

Opportunity costs calculation in agent-based vehicle routing and scheduling

Martijn Mes*, Matthieu van der Heijden, Peter Schuur†

11 May 2006

Abstract

In this paper we consider a real-time, dynamic pickup and delivery problem with time-windows where orders should be assigned to one of a set of competing transportation companies. Our approach decomposes the problem into a multi-agent structure where vehicle agents are responsible for the routing and scheduling decisions and the assignment of orders to vehicles is done by using a second-price auction. Therefore the system performance will be heavily dependent on the pricing strategy of the vehicle agents. We propose a pricing strategy for vehicle agents based on dynamic programming where not only the direct cost of a job insertion is taken into account, but also its impact on future opportunities. We also propose a waiting strategy based on the same opportunity valuation. Simulation is used to evaluate the benefit of pricing opportunities compared to simple pricing strategies in different market settings. Numerical results show that the proposed approach provides high quality solutions, in terms of profits, capacity utilization and delivery reliability.

Keywords: Transportation; Multi-agent systems; Auctions/bidding; Scheduling;

1 Introduction

Most techniques and models used in transportation planning, scheduling and routing use centralized solution approaches with static input data. Although such techniques have successfully been

*Corresponding author. Address: Department of Operational Methods for Production and Logistics, Faculty of Business, Public Administration and Technology, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands; phone +31-53-489-4062; fax: +31-53-489-2159; e-mail: m.r.k.mes@utwente.nl.

†Research supported by the BSIK project “Transition Sustainable Mobility (TRANSUMO)”

implemented, they are less suitable in a dynamic environment and in an environment with multiple independent stakeholders. An example of such an environment are online freight and vehicle exchange services such as Teleroute (www.teleroute.com).

Dynamic environments are characterized by frequent and unpredictable modifications in the relevant planning information and the ability (or even necessity) to update the planning based on this additional information. For example, carriers may know only a fraction of the shipments to be served when the first plan is constructed whereas additional (rush) transportation jobs arrive during execution of the original plan. Also, job characteristics as the volume may change during execution. Further, additional information on processing times (travel time updates in case of congestion) and equipment failures may arrive during execution. A proper transport planning approach should be able to construct initial plans taking into account all such uncertainties and to update the plans reacting on real-time information updates.

Particularly for real-time planning updates a central approach is not suitable, because a global re-optimization may lead to a completely different plan in response to a relatively minor information update. A decentralized approach where local problems are solved locally as much as possible to limit schedule disruption has certainly advantages. Besides, it is known that the added value of global optimization versus local planning heuristics decreases in an uncertain, dynamic environment, see e.g. (Heijden et al. 2002). Finally, a distributed solution is required when multiple independent organizational units (multiple fleet managers and shippers) are working in an autonomous, self-interested and not necessarily cooperative way. Then a distributed approach is needed to optimize the network performance (maximize profits) while reckoning with the individual competences, goals and information access.

In the literature, our transportation problem is known as a dynamic multi-vehicle pickup and delivery problem with time windows. We consider a variant with full truckloads and stochastic arrivals of jobs. Within the transportation network multiple shippers offer loads for transportation and multiple fleet managers are competing for these jobs. We propose an agent-based solution approach where vehicle agents are responsible for their own routing and scheduling. The assignment of jobs to vehicles will be done by using an auction. Therefore a proper pricing mechanism is needed to optimize the system wide performance, such as the minimization of the total costs, consisting of transportation costs and penalties on lateness.

In an earlier paper (Mes et al. 2006) we presented a basic multi-agent system and compared its performance with two traditional scheduling heuristics. This paper aims to enhance the

performance of this agent-based transportation system by improving the pricing and scheduling techniques of the vehicle agents. We will focus on two key questions. First, how can we use information on historic job patterns to improve planning and scheduling? Second, what is the impact of such additional intelligence on the overall system performance?

The remainder of this paper is structured as follows. In the next section, we give an overview of related literature and we explain the contribution of our paper. Our model is presented in Section 3. In Section 4 we present our solution method to estimate the value of a schedule using value functions. In Section 5 we explain how these value functions can be used to determine a waiting strategy. In Section 6 we discuss some extensions of our method. Experimental settings and simulation results are presented in Section 7. We end up with conclusions, remarks on generalizations and directions for further research (Section 8).

2 Literature

Our problem is well known in the area of vehicle routing problems (VRP). The VRP and its variants have been studied extensively; see (Laporte et al. 1992; Toth and Vigo 2002) for a survey. Most work focuses on static and deterministic problems in which all information is known in advance, see for example (Desrosiers et al. 1995; Fischer 1995). In dynamic and stochastic vehicle routing problems (also known as real-time routing and dispatching problems) the input data (travel times, demands) are stochastic and change over time. Therefore, the output of a dynamic VRP (DVRP) is not a set of routes, but rather a policy that prescribes how the routes should evolve as a function of those inputs that evolve in real-time (Psaraftis 1988).

Routing and scheduling in a dynamic environment has been studied by a number of authors, see for example (Psaraftis 1988; Gendreau and Potvin 1998). The most common approach to handle these problems is to solve a model using the data that are known at a certain point in time, and to re-optimize as soon as new data become available. Because a fast response is required in a real-time environment, a solution is usually achieved by using relatively simple heuristics or by parallel computation methods, see (Giani et al. 2003) for an overview of approaches.

Another line of research is on dynamic fleet management problems (DFMP), or more general the dynamic assignment problem. These problems ask for a dynamic assignment of resources (trucks) to tasks (loads). Truly stochastic models decompose the DFMP with respect to time periods and assess the impact of the current decisions on the future through a recourse or value function. Examples can be found in (Carvalho and Powell 2000; Godfrey and Powell 2002; Topaloglu and

Powell 2002).

A more recent development is the applicability of multi-agent systems (MAS) in the field of transportation control. This research aims at the development of robust, distributed market mechanisms. In an early paper Sandholm (1993) applied a bidding protocol, called Contract Net Protocol, to a transportation system, where dispatch centers of different companies cooperate automatically in vehicle routing. Fischer et al. (1996) developed a system for cooperative transportation scheduling and a simulation test bed for multi-agent transport planning, called MARS. Another interesting contribution comes from Figliozzi et al. (2003), who present a framework for the study of carriers' strategies in an auction marketplace for dynamic, full truckload vehicle routing with time windows. As in this paper, they use a Vickrey auction and a simple heuristic for generating bids, namely the additional costs of serving a shipment by appending it to the end of the vehicle schedule. They focus on profit allocation rather than on the efficiency of assignment decisions. In (Figliozzi et al. 2004) they study the impact of different assignment strategies on the travel costs under various demand conditions. They consider four fleet assignment methods that are related to the agent-based approaches considered in this paper. In (Hoen and Poutré 2004) a multi-agent system is presented for real-time vehicle routing problems with consolidation in a multi-company setting, where vehicles have the option to break an agreement in favor of a better deal.

The main contribution of our paper is that we combine the basic ideas behind DFMP to enhance the bidding strategy of vehicle agents in a multi-agent setting for transport planning. We cannot use the DFMP algorithms directly, because in our case vehicles may schedule multiple transportation jobs in advance. Also the price of a job is not given externally but subject to negotiation. Moreover, the arrival intensity of jobs at a company is not described by an exogenous information process, but can be influenced by better repositioning of vehicles. Further, DFMP is only developed for central optimization and not for a multi-actor setting.

Compared to the research on MAS in the field of transportation management, we focus on the intelligence of agents instead of the MAS architecture as is often subject of research. In our case, this intelligence is mainly concerned with the bid price calculation and the valuation of future implications of new job insertions. This certainly distinguishes our approach from other research in agent-based transportation planning, such as (Figliozzi et al. 2003) and (Hoen and Poutré 2004).

We focus on the bidding strategy of vehicle agents in a multi-agent setting where the intelligence

of agents arises from historical information on job patterns and prices. That is, the bidding strategy should account for the attractiveness of remaining capacity in a vehicle route. This consists of two issues: First, the attractiveness of a schedule destination, based on the probability that during schedule execution a profitable job may arise to be picked up nearby. Second, the attractiveness of so-called “gaps” in a schedule that may arise because two jobs with very distinct time windows are assigned to a vehicle so that time between execution of these two jobs is available for an additional job.

To model this, we will use the basic ideas behind DFMP as discussed in (Powell et al. 1988). Because this is already a hard problem as we will see, we limit the dynamism in our model to real-time arrival of jobs during schedule execution, whereas we exclude other sources of uncertainty like congestion influenced travel times and equipment failures.

3 Model and notation

We consider a pickup and delivery problem with full truckloads, deterministic travel times, and stochastic arrival of jobs. To present our model we subsequently discuss the network and cost structure (3.1), job characteristics (3.2), the market mechanism (3.3) and the vehicle scheduling and bid calculation (3.4) in the next subsections.

3.1 Network description

Our transportation network consists of a set of nodes \mathcal{N} and a set of arcs \mathcal{A} connecting these nodes. In this network multiple fleet managers and shippers operate. The system dynamics is driven by the incoming jobs from shippers that are not known beforehand. These jobs consist of unit loads (full truckloads) which have to be transported between nodes in the network. A set of vehicles \mathcal{V} , belonging to the different fleet managers is available to transport these loads.

We define the handling time of an order as the sum of the loaded travel time and time for loading and unloading. The handling time for a job from node i to node j is deterministic and given by τ_{ij}^f . The empty travel time from node i to j is given by τ_{ij}^e and we assume that $\tau_{ij}^e < \tau_{ij}^f$ for all routes. We further use the notation τ_{ijk}^e to indicate the trip $\tau_{ij}^e + \tau_{jk}^e$.

