Improving Multi-actor Production, Inventory and Transportation Planning through Agent-Based Optimization

Johan Holmgren[1], Jan A. Persson[1], and Paul Davidsson[1,2]

[1] School of Computing, Blekinge Institute of Technology, SE-374 24 Karlshamn, Sweden
[2] School of Technology, Malmö University, SE-205 06 Malmö, Sweden
{johan.holmgren,jan.persson,paul.davidsson}@bth.se

Abstract. We present an agent-based optimization approach that is built upon the principles of Dantzig-Wolfe column generation, which is a classic reformulation technique. We show how the approach can be used to optimize production, inventory, and transportation, which may result in improved planning for the involved supply chain actors. An important advantage is the possibility to keep information locally when possible, while still enabling global optimization of supply chain activities. In particular, the approach can be used as strategic decision support to show how the involved actors may benefit from applying Vendor Managed Inventory (VMI). In a case study, the approach has been applied to a real-world integrated production, inventory and routing problem, and the results from our experiments indicate that an increased number of VMI customers may give a significant reduction of the total cost in the system. Moreover, we analyze the communication overhead that is caused by using an agent-based, rather than a traditional (non agent-based) approach to decomposition, and some advantages and disadvantages are discussed.

1 Introduction

Supply chain management is an area in which actors may experience great potentials by the use of efficient e-business solutions [1]. The introduction of powerful computers and efficient methods for formulating and solving complex optimization problems has made it possible to improve the operations in supply chains. In traditional central approaches for solving optimization problems, all information that need to be used when formulating and solving a problem has to be shared with a central node of computation. However, if multiple organizations are involved there is often a wish to keep sensitive information local. We propose an agent-based approach for integrated optimization of production, inventory and transportation, which has the potential to offer increased confidentiality for the involved organizations.

In this paper we describe how Dantzig-Wolfe (DW) decomposition [2], which is a classic reformulation technique, can be incorporated in a multi-agent system. The main purpose is to give a detailed account to how a classic optimization approach can be "agentified". Another purpose is to validate that some positive characteristics can be achieved by using this type of approach. The main characteristic is confidentiality of information, which is of particular importance in applications where different, potentially

A. Håkansson & R. Hartung (Eds.): *Agent & Multi-Agent Syst. in Distrib. Syst.*, SCI 462, pp. 1–31.
DOI: 10.1007/978-3-642-35208-9_1

competing, organizations are represented. Moreover, we investigate the possibility to achieve performance improvements by distribution and parallelization of the approach.

In a case study we apply our approach to a real-world integrated production, inventory and routing problem. The problem includes production planning, vehicle routing and inventory planning, and the objective is to minimize the total cost for production, transportation and inventory holding, while meeting the customers' demands for products. In particular, we show how the approach can be used for strategic decision making by quantifying the economic benefits that can be achieved by introducing *Vendor Managed Inventory* (VMI) for one or more supplier-customer relations. In VMI [3], the supplier is responsible for replenishing the customers' inventories, while given continuous access to information about forecasted customer demand and storage levels. Typically, the supplier owns the products until the customer removes them from inventory. The supplier benefits from being able to get updated information about forecasted customer demand and storage levels in contrast to being forced to deal with often late arriving, and changing customer orders. This provides flexibility in production and transportation planning. The customer potentially benefits from being able to pay later (and typically less) for products. Also, since it is the responsibility of the supplier to decide about deliveries, the customer does not have to bother about ordering products. For the studied problem we have performed simulation experiments, which indicate that an increased number of VMI customers may give a significant reduction of the total cost in the system. Moreover, we discuss how the approach can be used for operational planning, e.g., by providing decision support for real-world planners concerning how to perform supply chain activities.

In summary, we focus on the supply chain cooperation dimension of e-business by: (1) exploring an approach for integrated planning (involving multiple actors), which supports the keeping of planning related information confidential, and (2) from a strategic perspective, analyzing and quantifying the benefit of increased cooperation by VMI.

In next section we introduce the reader to the fields of agent-based optimization and decomposition, and in Section 3 we given an account to integrated planning of production, inventory and transportation, including some related work. In Section 4, we present a real-world case problem, and for the case problem, in Section 5 we describe an agent-based decomposition approach. Finally, in Section 6 we present some computational experiments before concluding the paper with a discussion on confidentiality in Section 7 and some conclusions and directions for future work Section 8.

2 Agent-Based Optimization and Decomposition

Agent-based approaches to optimization can be built in many ways, e.g., by using classical agent concepts, such as auctions and negotiation, as in the examples provided by Karageorgos et al. [4] and by Dorer and Calisti [5]. However, the focus here is on agent-based approaches that make use of techniques and concepts from classical optimization, e.g., methods for formulating and solving complex optimization problems.

It has been argued that the strengths and weaknesses of agent-based approaches and classical optimization techniques complement each other well for dynamic resource allocation problems [6]. The strengths and weaknesses of agent-based approaches and mathematical optimization techniques were compared for resource allocation in the domain of production and transportation. The comparison indicated that agent-based approaches tend to be preferable when: the size of the problem is large, communication and computational stability is low, the time scale of the domain is short, the domain is modular in nature, the structure of the domain changes frequently and there is sensitive information that should be kept locally, and classical optimization techniques when: the cost of communication is high, the domain is monolithic in nature, the quality of the solution is important, and it is desired that the quality of the solution can be guaranteed. Moreover, the comparison indicated that the properties of the two approaches are complementary and that it can be advantageous to combine them. In a case study, two hybrid approaches were tested:

1. Optimization was embedded in the agents to improve their abilities to take good decisions.
2. Optimization was used for creating long-term coarse plans, which were refined dynamically by the agents.

Another hybrid approach is referred to as *distributed constraint optimization*. According to Petcu [7], a *Constraint Optimization Problem* (COP) is defined as a set of variables with corresponding discrete and finite variable domains and a set of utility functions. Each utility function assigns a utility to each possible assignment of the variables, and the purpose is to find the variable instantiation that maximizes the sum of utilities of the utility functions. An interesting property is that variable domains are not restricted to numerical values, e.g., a variable domain may refer to colors or whatever is relevant for a particular problem. A *Distributed Constraint Optimization Problem* (DCOP) is defined as a set of agents, where each agent owns a centralized COP (i.e., a local subproblem), and a set of inter-agent utility functions, which are defined over variables from the local subproblems. An inter-agent utility function represents the award that is assigned to the involved agents when they take a joint decision.

Decomposition approaches, such as Dantzig-Wolfe column generation [2] has been developed for solving linear problems, and Lagrangean relaxation [8] and Benders' decomposition [9] have been developed for solving *Mixed Integer Linear optimization Problems* (MILPs). Moreover, decomposition approaches for linear problems have been used together with *branch-and-price* [10] to solve MILPs.

Combining agents with decomposition approaches is another, relatively new approach, which we find particularly interesting to investigate. An example is provided by Hirayama [11], who proposed an agent-based approach for solving the *Generalized Mutual Assignment Problem*. In the approach, which was built using a distributed solution protocol based on Lagrangean decomposition and distributed constraint satisfaction; agents were used to solve individual optimization problems, which were coordinated in order to improve a global solution.

In this paper the focus is on another decomposition approach, and we describe how Dantzig-Wolfe decomposition can be agentified into a multi agent system. In Dantzig-Wolfe decomposition, a linear *Master Problem* (MP) is reformulated into a *Restricted Master Problem* (RMP) containing only a subset of the variables in MP, and a set of subproblems which produce new solutions (columns) that are coordinated by RMP. In an iterative process, subproblem solutions based on so-called dual variables (which is a control mechanism) are added as improving variables to RMP. A dual variable is often interpreted as a value or price, e.g., for obtaining an additional unit of a scarce resource, or for producing one more unit of a particular product type.

In contrast to approaches to agent-based optimization, in which agents are able to communicate directly with each other in a peer-to-peer fashion, agent-based decomposition requires a coordinator agent who is responsible for managing the problem solving process. In DW decomposition, the coordinator agent corresponds to the RMP, and planner agents are responsible for providing plans (solutions) to the coordinator by solving subproblems. A conceptual illustration of agent-based DW decomposition is provided in Fig. 1, in which it can be seen that the coordinator sends dual variables to the planner agents, who return plans to the coordinator.

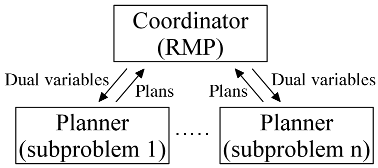

Fig. 1. A conceptual model of agent-based Dantzig-Wolfe decomposition. In an iterative process, the coordinator sends dual variables to the planner agents, who return improving plans to the coordinator.

For a case problem, in Section 5 we provide a detailed example of an agent-based decomposition approach for an integrated production, inventory, and routing problem. We have chosen a decomposition formulation that we find attractive, in particular since it allows for a natural interpretation of dual prices and subproblems. The studied problem class captures the difficulties with distributed decision-making since information and resources typically are distributed and the exact conditions, e.g., the demand and the availability of resources, are unknown in advance.

Planning tasks in supply chains are often performed by different organizations, which is why confidentiality is an important concept. Traditionally, a decomposition algorithm runs in a single process (on a single computer), which needs access to all information that is required for formulating and solving the optimization problem. In such an approach it is typically not possible to achieve confidentiality due to the fact that information, which might be considered sensitive for the planners may have to be shared. With an agent-based approach, where different problems are represented by different agents, it is often possible to run the optimization with less need for sharing sensitive information. In our case problem there is an agent who coordinates production, transportation and inventory. The coordinator agent needs access to customer demand forecasts, but it does not need to know any underlying details about how transportation plans and production plans are created. Hence, the agents responsible for solving subproblem do not need to share all information to the coordinator. Obviously, the use of agents may have a negative impact on the performance, i.e., concerning the execution time due to an increased need for communication. However, a possible advantage over classic decomposition approaches is that it is straightforward to distribute an agent system over several computers. In a distributed solution approach, there is a potential for reduced computation time and in some cases improved solution quality since more computing power allows for solving more complex subproblems. The reason is that the subproblems can be solved in parallel, which is impossible when the approach runs on a single processor. Further, a decentralized approach makes the system less vulnerable to single point failures. In the agent-based approach, where the coordinator agent typically retains control of all decisions, a failure of the coordinator agent is fatal. However, in case of a planner agent failure, the rest of the planners will still be able to produce new (improving) plans, which can be considered by the coordinator.

3 Integrated Planning of Production, Inventory and Transportation

Raw material supply, production and transportation have often been separated by large inventory buffers allowing different supply chain activities, such as production and transportation, to be planned separately. Various planning problems for different parts of the supply chain have been studied; a survey of lot sizing and scheduling problems is provided by Drexl and Kimms [12], and an overview of the vehicle routing problem and variations is given by Toth and Vigo [13]. The importance of inventory reduction has led to an increased interest in integrated planning of different logistical activities [14], and a review covering efforts in the area of integrated supply chain planning is provided by Sarmiento and Nagi [15].

