Autonomous multi-vehicle formations for cooperative airfield snow shoveling

Martin Hess* Martin Saska* Klaus Schilling*
*University of Wuerzburg, Computer Science VII: Robotics and Telematics, Germany

Abstract— **In this paper an approach for snow shoveling of airfields using formations of snowploughs is introduced. The vehicle formations split and reunite depending on the varying width of the roads to clean. An optimal schedule is derived for the vehicles guaranteeing complete coverage of the graph which is used to model the airfield. Based on the schedule a simple path is generated for the formation centers. Even during turns the generated motion for the car-like robots within the formation ensures proper breadthways coverage. Finally the method's applicability is verified by simulations as well as hardware experiments.**

Index Terms— **Cooperative sweeping, autonomous robot formations, optimal coverage**

I. Introduction

With increasing air traffic in the whole world, main international airports are compelled to serve the utmost amount of flights also during contrary weather conditions. Better sensors, communication and navigation systems enable to launch and land even during heavy snowfall. But the condition of the runways is a limiting factor for the uninterrupted full-load running of the airport.

The novel airport snow shoveling approach introduced in this paper offers a periodical autonomous system based on formation driving of car-like mobile robots. The uninterrupted airport operations are ensured by dividing all service roads to protective zones and by switching the runways between active and shoveling mode. Due to the formation driving, all roads are cleaned always completely, which is important for unexpected rescue operations or even for an emergency landing. In the presented algorithm minimal cleaning time in each zone of the airport and complete cleaning of the whole airport during one loop is guaranteed. The short execution time is accomplished by splitting formations for shoveling smaller roads and by reuniting to a significant number for the main runways.

The developed method has been intensively tested in various simulations as well as in laboratory hardware experiments. As a testing scenario for the proposed task scheduling algorithm Frankfurt airport was chosen, which is one of the largest airports in Europe. Bigger airports in the rest of world may differ in the number of runways but the complexity of the service roads is comparable. Due to the decomposition of the optimization problem to subproblems focusing on each runway and its environment, we suppose that our approach is useful in general.

For the hardware experiments and for the snow shoveling simulator a smaller scenario with one runway and two service roads was arranged, that contains all significant parts of a larger airport map (splitting and reuniting of formations, straight movement and curves with split-up and joint formations). This restriction is necessary due to the limited workspace in our laboratory and also due to the computing power needed for simulating the snow particles movement. The calculations required for communication, localization, motion planning and trajectory tracking used in the real world experiment as well need only an inconsiderable part of processor time.

The rest of the paper is organized as followed: Related work is described in Section II. Section III provides the description of the task planning algorithm. In subsection III-A a novel approach for optimal route scheduling is proposed, while subsection III-B describes, how the final path and the desired inputs for the formation movement are derived. In Section IV it is explained how the robots are controlled within the formation and how a desired shape of the formation can be obtained. After this the results of accomplished simulations and hardware experiments are presented in Section V followed by concluding words and plans for future work in Section VI.

II. RELATED WORK

In the recent years, issues involving multi-robot coverage have been addressed by many researchers. Most of the research in this area can be classified into two categories. On one hand is the *static coverage*, which aims to find a configuration of the nodes in order to optimally cover or observe a predefined area. Theoretical results for optimal static coverage are well described in [3] and [12]. More related to the work in this paper are approaches concerning *dynamic coverage*, whose goal it is to visit each place in a predefined area by at least one node in order to cover it completely. Often it is desired to coordinate the agents in a way that time optimality is achieved. E.g. in [9], distributed coverage is carried out by sharing an incrementally created adjacency graph of the decomposed area between the robots.

The work in this paper is mostly related to cooperative sweeping, which is a well-known subfield of dynamic coverage. The goal of sweeping is to find a specific motion for a robot in order to cover a 2-dimensional area by its effecter. Fundamental research in the field of cooperative sweeping was done by Kurabayashi et al. [11, 10]. A decentralized approach using a negotiation mechanism was proposed in [15]. Later, these basic off-line methods were extended for real time applications with an obstacle avoidance algorithm by Luo et al. [13, 14]. In contrast to these papers which mainly concentrate on distributing a given area to the agents, our work considers also nonholonomic vehicle kinematics as well as the fact that the effecter of snowploughs is displaced from the vehicles barycenter.