Objective of the shippers is to minimize their costs. Objective of the fleet managers is to maximize their profits. We consider two costs functions, namely the travel costs $c^t(t)$ as function of the travel time t and the penalty costs $c^p(t)$ in case of lateness ($t > 0$) which is an arbitrary

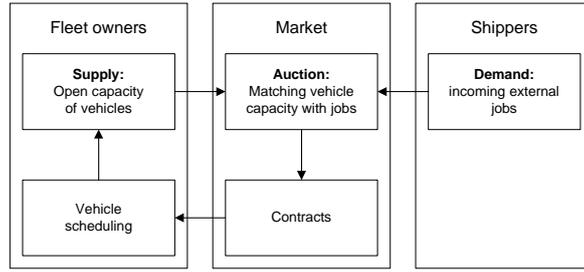


Figure 1: Demand driven online order assignment

positive non-decreasing function of the time t . Of course the travel cost function can easily be extended to reckon with different types of moves (full, empty or pro-active).

In order to cover the transportation costs, the fleet managers will charge the shippers for their transportation services. The total costs for a shipper are given by the sum of all prices paid to the fleet managers for transporting their loads. The profits for the fleet managers are given by their income from all transportation jobs minus the transportation costs and cost for lateness.

Matching of jobs with open vehicle capacity is done using an auction procedure which leads to a contract between a fleet manager and a shipper. Execution of the resulting contracts requires scheduling of the vehicles while taking the contract terms into account. Vehicle scheduling has its impact on the future availability of vehicle capacity and on the system dynamics and hence impact on the profitability of the companies. A general impression of the situation is given in Figure 1.

As for the dynamics and control, we assume that the system is stable in the long run, so that all jobs can be handled. As for communication, we assume that at any moment in time communication between shippers, vehicles and fleet managers is possible.

3.2 Job characteristics

Jobs to transport unit loads (full truckloads) between nodes in the network arrive one-by-one according to a stationary Poisson process with arrival intensity λ_{kl} of jobs from k to l . We define a job φ by the following characteristics:

- the announcement time $a(\varphi)$;
- its origin: node $o(\varphi)$;
- its destination: node $d(\varphi)$;
- the earliest pickup time of the load at its origin: $s(\varphi)$;

- the latest pickup time of the load at its origin: $e(\varphi)$;

The contract terms describe whether the constraints $s(\varphi)$ and $e(\varphi)$ may be violated and the penalty on violations. We assume that these terms are equal for all jobs. We further assume that the fleet manager agrees with the earliest pickup time as a hard restriction and the latest delivery time as soft restriction with penalty costs $c^p(t)$.

We consider two types of contracts, namely contracts with fixed and flexible pickup times. In case of fixed pickup times, the exact pickup and delivery time are agreed upon in the contract and fleet managers are not allowed to change these times later on. In case of flexible pickup times, the pickup and delivery time may be modified during schedule execution where the pickup time is bounded by $s(\varphi)$ time and penalty costs are charged if the latest delivery time $e(\varphi)$ is exceeded. To keep the presentation simple, we first consider fixed contracts and discuss in Section 8 how the results can be extended to flexible contracts.

The time between the earliest pickup time $s(\varphi)$ and the announcement time $a(\varphi)$ is called the look-ahead of job φ . To simplify our presentation we assume that the *look-ahead* of all jobs is zero (which will be generalized in Section 8). The time between the earliest pickup time and latest pickup time is referred to as the *time-window length* of job φ . We assume that the time-window length of future jobs ($e(\varphi) - s(\varphi)$) can be estimated based on the route. Therefore we introduce the notation z_{kl} for the expected time-window length for jobs on route kl . Finally, we assume that an external job in process cannot be interrupted (no preemption). That is, a vehicle may not temporarily drop a load in order to handle a more profitable load and return later.

3.3 Market mechanism

The shippers and fleet managers meet at a market place where jobs are assigned using some auction mechanism. When all bids are known to each bidder we speak of an open auction, otherwise a closed (sealed-bid) auction. An example of an open auction is the well known English auction in which bidders sequentially raise their bids until nobody is willing to bid higher; the object is then sold at that final price. An example of a closed auction is the Vickrey auction in which every bidder submits a single sealed bid and the bidder with the highest bid will receive the object at the price of the second highest bid.

We will use a open second-price auction which is frequently used in online auctions (Caserta 2002) as a mixture between the English auction and the Vickrey auction. Bidders submit their maximum valuation like in the Vickrey auction, and leave sequential bidding rounds (also called

proxy bidding) to a software program. In this open second-price auction bidders only have to place their true valuation of the object once and bids are publicly accessible. Note that for our purposes it is sufficient if bidders have information about the best bid, which equals the final price paid by the shipper. Generalization towards closed auctions is presented in Section 8.

We use this open second-price auction because of its simplicity. Under general conditions, the optimal bidding strategy under such an auction is to bid your true valuation, see (Vickrey 1961). Note that this is not as simple as it seems as we aim to take into account opportunity costs in this paper. Let us illustrate this using an example. Suppose that a vehicle has accepted a job from A to B and has to bid on a job from C to D. The net additional costs are costs of driving empty from B to C and driving loaded from C to D plus possibly lateness costs plus possibly waiting costs if the time windows do not match. However, when taking into account opportunity costs, it is possible that the new schedule destination D is less attractive than the old schedule destination B, because the possibility of getting an attractive job is simply lower. We aim to include this in our cost calculations by adding an estimate for the expected profit loss when staying in location D instead of B and taking into account the timing at which the vehicle is available. As it will appear, this is not a trivial matter.

We implement the market mechanism as follows. When an job φ arrives at some shipper he starts an auction by sending an announcement to all fleet managers. All fleet managers respond with a bid confirming agreement with the contract terms. The shipper evaluates all bids and the winning fleet manager will receive a grant message while the others receive reject messages.

In this paper we will focus on the transportation side: the fleet managers and their vehicles. A fleet manager has to face two decision problems, namely the bidding decision and the assignment decision. The *bidding decision* is concerned with the calculation of the minimum price for which the fleet manager is willing to handle the job. This bidding policy uses information about the state of the fleet manager and his vehicles, the characteristics of the load, the marginal cost of serving this load, and beliefs about the competitors and environment. Such a policy is important because it is critical to the revenue of a company. The calculation of a bid should be made in a short time. The *assignment decision* refers to the assignment of the job to a vehicle and the modification of the vehicle schedule to handle this job. This decision is not necessarily irreversible but can be modified (reassignment) when information is updated.

The bidding decision problem requires the fleet managers to calculate the expected cost for doing this job. However, complete assessment of the feasibility and the expected profit of a job in

real-time is hard. In order to respond fast to these auctions we choose for a distributed structure where every vehicle calculates a bid for this job and sends it to its fleet manager. Therefore vehicles are modeled as intelligent agents that determine their bidding and scheduling strategy based on historic data (experience), learning and on expectations of future consequences of current actions. They have the opportunity of learning about the environment and about other players with each auction. Each vehicle agent is responsible for the planning and scheduling decisions for its corresponding vehicle.

When a fleet manager receives an announcement he will send a request to all his vehicles to calculate the expected cost for handling this job. After receiving all bids from his vehicles he selects the bid with the lowest costs and sends it to the auction. When this fleet manager receives a grant for this job, it will be assigned to the vehicle whose bid was submitted to the auction. We assume that there is no exchange of jobs between vehicles.

3.4 Vehicle scheduling and bid calculation

Vehicle agents face three types of decisions: bidding decisions, scheduling decisions and waiting decisions. In this section we explain how these decisions can be made. Because we focus on the strategies of individual vehicle agents, we omit any subscripts indicating specific vehicles or companies.

3.4.1 Schedule definition

At each point in time, a vehicle has a job schedule, i.e., a list of jobs with scheduled starting times. The destination of the last job in the schedule will be referred to as schedule destination and the time until the expected arrival time at the schedule destination is referred to as length of a schedule.

Formally, we define a schedule Ψ by an ordered list of 2-tuples $\Psi_n = (\psi_{n,1}, \psi_{n,2})$ where $\psi_{n,1}$ refers to a specific job φ and $\psi_{n,2}$ to the scheduled pickup time of this job. We can see a schedule as a sequence of loaded moves and gaps. The loaded move for job n goes from $o(\psi_{n,1})$ to $d(\psi_{n,1})$, starting at time $\psi_{n,2}$ and being delivered at time $\psi_{n,2} + \tau_{o(\psi_{n,1})d(\psi_{n,1})}^f$. A *gap* appears between two consecutive loaded moves whenever the pickup time of the second job is later than the delivery time of the first job. Such a gap may be needed for an empty move from the destination of the first job to the origin of the second job. Further, we introduce the phrase *end-gap* for the difference between the planning period T (which we choose to be much larger than the length of a schedule)

Location	A	B	C	A	D	B	D	
Action	Job 1	Gap 1		Job 2	Job 3	Gap 3	Job 4	End
Time	0	2	7	9	11	13	15	T

Figure 2: Example of a vehicle schedule

and the length of a schedule. The gaps and the end-gap are important to value a schedule, because future jobs can only be inserted in these periods. An illustration of a possible schedule can be found in Figure 2.

3.4.2 Scheduling and waiting decisions

When an auction for a job is started, each vehicle agent creates a temporal schedule combining its current schedule with the new job. In this paper, we assume that vehicle agents can insert a new job at any position in the current schedule without altering the order of execution for the other jobs. Since the fixed contract agreements do not permit moving already assigned jobs, a new job can only be inserted in gaps for which the agreed pickup times for other jobs are not violated.

If the gap is larger than the time needed to insert the job, the vehicle agent has flexibility to select the best pick-up time. One option is to schedule the new job as early as possible, but in some situations it may be advantageous to postpone. Let us illustrate that using an example referring to Figure 2. Suppose that a job from A to C should be inserted in the schedule and that this job can be started at any time between $t = 3$ and $t = 5$, because the job handling time equals $\tau_{AC}^f = 2$ and it takes $\tau_{BA}^e = 1$ to drive empty from B to A. If we schedule the job as early as possible (start at $t = 3$), the vehicle will arrive in C at $t = 5$. Probably, the vehicle has to wait there until the next job can be picked up at $t = 7$. After all, the probability that another job can be inserted in this time interval is low, because an empty ride is needed anyway. If on the other hand the start of the job is scheduled at $t = 5$, the vehicle will arrive in C at $t = 7$ so that the next job can be started immediately. Then the vehicle has three time units to drive from B to A. Then it is very well possible that another job can be inserted in this interval, either from B to A or between other locations that require only little additional empty driving time.

