AGENT-BASED SCHEDULING FOR AIRCRAFT DEICING [1]

Xiaoyu Mao [ab]   Adriaan ter Mors [ac]   Nico Roos [b]   Cees Witteveen [c]

[a] *Almende B.V.,*
Westerstraat 50, 3016 DJ Rotterdam, The Netherlands
[b] *IKAT, University of Maastricht,*
P.O. Box 616, 6200 MD Maastricht, The Netherlands
[c] *EWI, Delft University of Technology,*
P.O. Box 5031, 2600 GA Delft, The Netherlands
`{xiaoyu, adriaan}@almende.com`

Abstract

The planning and scheduling of the *deicing* and *anti-icing* activities is an important and challenging part of airport departure planning. Deicing planning has to be done in a highly dynamic environment involving several autonomous and self-interested parties. Traditional centralized scheduling approaches do not acknowledge the autonomy of parties involved. We therefore propose an agent-based scheduling approach for solving the aircraft deicing problem where stakeholders are modelled as autonomous agents. Based on our view that multi-agent scheduling is scheduling combined with agent coordination, we propose a simple *first come, first served* heuristic combined with the concept of decommitment penalties. We demonstrate the feasibility of the approach through a series of experiments.

1 Introduction

The planning and scheduling of the *deicing* and *anti-icing* activities at airports is an important and challenging part of airport departure planning. Aircraft deicing and anti-icing is required in winter time when frost, snow, and ice can form on the wings and fuselage of an aircraft. Such a layer of frost or ice on aircraft surfaces influences the aircraft's aerodynamic properties which may cause a loss of lift that could result in a crash. Deicing refers to the removal of frost, snow, or ice from aircraft surfaces, while anti-icing is the application of a layer of viscous fluid onto aircraft surfaces that should prevent snow or ice from accumulating. Since the deicing and anti-icing operations are always performed together, in the remainder of this paper we will not distinguish them and will use the term *deicing* to refer to both deicing and anti-icing.

Planning and scheduling the deicing of aircraft has to be done in a highly dynamic environment involving several autonomous and self-interested parties. The dynamic nature of the aircraft deicing problem stems from the fact that in many temperate climate zones as found in Western Europe, the process of deicing is not part of the original flight plan, and thus it has scheduled as part of *operational* (i.e., short-term) planning. Moreover, during wintry conditions involving snow and ice, airport capacities will be greatly reduced — again, in temperate climate zones, this is not taken into account in the flight schedules — putting a great strain on the re-planning capabilities of all parties involved. The parties involved are self-interested and often have conflicting interests. For instance, airlines and pilots will be concerned with the effects of deicing on their flight schedules, air traffic control will be responsible for safe flight movements, the airport itself will strive for a maximum utilization of its facilities (runways, gates, etc.), and the ground servicing companies performing the deicing will want to operate as efficiently as possible.

In this paper, we propose an agent-based scheduling architecture for solving the aircraft deicing problem where

- multiple stakeholders are modelled as autonomous agents having their own interests and value systems,

---

[1] This research is supported by Dutch technology exchange program Casimir, funded by NWO (Netherlands Organization for Scientific Research). Project No.CSI4006: Hybrid Human-Agent Networks.

- a fall-back to local agent planning is allowed to deal with the dynamic nature of the problem, i.e., in case the global, coordinated plan is rendered infeasible by incidents.

Furthermore, we investigate how we can combine traditional scheduling heuristics with agent coordination. For the local agent strategies we simply choose First Come, First Served (FCFS), as it is quite representative of scheduling heuristics, and to coordinate the agents we will introduce the notion of *decommitment penalties*: in case an agent has reserved a specific slot for deicing at the *deicing station*, but due to a delay at the gate it cannot make this slot, it will have to pay a decommitment penalty. In a FCFS environment, the first agent to request a resource will be assigned the use of it. This will encourage agents to reserve resources as quickly as possible. This is, however, not what we want in an environment where long-term plans are unstable, and will be subject to re-planning. The decommitment penalties should counter this effect by creating an incentive only to reserve a resource when an agent is sufficiently sure that it does not need to decommit later on. Hence, the introduction of decommitment penalties requires agents to reason about uncertainty, and to reserve slots only when they have a certain degree of confidence they can honour their agreements. Decommitment penalties are therefore a way to tackle the problem of sharing the use of scarce resources among self-interested agents.

