
$\tilde{\theta}_p^{\max}$	maximum minimum or predefined stay time in a mode of plant p [Δt^{f}]
π_t	level-2 time period that begins at the same time point as level-1 time period t
ρ_{pit}	unit inventory cost for storing product i at plant p in time period t [\$/kg]
$\bar{\rho}_{ict}$	unit inventory cost for storing product i at customer c in time period t [\$/kg]

Continuous variables

$\overline{DL}_{ipts}$	amount of product i delivered on route $s \in S_{ip}$ in time period t [kg]
$\widehat{DL}_{iptsc}$	amount of product i delivered to customer c on route $s \in S_{ip}$ in time period t [kg]
DL_{ict}	amount of product i delivered to customer c in time period t [kg]
IV_{pit}	inventory of product i at plant i at time t [kg]
$\overline{IV}_{ict}$	inventory of product i at customer c at time t [kg]
LD_{ipt}	amount of product i delivered from plant p in time period t [kg]
PC_{ict}	amount of product i purchased for customer c in time period t [kg]
PD_{pit}	amount of product i produced at plant p in time period t [kg]
$\overline{PD}_{pmrit}$	amount of product i produced in subregion $r \in R_{mp}$ in time period t [kg]
TC	total cost [\$]
λ_{pmrjt}	coefficient for vertex j in subregion $r \in R_{pm}$ in time period t

Binary variables

x_{ipts}	1 if route $s \in \overline{S}_{ipt}$ is used to deliver product i
y_{pmt}	1 if plant p operates in mode m in time period t
$\bar{y}_{pmrt}$	1 if plant p operates in subregion $r \in R_{pm}$ in time period t

$z_{pmm't}$ 1 if operation at plant p switched from mode m to mode m' at time t

Acknowledgments

The financial support from the National Science Foundation under Grant No. 1159443 and from Praxair is gratefully acknowledged. Furthermore, the authors would like to thank Prof. Carlos A. Méndez from the National University of Litoral for early discussions on potential solution approaches.

Appendix A. DCVRP formulation

The routing problem in Phase 2 of Heuristic H1 can be solved for each product i, plant p, and time period $t \in \overline{T}^{\text{c}}$ independently. For each (i, p, t), we consider plant p and the customers in C_i that receive deliveries from plant p in time period t. We create a complete directed graph $G = (N, A)$ where $N = \{0, 1, \ldots, n^{\text{C}}\}$ represents the set of the plant (node 0) and the customers and $A = \{(n, n') : n \in N,\, n' \in N,\, n \neq n'\}$ is the set of arcs. We further define the set of customers $\widetilde{N} = N \setminus \{0\}$ and the set of arcs between customers $\widetilde{A} = \{(n, n') : n \in \widetilde{N},\, n' \in \widetilde{N},\, n \neq n'\}$. The fixed order from customer n, denoted by O_n, is set to DL_{ict} (for the corresponding c in the customer set), which is obtained in Phase 1. The time required for visiting node n' from node n is denoted by $\hat{\tau}_{nn'}$, which consists of the travel time from n to n' and the average stay time at one node. Similarly, the corresponding distribution cost, denoted by $\hat{\beta}_{nn'}$, includes the travel cost and the fixed cost for loading and unloading.

For each (i, p, t), we solve the following MILP model of the DCVRP with product purchase:

$$\min \quad \sum_{(n,n') \in A} \hat{\beta}_{nn'}\, w_{nn'} + \sum_{n \in \widetilde{N}} \bar{\alpha}_n\, \overline{PC}_n \tag{A.1a}$$

$$\text{s.t.} \quad \sum_{n' \in N, n' \neq n} w_{nn'} = \sum_{n' \in N, n' \neq n} w_{n'n} \quad \forall\, n \in \widetilde{N} \tag{A.1b}$$

$$\sum_{n \in \widetilde{N}} w_{0,n} \leq L_{ip} \tag{A.1c}$$

$$u_n - u_{n'} + V_i\, w_{nn'} + (V_i - O_n - O_{n'})\, w_{n'n} \leq V_i - O_{n'} \quad \forall\, (n, n') \in \widetilde{A} \tag{A.1d}$$

$$O_n + \sum_{n' \in \widetilde{N}, n' \neq n} O_{n'}\, w_{n'n} \leq u_n \quad \forall\, n \in \widetilde{N} \tag{A.1e}$$

$$u_n \leq V_i - \sum_{n' \in \widetilde{N}, n' \neq n} O_{n'}\, w_{nn'} \quad \forall\, n \in \widetilde{N} \tag{A.1f}$$