The choice of the starting time for a job in a gap in the current schedule is equivalent to finding the optimal length of the newly created gap before the new job. We will refer to this choice as the *waiting strategy*. Upon arrival at a gap, the vehicle agent also faces an *operational waiting decision*. Suppose the vehicle is currently located in node i and has to be at node j at time t , then he has three options. First, he can drive immediately to node j and wait over there. Second, he

can wait at node i until he wins another job or at last drives to node j at time $t - \tau_{ij}^e$. Third, he can drive pro-actively to another node k in anticipation of a future job and if he does not receive such a job, he can move at time $t - \tau_{kj}^e$ to node j . The waiting strategy and the operational waiting decisions are presented in Section 6.

The scheduling decision is concerned with choosing the best position and pickup time for the new job in the current schedule given a certain waiting strategy. A vehicle agent updates its schedule when (i) the first loaded move in a schedule has been completed, and (ii) an auction for a new job is won. In the first case, the vehicle agent applies its waiting strategy, i.e., the vehicle either starts immediately with the next loaded move or it waits at its current location or it moves empty to a better location. In the second case, the vehicle agent replaces its current schedule with the temporal schedule that had been constructed for the auction.

3.4.3 Bid calculation

A vehicle agent bids the marginal cost of a job insertion, taking into account the expected revenues due to future job insertions in both the current and the temporal schedule. To construct a temporal schedule, a vehicle agent evaluates all possible insertions of the new job in the schedule. Because the first job of the schedule is always in execution, the number of options equals the number of jobs in the current schedule $|\Psi|$. For a given position m in the schedule (i.e., after the m^{th} job in the current schedule), the vehicle agent has to select the most profitable pickup time ω for the new job. Therefore, the bid price $b(\varphi, \Psi)$ for a new job φ given the current schedule Ψ is given by:

$$b(\varphi, \Psi) = \min_{m, \omega} (C^d(\varphi, \omega) + OC(\varphi, m, \omega, \Psi)) \quad (1)$$

where $m = 1..|\Psi|$ and $\omega \geq s(\varphi)$. Given the fixed contract agreements (see Section 3.2) this pickup time ω is also bounded by the starting time of the next job in the schedule. $OC(\varphi, m, \omega, \Psi)$ denotes the opportunity costs of scheduling a new job φ at position m and pickup time ω in the current schedule Ψ (see the next section for the calculation of this function). The direct costs $C^d(\varphi, \omega)$ of picking up the new job φ at time ω is given by the travel costs for the loaded move and possibly penalty costs:

$$C^d(\varphi, \omega) = c^t \left(\tau_{o(\varphi)d(\varphi)}^f \right) + c^p \left((\omega - e(\varphi))^+ \right) \quad (2)$$

The costs for a possible empty move towards the origin of the new job φ are not included in the direct costs because these costs are not certain but can possibly be replaced by a loaded move. Empty moves are always part of a gap and are therefore described by the opportunity costs.

The vehicle agent selects schedule option m with pickup time ω corresponding to the lowest bid price $b(\varphi, \Psi)$ and submits this bid with the expected pickup time ω to the auction.

4 Opportunity costs and value functions

The opportunity costs describe the loss in expected future revenues due to a job insertion, taking into account the stochastic job arrival process. That is, the vehicle agent does not know which jobs will arrive, when they arrive, and which auctions will be won. However the agent has information about past jobs and auctioning processes that can be used to estimate the attractiveness of a specific time slot at a specific location for the vehicle. We quantify the attractiveness using value functions. First, the value of the end-gap (shorthand end value) $V^e(i, \sigma, t)$, which is the expected revenue during a period t after arrival at schedule destination i at time σ from now on. Second, the gap value $V^g(i, j, \sigma, t)$ which is the expected value of all future moves in a gap defined by node i at the beginning of the gap, node j at the end of the gap, σ the time until the delivery at node i and t the length of the gap.

The parameter t in the end value function is the length of the end-gap. In Section 3.4.1 we introduced a planning horizon T . Initially we consider the remaining time $t = T - \sigma$ until the end of the planning horizon. Later we will iterate on this time t so that the remaining horizon will be shorter after each transition. In the remainder we use the word time-to-go to indicate the time σ from now till the arrival at the schedule destination i or the starting node i of the gap.

4.1 Opportunity costs calculation

Suppose that at current time θ an auction is started for a new job φ . The vehicle agent calculates the opportunity costs for this job as follows:

- Suppose that the new job is added to the end of the current schedule (at position $m = |\Psi|$) that is characterized by schedule destination i , time-to-go σ and a remaining planning horizon $T - \sigma$. Recall that we have chosen T much larger than the length of a schedule. Then, the earliest time at which the new job can be scheduled is given by the end time of the current schedule $\theta + \sigma$ plus possibly empty travel time to the origin of the new job $\tau^e(i, o(\varphi))$. The

opportunity costs $OC(\varphi, m, \omega, \Psi)$ of scheduling job φ at time ω ($\omega \geq \theta + \sigma + \tau^e(i, o(\varphi))$) are given by the old end value minus the new end value and possibly minus the value of a gap before this new job:

$$OC(\varphi, m, \omega, \Psi) = V^e(i, \sigma, T - \sigma) - V^g(i, o(\varphi), \sigma, \omega - \sigma - \theta) - V^e(d(\varphi), \omega + \tau_{o(\varphi)d(\varphi)}^f - \theta, T - \omega - \tau_{o(\varphi)d(\varphi)}^f) \quad (3)$$

- Suppose the new job is inserted in a gap $\{i, j, \sigma, t\}$ scheduled to be picked up at time ω . Again, the earliest time at which the new job can be scheduled is given by the start time of the gap $\theta + \sigma$ plus possible empty travel time to the origin of the new job $\tau^e(i, o(\varphi))$. The latest time at which the new job can be scheduled is given by the end time of the gap $\theta + \sigma + t$, possibly minus empty travel time from the destination of the new job to the origin of the next job in the schedule $\tau^e(d(\varphi), j)$. So, we have that $\theta + \sigma + \tau^e(i, o(\varphi)) \leq \omega \leq \theta + \sigma + t - \tau^e(d(\varphi), j)$. If the lower bound exceeds the upper bound, the job cannot be scheduled in the gap. The opportunity costs are given by the value of the old gap minus the value of two new gaps that are possibly created by this insertion (before and after the planned execution of φ):

$$OC(\varphi, m, \omega, \Psi) = V^g(i, j, \sigma, t) - V^g(i, o(\varphi), \sigma, \omega - \sigma - \theta) - V^g(d(\varphi), j, \omega + \tau_{o(\varphi)d(\varphi)}^f - \theta, \theta + \sigma + t - \omega - \tau_{o(\varphi)d(\varphi)}^f) \quad (4)$$

4.2 Value functions

In our multi-agent setting, the vehicle agent should find a sequence of decisions such that his expected trajectory of future states within gaps or at the end-gap yields the maximum expected reward. The values of these trajectories are given by the value functions V^e and V^g for which we can derive recursive relations. After discretization of the time, we obtain a Stochastic Dynamic Programming (SDP) recursion.

The recursive relations are described by four types of information; state space, decision set, transition probabilities and expected rewards. We use the state variables to capture all necessary information to value the future behavior of the system to be controlled. Derived directly from the value functions at the beginning of Section 4 we use $\{i, j, \sigma, t\}$ to describe the state within a gap and $\{i, \sigma, t\}$ for the state in an end-gap. We present the recursion for end values only because the gap values are derived in a similar manner. The necessary modifications for the gap values will be

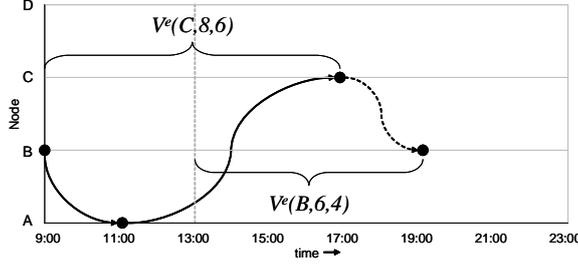


Figure 3: Transition of end states

discussed at the end of this section.

An illustration of the transition of states in the end-gap can be found in Figure 3. In this example, the current time is $\theta = 9$, the planning horizon is $T = 14$ and the vehicle schedule initially ends in location C at time 17, so the state at time $\theta = 9$ is given by $\{i, \sigma, t\} = \{C, 8, 6\}$. Suppose that the vehicle agent wins a next job at time 13 with origin C and destination B , to be picked up at time 17 and to be delivered at time 19. Then his state at the auctioning time 13 is given by $\{B, 6, 4\}$.

As mentioned in section 3.4.2, a vehicle has two decision moments: (i) at an announcement of a new order and (ii) after delivery of an order. At an announcement vehicles do not have to decide about their bids prices that are simply given by Equation 1. However they have to decide about the pickup and delivery time of this order. In our recursion, we assume that new jobs are scheduled as early as possible, and therefore are always added to the end of the schedule. We further assume in our recursion that if a vehicle finished an order and his schedule is not empty, he will drive directly towards the origin of the next order.

Whenever he finished a job and his schedule is empty he has to decide where to wait. Therefore we consider the decision $\delta(i) \in \mathcal{N}$ to move pro-actively to node $\delta(i)$ directly upon arrival at node i . If $\delta(i) = i$, the decision is to wait at the current node i . In the remainder we will use the shorthand notation δ instead of $\delta(i)$. Of course a vehicle will only make this decisions when it is waiting at some node ($\sigma = 0$). So if his current state is given by $\{i, 0, t\}$ then his next state after making decision δ is given by $\{\delta, \tau_{i\delta}^e, t - \tau_{i\delta}^e\}$.