Our focus is on optimization approaches that consider planning of production, inventory, as well as transportation. There is research focusing on only two of these aspects (e.g., [16,17,18]). According to our knowledge, the earliest contribution that combines all three problem aspects, i.e., planning of production, inventory and transportation, was presented by Chandra and Fisher [19]. For a multi-period planning horizon, the authors solved a combination of the production scheduling problem and the vehicle routing problem where multiple products were distributed from a single production facility to a number of customers.

A more recent approach was presented by Lei et al. [20], who considered an integrated production, inventory, and routing problem with a single product, multiple heterogeneous production plants, multiple customer demand centers and heterogeneous vehicles. Inventory management was considered both at the production plants and at the customer demand centers. The model was approached by formulating a large mixed-integer linear problem, which was solved using a 2-phase solution method. In phase one, the problem was reformulated to only include direct transportation between the producers and the customer demand centers. This restricted problem gives a feasible but non-optimal solution to the original problem. In phase two, a heuristic approach was used to improve the solution from phase one by also considering transport routes involving multiple customer demand centers. The main differences compared to their approach are that we model transportation in more detail, and that we use a different solution approach.

Another recent contribution is provided by Persson and Göete-Lundgren [21], who formulated and solved an optimization model for planning the production at a set of oil refineries and shipments of finished products while considering inventories both at producer and customer depots. Their problem is similar to the one we are considering. However, they used a longer planning horizon, but with longer individual time periods, there was no separation of the production planning into subproblems, and they used ships for transportation whereas we use trucks including explicit modeling of driving time restrictions. They proposed a solution approach based on column generation and valid inequalities, and integer solutions were obtained by using a fixing strategy in which vehicles were fixed to visit certain depots at specific times. To determine how much of each product should be delivered to each depot, an integer model was formulated and solved.

Bilgen and Ozkarahan [22] considered a problem for optimization of blending and shipment of grain (bulk) products from a set of producers to a set of customers. Whereas our approach is built on decomposition, Bilgen and Ozkarahan formulated a single MILP model with the objective to minimize the costs for blending, loading, transportation and inventory. The optimization problem was solved with the ILOG CPLEX solver for a rather short planning horizon.

4 Real World Case Problem Description

We consider a real-world case problem with a producer of vegetable oils and a single hauler that manages a fleet of bulk trucks that take care of deliveries of finished products to a set of customers. Production, and hence the planning of production and transportation is driven by the arrival of customer orders. Production is performed on multiple production lines, which are scheduled (individually) to match the shipping times that are chosen to match the delivery time windows of the customer orders. A typical horizon for production and transportation planning is usually less than a week.

Before loaded onto vehicles, finished products are stored in short-term inventory at the producer depot. The capacity of this short-term inventory is rather limited, and shipping is typically initiated the same day as the last production step is finished. At delivery, products are stored in customer inventories. Starting with a full (or close to full) truck load at the producer, sometimes a truck visits only one customer before returning but sometimes deliveries are grouped together and the truck visits multiple customers in a trip. Occasionally, a vehicle can be scheduled for a non-empty transport on the return trip from a customer to the producer. When the transportation demand exceeds the available transportation capacity, it is possible to call in extra capacity to a higher cost. Furthermore, time and costs for loading and unloading are considered, and the drivers have to follow the European Economic Community (EEC) regulations (EEC 3820/1985) for working and resting hours.

Raw material arrive to the production plant by boat or truck in quantities based on long term forecasts since the order-to-delivery lead time of raw material considerably exceeds the production planning horizon. Therefore, an unlimited supply of raw material can be assumed, and the considered operational production planning can be separated from ordering of raw material.

Currently, the considered real-world producer is considering an introduction of VMI for some of its customers. An introduction of VMI requires that accurate forecasts of consumption can be made available, which is also assumed to be possible. VMI might lead to a higher flexibility in the production and transportation planning, and the expectation is that a higher utilization of (often limited) production and transportation resources can be achieved.

In Fig. 2 we provide a conceptual illustration of the case problem with a small example of a transportation network containing a factory, inventories, customer depots and trucks, and planners who are responsible for taking decisions about physical resources.

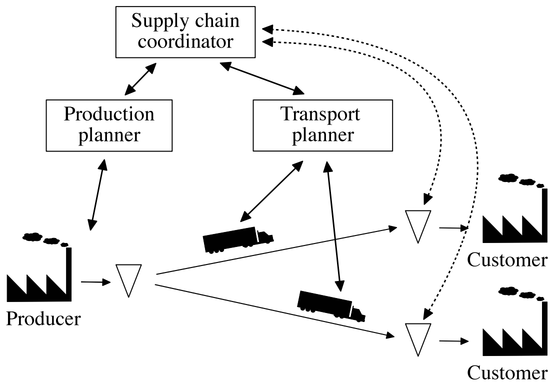

Fig. 2. An illustration of the case problem, with physical resources, planners and a supply chain coordinator. The dashed arrows represent connections that are present only for VMI customers.

5 Decomposition Formulation

The identified real world problem is modeled as a MILP, which is built using the principles of Dantzig-Wolfe decomposition. The decomposition formulation includes a linear DW master problem, a set of transportation subproblems and a set of production scheduling subproblems for construction of transportation plans and production plans. Even though MP is a linear problem, there are no restrictions regarding which types of variables the subproblems are allowed to model. It should be mentioned that DW decomposition is a technique for reformulating optimization problems, and in many cases it is necessary to apply a branch and bound method, i.e., branch-and-price [10], to be able to use it for solving problems with integer variables.

In addition to the practical issues, like for example confidentiality, reasons for not formulating the problem as a large MILP include: 1) a large MILP would most probably be difficult, or impossible, to solve to optimality due to its size and high complexity, and 2) a decomposition approach allows the subproblems to be reformulated without modifying the master problem.

It is possible to use alternative decomposition approaches for modeling the studied problem, e.g., Lagrangean relaxation [8] and Benders' decomposition [9]. In our opinion, the choice of which decomposition technique to use can often be seen as a matter of preference. Important reasons for using DW decomposition for modeling the studied problem is that it allows us to benefit from the special block structure that characterize the problem. According to Fumero and Vercellis [23], problems in the studied area tend to use an underlying network structure that may be exploited by decomposition approaches. The objective is to minimize the costs for production, distribution, and inventory holding, while satisfying the customers' product demand throughout a given planning horizon. The problem formulation includes production planning at producer depots, transportation planning including route choice, quantities to ship and times for deliveries to customers and pickups from the producers, and inventory planning of finished products. The presented approach focuses on the VMI situation due to its potential for improved resource utilization. However, it is not limited to VMI since a tight specification of customer inventory constraints mimics a non-VMI situation.

We have built the agent-based decomposition approach using a hierarchical agent model for decision making in a multi-agent-based supply chain simulation model called TAPAS, which was presented by Davidsson et al. [24]. The model contains a supply chain coordinator, a transport buyer, a product buyer, production planners, transport planners, and customers. However, in the suggested decomposition approach we have chosen not to model the product and transport buyer agents. The downscaled version of the hierarchical agent model in TAPAS corresponds to the agent model in Fig. 2. In the agent system, the supply chain coordinator agent represents MP, each production planner agent handles a set of production scheduling subproblems (one subproblem for each production line in one producer depot), and each transport planner agent handles a set of transportation subproblems (one subproblem for each vehicle in its vehicle fleet).

In the problem formulation, we let D^P denote the set of producer depots, D^C the set of customer depots, $D = D^P \cup D^C$ the set of all depots, V the set of (inhomogeneous) vehicles, P the set of product types, and L the set of production lines. The planning horizon is represented by an ordered set $T = \{1, 2, \ldots, \bar{t}\}$ of discrete time periods with uniform length τ. A *transportation plan* for a vehicle is defined as the amount of each product delivered to each customer depot and picked-up from each producer depot in each time period throughout the planning horizon. The set of all feasible transportation plans for a vehicle $v \in V$ is denoted R_v, and the cost for using plan $r \in R_v$ is denoted ψ_r. If d is a producer depot (i.e., $d \in D^p$) we let variable x_{dptr} denote the amount of product p that is picked-up from depot d for plan r by vehicle v in period t, otherwise ($d \in D^C$) x_{dptr} represents a delivery to depot d. Similarly, we define a *production plan* for a production line as the amount of each product that is produced in each time period throughout the planning horizon. We let S_l denote the set of all valid production plans for production line $l \in L$, and ω_s denotes the cost for using plan $s \in S_l$. For production plan $s \in S_l$, which represents a production line located at some depot $d = d(s)$, we let y_{dpts} represent the amount of product p produced in period t. Furthermore, parameter ϱ_{dpt} denotes the demand for product $p \in P$ at customer depot $d \in D^C$ in time period $t \in T$. Hence, the parameter ϱ_{dpt} specifies the amount of product p removed from the customer inventory in time period t. It should be noted that, in our methodological approach, only a subset of all the plans for the available resources will be generated and represented in the model.

Each depot $d \in D$ has an inventory level modeled by variable z_{dpt}. An inventory cost ϕ_{dp} is considered for each unit of product $p \in P$ in inventory at depot $d \in D$

between two subsequent periods. For a depot d, the inventory level of product p in time period t must not fall below a lower bound $\underline{z}_{dpt}$ (which typically corresponds to a safety inventory level) and must not exceed an upper bound (typically a maximum capacity) of $\overline{z}_{dpt}$ units. To allow violating the safety inventory and maximum allowed inventory levels, we let variable u_{dpt} represent how much the inventory level of product p falls below the safety inventory level at depot d in period t, q_{dpt} how much it exceeds the maximum allowed inventory level, and M^u_{dpt} and M^q_{dpt} corresponding penalty costs for violating the inventory constraints. It is assumed that, if any of the u:s or q:s are greater than zero in a period (i.e., an inventory constraint has been violated), a penalty is applied for resolving the inventory level infeasibility. There are other ways to model violation of inventory constraints, e.g., to include the u:s and q:s in constraint set (4) instead of in constraint sets (2) and (3). However, it is not obvious if one of these approaches is better than the other.