III. TASK PLANNING

In this section a novel path planning algorithm for snow shoveling of an airport by formations of mobile robots is introduced. The main idea of the approach is to describe the roads in a map of the airport by edges of a graph. Furthermore the nodes represent the centers of all crossings. An example of such graph is depicted in Fig. 4. The first subtask of the path planning is to find a time optimal sequence of the edges covering all roads that need to be cleaned. In the second phase the graph edges are replaced by a path assigned with a certain speed function that acts as a reference trajectory for the vehicle formations. The path has to be feasible for the car-like robots and should be optimal with respect to maximum breadthways coverage of the road by the formation.

A. Route scheduling

A state in the presented method is described by the actual positions of all robots and the history of their movements. Additionally it consists the time needed for the edge executions, and the set of edges that remain to be cleaned. The simplest way to find an optimal schedule for all vehicles is to evaluate the complete space of states, where the state with minimum execution time that contains no more edges to clean is the optimal solution.

The following base rules for the scheduling were fixed after discussions with a specialist from an airport:

1) *Each road should always be entirely cleaned.* The reason is that partly cleaned paths with snow hills remaining on the surface could be dangerous in emergency situations. For rescue and fire fighting vehicles and for planes making a forced landing, roads with constant layer of snow are safer than those with a cluttered surface.
2) *A formation is allowed to turn about only in dead ends* (nodes 14, 43 and 79 in Fig. 4). This rule prevents complications with snow remaining after the turning.

One possibility to find the optimal solution is the ”breadth first” search algorithm, that systematically evaluates the whole space. It is interesting to think about the total number of states that need to be expanded, where expanding means to evaluate all sons of a state in the state tree. In the shoveling scenario the minimum number of crossing lines in each node is three (nodes with only two edges can be replaced by one long edge between it’s neighbors). Thus, each vehicle situated in an expanding node has to choose one out of at least three possible actions. The snowploughs are not allowed to use the same edge again (due to the second rule), but they may stay in place and wait for a sufficient number of vehicles (the first rule).

One state, providing that the sufficient number of vehicles for each adjacent edge of the current node is one (or the roads are already cleaned) can be expanded to

$$S_i^n = \frac{(n+i-1)!}{i!(n-1)!}, \tag{1}$$

states, where i denotes the number of vehicles and n is equal to the adjacent edges in the current node. To estimate a lower bound for the number of states that need to be evaluated we assume that all nodes have 3 adjacent edges and 1 snowplough is applied for shoveling. When applying 100 transitions, i.e. movements from a node to an adjacent node, we get

$$(S_1^3)^{100} = 3^{100} \approx 5.15 * 10^{47} \tag{2}$$

states to evaluate. It is easy to see, that such an amount cannot be evaluated by conventional computers in appropriate time.

Fortunately not all of the states are feasible. Usually each airport has several parallel runways with appropriate service roads. In order to prevent collisions between snowploughs and the airport traffic, such parts should be cleaned separately. During the cleaning phase airport traffic is not allowed to enter the cleaning area and conversely the other parts of the airport are forbidden for the ploughs.

Frankfurt airport is usually using two parallel runways labeled A and B. For the scheduling algorithm we divided all roads into the three non-overlapping sets A, B and AB (see Fig. 4). The edges of set A (resp. B) need to be cleaned up completely while runway A (resp. B) is in shoveling mode, because these roads are used by airplanes, when the appropriate runway is in active mode. Edges of set AB can be cleaned with set A or B, but together at least with one of them. These auxiliary roads are only used by the ground crew, that are able to avoid the snowploughs. Therefore they do not need to be closed for the robots.