To derive a recursion for the end values $V^e(i, \sigma, t)$, we consider the following three cases (1) we win a job during the time-to-go σ (2) otherwise we end up at node i , we decide to move pro-actively to node δ and we win a job during this time $\tau_{i\delta}^e$ (3) otherwise we end up at node δ and we wait until we win the next job over there. That is, we *only* face a decision δ if the vehicle arrives

at the schedule destination and no auction has been won in between. In the remainder of this section, we will show that we can express the corresponding value function for each case using a *partial* value function $V^p(i, \sigma, \eta, t)$, which is defined as the expected future revenue during a finite period $[\theta + \sigma, \theta + \sigma + t]$ for a truck ending in location i given he wins an order at time η during his time-to-go σ . The end value is then derived by combining the three partial value functions. First, we will derive the expression for case (1), next we will derive the expression for case (2) and (3).

In *case (1)*, the next job is won within the time-to-go σ , say at time $\theta + \eta$ ($0 \leq \eta \leq \sigma$). We define $p_{ikl}(\sigma - \eta)$ as the conditional probability that a truck ending in location i will have a trip from k to l as next job, given that the corresponding agent wins a job at time $\theta + \eta$. Here $\sigma - \eta$ is the time between winning an order and the earliest time he can actually start this order. Note that the probability of winning a specific order may depend on this time because orders have time-window restrictions and these time-windows may differ per route. We define the expected rewards of a job from k to l that is won at time $\theta + \eta$ by $r_{ikl}(\sigma - \eta)$. Now the next job from k to l ends at time $\sigma + \tau_{ik}^e + \tau_{kl}^f$ from now on. If $\tau_{ik}^e + \tau_{kl}^f \leq t$ (the job is handled within the time horizon $T = \sigma + t$), then we include the full profit in the value function. Otherwise we include a fraction $\frac{t}{\tau_{ik}^e + \tau_{kl}^f}$ corresponding to the percentage of the job that is completed within the time horizon. The next end value is the value at the new end location l at time $\sigma + \tau_{ik}^e + \tau_{kl}^f$, so the time to go from η on is $\sigma + \tau_{ik}^e + \tau_{kl}^f - \eta$ and the remaining time horizon is $\max\left\{t - \tau_{ik}^e - \tau_{kl}^f, 0\right\}$. By summation over all possible routes kl we get the following partial value function:

$$V^p(i, \sigma, \eta, t) = \sum_{\forall k, l \in N} p_{ikl}(\sigma - \eta) \left[\begin{array}{c} \alpha_{ikl}(t) r_{ikl}(\sigma - \eta) + \\ V^e\left(l, \sigma + \tau_{ik}^e + \tau_{kl}^f - \eta, t - \tau_{ik}^e - \tau_{kl}^f\right) \end{array} \right] \quad (5)$$

where $\alpha_{ikl}(t)$ is the fraction of the profit that we include in the value function. This fraction is given by:

$$\alpha_{ikl}(t) = \min\left\{\frac{t}{\tau_{ik}^e + \tau_{kl}^f}, 1\right\}, t > 0 \quad (6)$$

Obviously we put $\alpha_{ikl}(t) = 0$ if $t \leq 0$. By weighing over the time at which the next order is won, which we describe using a probability density function $f_{i\sigma}(\eta)$ and corresponding distribution function $F_{i\sigma}(\eta)$, we find that the first part of the value function for the end-gap is given by $\int_0^\sigma f_{i\sigma}(\eta) V^p(i, \sigma, \eta, t) d\eta$. Note that $f_{i\sigma}(\eta)$ is an exponential density because we assumed Poisson arrivals, see Section 3.2.

In *case (2) and (3)*, we do not win an order during the time-to-go σ . This happens with probability $1 - F_{i\sigma}(\sigma)$. Then we have to find the best option for the pro-active move to location δ

which takes a time $\tau_{i\delta}^e$ and which costs $c^t(\tau_{i\delta}^e)$ (if $\delta = i$ we wait at node i without costs for an pro-active move). Therefore, we compute the expected revenues and costs if we move pro-actively to δ and we select the option with maximum revenues in our recursion:

- In case (2), we win the next job at time $\theta + \eta'$ before arrival at node δ ($0 \leq \eta' \leq \tau_{i\delta}^e$). The remaining time horizon directly after arrival at the end-node δ is given by $t - \tau_{i\delta}^e$. Therefore, we find that the partial value function is given by $V^p(\delta, \tau_{i\delta}^e, \eta', t - \tau_{i\delta}^e)$, which we have to weigh over the time at which the next order is won, having density function $f_{\delta\tau_{i\delta}^e}(\eta')$.
- In case (3), we win the next job at time $\theta + \eta''$ after arrival at node δ ($\eta'' > \tau_{i\delta}^e$). The new time-to-go is therefore also η'' and the remaining time horizon after winning this new job is $t - \tau_{i\delta}^e - \eta''$. The partial value function for this case is given by $V^p(\delta, \eta'', \eta'', t - \tau_{i\delta}^e - \eta'')$. Again, we have to weigh this function over the time at which the next order is won, having density function $f_{\delta 0}(\eta'')$. Note that this probability density function is slightly different because we do not have the condition that we win during the time-to-go σ which is zero in this case.

By combining the value functions for the three cases, we find the following relation for the end value:

$$V^e(i, \sigma, t) = \int_0^\sigma f_{i\sigma}(\eta) V^p(i, \sigma, \eta, t) d\eta + \quad (7)$$

$$(1 - F_{i\sigma}(\sigma)) \max_{\delta} \left\{ \begin{array}{l} -c^t(\tau_{i\delta}^e) + \int_0^{\tau_{i\delta}^e} f_{\delta\tau_{i\delta}^e}(\eta') V^p(\delta, \tau_{i\delta}^e, \eta', t - \tau_{i\delta}^e) d\eta' \\ + (1 - F_{\delta\tau_{i\delta}^e}(\tau_{i\delta}^e)) \int_0^\infty f_{\delta 0}(\eta'') V^p(\delta, \eta'', \eta'', t - \tau_{i\delta}^e - \eta'') d\eta'' \end{array} \right\}$$

We have the boundary constraint that for $t \leq 0$ both $V^e(i, \sigma, t) = 0$ and $V^p(i, \sigma, \eta, t) = 0$.

The recursion for the gap value $V^g(i, j, \sigma, t)$ is derived in a similar manner with three exceptions. First the state is given by (i, j, σ, t) (see beginning of Section 4). This implies that the node j at the end of the gap has to be passed to the next iteration. Second, we use another boundary condition $V^g(i, j, \sigma, t) = -\infty$ if $t \leq 0$ and $i \neq j$, meaning that we have to arrive on time at the gap-destination j . Third, because transitions and corresponding revenues are dependent on the restrictions at the end of the gap, we also have to pass the end-node j and remaining gap time t in the transition functions $p_{ikl}(\sigma - \eta, j, t)$ and revenue functions $r_{ikl}(\sigma - \eta, j, t)$.

Unfortunately, the exact calculation of the value functions is very time-consuming. Two main reasons are: (1) the state space can be very large (2) it is not possible to solve the equations recursively (even not if we discretize time), because a value function depends on other value

functions with both a larger and smaller time-to-go σ . So in fact, we have to solve a large system of equations. Therefore, we present some approximations in the next section.

4.3 Value function approximations

Even with perfect knowledge of the states, solving the recursions of the previous section is a complex and time-consuming process. This is especially due to the integration over all winning moments and the large state space. Computation time is an issue because decisions have to be made in real time. In this section we describe three approximations: (1) discretization of time, (2) replacing the time-to-go σ with its expectation and (3) approximation of gap values by using the same parameters as for the end values. In section 4.3.1, we discuss the approximation of the end values. Next, we address the approximation of the gap values in Section 4.3.2.

4.3.1 End value approximation

As a first approximation we discretize time into intervals of length ε in order to reduce the problem to a stochastic dynamic programming problem. For ease of notation, we assume that the time dimension is chosen such that $\varepsilon = 1$. Then, we use a discrete probability density $q_{i\sigma}(\eta)$ instead of the continuous density $f_{i\sigma}(\eta)$, where $q_{i\sigma}(\eta)$ denotes the probability that the vehicle will receive an order in the time interval $(\theta + \eta, \theta + \eta + 1]$ if the schedule ends at location i and the time-to-go is σ .

In the next approximation we replace the time-to-go σ with an average $\bar{\sigma}$. Because the state is now independent of the time-to-go σ , we can reduce this state to $\{i, t\}$ and we are able to derive approximate value functions recursively. However we do not use this approximation for the first step of the recursive value functions. Therefore we distinguish between the value of the first uncertain move after the end of the current schedule and all further uncertain moves (an idea from Powell et al. (1988)). The first uncertain move occurs directly after arrival at node i , which is the schedule destination or the starting node of a gap. All further uncertain moves occur after that. This process is illustrated for the end value of a schedule in Figure 4 with current time 9:00. The destination of the schedule is node C with time-to-go 4 hours. Because we are looking at the end value, all moves before 13:00 are treated as certain moves. There is a probability (case 1) that we receive a next job before arrival at our current schedule destination. In this case we will have a first uncertain move directly after 13:00 consisting of a full move and possibly preceded by an empty move. Otherwise (case 2) we will make a pro-active move towards node B and receive a

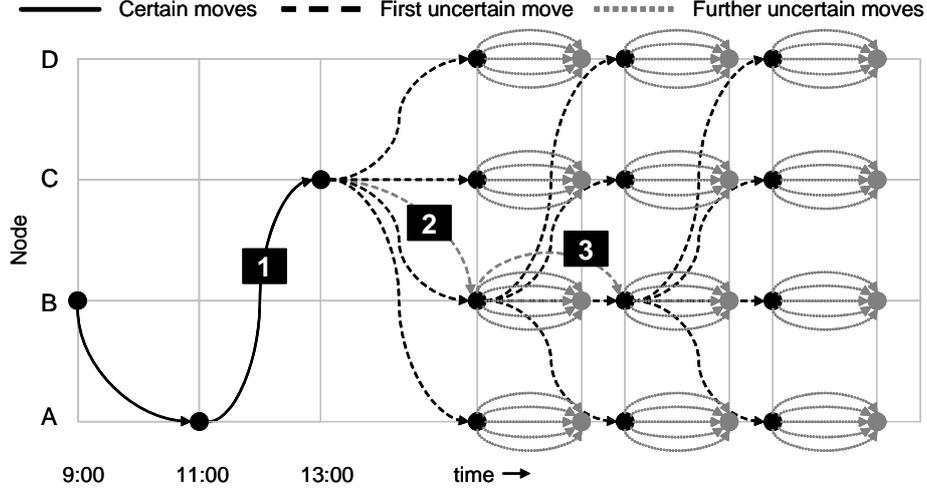


Figure 4: Approximation of end values

job during this pro-active move. Otherwise (case 3) we wait at node B until we win a job. After the first uncertain move we approximate all further possible moves, indicated by the grey dotted lines.