The remainder of this paper is organized as follows. In Section 2, we describe the background of the airport deicing scheduling problem, and we link it to the problem of multi-agent scheduling. In Section 3 we will give a formal model of the deicing scheduling problem and we will introduce a simple solution scheme. The agent coordination mechanism — the decommitment penalty — will be discussed in Section 4; in Section 5 we will discuss some experimental results obtained with the use of decommitment penalties. Section 6 concludes with a look to the future.

2 Background Description

As soon as an aircraft has safely landed at the airport, a sequence of tasks must be planned and scheduled before the aircraft can take off again. By reducing the time an aircraft spends on the ground between flights (the *turnaround* time), an airline can handle more flights a day, thereby increasing its revenues. Aircraft deicing is just one of the *ground servicing tasks* that an aircraft must undergo, but it plays a significant role in the turnaround time, due to its close relation with departure planning: after an aircraft has been deiced (and in particular, *anti-iced*) it needs to take off quickly before its 'anti-icing' layer wears off, and new ice will re-form on the aircraft surfaces. The maximum amount of time that may elapse between deicing and take-off (called the *holdover time*) will typically be around 15 minutes (though it depends on the severity of the weather conditions), which is quite short when you realize that in this time the aircraft still has to taxi from the deicing station to the runway.

An aircraft may have several alternatives to receive deicing treatment: most commonly, it will taxi to one of the deicing stations, located (hopefully) at strategic positions around the airport; second, it can also be deiced at the gate, in which case a *deicing vehicle* will drive to the gate at which the aircraft is docked. Since the total deicing capacity at an airport is usually limited, careful planning and scheduling of these resources is of crucial importance to efficient departure planning.

Like many real world problems, the problem of deicing resource management exhibits characteristics of both planning[2] and scheduling[3]. It is a scheduling problem in the sense that aircraft tasks have to be located in time on resources, and it is a planning problem in the sense that an aircraft has a number of choices with regard to which deicing resource to make use of — and this choice of deicing resource has implications for other airport planning problems like arrival planning, departure planning, and taxiway planning. Nevertheless, the management of deicing resources can best be characterized as a scheduling problem as it involves only a small, fixed number of choices, and because the focus is more on time and resource constraints, rather than ordering of actions (cf. [6]).

Based on a classification of scheduling problems by Yang et al.[10], we classify the airport deicing scheduling problem as a Multi-mode Resource-Constrained Project Scheduling Problem(MRCPSP), with the following characteristics:

---

[2] The *planning* problem is usually defined as finding a sequence of actions that will transform the initial state of the world into a state where the goals are attained.

[3] The *scheduling* problem is to assign limited resources to tasks over time to optimize one or more objectives.

- *non-regular objective function*: rather than the traditional objective of minimizing the total makespan of a project, each aircraft aims to depart close to its Target Time of Departure (TTD);
- *generalized precedence constraints*: between deicing and take-off not only exists a precedence constraint, but also an additional temporal constraint — the holdover time;
- *stochastic processing time*: the duration of deicing depends on how much ice has accumulated on the aircraft, which often only becomes clear as the aircraft arrives at the deicing station;
- *transportation delays*: an aircraft must taxi between deicing and take-off (the jobs), but this taxi time depends on its choice of runway and deicing station;
- *non-preemptive operations:* once the deicing process has commenced, it is not interrupted to make way for another aircraft

Scheduling literature surveys [2, 3, 4] reveal that all algorithms for determining an optimal solution of the MRCPSP as well as most heuristic approaches are based on centralized problem solving. A centralized approach is impractical, however, when jobs and resources are distributed over different stakeholders. We therefore propose that in this domain, scheduling should not be done *for* multiple agents (or at least, not exclusively), but *by* different agents. Moreover, as different stakeholders can have different and often conflicting interests, there will be a limit to how much they are willing to accommodate the scheduling of other agents, and how much planning information they are willing to exchange.[4]