Binary decision variables are used to determine which transportation plans and production plans to use, and for obvious reasons exactly one transportation plan for each vehicle and exactly one production plan for each production line are allowed. Decision variable v_r determines if transportation plan $r \in R_v$ is used ($v_r = 1$) or not ($v_r = 0$), and w_s if production plan $s \in S_l$ is used ($w_s = 1$) or not ($w_s = 0$). We formulate our main problem (IMP) as the following MILP (MP is the LP-relaxation of IMP):

$$\min \quad \sum_{p\in P}\sum_{d\in D}\sum_{t\in T}\left(\phi_{dp} z_{dpt} + M^q_{dpt} q_{dpt} + M^u_{dpt} u_{dpt}\right) + \sum_{v\in V}\sum_{r\in R_v} \psi_r v_r + \sum_{l\in L}\sum_{s\in S_l} \omega_s w_s, \quad (1)$$

$$\text{s.t.} \quad z_{dpt-1} + \sum_{v\in V}\sum_{r\in R_v} v_r x_{dptr} + u_{dpt} - q_{dpt} - \varrho_{dpt} = z_{dpt}, \quad t \in T, d \in D^C, p \in P, \quad (2)$$

$$z_{dpt-1} - \sum_{v\in V}\sum_{r\in R_v} v_r x_{dptr} + \sum_{l\in L}\sum_{s\in S_l} w_s y_{dpts} + u_{dpt} - q_{dpt} = z_{dpt}, \quad t \in T, d \in D^P, p \in P, \quad (3)$$

$$\underline{z}_{dpt} \leq z_{dpt} \leq \overline{z}_{dpt}, \quad d \in D, p \in P, t \in T, \quad (4)$$

$$\sum_{r\in R_v} v_r = 1, \quad v \in V, \quad (5)$$

$$\sum_{s\in S_l} w_s = 1, \quad l \in L, \quad (6)$$

$$v_r \in \{0,1\}, \quad v \in V, r \in R_v, \quad (7)$$

$$w_s \in \{0,1\}, \quad l \in L, s \in S_l, \quad (8)$$

$$z_{dpt}, u_{dpt}, q_{dpt} \in \mathbb{R}^+, \quad d \in D, p \in P, t \in T. \quad (9)$$

In IMP, the first component (i.e., the triple sum) of the objective function (1) models the inventory costs with penalties for violating the inventory constraints. The second component of (1) represents the cost for the transportation plans, and the third component the cost for production plans. Constraint sets (2) and (3) express the customer and producer depot inventory balances. For a customer depot, an inventory level at the end of a period equals the inventory level at the end of the previous period, plus the deliveries minus the consumption in the current period. For a producer depot, an inventory level at the end of a period equals the inventory level in the previous period, minus the pickups plus the produced amount in the current period. Constraints (4) assure that the inventory levels are kept between the minimum and maximum allowed levels, and (5) and (6) that exactly one plan is used for each resource (vehicle or production line).

For MP (as well as for RMP, which will be introduced below), we let λ denote the dual variables for constraints (2), μ the dual variables for constraints (3), δ the dual variables for constraints (5), and θ the dual variables for constraints (6). A dual variable expresses the change of the optimal objective function value per unit increase of the right hand side of the corresponding constraint. For example, $\bar{\lambda}_{dpt} > 0$ represents the value of having one extra unit of product p in depot d in period t, and a positive dual variable $\bar{\mu}_{dpt} > 0$ the value of having one extra unit of product p at depot d in period t. A value $\bar{\mu}_{dpt} < 0$ on the other hand, means that we want to decrease the production or reduce the inventory by transporting products away from depot d.

As mentioned above, to solve the IMP we relax the integer constraints and get an MP, i.e., a master problem. Furthermore, we only consider a subset of the potentially huge number of production and transportation plans, which gives an RMP, i.e., a restricted master problem. At initiation, RMP contains only a small number of plans for each resource (vehicle or production line), typically the procedure starts with an empty plan for each resource. For instance, for a vehicle an empty plan is one that has no deliveries and pickups. It should be emphasized that, to be able to satisfy constraints (5) and (6), RMP needs to contain at least one plan for each resource. We let $R'_v \subseteq R_v$ denote a set of all currently known transportation plans for vehicle $v \in V$, and $S'_l \subseteq S_l$ a set of currently known production plans for production line $l \in L$. That is, RMP is identical to MP, except for that we replace all occurrences of R_v with R'_v, all occurrences of S_l with S'_l.

In the solution approach, RMP is iteratively updated with new improving production and transportation plans that are generated by the planner agents based on the current optimal values of the λ, μ, δ, and θ dual variables. Dual variables are obtained from the current optimal solution (to RMP) when using a standard solver, which is typically based on the simplex method. A flow chart describing the main algorithm used by the coordinator agent is presented in Fig. 3, and a diagram showing how the coordinator agent communicates with the planners is provided in Fig. 4. In a distributed approach it is possible, and in many cases preferable, to send the transportation and production requests in parallel to allow the planners to solve subproblems simultaneously. Plans with negative reduced cost are added as improving columns/variables to RMP, and when no improving plan can be generated, the optimal solution to MP is obtained. Since the optimal solution to MP typically contains fractional combinations of plans, it is typically infeasible (due to the integer properties) in the original problem IMP. One approach for finding an optimal, or at least a heuristically "good" integer solution to IMP, some delivery/pickup (depot, period, product, and vehicle) or production (period, product and production line) must be restricted (fixed) to some integer quantity whenever a fractional optimal solution to RMP is obtained. To handle the integer characteristics of the studied problem, we apply branch-and-price [10], however in a limited form. To guarantee that the optimal solution to IMP will be generated, it is necessary to use a search tree, in which each node represents a restricted version of the original MP due to additional restrictions. In the branching approach, all possible combinations of fixings are represented somewhere in the search tree, and in theory the optimal solution of the original problem is guaranteed. However, since time and memory are limited, this is not possible for the studied problem. Due to its characteristics of being NP-hard and the size of real-world scenarios, we choose to explore only one branch in the search tree. Variable restrictions are determined by the coordinator and communicated to the planners to prevent them from generating future plans that violate variable fixings. Throughout the procedure of the algorithm, more and more variables are fixed, and eventually the algorithm terminates with an integer solution to IMP. In Section 5.1 we give a more detailed account to our strategy for variable fixings and termination.

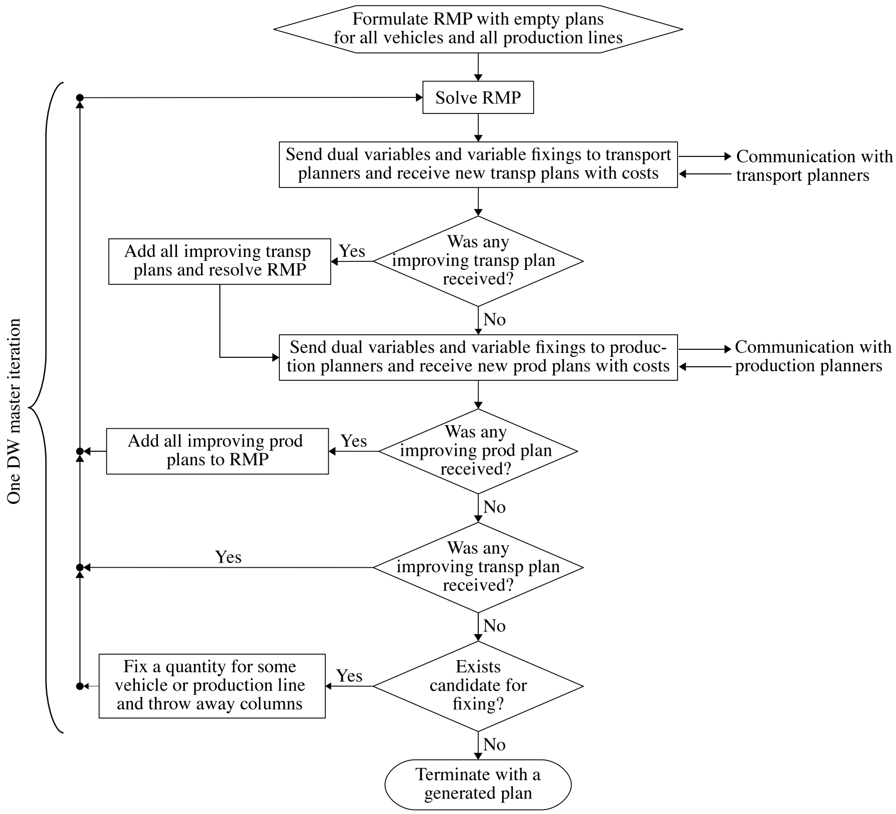

Fig. 3. A flow chart of the main algorithm used by the supply chain coordinator agent

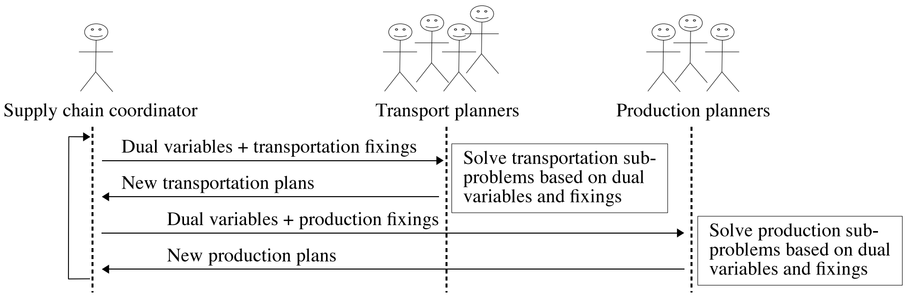

Fig. 4. Diagram describing the communication between the coordinator and planner agents

5.1 Heuristic Strategy for Variable Fixing and Termination

In our variable fixing strategy, which is inspired by the approach by Persson and Göete-Lundgren [21], we let $f^P_{ptl} \in \mathbb{Z}^+, p \in P, t \in T, l \in L$ represent production fixings (i.e., minimum quantities to produce), and $f^T_{dptv} \in \mathbb{Z}^+, d \in D, p \in P, t \in T, v \in V$ transportation fixings (i.e., minimum quantities to deliver and pickup). That is, there is one (production) fixing parameter for each combination of product, period and production line, and one (transportation) fixing parameter for each combination of depot, product, period and vehicle. All f^P:s and f^T:s are initialized to 0, meaning that no quantities are fixed when the algorithm starts. A fixing of a quantity f^P_{ptl} means that all subsequent production plans that are generated for production line l must contain a production of at least f^P_{ptl} units of product p in period t. Similarly, a transportation fixing f^T_{dptv} for vehicle v means that all subsequent plans for v must contain a pickup (if $d \in D^P$) or delivery (if $d \in D^C$) of at least f^T_{dptv} units of product p for depot d in period t. Hence, it follows that already fixed productions, pickups and deliveries may be re-fixed to higher values later in the process, as long as the capacities of vehicles and production lines are met when new plans are created. After a fixing has been determined, all columns (plans) violating the fixing (i.e., infeasible columns) are removed from RMP.

The main idea in the variable fixing strategy is that the production, pickup or delivery with the highest representation in the optimal solution to RMP should be chosen for fixing. For instance, to calculate how much a production is represented in RMP, the values representing how much the plans are used in the solution are added together for those plans (for the particular production line) that have a production strictly greater than zero units for the particular product and period. The calculation of how much a pickup or delivery is represented in RMP is made in the same way. Hence, a number between 0 and 1 is obtained for each production, pickup and delivery, and a higher number means a higher representation in RMP.