Resulting from the decomposition, the searched space is limited to about 40 edges for each area. Therefore the lower bound for the total number of states corresponding to the assumptions for (2) that need to be expanded during the ”breadth first” algorithm is equal to

$$(S_1^3)^{40} = 3^{40} \approx 1.21 * 10^{19} \tag{3}$$

which is still unsatisfactory.

If more than one vehicle is utilized for the shoveling task the number in (3) is even higher. To get around this another simplification of the search space, which results from the specific character of the airport, is made. The service roads are almost two times narrower than the main runways. Due to this we can expect that the optimal solution will be to firstly clean the runway by the whole formation and to divide it into two equal parts afterwards in order to clean up the remaining roads. Hence in the final solution there is no time wasted for robots waiting for others to form up.

To improve the algorithm we also changed the selection rule that determines the state that will be expanded in the next step. The ”breadth first” algorithm usually chooses the firstly found state that is yet unexpanded. But if instead the unexpanded state with the lowest waste time (time lost by using a path that does not need to be cleaned or by waiting for the second formation at the end of the plan) will be expanded, the firstly found solution is the optimal one. This is because all other states with possible lower waste time are already evaluated.

The complete cleaning loop for Frankfurt airport is described in 6 steps:

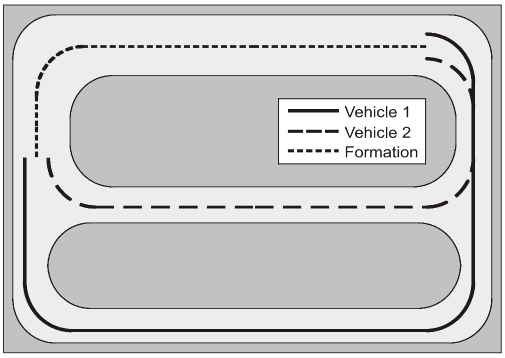


Fig. 1. Paths prepared for the testing scenario. This simplified scenario is solved by two snowploughs. The lines should be followed by the vertical center of the vehicle formation.

1) All robots clean up runway A in a big formation. The schedule is a sequence of the edges from node 22 to node 43.
2) Two smaller formations clean the remaining edges of A. Edges of AB can be also used.
3) Both formations move to node 19. The shortest schedule is found by the Dijkstra algorithm [8].
4) The reunited formation cleans runway B from node 19 to node 79.
5) The divided formations clean all edges $B + AB$ that were not cleaned in steps 2) - 4). The already cleaned edges $B + AB$ can be used during this step, too.
6) Both formations move to node 22. The sequence is restarted with step 1).

The optimal solution for Area A, that is depicted in Fig. 5 was obtained by the algorithm with 61520 state evaluations.

B. Path planning

The input for the path planning is a sequence of edges found by the route scheduling algorithm. To obtain a path, every two neighboring edges are replaced by two lines and one circle in between. The circle should have minimum radius with respect to feasibility for the whole formation, i.e. the curve is feasible for the robot that has to do the sharpest turn.

The final path is adjusted for the vertical center of the formation and not for the single robots. Thus problems can occur when the formation is divided or united. This was solved by splitting the vertical center of formation to two or more points or by uniting multiple ones. The new centers need to be chosen in terms of the new formations in order to keep a smooth path for each robot. Fig. 1 shows a simple example where one formation is split into two and later again reuniting.

IV. MOTION COORDINATION

In this section we will describe, how the motion of the robots is coordinated in order to fulfil the desired snow-cleaning task.

A. Motion generation for the vehicle formations

On parts of the airfield, e.g. on the wide runways it is desired to remove the snow in a preferably short time interval. A reasonable solution to this can be achieved by utilizing a formation of multiple snowplough vehicles for cleaning these areas. Since we assume car-like kinematics for the autonomous snowplough robots, a motion planning method applicable for this type of vehicles is needed. It should generate feasible trajectories for all vehicles, which result in a proper collective coverage of the ploughs shovels on the airfield. We developed a suitable approach for the trajectory generation based on [1] and [2], which will be shortly described below.