$$\bar{u}_n - \bar{u}_{n'} + \bar{\tau}^{\max}\, x_{nn'} + (\bar{\tau}^{\max} - \hat{\tau}_{nn'} - \hat{\tau}_{n'n})\, w_{n'n} \leq \bar{\tau}^{\max} - \hat{\tau}_{nn'} \quad \forall\, (n, n') \in \widetilde{A} \tag{A.1g}$$

$$\hat{\tau}_{0,n} + \sum_{n' \in N, n' \neq n} (\hat{\tau}_{0,n'} + \hat{\tau}_{n'n} - \hat{\tau}_{0,n})\, w_{n'n} \leq \bar{u}_n \quad \forall\, n \in \widetilde{N} \tag{A.1h}$$

$$\bar{u}_n \leq \bar{\tau}^{\max} - \hat{\tau}_{n,0} - \sum_{n' \in N, n' \neq n} (\hat{\tau}_{n',0} + \hat{\tau}_{nn'} - \hat{\tau}_{n,0})\, w_{nn'} \quad \forall\, n \in \widetilde{N} \tag{A.1i}$$

$$O_n \sum_{n' \in N, n' \neq n} w_{nn'} + \overline{PC}_n = O_n \quad \forall\, n \in \widetilde{N} \tag{A.1j}$$

$$\overline{PC}_n \geq 0 \quad \forall\, n \in \widetilde{N} \tag{A.1k}$$

$$w_{nn'} \in \{0, 1\} \quad \forall\, (n, n') \in A \tag{A.1l}$$

where the binary variable $w_{nn'}$ equals 1 if a vehicle travels from node n to node n', and $\overline{PC}_n$ denotes the amount of product purchased for customer n. Eq. (A.1a) states the objective function, which consists of the distribution cost and the product purchase cost, with $\bar{\alpha}_n$ denoting the unit cost for purchasing product for customer n. Eq. (A.1b) represents the vehicle flow conservation constraints, while constraint (A.1c) limits the number vehicles according to the availability at plant p. We adopt the lifted formulation of the Miller-Tucker-Zemlin (MTZ) subtour elimination constraints proposed by Desrochers and Laporte (1991), which are stated in Eqs. (A.1d)–(A.1i). Eqs. (A.1d)–(A.1f) further restrict the amount of product delivered on each trip to the vehicle capacity V_i, while Eqs. (A.1g)–(A.1i) prohibit the selection of trips that take longer than $\bar{\tau}^{\max}$. Finally, according to Eq. (A.1j), the order quantity for each customer is either fully met by delivery or by purchase.

References

Absi, N., Archetti, C., Dauzère-Pérès, S., Feillet, D., 2013. A two-phase iterative heuristic approach for the production routing problem. Transp Sci, Forthcoming (October 2015), 1–26.

Adulyasak, Y., Cordeau, J.-F., Jans, R., 2014a. Formulations and Branch-and-Cut Algorithms for Multivehicle Production and Inventory Routing Problems. INFORMS J 26 (1), 103–120.

Adulyasak, Y., Cordeau, J.-F., Jans, R., 2014b. Optimization-Based Adaptive Large Neighborhood Search for the Production Routing Problem. Transportation Science 48 (1), 20–45.

Adulyasak, Y., Cordeau, J.-F., Jans, R., 2015a. Benders Decomposition for Production Routing Under Demand Uncertainty. Operations Research 63 (4), 851–867.

Adulyasak, Y., Cordeau, J. F., Jans, R., 2015b. The production routing problem: A review of formulations and solution algorithms. Computers and Operations Research 55, 141–152.

Archetti, C., Bertazzi, L., Paletta, G., Speranza, M. G., 2011. Analysis of the maximum level policy in a production-distribution system. Computers & Operations Research 38 (12), 1731–1746.

Armentano, V. a., Shiguemoto, a. L., Løkketangen, A., 2011. Tabu search with path relinking for an integrated production-distribution problem. Computers and Operations Research 38 (8), 1199–1209.

Balinski, M. L., Quandt, R. E., 1964. On an Integer Program for a Delivery Problem. Operations Research 12 (2), 300–304.

Bard, J. F., Nananukul, N., 2009a. Heuristics for a multiperiod inventory routing problem with production decisions. Computers & Industrial Engineering 57 (3), 713–723.

Bard, J. F., Nananukul, N., 2009b. The integrated productioninventorydistributionrouting problem. Journal of Scheduling 12 (3), 257–280.

Bard, J. F., Nananukul, N., 2010. A branch-and-price algorithm for an integrated production and inventory routing problem. Computers & Operations Research 37 (12), 2202–2217.

Boudia, M., Louly, M., Prins, C., 2007. A reactive GRASP and path relinking for a combined productiondistribution problem. Computers & Operations Research 34 (11), 3402–3419.