For transportation, we find out which pickup or delivery (depot, product, period and vehicle) that is most represented in RMP by calculating

$$(d', p', t', v') = \underset{d \in D, p \in P, t \in T, v \in V}{\arg\max} \sum_{r \in R'_v} v_r \Lambda^T_{dptr}, \tag{10}$$

where

$$\Lambda^T_{dptr} = \begin{cases} 0 & \text{if } x_{dptr} = 0 \\ 1 & \text{if } x_{dptr} \neq 0. \end{cases} \tag{11}$$

To avoid fixing pickups and deliveries that are represented in all columns in RMP, and to disregard those pickups and deliveries that are not represented at all, in equation (10) we require that

$$0 < \sum_{r \in R'_{v'}} v_r \Lambda^T_{d'p't'r} < 1. \tag{12}$$

Since it is only relevant to consider pickups and deliveries that are represented by a strictly higher quantity than previously fixed, we also require that

$$\left| \sum_{r \in R'_{v'}} v_r x_{d'p't'r} \right| > f^T_{d'p't'v'}. \tag{13}$$

In the same way as for transportation, we find out which production (production line, product and period) is most represented in RMP by calculating

$$(l', p', t') = \underset{l \in L, p \in P, t \in T}{\arg\max} \sum_{s \in S'_l} w_s \Lambda^P_{d(s)pts} \tag{14}$$

where

$$\Lambda^P_{dpts} = \begin{cases} 0 & \text{if } y_{dpts} = 0 \\ 1 & \text{if } y_{dpts} \neq 0, \end{cases} \tag{15}$$

and where it is required in equation (14) that

$$0 < \sum_{s \in S'_l} w_s \Lambda^P_{d(s)p't's} < 1, \tag{16}$$

and

$$\sum_{s \in S'_{l'}} w_s y_{d(s)p't's} > f^P_{l'p't'}, \tag{17}$$

If only (d', p', t', v') exists (i.e., no candidate for production fixing exists), or if

$$\sum_{r \in R'_{v'}} v_r \Lambda^T_{d'p't'r} \geq \sum_{s \in S'_{l'}} w_s \Lambda^P_{d(s)p't's}, \tag{18}$$

then $f^T_{d'p't'v'}$ (i.e., a transportation fixing) is fixed (or re-fixed) for vehicle v' according to

$$f^T_{d'p't'v'} = \left\lceil \left| \sum_{r \in R'_{v'}} v_r x_{d'p't'r} \right| \right\rceil. \tag{19}$$

Otherwise, if only (l', p', t') exists, or if

$$\sum_{r \in R'_{v'}} v_r \Lambda^T_{d'p't'r} < \sum_{s \in S'_{l'}} w_s \Lambda^P_{d(s)p't's}, \tag{20}$$

then $f^P_{l'p't'}$ (i.e., a production fixing) will be fixed (or re-fixed) for production line l' according to

$$f^P_{l'p't'} = \left\lceil \sum_{s \in S'_{l'}} w_s y_{d(s)p't's} \right\rceil . \tag{21}$$

If neither a pickup or delivery pickup fixing candidate (d', p', t', v'), or a production fixing candidate (l', p', t') can be found, there might still exist any two columns $r'', r''' \in R'_{v'}$ or $s'', s''' \in S'_{l'}$ (for a vehicle $v' \in V$ or a production line $l' \in L$) with different coefficients in the optimal solution to RMP. This scenario can occur if equation (12) or (16) equals one for some delivery, pickup or production while $x_{d'p't'r''} \neq x_{d'p't'r'''}$ or $y_{d'p't's''} \neq y_{d'p't's'''}$ for any $d' \in D$, $p' \in P$ and $t' \in T$. Then we either set

$$f^T_{d'p't'v'} = \left\lceil \left| \sum_{r \in R'_{v'}} v_r x_{d'p't'r} \right| \right\rceil \tag{22}$$

or

$$f^P_{l'p't'} = \left\lceil \sum_{s \in S'_{l'}} w_s y_{d(s)p't's} \right\rceil , \tag{23}$$

to be able to converge towards an integer solution to IMP. Moreover, we remove from RMP all columns $r \in R'_{v'}$ with parameter

$$|x_{d'p't'r}| \neq f^T_{d'p't'v'} \tag{24}$$

if a delivery or a pickup was fixed, or all columns $s \in S'_{l'}$ with parameter

$$y_{d(s)p't's} \neq f^P_{l'p't'} \tag{25}$$

if a production was fixed.

From a few small-scale experiments, we realized that the convergence rate of the column generation approach was too slow to allow the RMPs to be solved to optimality before considering termination or variable fixing. Instead we use a heuristic termination strategy that is based on the relative improvement of generated plans, resulting in heuristic solutions to the RMPs (and hence MPs). The idea is that it still will be possible to find heuristically good integer solutions to IMP at termination of the algorithm.

A solution to RMP is considered to be good enough when: for each resource, the average reduced cost of the e (actually e^V for a vehicle or e^P for a production line) most recently added plans (for the particular resource) is less than g (g^V or g^P) percent better than the average reduced cost of the e (e^V or e^P) plans that were added immediately before that. That is, plans are added until the improvement rate has decreased to a certain level. At this point the algorithm either determines a variable to fix if the current solution of RMP is fractional, or terminates if the solution is integer. The choice of e^V, e^P, g^V, and g^P is typically a trade-off between solution time and quality.

5.2 Transportation Subproblems

We formulate a transportation subproblem for a vehicle $v \in V$ using a hierarchical approach with two separated subproblems; one routing problem and one product assignment problem. The routing problem is formulated as a shortest path problem with additional constraints for representing pickup and delivery fixings. The output is a sequence of depots with corresponding time-periods, which will be visited by the vehicle throughout the planning horizon. In other words, the output is a route serving as input to the product assignment problem. Then the product assignment problem decides how much of each product should be delivered for each customer depot visit in the route.

For the studied problem it was found reasonable to allow a maximum of two customer depot visits before visiting a producer. Formally this assumption can be described by introducing a set of network layers, denoted $H = \{0, \ldots, \bar{h}\}$, as a means to prevent vehicles from visiting more than $\bar{h}$ customers before visiting some producer. In our problem formulation $\bar{h} = 2$, and this is the largest number that makes it possible to calculate analytically correct dual variables for the routing subproblems.

The idea is that layer 0 belongs to the producer depots and layer 1 through $\bar{h}$ to the customer depots (each customer is represented in all layers from 1 to $\bar{h}$). A transport to a customer arrives in one layer higher than the departure layer, and a transport to a producer always arrives in layer 0 regardless of which is the departure layer. As an example, a transport from a producer depot to a customer depot starts in layer 0 and ends in layer 1. Accordingly, after $\bar{h}$ customer depot visits it is only possible to travel to a producer depot. However, it is possible to return to a producer before $\bar{h}$ customer depots have been visited. An *outbound trip* for a vehicle v is defined as a trip that starts at a producer depot, visits a number of customer depots, and ends when v returns to some producer. We require that an outbound trip must contain at least one customer, which is why a transport between two producers is not considered as an actual outbound trip even though it typically is allowed to travel directly between producers. The optimal route from the routing subproblem is defined as a set of outbound trips, which we denote O_v. A small example, including only a few transport options, of a time expanded transportation network is shown in Fig. 5. In the example, 3 network layers are used to allow a maximum of 2 customer depot visits before visiting a producer.

A time expanded transportation network for vehicle v is defined as a directed graph $(\mathcal{N}_v \cup \{a\}, \mathcal{A}_v)$ with a set of nodes $\mathcal{N}_v \cup \{a\}$ and a set of arcs $\mathcal{A}_v$. The set $\mathcal{N}_v = \{n_{dht} : d \in D, h \in H, t \in T\}$ is a set of network nodes corresponding to actual

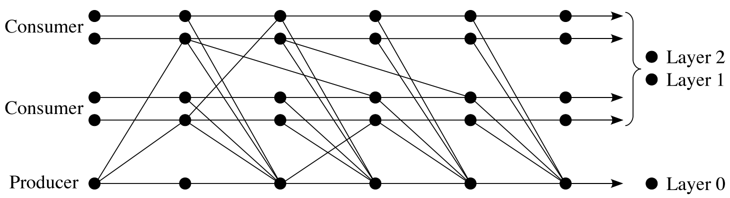

Fig. 5. An example of a time expanded transportation network, in which a maximum of two customer depot visits are allowed before returning to the producer depot

depots, and a is an artificial node that allows v to end its route anywhere in the network. An arc $(n_i, n_j) \in \mathcal{A}_v$, with travel cost $c^{\mathrm{RP}_v}_{(n_i,n_j)}$, is a connection between a starting node $n_i \in \mathcal{N}_v$ and an ending node $n_j \in \mathcal{N}_v \cup \{a\}$. We introduce parameter $\eta_{(n_i,n_j)}$ as a function of the actual dual variables μ and λ to describe an extra cost or discount that is added to the cost of arc (n_i, n_j). The calculation of the $\eta_{(n_i,n_j)}$ parameters is made in a way that the structure of the transportation network is being utilized, as will be detailed below. We introduce decision variable $x^{\mathrm{RP}_v}_{(n_i,n_j)} \in \{0, 1\}$ to determine the usage of arc $(n_i, n_j) \in \mathcal{A}_v$ so that $x^{\mathrm{RP}_v}_{(n_i,n_j)} = 1$ if arc (n_i, n_j) is used in the solution, otherwise $x^{\mathrm{RP}_v}_{(n_i,n_j)} = 0$.

The routing subproblem RP_v for vehicle v is formulated as a standard minimum cost flow problem as:

$$\min \sum_{(n_i,n_j)\in\mathcal{A}_v} (c^{\mathrm{RP}_v}_{(n_i,n_j)} - \eta_{(n_i n_j)}) x^{\mathrm{RP}_v}_{(n_i,n_j)} \tag{26}$$

$$\text{s.t.} \sum_{n_k:(n_k,n_i)\in\mathcal{A}_v} x^{\mathrm{RP}_v}_{(n_k,n_i)} - \sum_{n_j:(n_i,n_j)\in\mathcal{A}_v} x^{\mathrm{RP}_v}_{(n_i,n_j)} = b_{n_i}, \quad n_i \in \mathcal{N}_v \cup \{a\}, \tag{27}$$

$$x^{\mathrm{RP}_v}_{(n_i,n_j)} \in \{0, 1\}, \qquad (n_i, n_j) \in \mathcal{A}_v.$$

Constraint set (27) specifies the node balance constraints where b_{n_i} follows the rules in equation (28). Node $n_s \in \mathcal{N}_v$ denotes the node where v is situated at the beginning of the planning period, and a allows v to be at any node at the end of the planning period. The node balance parameter b_{n_i} for node n_i is defined as

$$b_{n_i} = \begin{cases} -1 & \text{if } n_i = n_s \\ 1 & \text{if } n_i = a \\ 0 & \text{otherwise.} \end{cases} \tag{28}$$

The arc set $\mathcal{A}_v$ can be described as a subset of the union

$$\mathcal{A}^{\mathrm{WP}}_v \bigcup \mathcal{A}^{\mathrm{PP}}_v \bigcup \mathcal{A}^{\mathrm{PC}}_v \bigcup \mathcal{A}^{\mathrm{CC}}_v \bigcup \mathcal{A}^{\mathrm{CP}}_v \bigcup \mathcal{A}^{\mathrm{A}}_v,$$

where the content of each of these sets will be detailed below. The estimated time $t^{\mathrm{link}}_{d'd''hv}$, that vehicle v needs for traveling the direct link from depot d' to depot d'', starting in layer h, includes actual driving time, estimated times for loading and unloading, and estimated resting time. How to calculate link-traveling times will be detailed below. We here represent a network node using three indices; depot, layer and time period.