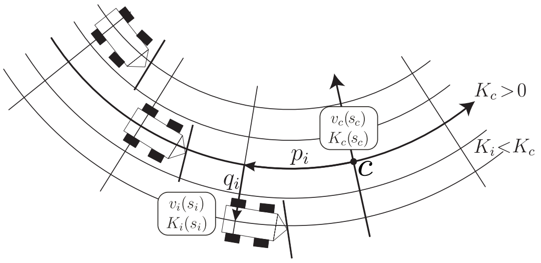


Fig. 2. Formation consisting of 3 snowploughs during a left turn.

A formation consists of N robots arranged around a reference point C, which follows the planned path with predefined speed (see Fig. 2). C was chosen on the vertical centerline in front of the robots, so the formation can deal with the switching mechanism mentioned in section III-B. A reference trajectory is fully defined by speed $v_c(d_c)$ and curvature $K_c(d_c)$, where d_c denotes the traveled distance of point C. The position of robot i, where $i \in \{1, ..., N\}$ is described by the relative coordinates $[p_i, q_i]$ with respect to the reference point. The horizontal component p_i is measured in curvelinear coordinates with the curvature of the reference trajectory at the origin, which is equal to the point C. In this description $[p_i, q_i]$ will stay constant, during periods when the formation's arrangement remains static. The euclidian distance between vehicles varies with changes of curvature and thus can lead to a collision between vehicles, if $[p_i, q_i]$ for $i \in \{1, ..., N\}$ are not chosen properly.

The control inputs for vehicle i are computed as

$$v_i(s_i) = Q v_c(s_i) \tag{4}$$

and

$$K_i(s_i) = \frac{S}{Q}\left(K_c(s_i) + \frac{K_c(s_i)\left(\frac{dq_i}{ds_i}(s_i)\right)^2}{Q^2} + \frac{\left(1 - q_i(s_i)K_c(s_i)\right)\frac{d^2q_i}{ds_i^2}(s_i) + q_i(s_i)\frac{dq_i}{ds_i}(s_i)\frac{dK_c}{ds_i}(s_i)}{Q^2} \right), \tag{5}$$

where

$$S = \operatorname{sgn}(1 - q_i(s_i)K_c(s_i)) \tag{6}$$

$$Q = \sqrt{\left(\frac{dq_i}{ds_i}(s_i)\right)^2 + \left(1 - q_i(s_i)K_c(s_i)\right)^2}, \tag{7}$$

and $s_i = d_c + p_i$ denotes the momentary traveled distance of vehicle i along the reference trajectory. To alter the formation in order to avoid obstacles or due to splitting or reuniting of the group it is possible to modify p_i and q_i continuously with the traveled distance along C1-continuous parts of the reference trajectory. Detailed information on the derivation of these equations and a simulative experiment can be found in [6].

The described approach results in optimal breadthways coverage for the snow shoveling task, because even in curves the vehicles are oriented in parallel at equally traveled distances along the reference trajectory. When a formation is divided the moment when splitting the reference point is time synchronized. On the other hand the moment of uniting formations is triggered by the arrival of the last subgroup, where groups that arrive earlier have to wait. This makes the system more robust against deviations from the desired trajectory.