We denote the approximate value functions of all moves after the first uncertain move by \tilde{V}^e and \tilde{V}^g , for the end value and the gap value respectively. The new value function $\hat{V}^e(i, \sigma, t)$ for all uncertain moves, including the first uncertain move, is given by:

$$\hat{V}^e(i, \sigma, t) = \sum_{\eta=0}^{\sigma} q_{i\sigma}(\eta) \tilde{V}^p(i, \sigma, \eta, t) + \quad (8)$$

$$(1 - F_{i\sigma}(\sigma)) \max_{\delta} \left\{ \begin{array}{l} -c^t(\tau_{i\delta}^e) + \sum_{\eta'=0}^{\tau_{i\delta}^e} q_{\delta\tau_{i\delta}^e}(\eta') \tilde{V}^p(\delta, \tau_{i\delta}^e, \eta', t - \tau_{i\delta}^e) \\ + (1 - F_{i\tau_{i\delta}^e}(\tau_{i\delta}^e)) \sum_{\eta''=0}^{\infty} q_{\delta 0}(\eta'') \tilde{V}^p(\delta, \eta'', \eta'', t - \tau_{i\delta}^e - \eta'') \end{array} \right\}$$

where \tilde{V}^p is the approximate partial value function. This function is exactly the same as Equation 5 with the only exception that the end value $V^e(i, \sigma, t)$ is replaced by the approximate end value $\tilde{V}^e(i, t) = \hat{V}^e(i, \bar{\sigma}, t)$ for all uncertain moves after the first uncertain move.

The value of the first uncertain move only is given by $\hat{V}^e(i, \sigma, t)$ where the approximate end values \tilde{V}^e are set equal to zero. This first uncertain move is calculated rather precisely because we still use the actual time-to-go σ . The value $\hat{V}^e(i, \sigma, t)$ of all uncertain movements is easily calculated once we know the approximate end values $\tilde{V}^e(i, t)$. In these approximate end values we reduced the state space to $\{i, t\}$ which enable us to solve it by using a simple backwards

stochastic dynamic programming recursion. At each single (discrete) point in time we calculate the probability that we win an auction during the average time-to-go $\bar{\sigma}$. If we do not win an auction, we make a proactive move (ending multiple time-units ahead) or we wait a single period at this node. We now have the following backwards recursion:

Algorithm 1 *Calculating the approximate end values*

init:

 given a planning horizon T

$$\tilde{V}^e(i, 0) = 0 \quad \forall i$$

for $t = 1$ to T do

 for $i = 1..N$ do

$$\begin{aligned} \tilde{V}^e(i, t) &= \sum_{\eta=0}^{\sigma} q_{i\bar{\sigma}}(\eta) \tilde{V}^p(i, \bar{\sigma}, \eta, t) + (1 - F_{i\bar{\sigma}}(\bar{\sigma})) \max_{\delta} \left\{ -c^t(\tau_{i\delta}^e) + \tilde{V}^e(\delta, t - \max(\tau_{i\delta}^e, 1)) \right\} \\ \tilde{V}^p(i, \bar{\sigma}, \eta, t) &= \sum_{k,l \in N} p_{ikl}(\bar{\sigma} - \eta) \left[\alpha_{ikl}(t) r_{ikl}(\bar{\sigma} - \eta) + \tilde{V}^e(l, t - \tau_{ik}^e - \tau_{kl}^f) \right] \end{aligned}$$

 end;

end;

To summarize, the approximate end values $\tilde{V}^e(i, t)$ provide the expected revenue during a period of length t after arrival at node i . We use this approximation only for the uncertain moves after the first uncertain move. The value of all uncertain moves is given by $\hat{V}^e(i, \sigma, t)$, where we use $\tilde{V}^e(i, t)$ in the approximate partial value function.

4.3.2 Gap value approximation

We approximate the gap values analogously to the end values. First, we replace the gap value function $V^g(i, j, \sigma, t)$ by $\hat{V}^g(i, j, \sigma, t)$ where time is discretized. For all uncertain moves after the first uncertain move within a gap we use the approximate gap values $\tilde{V}^g(i, j, t) = \hat{V}^g(i, j, \bar{\sigma}, t)$ where we replace the time-to-go σ by $\bar{\sigma}$.

At the end of Section 4 we mentioned that the transition and revenue functions for gap values differ from the end values because we have to take into account that a vehicle should end at node j in a remaining gap time t . This is a complication for the recursive expressions, because we have to store more information (opportunity costs for all possible states) and it requires more computation time. Therefore, as an approximation, we ignore the gap restrictions and we approximate the transition probabilities $p_{ikl}(\sigma - \eta, j, t)$ by $p_{ikl}(\sigma - \eta)$ and the revenues $r_{ikl}(\sigma - \eta, j, t)$ by $r_{ikl}(\sigma - \eta)$. This approximation also provides some notational convenience because the gap value function can

now be described by the same parameters as the end value function. Improvement by omitting these approximations is straightforward as we will see in Section 5.2.

A drawback of our approximation is that we may overestimate the winning probabilities, especially if we face a transition that involves a significant risk that we will violate the restrictions at the end of the gap. Then, it is possible that we make a non-profitable transition, e.g., if we had taken the restrictions at the end of the gap into account we would not have made a certain transition. To overcome this we multiply the transition probabilities with a decision variable δ_{kl}^a . This variable equals 1 if we accept the transition from k to l and otherwise it is zero. The logic behind this decision variable is that a vehicle always has the option to wait a single time unit if this seems to be more profitable than making this non-profitable transition. The approximate gap values are given by:

Algorithm 2 *Calculating the approximate gap values*

init:

given an end-node j and gap length t

$$\tilde{V}^g(i, j, s) = -\infty \text{ for all } i \neq j \text{ with } s < 0 \text{ and } \tilde{V}^g(j, j, 0) = 0$$

for $s = 1$ to t do

for $i = 1..N$ do

$$\tilde{V}^g(i, j, s) = \max_{\delta_{kl}^a, \forall k, l \in \mathcal{N}} \left\{ \begin{array}{l} \sum_{\eta=0}^{\sigma} q_{i\bar{\sigma}}(\eta) \tilde{V}^p(i, j, \bar{\sigma}, \eta, t) + \\ (1 - F_{i\bar{\sigma}}(\bar{\sigma}) + u(i, \eta)) \max_{\delta} \left(\begin{array}{l} -c^t(\tau_{i\bar{\sigma}}^e) + \\ \tilde{V}^g(\delta, j, s - \max(\tau_{i\bar{\sigma}}^e, 1)) \end{array} \right) \end{array} \right\}$$

$$\tilde{V}^p(i, j, \bar{\sigma}, \eta, t) = \sum_{k, l \in \mathcal{N}} \delta_{kl}^a p_{ikl}(\bar{\sigma} - \eta) \left[\alpha_{ikl}(t) r_{ikl}(\bar{\sigma} - \eta) + \tilde{V}^g(l, j, t - \tau_{ik}^e - \tau_{kl}^f) \right]$$

$$u(i, \eta) = q_{i\bar{\sigma}}(\eta < \bar{\sigma}) \left(1 - \sum_{k, l \in \mathcal{N}} \delta_{kl}^a p_{ikl}(\bar{\sigma} - \eta) \right)$$

end;

end;

where $u(i, \eta)$ is the probability that we did not accept a transition for a job won at time η after node i .

5 Parameter estimation

Because we have deterministic travel times, we assume that all vehicles are aware of the loaded travel times τ_{ij}^f and empty travel times τ_{ij}^e for all routes $i, j \in \mathcal{N}$. This leaves us with the following parameters that we need to estimate in order to calculate the (approximate) value functions:

- The average time-to-go $\bar{\sigma}$
- The conditional probability $p_{ikl}(\sigma - \eta)$ that a truck ending in location i will have a trip from k to l as next job, given that the corresponding agent wins a job at time $\theta + \eta < \theta + \sigma$.
- The expected rewards $r_{ikl}(\sigma - \eta)$ of a job from k to l that is won at time $\theta + \eta < \theta + \sigma$ with starting node i .
- The conditional probability $q_{i\sigma}(\eta)$ that we win a new job at time $\theta + \eta < \theta + \sigma$ if the schedule ends at node i , given that we win a job during the time-to-go σ .
- The distribution function $F_{i\sigma}(\sigma)$ that we win a new job during the time-to-go σ if the schedule ends at node i .

We estimate these parameters and functions based on historic data. A possibility is to store all historic waiting times, revenues, travel times and transition percentages for all possible states. Even when these parameters do not depend on the time-to-go σ , we must store a lot of information. Therefore we propose to estimate these parameters based on auction data for certain routes and the job arrival intensity for these routes. To enable the vehicle agents to estimate these parameters for their bid price calculation, they will receive the following information from their fleet manager:

- The sample mean \bar{x}_{ij} and sample variance s_{ij}^2 of all observations of the winning price for all routes $i, j \in \mathcal{N}$
- Average job arrival intensities λ_{ij} for all routes $i, j \in \mathcal{N}$
- Average time-window length z_{ij} for jobs on all routes $i, j \in \mathcal{N}$
- Average time-to-go $\bar{\sigma}$

These parameters are estimated by the fleet manager because he just receives more information than a single vehicle. The arrival intensities λ_{ij} and time-window lengths z_{ij} are estimated based on the job announcement characteristics by taking the average of past observations. The average time-to-go $\bar{\sigma}$ is estimated as the average time between winning and picking up an order. This value coincides with the average length of a schedule in the value function recursions because we assumed there that all orders are scheduled at the end of the schedule.