Of course, if agents were to schedule completely independently of each other, the union of their plans would show many conflicts. In the airport deicing domain, these conflicts will concern the simultaneous use of scarce resources. We therefore define the problem of multi-agent scheduling as follows (cf. [1]):

Definition 1 (Multi-Agent Scheduling) *Given a set of agents each with a set of jobs to schedule, and a set of resources to schedule them on, each agent should find an individual schedule for its jobs in such a way that none of the resource capacity constraints are violated.*

Obviously, satisfying all resource constraints will not happen by magic; the agents will need some coordination mechanism that will safeguard these constraints. Therefore, we can summarize the multi-agent scheduling problem as follows:

$$\boxed{\text{Multi-Agent Scheduling = Scheduling + Coordination}}$$

3 Formal Modelling

In this section we will present a formal model of the aircraft deicing scheduling problem. The model presented below is a simplified version of the problem described in the previous section, as we leave out other airport planning problems as runway planning and gate allocation, and as a result certain 'inter-planning-problem constraints' such as the holdover time are not taken into account. The following model has also been used as the basis of the experiments, which are described in Section 5.

Definition 2 (Aircraft Deicing Scheduling Problem) *The aircraft deicing scheduling problem is a tuple* $\langle A, D, c, \tau, d, l \rangle$ *where*

- A *is a set of* n *aircraft agents,*
- D *is a set of* m *deicing station resources,*
- $c : D \to \mathbb{N}$ *is a capacity function specifying the number of aircraft that can simultaneously be serviced at the deicing station (i.e., the number of deicing* bays*),*
- $\tau : A \to \mathbb{N}$ *is a function indicating the Target Start Deicing Time (TSDT) for each aircraft,*
- $p : A \to \mathbb{N}$ *is function that specifies the deicing process duration for a certain aircraft,*
- $l : \mathbb{N} \times A \to \mathbb{N}$ *is a function that assigns a cost to the delay of an aircraft,*

---

[4] In [9], this problem is studied from a multi-agent *planning* perspective.

The target start deicing time $\tau(a_i)$ defined in Definition 2 is the earliest possible start deicing time for aircraft agent a_i, which is in fact the time when all other ground services for this agent are assumed to be finished. A solution to an instance $\langle A, D, c, \tau, d, l\rangle$ of the deicing problem is a multi-agent schedule given by the vector $S = \langle(d_1, I_1), \dots (d_n, I_n)\rangle$ where (d_i, I_i) is a tuple in which d_i is the deicing station assigned to agent a_i during interval I_i such that

$$I_i = [s_i,\ s_i + p(a_i)] \quad \wedge \quad s_i \geq \tau(a_i) \tag{1}$$

A feasible schedule satisfies the following resource constraints: at every point in time t, the deicing resource utilization for each resource agent does not exceed the resource capacities. We have:

$$\forall t \forall d \in D |\{a_j \in A \mid (d, I_j) \in S \wedge t \in I_j\}| \leq c(d) \tag{2}$$

Given a Target Start Deicing Time for each aircraft agent a_i, the objective is to find a schedule S where the delay cost of all aircraft is minimized:

$$\min \sum_{a_i \in A} l(s_i - \tau(a_i),\ a_i) \tag{3}$$

where $s_i - \tau(a_i)$ is the delay of aircraft a_i.

4 Decommitment in FCFS

A standard heuristic approach to scheduling processes uses a *First Come, First Served* (FCFS) queue. It is starvation-free, very simple, and in some cases reasonably efficient. Therefore, it seems a reasonably choice for Air Traffic Control (ATC) to process aircraft requests for services as deicing.