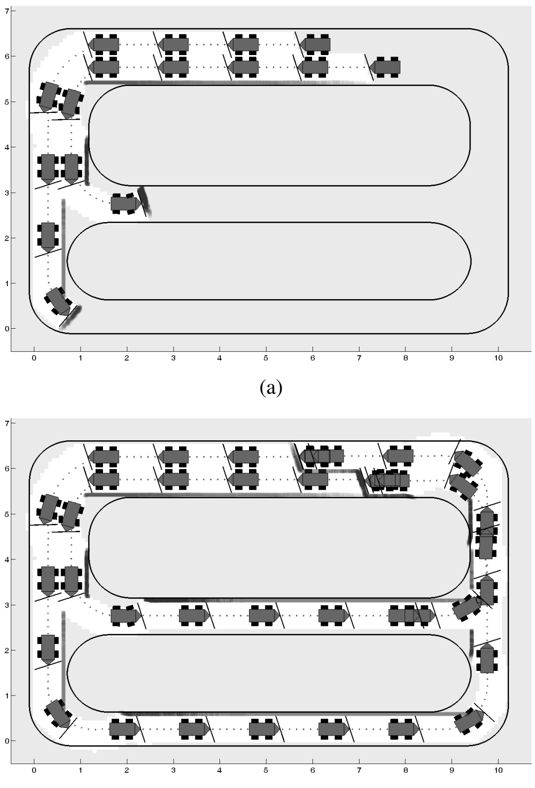
B. Trajectory tracking

In order to make the motion of the single snowplough robust against uncertainties and perturbations it is useful to apply a trajectory tracking control. For this work an approach for car-like robots utilizing full-state linearization via dynamic feedback was used. With appropriate gains for the controller, the barycenter of the applying vehicle located at the center of its rear axle will be stabilized along the desired trajectory which should be 3 times differentiable almost everywhere to receive accurate results. When following a path of the form described in III-B, small negligible deviations occur in the connection points of circles and lines. After passing such a point, the actual state will converge exponentially back to the desired state. To track a trajectory the robots always need to know their own position in a fixed coordinate system. This can be achieved by utilizing an accurate positioning system (e.g. DGPS) which is installed at the airfield. More information as well as mathematical description and analysis can be found in [4].

V. EXPERIMENTAL RESULTS

The functionality of the route scheduling algorithm has been verified by an experiment prepared for Frankfurt airport. The schedule depicted in Fig. 5 is the optimal plan for shoveling the service roads of A. In this part of cleaning the big formation is split into two parts with an equal number of robots (see step 2) of the cleaning loop in section III-A). The two sequences show order and time in which each node should be visited. The schedules for both subformations begin and end in the same node, which are the ones where the formations are divided and united (Nodes 41 and 24).

As mentioned before all significant parts of the airport snow shoveling scenario have been tested in smaller scenarios. Figure 3 shows two snapshots from the simulations performed. The size of the scenario is adjusted to the hardware experiment described later as we expect similar performance for large-scale vehicles and environment. The two snowploughs start to remove the snow from the main runway in a formation. After shoveling the first turn cooperatively the formation is split to clean the smaller roads separately (see Fig. 3(a)). Before the snowploughs reunite the vehicle on the center road has to wait for the other one to pass by. Otherwise the robot coming from the south would leave an amount of snow in the turnoff. Once the robots are reunited, they split vertically during the next turn in order to proceed to the starting point along the main runway (see Fig. 3(b)). It can be observed that the snow is not optimally removed during turns of the snowploughs. This results form the fact that the path is followed by the robots barycenter and not by the center of its shovel.



(a)

(b)

Fig. 3. Simulation snapshots (units in meter): (a) Splitting of formation. (b) One cleaning cycle complete.

Since the robots dynamics are left out completely in the simulation, we carried out hardware experiments with identical experimental setup. The snow is made of small pieces of polystyrene and the shovels are simply straight bars mounted transversely to the bumper of the car-like MERLIN robots [17]. Fig. 6 depicts three different states of one shoveling cycle. The position determination of the robots in the experiment relies on dead reckoning which is based on information obtained from wheel encoders and a gyroscope. Therefore the position error accumulates with time, which explains why the shovels of the two robots do not overlap anymore after one shoveling cycle (see Fig. 6(c)). Furthermore it is not possible to correct the initial position and orientation error. To avoid this in a real airfield scenario an accurate positioning system should be utilized to obtain external feedback for the robots absolute position. The phenomenon that snow particles remain in the intersection of the roads results from the simple design of the shovels used and will therefore not occur when utilizing modern snowploughs.