In the next section we describe how the vehicles estimate the distribution of the lowest bid using the sample mean and variance provided by their fleet manager. In Section 5.2 we describe how

the vehicles calculate the transition probabilities $p_{ikl}(\sigma - \eta)$ and expected revenues $r_{ikl}(\sigma - \eta)$. Calculation of the winning probabilities $q_{i\sigma}(\eta)$ and distribution $F_{i\sigma}(\sigma)$ of winning moments is given in Section 5.3.

5.1 Distribution of lowest bids

The estimation of the distribution parameters of the lowest bids depends on the structure of the auction. Here, we use an open second-price auction, where only the winning price, i.e. the one but lowest bid, is published. To estimate the distribution of the lowest bid, fleet managers can use information about all winning prices together with their own bid history. Because your own bids provide little information about the bids of your competitors we only use information about the winning prices. This causes a problem, because we need the probability distribution of the *lowest* price whereas we only have observations of the *one but lowest* price. In this section, we discuss how we deal with this problem.

We will use the theory of the so-called Extreme Value Distributions (EVD), being a class of probability distributions for the order statistics of a large set of random observations from the same (arbitrary) distribution. Particularly, the EVD cover the minimum and maximum value, but also a limiting distribution for the one but lowest (highest) observation is known. These limiting distributions have the same parameter set. Therefore, we estimate the parameters of the limiting distribution for the one but lowest bid and we use these estimated parameters for the limiting distribution of the minimum bid. Below, we elaborate this approach in formulas.

Suppose that the bids b_i for a single job from competitor i ($i = 1..n$) are independent and identically distributed with a cumulative distribution function $H(x)$. We denote the probability distribution functions of the corresponding order statistics by $H_i(x)$. The probability distribution of the first two order statistics are given by the following expressions:

$$\begin{aligned} H_1(x) &= 1 - \Pr(b_1 > x, b_2 > x, \dots, b_n > x) = 1 - (1 - H(x))^n \\ H_2(x) &= nH^{n-1}(x)[1 - H(x)] + H^n(x) \end{aligned} \tag{9}$$

Except for special cases, it is not possible to express these distributions as closed form expression with parameters that can easily be estimated. It is shown in (Gumbel 1958) that for any well-behaved initial distribution (i.e., $H(x)$ is continuous and has an inverse), limiting distributions

for $n \rightarrow \infty$ can be derived. In this paper, we use the Gumbel distribution which only request that the tail of the distribution of $H(x)$ declines exponentially (normal, log-normal, exponential, and gamma). The Gumbel distribution for the minimum value in a sample is given by:

$$G_1(x) = 1 - e^{-e^{\frac{x-\alpha}{\beta}}} \quad (10)$$

where α and $\beta > 0$ are the location and scale parameters. The Gumbel distribution for the second lowest value has exactly the same parameters and is given by:

$$G_2(x) = 1 - \left(e^{-e^{\frac{x-\alpha}{\beta}}} \left(1 + e^{\frac{x-\alpha}{\beta}} \right) \right) \quad (11)$$

We will use G_1 and G_2 to approximate H_1 and H_2 respectively. We see that then we can estimate the parameters α and β using observations from $G_2(x)$ and insert these parameters in $G_1(x)$. Various statistical methods can be used to estimate α and β , depending on the observed data. In case of an open auction we can simply use the Method of Moments. From the moments of the standard Gumbel distribution $G_n(x)$, see Reiss and Thomas (1997), we derive the following values for the location and scale parameter of $G_2(x)$:

$$\begin{aligned} \beta_{ij} &= \sqrt{\frac{s_{ij}^2}{\frac{1}{6}\pi^2 - 1}} \\ \alpha_{ij} &= \bar{x}_{ij} + \beta_{ij}(\gamma - 1) \end{aligned} \quad (12)$$

where $\gamma = 0.577216\dots$ is Euler's constant, \bar{x}_{ij} and s_{ij}^2 are respectively the sample mean and sample variance of all historical winning prices (the second-lowest bids) for orders on route i, j . The sample mean and sample variance are provided to the vehicle agents by their fleet manager so that they are able to calculate the location and scale parameters of the Gumbel distribution of the lowest bid. In the remainder we will indicate this distribution by $H_{ij}^{\min}(x)$ which in fact describes the probability that a vehicle will loose an auction given his bid price x .

Note that we used the assumption that bids in successive rounds are independent and identically distributed (iid) random variables. We may question whether that assumption is realistic, because the system state (state of the vehicles) will be quite similar in successive auctioning rounds unless the order arrival frequency is very low. However, Reiss and Thomas (1997) states that even if the distributions $H(x)$ are not exactly known or the iid condition of the bids fails, then H_1 may still be an accurate approximation of the actual distribution of the minimum. Because H_2 can be

expressed as a function of H_1 the same holds for the distribution of the second lowest bid.

5.2 Estimating revenues and transition probabilities

In order to estimate the revenues and transition probabilities, the vehicle agent uses so-called winning intensities. The winning intensities provide for all routes the intensity at which the fleet manager expects to win jobs given a certain state. We define the winning intensity $\xi_{ikl}(\sigma)$ as the mean number of winning jobs per time unit from k to l after arrival at node i with time-to go σ . The winning intensities are given by:

$$\xi_{ikl}(\sigma) = \lambda_{kl} \cdot p_{ikl}^{win}(\sigma) \quad (13)$$

where $p_{ikl}^{win}(\sigma)$ is the probability of winning a job from k to l with starting node i and time-to-go σ , given by

$$p_{ikl}^{win}(\sigma) = 1 - H_{kl}^{\min}(b_{ikl}(\sigma)) \quad (14)$$

Here $b_{ikl}(\sigma)$ is the bid price for the vehicle given a job on route kl , location i and time-to-go σ . This bid price consists of the direct costs $c_{kl}(\sigma)$ and the opportunity costs $OC_{ikl}(\sigma)$:

$$b_{ikl}(\sigma) = c_{kl}(\sigma) + OC_{ikl}(\sigma) \quad (15)$$

The direct costs (see Equation 2) are given by:

$$c_{kl}(\sigma) = c^t \left(\tau_{kl}^f \right) + c^p (\sigma - z_{kl})^+ \quad (16)$$

The opportunity costs however are not known yet because this whole approach is focused on finding them. Therefore we use an estimate based on previous opportunity costs charged in this state. In case of end values, we approximate the opportunity costs by:

$$OC_{ikl}(\sigma) = \tilde{V}^e(i, \sigma) + c^t (\tau_{ik}^e) - \tilde{V}^e \left(l, \sigma + \tau_{ik}^e + \tau_{kl}^f \right) \quad (17)$$

Here we used the approximate value functions to reduce computation time. A difference with original opportunity costs function (Equation 3) is that the value of the gap before the new job is replaced by the costs for an empty move. This because the time for an empty move can not be used for a loaded move.

In Section 4.3 we mentioned that we use the same parameters for the gap value function. If we choose to include the gap restrictions in our transition- and revenue functions, then all functions above are also dependent on the end-node j and the gap length t . However, this change will only affect the calculation of the opportunity costs, which in case of gap values are given by:

$$OC_{ikl}(j, \sigma, t) = V^g(i, j, \sigma, t) + c^t(\tau_{ik}^e) - V^g(l, j, \sigma + \tau_{ik}^e + \tau_{kl}^f, t - \tau_{ik}^e - \tau_{kl}^f) \quad (18)$$

In the end value function (Equation 7) we integrate the expected revenues over all possible winning moments η . Given that we win a job during time-to-go σ at time η , the remaining time-to-go is $\gamma = \sigma - \eta$. Then the probability $p_{ikl}(\gamma)$ that the winning job has origin k and destination l given location i is given by:

$$p_{ikl}(\gamma) = \frac{\xi_{ikl}(\gamma)}{\sum_{kl} \xi_{ikl}(\gamma)} \quad (19)$$

The expected revenue as a function of the winning time-to-go γ is given by the difference between the expected lowest bid of the competitors (given that the own bid is lower) and the costs that are made:

$$r_{ikl}(\gamma) = \frac{1}{1 - H_{ikl}^{\min}(b_{ikl}(\gamma))} \int_{x=b_{ikl}(\gamma)}^{\infty} x dH_{ikl}^{\min}(x) - c_{kl}(\gamma) - c^t(\tau_{ik}^e) \quad (20)$$

In our simulation experiments (see Section 7) we solve this function numerically.

5.3 Distribution of winning moments

We described the winning moments η by a distribution function $F_{i\sigma}(\eta)$. Difficulty is that the winning moments follow a so-called nonhomogeneous Poisson process (NHPP), see (Ross 2003). When time-windows, penalty costs or travel costs differ per route, than also transition probabilities will be time dependent. At different points in time, different jobs will have the highest winning probability. To ease our notation we introduce the following definition:

$$\Lambda_{i\sigma}(t) = \int_0^t \sum_{kl} \xi_{ikl}(u) (\sigma - u) du \quad (21)$$

The value $\Lambda_{i\sigma}(t)$ is called the rate function of the nonhomogeneous Poisson process because the counting process of winning jobs can be described by a Poisson process with mean $\Lambda_{i\sigma}(t)$. The distribution function of the random amount of time η until the first time we win an auction, given

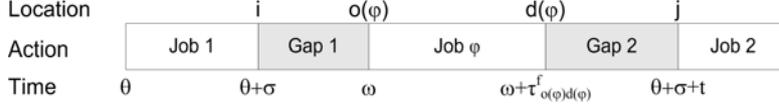


Figure 5: Job insertion in a gap

we win this order during time-to-go σ with starting node i , is given by:

$$F_{i\sigma}(\eta) = \frac{1 - e^{-\Lambda_{i\sigma}(\eta)}}{1 - e^{-\Lambda_{i\sigma}(\sigma)}}, \quad 0 \leq \eta \leq \sigma, \sigma > 0 \quad (22)$$

For the special case with time-to-go zero, the winning moments follow a homogeneous Poisson process. Then we have the following distribution function:

$$F_{i0}(\eta) = 1 - e^{-\eta \sum_{kl} \xi_{ikl}^{(0)}} \quad (23)$$

Because we discretized time we use the following discrete probabilities:

$$q_{i\sigma}(\eta) = F_{i\sigma}\left(\eta + \frac{\varepsilon}{2}\right) - F_{i\sigma}\left(\eta - \frac{\varepsilon}{2}\right) \quad (24)$$

Recall that we standardize the time dimension by choosing $\varepsilon = 1$.