$\mathcal{A}^{\mathrm{WP}}_v = \{(n_{d0t}, n_{d0,t+1}) : d \in D^P, t \in T \setminus \{\bar{t}\}\}$ contains arcs going from a producer to the same producer. This allows v to wait at a producer depot between to subsequent time periods.

$\mathcal{A}^{\mathrm{PP}}_v = \{(n_{d'0t}, n_{d''0,t+\lceil t^{\mathrm{link}}_{d'd''0v}\rceil}) : d', d'' \in D^P, d' \neq d'', t \in \{0, \ldots, \bar{t} - t^{\mathrm{link}}_{d'd''0v}\}\}$ contains arcs going from one producer to a different producer to allow v to travel between producer depots.

$\mathcal{A}^{\mathrm{PC}}_v = \{(n_{d'0t}, n_{d''1,t+\lceil t^{\mathrm{link}}_{d'd''0v}\rceil}) : d' \in D^P, d'' \in D^C, t \in \{0, \ldots, \bar{t} - t^{\mathrm{link}}_{d'd''0v}\}\}$ contains arcs going from a producer to a customer, allowing v to travel from producer depots to customer depots.

$\mathcal{A}_v^{\text{CC}}$ $= \{(n_{d'ht}, n_{d'',h+1,t+\lceil t^{\text{link}}_{d'd''hv}\rceil}) : d', d'' \in D^C, d' \neq d'', h \in \{1, \ldots, \bar{h}-1\}, t \in \{0, \ldots, \bar{t} - t^{\text{link}}_{d'd''hv}\}\}$ contains arcs going from a customer to a different customer allowing v to travel between customer depots.

$\mathcal{A}_v^{\text{CP}}$ $= \{(n_{d'ht}, n_{d''0,t+\lceil t^{\text{link}}_{d'd''hv}\rceil}) : d' \in D^C, d'' \in D^P, h \in H \setminus \{0\}, t \in \{0, \ldots, \bar{t} - t^{\text{link}}_{d'd''hv}\}\}$ contains arcs going from a customer to a producer to allow v to travel from customer depots to producer depots.

$\mathcal{A}_v^{\text{A}}$ $= \{(n, a) : n \in \mathcal{N}_v\}$ contains one arc from each network node to the artificial node a to allow v to stop its route at any location, in any layer, and in any time period. The transportation cost for an "artificial arc", starting in node n, corresponds to the cost for traveling from n to the "home base" of v. The reason for adding costs to the artificial arcs, even though they do not correspond to actual transports, is that we do not want any particular location to be favored at the end of the planning period.

If there is no direct connection between two depots, $\mathcal{A}_v$ will not contain any arcs between the corresponding network nodes. Moreover, the time expanded transportation networks contain no arcs allowing vehicles to wait at customer depots. We consider this modeling assumption reasonable, because in the considered problem there is no need for waiting at customers.

Transportation costs for arcs in the routing subproblems are composed of three types of costs, which in the model are represented by link-based costs:

1. Time-based costs (e.g., driver salary, capital cost, and administration) are assumed for the time the vehicle spends away from a producer depot. The driver is assumed to receive salary when the vehicle is on the road and during unloading of products. Loading of products is performed by ground staff, who has the same salary as the drivers. Unloading, on the other hand, is assumed to be performed by the drivers. Therefore, the driving time, as well as the time for unloading need to be compensated for by resting time, and salary is not considered during resting.
2. Distance-based costs (e.g., fuel, vehicle wear, and kilometer taxes) are based on the distance the vehicles travel.
3. Link-based costs (e.g., road tolls) are fixed costs that are charged when vehicles travel on certain links.

When a routing subproblem is formulated there exists no information about loadings and unloadings (e.g., concerning quantities). Therefore, loading and unloading times have to be estimated, and we have chosen to base these estimations on average loading and unloading times taken over all available product types. We assume loading times for a full vehicle at producer nodes, and unloading times for a 50% load at customer depots, i.e., loading time is added to arcs leaving producers and unloading time to arcs arriving to customer depots. Accordingly, we model fixed times for loading and unloading, which are independent on actual transportation volumes.

The working and resting hour regulation used in the routing subproblem allows for a maximum of $t^{\text{max working}}$ hours of working (driving) before there must be a minimum of $t^{\text{min resting}}$ hours of resting. However, we assume that two outbound trips that in sequence violate the working hour regulation are performed by different drivers. Hence, working hour restrictions are only considered within a single outbound trip.

To be able to estimate the time that the driver needs to rest when traveling on a link from depot d' to depot d'', starting in layer h, we introduce a working hour estimation $t^{\text{est working}}_{d'hv}$. This is a lower bound estimation of the time it takes for v to drive from any producer depot to the starting depot d', with the requirement that it must pass exactly h different customer depots. The estimation is actually the shortest path from any producer to d' with the additional requirement of exactly h customer depot visits. For small networks, and small numbers of $\bar{h}$, these lower bound estimations can be found rather easily.

The estimated time $t^{\text{link}}_{d'd''hv}$ that vehicle v needs to travel a link from depot d' to depot d'', when starting in layer h, is calculated according to equation

$$t^{\text{link}}_{d'd''hv} = t^{\text{working}}_{d'd''hv} + t^{\text{resting}}_{d'd''hv}, \tag{29}$$

where the link traveling time is decomposed into working time $t^{\text{working}}_{d'd''hv}$ (i.e., time for driving and unloading) and resting time $t^{\text{resting}}_{d'd''hv}$. The resting time $t^{\text{resting}}_{d'd''hv}$ is estimated according to equation

$$t^{\text{resting}}_{d'd''hv} = k^d \cdot t^{\text{min resting}}, \tag{30}$$

as the number of resting periods k^d times the minimum resting time $t^{\text{min resting}}$ of each such resting period. The number of resting periods depends on the link working time $t^{\text{working}}_{d'd''hv}$ and on the working time estimation $t^{\text{est working}}_{d'hv}$ for traveling to depot d', and it is calculated as

$$k^d = \max\left\{ \left\lfloor \frac{\left(t^{\text{est working}}_{d'hv} - t^{\text{est load}}\right) \bmod t^{\text{max working}} + t^{\text{working}}_{d'd''v} - \varepsilon}{t^{\text{max working}}} \right\rfloor, 0 \right\}, \tag{31}$$

where the expression

$$\left(t^{\text{est working}}_{d'hv} - t^{\text{est load}}\right) \bmod t^{\text{max working}} \tag{32}$$

denotes the remaining portion of the estimated working time $t^{\text{est working}}_{d'hv}$ that has not been accounted for by resting time when traveling to d'. The loading time is subtracted from the estimated working time $t^{\text{est working}}_{d'hv}$ because loading is assumed to be performed by ground staff. The parameter $0 < \varepsilon < 1/t^{\text{max working}}$ is introduced in order to avoid a special case that occurs in equation (31) whenever the denominator is a devisor of the nominator, causing k^d to take on a value that is 1 unit too large. It is worth noting that all times are expressed in minutes represented by integer numbers. It should also be noted that the estimation of resting times assumes that vehicles never wait at customer depots, which is also prohibited in the routing problem.

As mentioned above, the η cost parameters that are used in the routing subproblems are calculated from the μ and λ dual variables in a way that the structure of the transportation network is utilized. For each arrival customer depot, network layer and time period, the calculation of an analytically correct value of the η parameter requires knowledge about when and from which producer depot the vehicle has departed. To enable correct costs and strong estimates on time, we make the following assumptions:

1. An outbound trip is allowed to contain at most two customer depot visits.
2. Each customer receives products from exactly one producer depot.
3. Waiting at customer depots is not allowed.

With these assumptions, the estimated resting times will also be analytically correct, i.e., not estimated. Otherwise it is impossible to know how vehicles travel in the network. The assumptions listed above are essential because driving time, which has not been "rested for" accumulates over traveled links. A small example of a driving and resting time approximation is given in Fig. 6.

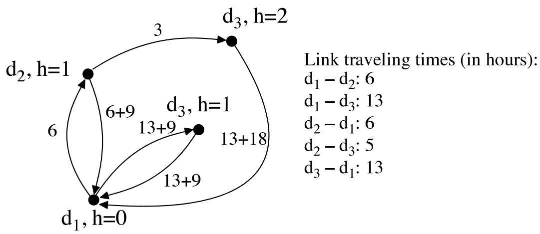

Fig. 6. An example of a working and resting time approximation with $t^{\text{max working}} = 10$ and $t^{\text{min resting}} = 9$

From the maximum capacity of vehicle v (weight capacity φ_v^{weight} and volume capacity φ_v^{volume}), for each $p \in P$ we estimate the maximum number

$$\Phi_{vp} = \left\lfloor \min \left\{ \frac{\varphi_v^{weight}}{weight(p)}, \frac{\varphi_v^{volume}}{volume(p)} \right\} \right\rfloor \tag{33}$$

of items of product p that can be loaded on v. The Φ_{vp} :s are restricted either by the volume capacity or by the weight capacity of v, and they will be used when calculating the η parameters, as well as in the product assignment problem presented below.

In order to compute the parameter η we first consider transports from producer depots to customer depots. A transport is assumed to start in a network node n_s representing producer depot $d(n_s) \in D^P$ and period $t(n_s)$, and it is assumed to end in a network node n_e representing customer depot $d(n_e) \in D^C$ and period $t(n_e)$. This case is rather straightforward and the $\eta_{(n_s,n_e)}$ value is calculated according to

$$\eta_{(n_s,n_e)} = \max \left\{ 0, \max_{p \in P'} \left(\left(\lambda_{d(n_e)pt(n_e)} - \mu_{d(n_s)pt(n_s)} \right) \cdot \Phi_{vp} \right) \right\}, \tag{34}$$

where $P' \subseteq P$ describes the set of products that can be produced in $d(n_s)$ and consumed in $d(n_e)$. Next we consider transports between two different customer depots. Here a transport is assumed to start in a network node n_e representing customer depot

$d(n_e) \in D^C$ and period $t(n_e)$, and it is assumed to end in a network node n_c representing customer depot $d(n_c) \in D^C$ and period $t(n_c)$. Knowledge about the network node n_s representing the producer depot $d(n_s) \in D^P$ and the period $t(n_s)$ from where the transport was assumed to start is required for a correct calculation of $\eta_{(n_e,n_c)}$. As mentioned above, such information is accessible since: at most two customer visits is allowed in an outbound trip, each customer can be reached from at most one producer and waiting at customer depots is forbidden. The value of $\eta_{(n_e,n_c)}$ can be calculated as

$$\eta_{(n_e,n_c)} = \max\left\{0, \eta_{(n_s,n_c)} - \eta_{(n_s,n_e)}\right\}, \tag{35}$$

where $\eta_{(n_s,n_e)}$ and $\eta_{(n_s,n_c)}$ is calculated according to equation (34). Hence the value for going from n_e to a customer depot n_c is equivalent to the potential extra value that can be obtained at n_c compared to the value at n_e.