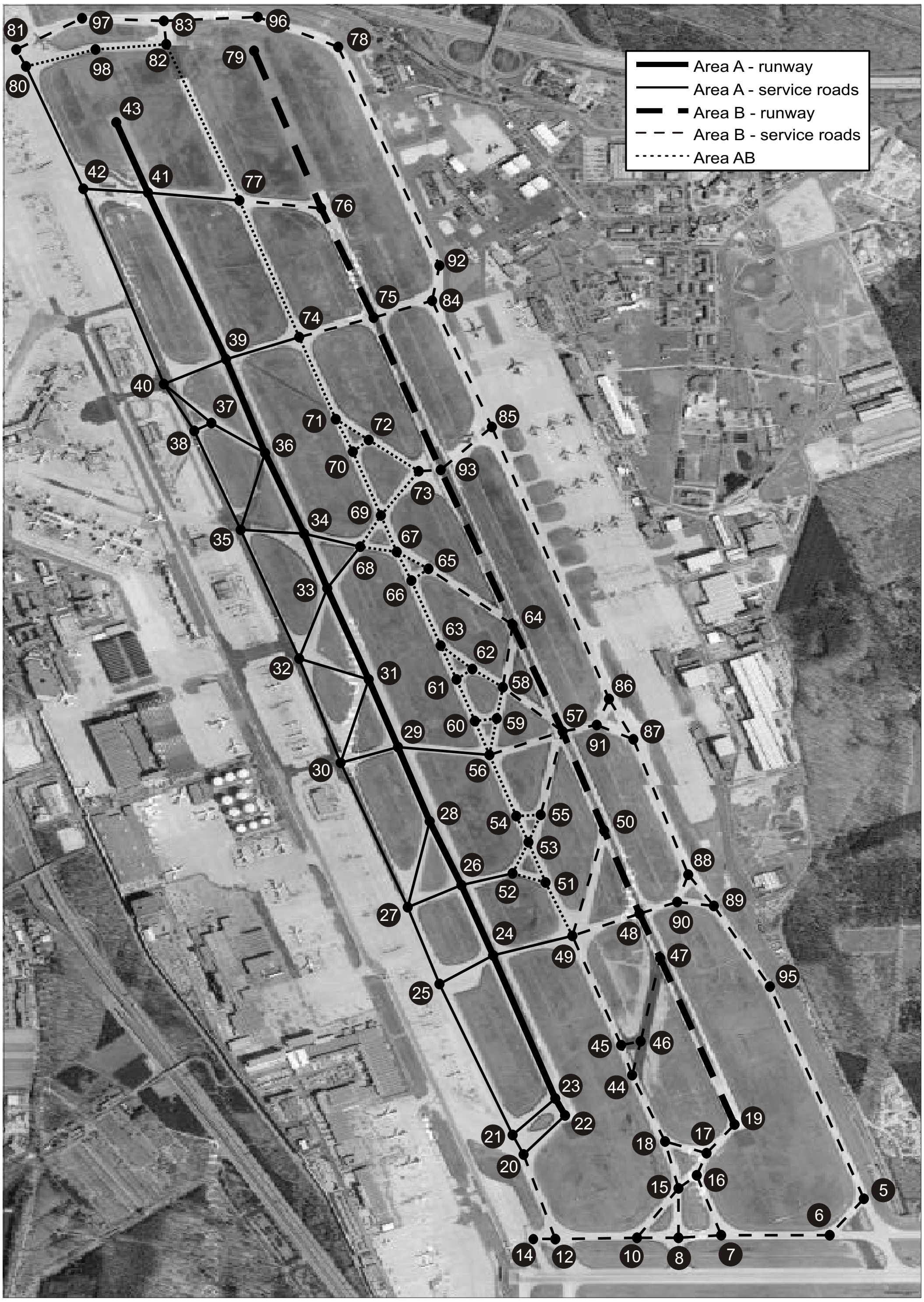


Fig. 4. Map of Frankfurt Airport with roads and runways partitioned into 3 sets. The satellite map was obtained from [7].