If we apply the FCFS policy in a multi-agent environment, agents are encouraged to reserve a slot as soon as possible. In a highly dynamic environment, in which incidents frequently occur that invalidate existing plans, this is not what we want; in fact, it is almost the opposite of what we want. Instead, we would encourage agents to schedule slots only in a future of sufficiently clarity. Therefore, we investigate whether we can design a coordination mechanism that can provide the correct incentives by making use of *decommitment penalties*.

In previous research, decommitment has been primarily used to enable agents to explore new opportunities from the domain or from other agents [7, 5, 8]; an example might be a package-delivery agent that decommits the contract for one package so that it is able to accept a more profitable package to deliver [7]. We propose that the concept of decommitment penalties can also be used to coordinate agents.

When an aircraft agent reserves a particular time slot at a resource such as a deicing station, it will commit to turn up at that deicing station at the specified time. If the aircraft fails to show up (possibly because of some incident that occurred on the airport, such as taxiway congestion, or delayed refuelling), it has to pay a decommitment penalty to the deicing station. Hence, with the introduction of decommitment penalties, agents have an incentive to reserve as late as possible; after all, if it reserves a slot five minutes from now, it will be fairly certain it can make this slot. On the other hand, the old incentive for scheduling as early as possible remains: agents will still want to schedule slots before other agents get them.

4.1 The Aircraft Problem

We assume that an aircraft has to obtain exactly one slot for deicing. It can obtain this slot by reserving a free slot at a deicing resource. However, with a certain probability incidents occur that make it impossible for the aircraft to be present at the deicing station at the agreed time. When such an event occurs, it must decommit and pay a decommitment penalty, which we assume to be an airport-wide constant δ. If the aircraft decommits, then it has to try again to reserve a slot at a deicing station. We assume that the agent can see when the first available slot is at all deicing stations (we will refer to this time as the Earliest Available Slot Time (t_{EAST})). To simplify the aircraft agent strategy, we assume that a deicing station never has to decommit.

As an aircraft agent can see the earliest available slot, it has to solve the following decision problem:

Do I reserve the currently available first slot, or do I reserve a slot at a later time?

To answer this problem, the agent a has to be able to evaluate his two different options. To judge whether the decision to reserve now has any merit, the agent needs to value the current slot based on how close it is to the $\tau(a)$, but it also needs to estimate the probability it will have to decommit from the slot. We assume that this probability of having to decommit is dependent on how far in the future the reserved slot is. Judging the option of reserving a slot at a later time is more difficult, as it needs to predict the availability of deicing slots in the future. This availability depends on at least the following factors:

1. incidents may occur to other aircraft, freeing up deicing resources,
2. the passage of time; if a slot is available 11 minutes from now, then, if no-one else takes it, there will be a slot 10 minutes in the future one minute from now,
3. other agents reserve slots.

Trying to incorporate all these factors into a realistic model is a formidable task, especially as the slot-reserving behaviour of agents may be subject to their perception (and prediction) of other agents' behaviour. Therefore, we will make the following simplifying assumptions to make the task of foretelling the future a more tractable one:

- If an agent has to decommit from a slot, then it will have to find a new slot. Apart from the time lost in decommitment, we assume that deciding whether or not to take the current slot is independent of time. Hence, we assume no peak hours in which finding a slot is more difficult.
- The delay an agent suffers when it has to decommit will mainly depend on the time it decommits; here we assume a constant value for this delay.
- When an aircraft opts to postpone its decision to reserve a slot until the next round, and it turns out that another agent has reserved the previously earliest slot, then the new t_{EAST} is simply the old t_{EAST} plus some constant value.

Armed with these simplifications, we can develop a strategy for an aircraft agent.