Boudia, M., Prins, C., 2009. A memetic algorithm with dynamic population management for an integrated productiondistribution problem. European Journal of Operational Research 195 (3), 703–715.

Brown, G., Keegan, J., Vigus, B., Wood, K., 2001. The Kellogg Company optimizes production, inventory, and distribution. Interfaces 31 (6), 1–15.

Campbell, A., Clarke, L., Kleywegt, A., Savelsbergh, M., 1998. The Inventory Routing Problem. Fleet Management and Logistics 46 (Ddm), 95–113.

Çetinkaya, S., Üster, H., Easwaran, G., Keskin, B. B., 2009. An integrated outbound logistics model for frito-lay: Coordinating aggregate-level production and distribution decisions. Interfaces 39 (5), 460–475.

Chandra, P., Fisher, M. L., 1994. Coordination of production and distribution planning. European Journal of Operational Research 72 (3), 503–517.

Coelho, L. C., Cordeau, J.-F., Laporte, G., 2013. Thirty Years of Inventory Routing. Transportation Science 48 (1), 1–19.

Desrochers, M., Laporte, G., 1991. Improvements and extensions to the Miller-Tucker-Zemlin subtour elimination constraints. Operations Research Letters 10 (1), 27–36.

Dong, Y., Pinto, J. M., Sundaramoorthy, A., Maravelias, C. T., 2014. MIP Model for Inventory Routing in Industrial Gases Supply Chain. Industrial & Engineering Chemistry Research 53 (44), 17214–17225.

Erengüç, S., Simpson, N., Vakharia, A. J., 1999. Integrated production/distribution planning in supply chains: An invited review. European Journal of Operational Research 115 (2), 219–236.

Fumero, F., Vercellis, C., 1999. Synchronized Development of Production, Inventory, and Distribution Schedules. Transportation Science 33 (3), 330–340.

GAMS Development Corporation, 2015. GAMS version 24.4.6.

Glankwamdee, W., Linderoth, J., Shen, J., Connard, P., Hutton, J., 2008. Combining optimization and simulation for strategic and operational industrial gas production and distribution. Computers & Chemical Engineering 32 (11), 2536–2546.

IBM ILOG, 2015. CPLEX version 12.6.2.

Karimi, B., Fatemi Ghomi, S., Wilson, J., 2003. The capacitated lot sizing problem: a review of models and algorithms. Omega 31 (5), 365–378.

Laporte, G., 2009. Fifty Years of Vehicle Routing. Transportation Science 43 (4), 408–416.

Lei, L., Liu, S., Ruszczynski, A., Park, S., 2006. On the integrated production, inventory, and distribution routing problem. IIE Transactions 38 (11), 955–970.

Marchetti, P. a., Gupta, V., Grossmann, I. E., Cook, L., Valton, P.-M., Singh, T., Li, T., André, J., 2014. Simultaneous Production and Distribution of Industrial Gas Supply-Chains. Computers & Chemical Engineering 69, 39–58.

Mitra, S., Grossmann, I. E., Pinto, J. M., Arora, N., 2012. Optimal production planning under time-sensitive electricity prices for continuous power-intensive processes. Computers & Chemical Engineering 38, 171–184.

Pochet, Y., Wolsey, L. A., 2006. Production Planning by Mixed Integer Programming. Springer Science+Business Media.

Ruokokoski, M., Solyali, O., Cordeau, J., Jans, R., Süral, H., 2010. Efficient Formulations and a Branch-and-Cut Algorithm for a Production-Routing Problem. Tech. rep., HEC Montreal.

Sung, C., Maravelias, C. T., 2009. A Projection-Based Method for Production Planning of Multiproduct Facilities. AIChE Journal 55 (10), 2614–2630.

Thomas, D. J., Griffin, P. M., 1996. Coordinated supply chain management. European Journal of Operational Research 94 (1), 1–15.

Toth, P., Vigo, D. (Eds.), 2002. The Vehicle Routing Problem. Society for Industrial and Applied Mathematics.

Zamarripa, M., Marchetti, P. A., Grossmann, I. E., Singh, T., Lotero, I., Gopalakrishnan, A., Besancon, B., André, J., 2016. Rolling Horizon Approach for ProductionDistribution Coordination of Industrial Gases Supply Chains. Industrial & Engineering Chemistry Research, acs.iecr.6b00271.

Zhang, Q., Grossmann, I. E., Sundaramoorthy, A., Pinto, J. M., 2015. Data-driven construction of Convex Region Surrogate models. Optimization and Engineering.

Zhang, Q., Sundaramoorthy, A., Grossmann, I. E., Pinto, J. M., 2016. A discrete-time scheduling model for continuous power-intensive process networks with various power contracts. Computers & Chemical Engineering 84, 382–393.