6 Waiting strategies

Until now, we assumed that a new job is scheduled as early as possible. However, this does not need to be optimal. The vehicle agents may use a more advanced waiting strategy consisting of two types of waiting decisions, namely (1) the choice for a suitable pick-up time for a given job and (2) a pro-active move policy that tells the vehicle agent where to wait after arrival at some node and for how long.

Decisions regarding the pick-up time for jobs have to be taken while calculating a bid. Extra waiting time can be incorporated in anticipation of future job insertions. We consider the following two situations: (1) the new job will be scheduled in a gap and (2) the new job will be scheduled at the end of the current schedule.

Consider a load φ and a gap with start-node i , end-node j , time-to-go σ and gap length t (see Figure 5). When the new order is inserted in this gap, we possibly get two new gaps. The optimal pickup-time ω^* is calculated by maximizing the value of these new gaps:

$$\omega^* = \max_{\omega} \left\{ \begin{array}{l} \hat{V}^g(i, o(\varphi), \sigma, \omega - \sigma - \theta) - c^p(\max(0, \omega - e(\varphi))) + \\ \hat{V}^g(d(\varphi), j, \omega + \tau_{o(\varphi)d(\varphi)}^f, t - \omega + \sigma + \theta - \tau_{o(\varphi)d(\varphi)}^f) \end{array} \right\} \quad (25)$$

where we use discrete time intervals with length ε for ω . Note that ω lies between the earliest pickup-time $\theta + \sigma + \tau_{i,o(\varphi)}^e$ and the latest pickup-time $\theta + \sigma + t - \tau_{o(\varphi)d(\varphi)}^f - \tau_{d(\varphi),j}^e$. The pickup-time ω^* for a job at the end of the schedule is calculated in a similar manner:

$$\omega^* = \max_{\omega} \left\{ \begin{array}{l} \hat{V}^g(i, o(\varphi), \sigma, \omega - \sigma - \theta) - c^p(\max(0, \omega - e(\varphi))) + \\ \hat{V}^e(d(\varphi), \omega + \tau_{o(\varphi)d(\varphi)}^f) \end{array} \right\} \quad (26)$$

where $\omega \geq \theta + \sigma + \tau_{i,o(\varphi)}^e$.

To reduce computation time, we can use the approximations \tilde{V}^e and \tilde{V}^g . Therefore we only need to calculate $\tilde{V}^g(i, o(\varphi), t - \tau_{o(\varphi)d(\varphi)}^f - \tau_{d(\varphi),j}^e)$, $\tilde{V}^g(d(\varphi), j, t - \tau_{i,o(\varphi)}^e - \tau_{o(\varphi)d(\varphi)}^f)$, and $\tilde{V}^e(d(\varphi), T)$, and store all intermediate values from the dynamic programming recursions.

Whenever a vehicle becomes idle, it has to decide upon a pro-active move. This decision can simply be found by using the parts of the value functions for which we assume we have to wait. Using Equation 8, we find that the best waiting decision $\delta^*(i)$ at node i in case of an end-gap is given by:

$$\delta^*(i) = \max_{\delta \in \mathcal{N}} \left\{ \begin{array}{l} -c^t(\tau_{i\delta}^e) + \sum_{\eta'=0}^{\tau_{i\delta}^e} p_{\delta} \tau_{i\delta}^e(\eta') \tilde{V}^p(\delta, \tau_{i\delta}^e, \eta', t - \tau_{i\delta}^e) \\ + (1 - P_i(\tau_{i\delta}^e)) \sum_{\eta''=0}^{\infty} p_{\delta 0}(\eta'') \tilde{V}^p(\delta, \eta'', \eta'', t - \tau_{i\delta}^e - \eta'') \end{array} \right\} \quad (27)$$

If we are at the end of our schedule and we have a stationary order arrival process, than this decision has to be taken only once. If there is a more profitable node δ than the current node i , then it will move directly towards this node, otherwise it will wait at node i until the next job.

In case of a gap, this decision has to be taken at every moment in the gap for which the vehicle is not active. However, an optimal waiting strategy can be found in advance using the approximate gap values which are already calculated with Algorithm 2. In this recursion we established the best waiting decision for each node i and remaining gap length δ . Searching these values will provide the time at which the vehicle has to move to the end-node of the gap, given that it did not receive another job in this gap.

7 Simulation

In this section we discuss the results of a concise simulation study. The goal of this study is to provide some insight in the performance of opportunity based bid pricing compared to a pricing strategy where only the direct costs are taken into account. An extensive study on the effects in different market settings (i.e. number of companies, company size and different strategies of players) is out of the scope of this paper and subject for further research.

We compare opportunity based bid pricing to a naïve pricing strategy where only the direct costs of job insertions is taken into account. Because this naïve strategy is not able to value opportunities, we use a simple waiting strategy where new jobs are scheduled as early as possible at a certain place in the schedule.

When we use fixed contracts with this naïve strategy, new jobs will always be added to the end of the schedule. Therefore we will also consider flexible contract agreements in order to provide a 'fair' conclusion about the benefit of using opportunity costs. With flexible contracts vehicles are not committed to agreed pickup and delivery times, so that the scheduled times can be modified later on, possibly at the costs of additional penalties due to time-window violation. Because we assumed fixed contracts throughout this paper we have to modify the opportunity based bid pricing approach for flexible contracts. We do this by simply ignoring the gap values, i.e. the price of a new job insertion is given by the change in direct costs and the change in the end values.

An important aspect of the implementation of the agent-based planning concept is the market situation. We consider two simplified market structures. First an open network where we apply the opportunity based bid pricing approach to a single company while the other companies use the naïve pricing strategy. Second a closed network where all companies will use the same planning method.

The opportunity based bid pricing approach was developed for open markets where we apply this strategy to an individual company and assume that it does not affect the behavior of the other companies. This is a basic assumption of our approach because current decisions (bid price calculation) are based on historical observations of auction data. The closed market structure can be seen as an internal application of our approach. For example if we apply our approach to a closed consortium of transportation companies or even to a single transportation company where the trucks compete against each other. When all players are using the same value functions this may have serious influence on the prices in the system. This is especially true if the behavior of all players is exactly the same, that is, if they use the same historical data, update their parameters

Factor	Range
Contracts	fixed / flexible
Using opportunity costs	yes / no
Market	open / closed
Time-windows	short / long

Table 1: Experimental factors

at the same time etc. Then the winning intensities of all players will remain constant. We include this behavior by setting the opportunity costs in the expected bid price (Equation 15) in the dynamic programming recursions equal to zero.

7.1 Settings

We consider a raster network with 9 (3x3) nodes. The horizontal and vertical distances between nodes are 25 km. The travel distances are Euclidian and we use three companies each having three vehicles. Orders arrive according to a Poisson process with mean interarrival time of 9 minutes. The network is unbalanced in the sense that some nodes are more popular than others. In our simulation model, we proceed as follows. For a given order, we first generate an origin and next a destination that is different from the origin. The probability of being an origin for nodes on row 1 is 5 times higher than row 2 and 25 times higher than row 3. Given an origin node, all destination nodes are generated with equal probability.

In our simulation experiments, we vary the experimental factors as shown in Table 1. The short (long) time-window is 4 (6) hours. The travel speed of vehicles is 72 km/hour, the travel cost function is given by $c^t(t) = t$ and the penalty costs function by $c^p(t) = 10t$. The loading and unloading times are both 5 minutes per job. For the dynamic programming recursions we discretize time into periods of 1 minute.

We use a replication / deletion approach for our simulations, cf. (Law and Kelton 2000), where each experiment consists of a number of replications (each with different seeds) of 16 days, each including a one-day warm-up period. To determine the number of replications we consider the percentage of driving loaded (DL) and the service levels (SL) of all experiments. The maximum number of replications needed with a confidence level of 95% and a relative error of 5% (for both performance indicators) is 10. To facilitate comparison we use 10 replications for all experiments.

7.2 Results

We use the following performance indicators: (i) the percentage of driving loaded (DL) which is the percentage of the total distance we have travelled loaded and (ii) the service level (SL) which is the percentage of jobs that are delivered on time. In the open network setting we also use the relative profit (RP) which is the profit of the smart company compared to the average profit of his competitors. The results for the closed network can be found in Table 2.

OC	Contract	Time-windows	DL	SL
false	flexible	short	61	90
true	flexible	short	61	98
false	flexible	long	62	97
true	flexible	long	62	100
false	fixed	short	59	87
true	fixed	short	60	94
false	fixed	long	59	96
true	fixed	long	60	99

Table 2: Triangular network with closed market

We find that using opportunity costs always yields an increase in service level. This is caused by three things. First of all the workload is spread more equally among the vehicles. The average schedule length of vehicles is roughly 20% lower for all 4 experimental settings. Second, vehicle agents tend to schedule unattractive jobs (jobs with high probability of an empty move towards the origin or from the destination node) relatively later in the schedule. This leads to an increase in probability that the empty move can be replaced by a new loaded move. For example orders to a node on the third row (the row with lowest probability of becoming origin of an order) are scheduled with a time-to-go 40% longer than for jobs to a node on the first row. Third, in case of fixed contracts, the vehicles create gaps to avoid empty moves. With short time-windows we have 10 gaps with 8 filled. In case of long time-windows there 83 gaps created that are filled with 118 orders. Approximately 90% of the gaps are created before orders leaving from a node on the first row that are scheduled a long time-to-go in advance.