4.2 Aircraft Agent Strategy

The heuristic we will investigate in this section can be described as follows:

> *Reserve the earliest available slot if the expected cost of reserving this slot is less than the expected cost of reserving a slot the next round; otherwise, postpone the reservation decision until the next round.*

We will now introduce a number of functions to be able to define the expected cost of reserving the earliest available slot. We assume that the probability of an incident occurring is a linear function of the time passed since the reservation time t. Hence, the probability that an incident will occur until the reservation is used equals:

$$p_d(t, t_{\text{EAST}}) = \min(1, \beta \cdot (t_{\text{EAST}} - t)) \qquad (4)$$

where $0 \leq \beta \leq 1$, t is the time the reservation is made, and t_{EAST} is the start time of the reserved slot. An aircraft can reserve a slot from the Target Start Deicing Time onwards, i.e., $\tau(a) \leq t_{\text{EAST}}$. If the aircraft obtains a slot that is later than this target time, it incurs additional cost ('an aircraft on the ground costs the airline money'). We assume that the delay cost per minute α is the same for all aircraft, resulting in the following cost function:

$$l(t) = \alpha \cdot t \qquad (5)$$

where t is the time in minutes that the aircraft is delayed; if an aircraft has reserved a slot at t_{EAST}, then its delay will be $t_{\text{EAST}} - \tau$. Ultimately, an aircraft's delay cost will be related to its take-off time, but as deicing is quickly followed, in general, by take-off, a delay in deicing will translate into a delay in take-off time.

We can now define the cost of being delayed as a result of having to decommit. First of all, an agent has to pay the decommitment penalty δ; second, if t_d stands for the time decommitment occurs, then the aircraft has wasted $(t_d - t)$ minutes (where t is the time at which the slot it reserved). Again, we assume that this quantity $(t_d - t)$ will in fact delay deicing — and therefore take-off — by $(t_d - t)$ minutes. For simplification purposes, we will assume (i) that $(t_d - t)$ is a fixed value, and so we define the cost of decommitment as:

$$dcp \quad = \quad \delta + l(t_d - t) \tag{6}$$

The cost specified in Equation 6 is the *immediate* cost that the aircraft incurs — this is the cost the aircraft will have to pay anyway — without taking into account that there is a chance of having to decommit again in the future. Using the above definitions, an agent can calculate the expected cost of reserving a slot at time t with earliest available slot time t_{EAST}:

$$E_{\text{res}}(t, t_{\text{EAST}}) \quad = \quad p_d(t, t_{\text{EAST}}) \cdot dcp + (1 - p_d(t, t_{\text{EAST}})) \cdot l(t_{\text{EAST}} - \tau) \tag{7}$$

Note that a more realistic model for the cost of reserving a slot would be forward recursive: in case an aircraft has to decommit, it will have to try to get a slot again in subsequent rounds, again with the possibility of having to decommit, adding to its cost. Equation 7 effectively cuts off this forward recursion after one step, by taking into account only the immediate cost for decommitment.

The expected cost of waiting until the next round is given by the following function:

$$E_{\text{wait}}(t) \quad = \quad p_T(t, t_{\text{EAST}}) \cdot E_{\text{res}}(t^+, t^+_{EAST}) + (1 - p_T(t, t_{\text{EAST}})) \cdot E_{\text{res}}(t^+, t_{EAST}) \tag{8}$$

in which $p_T(t, t_{\text{EAST}})$ stands for the probability of another agent having reserved between time t and t^+ the slot starting at t_{EAST}. This probability function is based on the number of aircraft in the system, and the scarcity of the deicing resources. We assume aircraft take-off times are independent of each other and are uniformly distributed over time, and so we model the probability $p_T(t, t_{\text{EAST}})$ is with a Poisson distribution $(f(k; \lambda) = \frac{e^{-\lambda}\lambda^k}{k!})$ where:

$$p_T(t, t_{\text{EAST}}) = 1 - f(0, \frac{t^+ - t}{|D| \cdot T}) = 1 - e^{-\frac{|A| \cdot (t^+ - t)}{|D| \cdot T}} \tag{9}$$

where T is the time in minutes over which these aircraft are distributed (e.g., we could have a simulation run of $T = 360$ minutes in which $|A| = 100$ aircraft have to be deiced using $|D| = 4$ deicing stations).