In a routing subproblem, quantities can be fixed either at producer depots or at customer depots and the two cases are handled differently. In routing problem RP$_v$, a constraint

$$\sum_{n_i:d(n_i)=d',t(n_i)=t'} \sum_{n_k:(n_i,n_k)\in\mathcal{A}_{v'}} x_{(n_i,n_k)}^{\text{RP}_{v'}} = 1 \tag{36}$$

is added for each producer depot fixing $f_{d'p't'v'}^T > 0$, and for each customer depot fixing $f_{d'p't'v'}^T > 0$ we add a constraint

$$\sum_{n_k:(n_k,n_j)\in\mathcal{A}_{v'}} \sum_{n_j:d(n_j)=d',t(n_j)=t'} x_{(n_k,n_j)}^{\text{RP}_{v'}} = 1. \tag{37}$$

Constraint (36) means that v' must depart from producer depot d' in period t', and constraint (37) forces vehicle v' to arrive at customer depot d' in period t'. It should be emphasized that these constraints remove the integrality property of the routing problems.

Implicitly the product assignment subproblem is solved already in the routing subproblem. However, we explicitly formulate an optimization problem in order to handle fixings, i.e., minimum quantities for pickups and deliveries. From the optimal route O_v determined by RP$_v$, defined as a set of outbound trips, a product assignment problem is formulated. The purpose of a product assignment problem is to decide how much of each product will be picked-up from each producer depot and delivered to each customer depot in the route. The problem separates into one subproblem (ASP$_o$) for each outbound trip $o \in O_v$.

We introduce an ordered index set $J_o = \{1, \ldots, j_o\}$ over the depot visits in outbound trip $o \in O_v$, starting with index 1 for the producer depot, index 2 for the first customer depot, etc. For simplified representation we let $d(j)$ and $t(j)$ refer to the depot and time period represented by visit j in the outbound trip. Moreover, we let decision variable $x_{d(j)p}^{\text{ASP}_o} \in \mathbb{Z}^+, p \in P, j \in J_o \setminus \{1\}$ represent the amount of product type p that is delivered to the j :th customer in outbound trip o.

We formulate the product assignment problem ASP_o for outbound trip $o \in O_v$ as:

$$\begin{aligned} \max \quad & \sum_{p \in P} \sum_{j \in J_o \setminus \{1\}} \left(\lambda_{d(j)pt(j)} - \mu_{d(1)pt(1)} \right) x^{\text{ASP}_o}_{d(j)p}, && (38) \\ \text{s.t.} \quad & \sum_{p \in P} \sum_{j \in J_o \setminus \{1\}} weight(p) \cdot x^{\text{ASP}_o}_{d(j)p} \le \varphi^{weight}_v, && (39) \\ & \sum_{p \in P} \sum_{j \in J_o \setminus \{1\}} volume(p) \cdot x^{\text{ASP}_o}_{d(j)p} \le \varphi^{volume}_v, && (40) \\ & x^{\text{ASP}_o}_{d(j)p} \in \mathbb{Z}^+, \qquad p \in P, j \in J_o \setminus \{1\}. \end{aligned}$$

The objective function (38) maximizes the utility of pickup and deliveries for the given route, and constraint sets (39) and (40) express the weight and volume restrictions on vehicle v.

In a product assignment problem, a quantity can be fixed either for a producer depot or for a customer depot. For each producer depot fixing $f^T_{d'p't'v'} > 0$, a constraint

$$\sum_{j \in J_{o'} \setminus \{1\}} x^{\text{ASP}_{o'}}_{d(j)p'} \ge f^T_{d'p't'v'} \tag{41}$$

is added to the assignment problem $\text{ASP}_{o'}$ that represents $f^T_{d'p't'v'}$. This forces v' to pickup at least $f^T_{d'p't'v'}$ units of product p' from producer depot d' in period t'. Moreover, for each customer fixing $f^T_{d'p't'v'} > 0$, we add a constraint

$$x^{\text{ASP}_{o'}}_{d'p'} \ge f^T_{d'p't'v'}, \tag{42}$$

to the assignment problem $\text{ASP}_{o'}$ that represents $f^T_{d'p't'v'}$. This guarantees that at least $f^T_{d'p't'v'}$ units of product p' will be delivered to customer depot d' in period t'.

The ASP_o subproblems assign products to the optimal route O_v, and together they form a transportation plan. After solving the routing subproblem and the product assignment subproblems, the optimal objective function value (i.e., the reduced cost in RMP) of the transportation subproblem for vehicle v can be calculated as

$$\sum_{(n_i,n_j) \in \mathcal{A}_v} c^{\text{RP}_v}_{(n_i,n_j)} x^{*\text{RP}_v}_{(n_i,n_j)} + \sum_{o \in O_v} \text{ASP}^*_o - \delta_v, \tag{43}$$

where $x^{*\text{RP}_v}_{(n_i,n_j)}$ denotes the optimal value of variable $x^{\text{RP}_v}_{(n_i,n_j)}$, ASP^*_o denotes the optimal objective value of ASP_o, and δ_v is the convexity constraint dual variable for vehicle v.

5.3 Production Scheduling Subproblems

The purpose of a production scheduling subproblem for a production line $l \in L$ is to find improving production plans for l guided by the values of the μ dual variables. We let c^{prod}_{lp} denote the cost for production line l to produce one unit of product $p \in P$, and c^{setup}_{lp} denote a fixed setup cost for each period product p is produced. The modeled real-world problem contains costs for startup and changeover, but we consider the problem formulation here to be detailed enough. A more advanced model for the studied real-world production problem is provided by Sohier [25]. The model by Sohier includes product sequencing on a set of production lines, but it is restricted to short planning horizons, which makes it difficult for us to use.

We let decision variable $x_{pt}^{\text{PSP}_l} \in \mathbb{Z}^+$ determine the amount of product p to be produced in period t. The products that are produced in a period are assumed to be available for pickup in the same period. Binary variable $y_{pt}^{\text{PSP}_l} \in \{0,1\}$ is used to indicate whether there will be a production of product p in period t ($y_{pt}^{\text{PSP}_l} = 1$) or not ($y_{pt}^{\text{PSP}_l} = 0$), and we let U denote the maximum number of different product types that can be produced in one period. The production scheduling subproblem PSP_l can be formulated as:

$$\min \quad \sum_{p\in P}\sum_{t\in T}\left(c_{lp}^{\text{prod}} - \mu_{pt}\right) x_{pt}^{\text{PSP}_l} + \sum_{p\in P}\sum_{t\in T} c_{lp}^{\text{prod}} y_{pt}^{\text{PSP}_l} - \theta_l \tag{44}$$

$$\text{s.t.} \quad \sum_{p\in P} t_{lp}^{\text{prod}} x_{pt}^{\text{PSP}_l} \le \tau, \qquad t \in T, \tag{45}$$

$$t_{lp}^{\text{prod}} x_{pt}^{\text{PSP}_l} \le \tau y_{pt}^{\text{PSP}_l}, \qquad p \in P, t \in T, \tag{46}$$

$$\sum_{p\in P} y_{pt}^{\text{PSP}_l} \le U, \qquad t \in T, \tag{47}$$

$$x_{pt}^{\text{PSP}_l} \in \mathbb{Z}^+, \qquad p \in P, t \in T,$$

$$y_{pt}^{\text{PSP}_l} \in \{0,1\}, \qquad p \in P, t \in T.$$

Constraints (45) models the capacity constraints, where t_{lp}^{prod} denotes the time needed for production line l to produce one unit of product p, and τ denotes the length of a time period. To be able to model setup costs, constraint set (46) forces each $y_{pt}^{\text{PSP}_l}$ variable to value one whenever the corresponding $x_{pt}^{\text{PSP}_l}$ is greater than 0. Constraint set (47) restricts the number of different product types can be produced in any period to U. Note that we subtract the convexity constraint dual variable θ_l in the objective function, and that the inventory balance constraints normally included in production scheduling problems here belong to the master problem, which is controlled by the supply chain coordinator. Therefore, the production scheduling problems separate over time. The objective function value of the optimal solution gives the reduced cost of the generated production plan.

To represent production fixings in the production scheduling subproblems, for each production fixing $f_{p't'l'}^{P} > 0$ we add a constraint

$$x_{p't'}^{\text{PSP}_{l'}} \ge f_{p't'l'} \tag{48}$$

to subproblem $\text{PSP}_{l'}$. This forces production line l' to include a production of at least $f_{p't'l'}$ units of product p' in period t' in all subsequently generated plans.

6 Computational Experiments

Our DW column generation algorithm has been implemented inside a multi-agent-based simulation tool called TAPAS [24]. TAPAS is implemented in the Java programming language using the Java Agent DEvelopment Framework (JADE) platform [26] and the MILP solver ILOG CPLEX[1] 10.0 is used for solving optimization problems. In the experiments TAPAS runs on a single computer, but a distributed implementation where agents run on different computers, is straightforward. This would increase the communication overhead while potentially reducing the overall solution time due to the availability of more processors and the potential for parallel processing. In our implementation, overhead is caused by JADE, e.g., by spending effort on coding and decoding messages. Our implementation approach simulates a situation with a coordinator agent and planning agents located at different locations, and it allows for investigation of the potential usage of the approach.

6.1 Scenario Description

The proposed solution approach has been used for solving 5 scenarios with a transportation network containing 2 producer depots d_1 and d_7 (with one production line each), 6 customer depots $d_2, \ldots, d_6$ and d_8, and a planning horizon of 72 time periods with uniform length 2 hours (i.e., 6 days). The scenarios were generated randomly, and they differ in customer demand, as well as minimum, maximum and initial storage levels. In each scenario, three different settings regarding VMI were tested, giving a total of 15 simulation runs. Producer depot d_1 provides customer depots $d_2, \ldots, d_6$ with products and the purpose of d_7 and d_8 is to model possibly non-empty return transports for a small portion of the vehicles returning to d_1. The routes between depots are represented by direct links, with distances given in Table 1. From depot d_1 it is possible to travel to depots $d_2, \ldots, d_7$, from $d_2, \ldots, d_6$ to all depots except d_8, from d_7 to d_1 and d_8, and from d_8 to d_1. The average speed is 70 km/hour on all links and the time-based transportation cost is estimated to 250 SEK/hour. Furthermore, the EEC regulations (EEC 3820/1985) for working and resting hours are approximated by allowing a maximum of 10 hours of working before a minimum of 9 hours of resting has to take place. Moreover, the distance-based cost during transportation is 7.69 SEK/km.