The results of the open market setting can be found in Table 3. The columns DL and SL consists of two values, the first is the value for the single company that apply opportunity based bid pricing, while the second value is the average value for the other two companies.

From these results we see that using opportunity based bid pricing always results in a better performance. There is always an increase in capacity utilization, service level and profits. Especially in case of long time-windows, the average profit is up to 4.16 times higher than the average

Contract	Time-windows	DL	SL	RP
flexible	short	69 / 59	98 / 95	195 %
flexible	long	69 / 59	100 / 98	362 %
fixed	short	68 / 57	92 / 87	124 %
fixed	long	68 / 57	100 / 99	416 %

Table 3: Open market settings for both networks

profit of the competitors.

The company using opportunity costs effectively uses gaps to reduce empty moves, thereby increasing capacity utilization and profits. In case of short time-windows this company created 29 gaps that are filled with 24 orders. In case of long time-windows there are 39 gaps filled with 36 orders. However, the main reason for the high profits of the company using opportunity costs is that it picks out the most profitable jobs. Because the 'smart' company is driving more effectively, it is able to handle more orders. For example, in case of fixed contracts with long time-windows, the 'smart' company has won 26% more orders. Given the relative profit of 416%, the profit per order is much higher. In this experiment we see that the 'smart' company wins 6 times more orders towards a node on row 1 and 5 times less to a node on row 3. Moreover, 84% of its transport takes place on the first row.

8 Relaxation of assumptions

Throughout this paper we made several assumptions. In this section we relax some of these assumptions by successively introducing look-ahead, closed auctions and gap evaluation in case of flexible contracts.

We defined the look-ahead of job φ as the time between announcement $a(\varphi)$ and earliest pickup time $s(\varphi)$. Suppose that the look-ahead for a job on route i, j can be estimated by a_{ij} . In order to incorporate look-ahead we need two modifications. First, the most suitable pickup time ω will now be $\omega' = \max(\omega, \theta + a_{ij})$. Second, we have to reckon with lookahead in the dynamic programming recursions. For example, the new time-to-go after accepting a new job has to be modified from $\sigma + \tau_{ik}^e + \tau_{kl}^f$ to $\max(a_{kl}, \sigma + \tau_{ik}^e) + \tau_{kl}^f$.

Throughout this paper we assumed that all bidders receive information about the winning prices. In case of a closed auction, bidders only receive information about whether they win or lose an auction. In fact, we have censored data for the distribution of the minimum price. In principle, we can handle this using e.g. Maximum Likelihood estimation.

Gap evaluation in combination with flexible contract agreements is slightly more difficult. Using the flexible contracts, new jobs can be inserted at any place at any time in the current schedule, possibly rescheduling other jobs. Therefore we have to evaluate all possible gap lengths between all successive jobs. To reduce computation time we propose the following. Given a schedule with m jobs, a new job can be inserted at m possible places. We indicate each alternative schedule m by Ψ^m , the current schedule of a vehicle by Ψ^* , and the value of a schedule Ψ is given by $W(\Psi)$. Then the bid price is given by the difference between the value of the best proposed schedule and the value of his current schedule:

$$b = W(\Psi^*) - \max_m W(\Psi^m) \quad (28)$$

The value of a schedule is given by the sum of the direct costs of all jobs, all gap values and the end value. Although all gaps are flexible, the vehicle agent will calculate the most profitable pickup time ω_i for all jobs i . These pickup times ω_i can be found using the expressions from Section 6, where the possible pickup times can be bounded by for example the latest pickup times for jobs $i + 1, \dots, |\Psi|$.

9 Conclusions

In this paper, we presented an opportunity based bidding concept. This method makes it possible to determine the value a schedule, more specifically the value of gaps in the schedule and the value of a schedule destination. This enables us to calculate the opportunity costs which are defined as the loss in expected future revenues due to a new job insertion. By including this value in our bid price we not only cover the direct costs of an insertion, but also its future implications. Therefore we prevent less profitable moves and increase our opportunities by better prepositioning of vehicles. In addition, the value functions that are used to calculate the opportunity costs can also be used to determine a waiting strategy.

From our simulation experiments, we conclude that an individual player using opportunity based bid pricing will perform significantly better than other players who use a more naïve pricing strategy. This holds not only in terms of profits, but also higher vehicle utilization and higher service levels. But also in closed environments (internal use of our approach), opportunity based bid-pricing can result in higher service levels. We explain these results from the following behavior. First, the vehicle agents tend to schedule unattractive jobs later increasing the probability of

combining this job with another job. Second, in case of fixed contract agreements, the vehicle agents tend to create gaps before unattractive jobs, possibly reducing empty moves. Third, in case of open markets, the vehicle agents tend to select out the most profitable jobs.

Further research includes an extensive simulation study on the emergent behavior of agent-based transportation networks. Therefore we will consider different market settings, other auction mechanisms (first price auctions and combinatorial auctions) and different strategies for different fleet managers (for example specialization in certain regions). Furthermore, some interesting model extensions need to be examined, such as stochastic travel times, learning with feedback and using information on the number of vehicles per nodes for bid pricing. Also, faster methods are useful for real-time decision making. Regarding the latter, we might think of functional approximations of the discrete value functions and aggregation methods in large networks, for example by using value functions per region (cluster of locations) instead of for each separate location.

References

- Carvalho, T.A. and W.B. Powell (2000). A multiplier adjustment method for dynamic resource allocation problems. *Transportation Science* 34, 150–164.
- Caserta, S. (2002). *Extreme Values in Auctions and Risk Analysis*. Ph. D. thesis, Erasmus University Rotterdam.
- Desrosiers, J., Y. Dumas, M.M. Solomon, and F. Soumis (1995). Time constrained routing and scheduling. In M. Ball, T. Magnanti, C. Monma, and G. Nemhauser (Eds.), *Network Routing*, Volume 8 of *Handbooks in Operations Research and Management Science*, pp. 35–139. Amsterdam, The Netherlands: Elsevier.
- Figliozzi, M.A., H.S. Mahmassani, and P. Jaillet (2003). Framework for study of carrier strategies in auction-based transportation marketplace. *Transportation Research Record* 1854, 162–170.
- Figliozzi, M.A., H.S. Mahmassani, and P. Jaillet (2004). Competitive performance assessment of dynamic vehicle routing technologies using sequential auctions. *Transportation Research Record* 1882, 10–18.
- Fischer, K., J.P. Muller, and M. Pischel (1996). Cooperative transportation scheduling: an application domain for dai. *Journal of Applied Artificial Intelligence. Special issue on Intelligent Agents* 10(1), 1–33.

- Fischer, M. (1995). Vehicle routing. In M. Ball, T. Magnanti, C. Monma, and G. Nemhauser (Eds.), *Network Routing*, Volume 8 of *Handbooks in Operations Research and Management Science*, pp. 1–33. Amsterdam, The Netherlands: Elsevier.
- Gendreau, M. and J.Y. Potvin (1998). Dynamic vehicle routing and dispatching. In T. Crainic and G. Laporte (Eds.), *Fleet Management and Logistics*, pp. 127–157. Kluwer, Academic Publishers.
- Giani, G., F. Guerriero, G. Laporte, and R. Musmanno (2003). Real time vehicle routing: solution concepts, algorithms and parallel computing strategies. *European Journal of Operational Research* 151, 1–11.
- Godfrey, G.A. and W.B. Powell (2002). An adaptive dynamic programming algorithm for dynamic fleet management, i: Single period travel times. *Transportation Science* 36(1), 21–39.
- Gumbel, E.J. (1958). *Statistics of extremes*. New York: Columbia University Press.
- Heijden, M.C. van der, A. van Harten, M.J.R. Ebben, Y.S. Saanen, E. Valentin, and A. Verbraeck (2002). Using simulation to design an automated underground system for transporting freight around schiphol airport. *Interfaces* 32(4), 1–19.
- Hoen, P.J. 't and J.A. la Poutré (2004). A decommitment strategy in a competitive multi-agent transportation setting. In D. P. P. Faratin and J. Rodriguez-Aguilar (Eds.), *Agent Mediated Electronic Commerce V (AMEC-V)*, pp. 56–72. Berlin: Springer - Verlag.
- Laporte, G., F. Louveaux, and H. Mercure (1992). The vehicle routing problem with stochastic travel times. *Transportation Science* 37(1), 161–170.
- Law, A.M. and W.D. Kelton (2000). *Simulation Modeling and Analysis*, Volume 3rd edition. Singapore: McGraw-Hill.
- Mes, M.R.K., M.C. van der Heijden, and A. van Harten (2006). Comparison of agent-based scheduling to look-ahead heuristics for real-time transportation problems. *Accepted for publication in European Journal of Operation Research*.
- Powell, W.B., Y. Sheffi, K.S. Nickerson, and S. Atherton (1988). Maximizing profits for north american van lines truckload division a new framework for pricing and operations. *Interfaces* 18(1), 21–41.
- Psaraftis, H. (1988). Dynamic vehicle routing problems. In B. Golden and A. Assad (Eds.), *Vehicle Routing: Methods and Studies*, pp. 223–248. Amsterdam, The Netherlands: Elsevier.

- Reiss, R.K. and M. Thomas (1997). *Statistical Analysis of Extreme Values*. Basel, Switzerland: Birkhauser Verlag.
- Ross, S.M. (2003). *Introduction To Probability Models* (8th ed.). Elsevier.
- Sandholm, T. (1993). An implementation of the contract net protocol based on marginal cost calculations. In *Proceedings of the 12th International Workshop on Distributed Artificial Intelligence*, Hidden Valley, Pennsylvania, pp. 295–308.
- Topaloglu, H. and W.B. Powell (2002). Dynamic programming approximations for stochastic, time-staged interger multicommodity flow problems. Technical report cl-00-02, Department of Operations Research and Financial Engineering, Princeton University.
- Toth, P. and D. Vigo (2002). *The Vehicle Routing Problem*. SIAM Monographs on Discrete Mathematics and Applications. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.
- Vickrey, W. (1961). Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance* 16(1), 8–37.