Note that we have assumed above in Equation 8 that the value of t^+_{EAST} is simply the value t_{EAST} plus some constant. Equation 8 basically expresses that by not reserving a slot this round, there is a chance that another agent reserves the previously earliest available slot, and you consequently have to schedule a later slot t^+_{EAST} (which will result in more delay); on the other hand, if no agent has reserved the slot starting from t_{EAST}, then this possibility is still open to you at time t^+. By this time, the probability of decommitment will have lowered (i.e., $p_d(t^+, t_{\text{EAST}}) < p_d(t, t_{\text{EAST}})$), and thus reserving this slot at time t^+ will have a lower expected cost.

The agent strategy we propose in this section is simple: in case $E_{\text{res}} < E_{\text{wait}}$, the agent will reserve at time t the slot starting at t_{EAST}, otherwise it will wait until the next round. In the next section, we will investigate whether reasoning about decommitment in this way results in improved performance.

5 Experimental Results

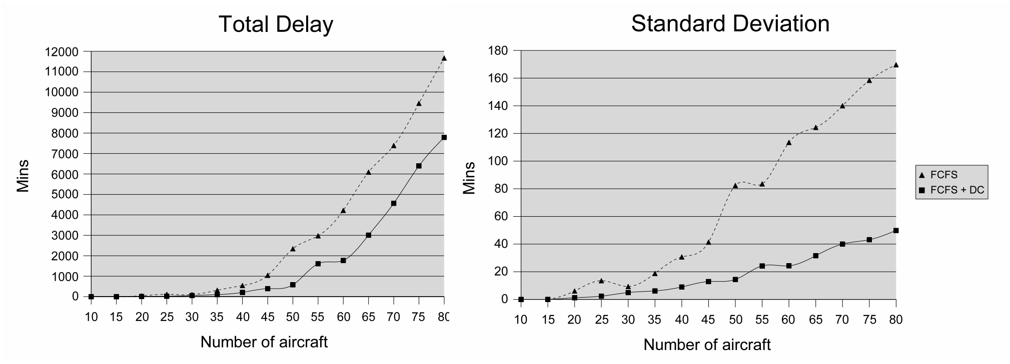
In this section, we will evaluate the efficiency of determining the time to reserve a slot using the decommitment formulas from the previous sections, assuming the deicing stations apply a FCFS queue. We compare this with a FCFS queue in which the aircraft agents request slots arbitrarily.

We conducted these experiments using only a single deicing station having 3 bays; deicing processing durations and aircraft delay cost functions are assumed to be the same for all aircraft (we chose a deicing time of 20 minutes), and the Target Start Deicing Times were randomly distributed over a six-hour interval. For these parameters, the maximum number of aircraft that can be serviced without delay equals $n = \frac{3 \times 6 \times 60}{20} = 54$, assuming a maximally convenient distribution of aircraft Target Start Deicing Times (τ). This means that with a random distribution of τ, we can expect some delays regardless of the scheduling strategy in case we have more than 54 aircraft. Some further parameter values are $\alpha = 1$, $\beta = \frac{1}{60}$, and $dcp = 50$. We have tested both strategies for different numbers of aircraft, and for each number of aircraft, we have performed three runs. The averaged results are displayed in Table 1.

	FCFS		FCFS + DC	
Number of aircraft	Total delay	Standard deviation	Total delay	Standard deviation
10	0	0	0	0
15	0	0	0	0
20	42	6.08	9	1.19
25	114	13.5	19	2.35
30	99	9.32	56	4.94
35	312	18.8	98	6.18
40	546	30.7	208	9
45	1044	41.4	391	12.9
50	2343	82.33	579	14.4
55	2973	83.5	1611	24.23
60	4220	113.5	1770	24.3
65	6090	124.33	3007	31.6
70	7392	140	4567	40
75	9453	158.33	6395	43.17
80	11667	169.67	7790	49.83

Table 1: Total delay and standard deviation in FCFS and FCFS+DC

In Figure 1.a, we see that using decommitment-penalty reasoning, we end up with a lower delay regardless of the number of aircraft. The reason for this is that the decommitment penalties encourage the agents to schedule only a relatively short period of time (no more than an hour) in advance of their Target Start Deicing Time. As a result, the Earliest Start Time (EST) algorithm emerges as the combination of aircraft agent strategies. From our experiments, this heuristic proves to be more efficient than arbitrary-order scheduling.



a) b)

Figure 1: Summarized results of total delay and standard deviation in FCFS and FCFS+DC

Figure 1.b shows the standard deviation for delays of individual aircraft. The higher standard deviation for the non-decommitment version is caused by the fact that some aircraft have very little delay, while others have great deal of it. Using decommitment reasoning, delay is distributed more evenly over the aircraft. In a sense, decommitment reasoning thus also improves the fairness of agent interactions.