Table 1. Distance (in km) between the depots in the scenario

d_1							
355	d_2						
353	216	d_3					
450	407	167	d_4				
531	487	248	141	d_5			
615	455	333	226	93	d_6		
209	130	232	362	442	522	d_7	
95	-	-	-	-	-	119	d_8

The scenarios use a fleet of 9 vehicles $v_1, \ldots, v_9$ with identical weight capacities of 35 tons. The volume restriction for the vehicles are dominated by the weight restrictions,

[1] `http://www.ilog.com/`

and there are no limitations concerning which types of products can be transported on the different vehicles. For each vehicle, we specify a depot and an earliest time period from which it is available for further planning. Vehicles v_8 and v_9 represent third party transport capacity, which can be called in to a cost approximately 10% higher than the cost for using $v_1, \ldots, v_7$.

The scenarios use 4 different product types $p_1, \ldots, p_4$, each with a mass of 5 tons per unit (or batch). Product type p_4 is used to model return transports and it is assumed that p_4 is produced only in d_7 and consumed only in d_8. Therefore, the production and setup costs for p_4 are assumed to be 0. Products $p_1, \ldots, p_3$ can be produced in production line l_1 with production costs of 1500 SEK/unit and setup costs of 1000 SEK. The production capacity of product types $p_1, \ldots, p_3$ is 2.5 batches/hour and the capacity for producing p_4 is 4 batches/hour. The cost for storing one unit of $p_1, \ldots, p_3$ between two subsequent periods is 300 SEK in d_1 and 350 SEK in $d_2, \ldots, d_6$. For p_4, the storage cost equals 150 SEK in d_7 and 175 SEK in d_8.

For each producer depot, an initial inventory level was chosen randomly between 0 and a maximum inventory level, which is 10 for products $p_1, \ldots, p_3$ in d_1, and 7 for product p_4 in d_7. The penalty cost for violating the allowed minimum and maximum inventory levels is 8200 SEK/unit for d_1 and 4120 SEK/unit for d_7. For customer depots, the penalty cost for exceeding the maximum allowed inventory level is 8200 SEK/unit for depots $d_2, \ldots, d_6$ and 4120 SEK/unit for d_8. For inventory shortages, the penalty is 9020 SEK/unit for depots $d_2, \ldots, d_6$ and 4520 SEK/unit for depot d_8.

In depots $d_2, \ldots, d_6$, the forecasted average consumptions (in units per 2 hour time period) of products $p_1, \ldots, p_3$ is 0.15, and in d_8, the average forecasted consumption of p_4 is 0.35. A demand-forecast used in a scenario is chosen randomly in steps of 0.05 units with equal probability between 0 and 2 times the averaged forecasted demand. For example, for an expected forecast of 0.2, values between 0 and 0.4 can be generated. Consumption is aggregated to integer values, which means, for instance, that an average consumption of 0.2 units per period gives a demand for 1 unit every 5:th period. The main reason for aggregating consumptions is that transportation is performed in integer quantities, and it simplifies our comparisons between scenarios with and without VMI.

In a customer depot, the safety inventory level for a product is chosen as the quantity that, according to the consumption forecast, is consumed during any period of random length between 1 and 2 days. In the same way, for a VMI customer, the maximum inventory level is chosen as the quantity that is expected to be consumed during a period of random length between 5 and 7 days. An initial inventory level is chosen randomly as an integer number between the safety and the maximum inventory levels. For non-VMI customers, we randomly generated delivery quantities (one for each customer and each product type) in steps of 5 between 5 and 35 tons (i.e., 1 to 7 units). This is justified by a historical, for the studied producer, average delivery size of approximately 20 tons. Hence, for a non-VMI customer the maximum inventory level for a product is assumed to be equal to the safety inventory level plus the delivery size, minus one (deliveries are made so that the inventory level never falls below the minimum level). To be able to compare the results for VMI and non-VMI settings, a maximum inventory level for a VMI customer is not allowed to be less than the maximum level for the same customer and product in the corresponding non-VMI setting. For VMI customers, the inventory levels always have to be kept between the safety and maximum allowed inventory levels, and for a non-VMI customer the inventory level of a product in a particular period equals (i.e., with equality in equation (4)) a certain level that depends on the initial inventory level, delivery size, order point, and forecasted consumption rate.

The time it takes to load a truck is estimated to 1 hour and the time it takes to unload a truck is estimated to 2 hours, with costs 250 SEK and 500 SEK respectively. Before loading takes place at a producer depot, the truck has to be cleaned, and cleaning takes 1 hour and costs 1300 SEK. Loading and unloading times are assumed for full trucks, which we consider to be reasonable since the studied transport operator indicates that it is common that vehicles are fully loaded (or close to being fully loaded).

6.2 VMI Analysis

For the set of 5 randomly generated scenarios, simulation experiments with 3 different VMI settings have been performed. We let *full* refer to a setting where all customers except d_8 use VMI, *none* to a setting with only non-VMI customers, and *one* to a setting where only customer d_3 use VMI. In the experiments we used a termination criteria with parameters $e^V = e^P = 3$ and $g^V = g^P = 10$.

The results from the experiments, with respect to different types of costs, are presented in Table 2. In Table 3 we present the reduction of each type of cost when taking the step from setting *none* to setting *one*, i.e., when going from zero to one VMI customer. The additional cost reductions when letting all customers except d_8 use VMI are presented in Table 4. The economic advantage for *full* compared to *one* is rather

Table 2. The costs for production, transportation, storage, and penalty, as well as the total cost in the system, for each scenario (Sc) and VMI setting. In the penalty cost column, the total penalty cost for violating safety stock levels is written first, followed by the total penalty cost for exceeding maximum inventory levels.

Sc	Setting	Production	Transportation	Storage	Penalty (short / exc)	**Total**
1	*full*	119000	220741	135738	0 / 0	**475479**
1	*one*	119500	283987	139617	0 / 28720	**571824**
1	*none*	140000	261653	139838	0 / 114800	**656291**
2	*full*	212500	300789	172210	27080 / 0	**712579**
2	*one*	233000	354719	174083	99280 / 65600	**926682**
2	*none*	234500	398659	170829	72220 / 98400	**974608**
3	*full*	166000	262935	174190	0 / 0	**603125**
3	*one*	173500	385060	176325	63140 / 114800	**912825**
3	*none*	145000	366564	163920	198880 / 271480	**1145844**
4	*full*	194500	264527	156875	0 / 0	**615902**
4	*one*	213500	304449	162915	54120 / 4120	**739104**
4	*none*	222500	342993	157340	81180 / 172200	**976213**
5	*full*	134500	252012	160033	0 / 0	**546545**
5	*one*	241000	295026	186117	36080 / 164000	**922223**
5	*none*	221000	305386	178754	63140 / 176320	**944600**

Table 3. The reduction of the production cost, transportation cost, storage cost, total penalty cost, total system cost, and total cost when the penalty cost is disregarded, when going from setting *none* to setting *one*. It should be noted that cost reductions are represented by positive numbers.

Sc	Production	Transportation	Storage	Penalty	**Total**	Total (no penalty)
1	20500	-22334	221	86080	**84467**	-1613
2	1500	43940	-3254	5740	**47926**	42186
3	-28500	-18496	-12405	292420	**233019**	-59401
4	9000	38544	-5575	195140	**237107**	41969
5	-20000	10360	-7363	39380	**22377**	-17003
Avg	-3500	10403	-5675	123752	**124980**	1228
Std Dev	20338	30917	4711	118332	**102897**	42877

Table 4. The reduction of the production cost, transportation cost, storage cost, total penalty cost, total system cost, and total cost when the penalty cost is disregarded, when going from setting *one* to setting *full*. Again cost reductions are presented with positive numbers.

Sc	Production	Transportation	Storage	Penalty	**Total**	Total (no penalty)
1	500	63246	3879	28720	**96345**	67625
2	20500	53930	1873	137800	**214103**	76303
3	7500	122125	2135	177940	**309700**	131760
4	19000	39922	6040	58240	**123202**	64962
5	106500	43014	26084	200080	**375678**	175598
Avg	30800	64447	8002	120556	**223806**	103250
Std Dev	43118	33540	10244	74551	**119273**	48755

obvious since it is observed that all costs in each scenario is lower in *full*. However, the economic advantage for *one* compared to *none* is less obvious. Penalty costs represent real costs for obtaining products too early or too late, e.g., by approximating costs for additional storage, wastage, and missed sales. Sometimes, slightly penalized (but cost efficient) plans might be accepted due to the flexibility of the customers. Therefore, it is relevant here to also compare the costs when penalties are disregarded. If penalty costs are included in the comparison of the total costs, *one* always performs better than *none*. However, if the penalty costs are disregarded, *none* sometimes performs better than *one* and sometimes the opposite holds. The average improvement of 1228 SEK does however indicate that *one* works better. In summary, the results indicate that the algorithm manages to find better solutions for the *full* setting than for the *one* and *none* settings.

6.3 Time Performance Analysis

In each iteration, the coordinator agent sends one plan request message to each planner agent (in our scenarios: one transport planner and two production planners), and each planner returns a response message to the coordinator. A response message can either contain a generated plan or a failure notification indicating that no plan could be generated at the time of the request. For our 15 simulation runs, which used an average of 2171 master problem iterations to reach the final solution, in average 13023 messages was be sent. The number of messages can be used as a measure of the overhead in the system in comparison to a non-agent-based implementation approach. The size of each message depends on the actual application and the problem size, since a bigger problem requires that more information (e.g., dual variables) need to be communicated.

For each of the 15 simulation runs, we estimated an upper bound of the overhead imposed by the agentification our decomposition algorithm. This upper bound was calculated as the total running time minus the estimated time for performing decomposition related tasks. We measured an average total running time of 9509 seconds and an average lower bound estimation of 6583 seconds for the time spent in the actual decomposition algorithm. This gives an average estimated overhead of approximately 35.7% of the total running time, which is taken as the average over the estimated overhead of all simulation runs.

Moreover, we estimated the expected performance improvement that can be achieved when all subproblems are distributed and solved in parallel on different computers. For the 15 simulation runs, we measured average estimated total solution times of 3856 seconds for the transportation subproblems and 17.1 seconds for the production scheduling subproblems. Assuming all problems of the same type need the same solution time, we get average times of 0.20 and 0.0039 seconds for solving one transportation and one production scheduling subproblem. The average transportation subproblem solution time is calculated as the average of the average transportation subproblem solution times taken over the 15 simulation runs, and the average production subproblem solution time is calculated in the same way.

If the communication overhead that is caused by a parallelization is disregarded, a theoretical average potential time reduction from $(3856 + 17.1) \approx 3873$ seconds to 429 seconds is estimated for our scenarios. The theoretical time reduction for a scenario is calculated as the number of master problem iterations times the maximum of the average solution time of one transportation subproblem and the average solution time of one production subproblem. In our scenarios, this would give an average time reduction of $3873 - 429 = 3444$ seconds of the total running time. Here, we assume a use of 12 computers; 9 for transportation subproblems, 2 for production scheduling subproblems and 1 for the master problem. Solving the production scheduling subproblems and the master problem on the same computer would give the same improved running time, but confidentiality of information would be weakened. Note that it is impossible to use the main algorithm as it is displayed in Fig. 3 to utilize a complete parallelization. Instead dual variables have to be sent to the transport planners and the production planners in parallel without resolving RMP in between.

7 Confidentiality of Information

An important advantage of the proposed agent-based approach to DW decomposition, compared to a centralized approach, is the possibility to achieve increased confidentiality of information. From the perspective of what type of information need to be communicated, we here provide a comparison between a centralized and an agent-based approach to DW decomposition.

In a centralized approach, a central node of computation needs to be given access to all information that will be used when formulating the problem. From a confidentiality perspective it is not important whether the problem is formulated using decomposition or using a different approach; basically the same information need to be shared. In an agent-based approach, the coordinator needs to obtain all information that need to be used when formulating the master problem, but subproblem specific information can be kept local. The exact information that needs to be shared is case specific, depending on how the problem is formulated with respect to what aspects are modeled in the master problem and what aspects are modeled in the subproblems. Moreover, in each iteration, dual variables need to be sent to the planner agents, who return plans to the coordinator. It follows that complete confidentiality is practically impossible to achieve. For instance, since dual variables and plans are iteratively communicated between the coordinator and the planners, it is often possible for the coordinator to create a model of how planner agents generate plans. However, this has to be done without using explicit information about the planner agents.

In addition to knowing about which depots, production lines and vehicles should be modeled, in our case specific solution approach, the coordinator needs the following information to be able to construct MP (or actually IMP): (1) demand forecast for VMI customers and orders for non-VMI customers, (2) inventory constraints for all depots, and (3) penalty costs (for each depot) for violating inventory constraints. In each iteration, dual variables are sent to the planner agents, who return improving production plans and transportation plans. A production plan (for a particular production line) consists of a price and the amount of each product type that is produced in each period throughout the planning horizon. A transportation plan (for a vehicle) defines a price and the amount of each product type that is delivered to, or picked-up from, each depot in each period during the planning period.

8 Concluding Remarks and Future Work

We have shown that it is possible to create an agent-based approach to optimization based on the principles of Dantzig-Wolfe decomposition. Some rather obvious positive effects of the agent-based approach are increased confidentiality, robustness and possibility of distributed computing. Negative effects are increased processing and communication. However, a parallelization of the approach will typically have a positive effect on solution time and quality. One advantage is the ability to locally specify and modify the subproblems, and another is that a local agent always is able to provide the most recent subproblem solution to the local planners even though it may be based on old dual variables.

In order to capitalize on the use of our multi-agent-based approach for achieving performance improvements, we find it interesting to experiment with a distributed system where agents run on different computers. This is a natural representation of the real world, where actors (agents) are geographically separated and an agent typically only has access to its own data and its own computer. The estimations presented in

Section 6.3 indicate that a parallelization may give a significant improvement on the time performance. Also, a parallelization would allow for solving more complex subproblems, which would make it possible to capture more details of the actual problem, as well as obtaining solutions with higher quality. Note that our time performance improvement estimation assumes perfect parallelization with no communication overhead, and further investigation needs to be done regarding communication overhead. Regarding the robustness of the approach, planner agents may fail (go down) temporarily, while the coordinator never is allowed to fail. Future work includes refining the algorithmic approach to enable permanent failures of all agent types to be dealt with. For instance, a possible solution to a coordinator failure is to let a planner agent take on the role as coordinator.

In a case study, we have applied the agent-based decomposition approach to a real-world integrated production, inventory and routing problem. The studied problem is rather general and it is independent of the choice of production and transportation subproblems. However, the planners (subproblems) must be able to take dual variables as input and be able to produce new production or transportation plans that can be communicated to the coordinator (master problem). By choosing customized production and transportation subproblems, our solution approach can be used to solve integrated production, inventory, and routing problems where:

1. Decisions about inventories are taken centrally and detailed decisions about transportation and production are taken locally.
2. The master problem can produce dual variables and receive production and transportation plans.

Hence, we conclude that our approach has the potential to work as a framework for this type of problem. Moreover, the ideas behind our decomposition approach are rather general, and we argue that it can be used to develop solution approaches for other types of problems than our case problem. However, it should be emphasized that a decomposition scheme including master/subproblem formulations, variable restriction strategy, and termination criteria needs to be designed in such a way that the special characteristics of the studied problem is utilized.

For a set of scenarios in the studied case, the presented decomposition approach has been used to conduct a quantitative comparison of different degrees of VMI utilization. The main purpose of the VMI comparison was to illustrate the use of our approach. To obtain results that are statistically significant, more experiments need to be conducted and the approach need to be further validated. The results indicate that the algorithm produces solutions with lower costs for scenarios with more VMI customers. Most likely, the reason is that a higher number of VMI customers increases the solution space of the problem. For instance, we have experienced solutions with high penalty costs for scenarios without VMI customers, which has not been the case for scenarios with VMI customers. A possible explanation is that the algorithm performs better for cases with VMI customers. One reason for this might be that the *full* setting in general uses more iterations than the other settings. Therefore it would be interesting to experiment with termination criteria that allow the different settings to use approximately the same number of iterations before termination. Another possible explanation is the restriction that allows vehicles to wait only in producer depots. This makes it more difficult to obtain transport solutions in which more than one customer is visited in the same outbound trip.

By enabling the economic effects of the introduction of VMI to be studied, we believe that the proposed problem specific solution approach has the potential to provide strategic decision support (concerning VMI) for the involved actors. Moreover, since the approach can be used for suggesting candidate plans for the modeled resources we further believe it has potential to be used as an operational decision support system. For instance, it can suggest plans to human planners in real time to facilitate their decision-making.

References

1. Johnson, E., Whang, S.: E-business and supply chain management: An overview and framework. Production and Operations Management 11(4), 413–423 (2002)
2. Dantzig, G., Wolfe, P.: The decomposition algorithm for linear programs. Econometrica 29, 767–778 (1961)
3. Daugherty, P.J., Myers, M.B., Autry, C.W.: Automatic replenishment programs: an empirical examination. Journal of Business Logistics 20(2), 63–82 (1999)
4. Karageorgos, A., Mehandjiev, N., Weichhart, G., Hämmerle, A.: Agent-based optimisation of logistics and production planning. Engineering Applications of Artificial Intelligence 16(4), 335–348 (2003)
5. Dorer, K., Calisti, M.: An adaptive solution to dynamic transport optimization. In: AAMAS 2005: Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 45–51. ACM, New York (2005)
6. Davidsson, P., Persson, J.A., Holmgren, J.: On the Integration of Agent-Based and Mathematical Optimization Techniques. In: Nguyen, N.T., Grzech, A., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2007. LNCS (LNAI), vol. 4496, pp. 1–10. Springer, Heidelberg (2007)
7. Petcu, A.: A class of algorithms for distributed constraint optimization. Ph.D. thesis, Swiss Federal Institute of Technology, Lausanne, Switzerland (2007)
8. Geoffrion, A.M.: Lagrangean relaxation for integer programming. Mathematical Programming Study 2, 82–114 (1974)
9. Benders, J.: Partitioning procedures for solving mixed-variables programming problems. Numerische Mathematik 4, 238–252 (1962)
10. Barnhart, C., Johnson, E.L., Nemhauser, G.L., Savelsbergh, M.W.P., Vance, P.H.: Branch-and-price: Column generation for solving huge integer programs. Operations Research 46, 316–329 (1998)
11. Hirayama, K.: Distributed lagrangean relaxation protocol for the generalized mutual assignment problem. In: AAMAS 2006: Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 890–892. ACM, New York (2006)
12. Drexl, A., Kimms, A.: Lot sizing and scheduling - survey and extensions. European Journal of Operational Research 99, 221–235 (1997)
13. Toth, P., Vigo, D. (eds.): The Vehicle Routing Problem. SIAM monographs on discrete mathematics and applications, Philadelphia (2002)
14. Thomas, D.J., Griffin, P.M.: Coordinated supply chain management. European Journal of Operational Research 94, 1–15 (1996)
15. Sarmiento, A.M., Nagi, R.: A review of integrated analysis of production-distribution systems. IIE Transactions 31, 1061–1074 (1999)
16. Federgruen, A., Zipkin, P.: A combined vehicle and inventory routing allocation problem. Operations Research 32(5), 1019–1037 (1984)

17. Chien, T.W., Balakrishnan, A., Wong, R.T.: An integrated inventory allocation and vehicle routing problem. Transportation Science 23(2), 67–76 (1989)
18. Campbell, A., Clarke, L., Kleywegt, A., Savelsbergh, M.: The inventory routing problem. In: Crainic, T.G., Laporte, G. (eds.) Fleet Management and Logistics, pp. 95–113. Kluwer Academic Publishers, Dordrecht (1998)
19. Chandra, P., Fisher, M.L.: Coordination of production and distribution planning. European Journal of Operational Research 72(3), 503–517 (1994)
20. Lei, L., Liu, S., Ruszczynski, A., Park, S.: On the integrated production, inventory, and distribution routing problem. IIE Transactions 38(11), 955–970 (2006)
21. Persson, J.A., Göthe-Lundgren, M.: Shipment planning at oil refineries using column generation and valid inequalities. European Journal of Operational Research 163, 631–652 (2005)
22. Bilgen, B., Ozkarahan, I.: A mixed-integer linear programming model for bulk grain blending and shipping. International Journal of Production Economics 107(2), 555–571 (2007)
23. Fumero, F., Vercellis, C.: Synchronized development of production, inventory, and distribution schedules. Transportation Science 33(3), 330–340 (1999)
24. Davidsson, P., Holmgren, J., Persson, J.A., Ramstedt, L.: Multi agent based simulation of transport chains. In: AAMAS 2008: Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 1153–1160. International Foundation for Autonomous Agents and Multiagent Systems, Richland (2008)
25. Sohier, E.: Modelling a complex production scheduling problem - optimization techniques. Master's thesis, Blekinge Institute of Technology, Sweden (2006)
26. Bellifemine, F.L., Caire, G., Greenwood, D.: Developing Multi-Agent Systems with JADE. Wiley Series in Agent Technology. John Wiley & Sons, Chichester (2007)