6 Conclusions & Future Work

In this paper we have discussed an agent-based model for the scheduling of airport deicing services. By introducing the idea of decommitment penalties for aircraft that reserve slots but fail to turn up, we enabled an agent-based scheduling of the problem that improves on a naive (and greedy) random-order First Come, First Served queue.

Given the early stage of our research, options for future work are too numerous to list exhaustively. We would like to investigate other scheduling strategies in conjunction with decommitment penalties. Also, our results currently rely on many simplifying assumptions, and it would be interesting to see whether the conclusions of this paper hold up if we relax some of these assumptions.

Another extension of high priority is to look at the relation with other airport planning and scheduling problems. In itself, the deicing problem as formulated in the formal model of Section 3 is not that exceptional. What makes the problem interesting to look into is its relation to other planning problems, possibly involving other planning agents. The challenge for airport deicers lies in inserting the deicing scheduling problem into other well-known scheduling problems — into existing plans, even — such as arrival and departure planning, which, under everyday conditions, do not take into account an extra trip to the deicing station.

Acknowledgments

This research is funded by NWO (Netherlands Organization for Scientific Research), Grant No. CSI4006, which is gratefully acknowledged. We are indebted to Peet van Tooren, Jeroen Valk, Tamás Máhr, Gerald de Jong, and Jan Peter Larsen (Almende B.V.) as well as Jaap van den Herik (University of Maastricht) for their help in this research.

References

[1] Mathijs de Weerdt, Adriaan ter Mors, and Cees Witteveen. Multi-agent planning: An introduction to planning and coordination. In Handouts of the European Agent Summer School, pp. 1-32, 2005.

[2] Willy Herroelen, Bert De Reyck, and Erik Demeulemeester. Resource-constrained project scheduling: a survey of recent developments. Comput. Oper. Res., 25(4):279-302, 1998.

[3] Rainer Kolisch and Sönke Hartmann. Heuristic algorithms for solving the resource-constrained project scheduling problem: Classification and computational analysis. In J. Weglarz, editor, Project scheduling: Recent models, algorithms and applications, pages 147 - 178, Kluwer, Amsterdam, the Netherlands, 1999. Kluwer academic.

[4] Rainer Kolisch and Sönke Hartmann. Experimental investigation of heuristics for resource- constrained project scheduling: An update. European Journal of Operational Research, 23, May 2005.

[5] Tuomas Sandholm and Victor Lesser. Leveled commitment contracts and strategic breach. Games and Economic Behaviour, 25:212-270, 2001.

[6] David E. Smith, Jeremy Frank, and Ari K. Jónsson. Bridging the gap between planning and scheduling. Knowl. Eng. Rev., 15(1):47-83, 2000.

[7] P. J. 't Hoen and J. A. La Poutré. A decommitment strategy in a competitive multi-agent transportation setting. In AAMAS '03: Proceedings of the second international joint conference on Autonomous agents and multi-agent systems, pages 1010-1011, New York, NY, USA, 2003. ACM Press.

[8] Vanessa Teague and Liz Sonenberg. Investigating commitment flexibility in multi-agent contracts. Game Theory and Decision Theory in Agent-Based Systems, pages 267-292, 2002.

[9] Jeroen Valk. Coordination among Autonomous Planners. PhD thesis, Delft University of Technology, December 2005.

[10] Bibo Yang, Joseph Geunes, and William J. O'Brien. Resource-constrained project scheduling: Past work and new directions. Technical report, Department of Industrial and Systems Engineering, University of Florida, June 2001.