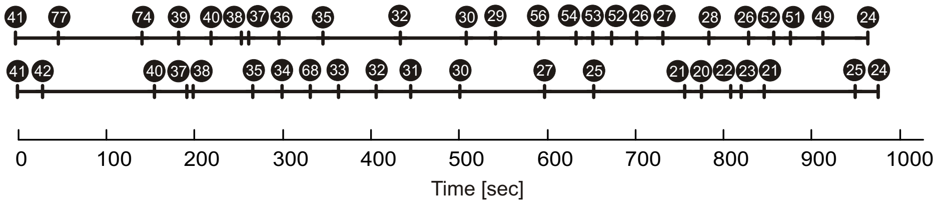
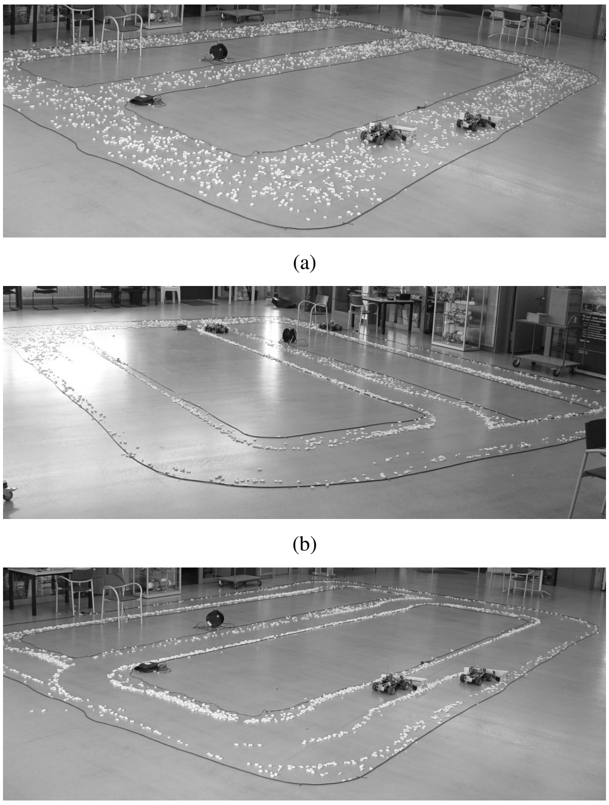


Fig. 5. Time schedule for cleaning the service roads in area A. The sequences for two formations show order and time in which each node should be visited.



(a)

(b)

(c)

Fig. 6. Snapshots from the hardware experiments: (a) Initial State. (b) The robot on the middle road waits for the other one passing by. This picture was taken from the opposing side. (c) One cleaning cycle complete.

VI. Conclusions

In this paper a novel snow shoveling method for airports was presented. The approach utilizes snowploughs driving in a formation and can be used for cleaning airports of different sizes. The autonomous car-like robots can be split and united to variously sized formations depending on the width of the roads to clean. The introduced algorithm offers optimal and complete shoveling during uninterrupted airport operations.

All parts of the method as well as the whole complex have been intensively tested in various simulation and hardware experiments. The results showed abilities and qualities of the method, but also emphasized some deficiencies and possible tasks for future work. At the moment the approach is lacking robustness with respect to unexpected events. Therefore we plan to enhance the vehicle coordination such that it can deal with robot failures as well as with unforeseen obstacles. On the other hand the task scheduling could be extended for finding a global optimal sequence for the whole airport. At the moment we can only verify a suboptimal solution for each region, but there may exist a better distribution of the unclassified paths (area AB). This task might be solvable by ”multiple traveling salesman” strategies [16]. In the task planning the composition of the final path should be improved to avoid the small deviations in the robots movement in the circle-line connection. This can be prevented by using spline paths similar to the ones in our previous work [5]. The suboptimal snow cleaning during turns that was observed in the snow shoveling simulation also needs to be improved. It might be possible to derive an appropriate path for the barycenter of the robot by applying a transformation to the optimal path of its shovel or by applying a trajectory tracking controller with respect to the center of the shovel.

Acknowledgment

This work was supported by the Elitenetwork of Bavaria through the program ”Identification, Optimization and Control with Applications in Modern Technologies”.

References

[1] T. D. Barfoot and C. M. Clark. Motion planning for formations of mobile robots. *Robotics and Autonomous Systems*, 46:65–78, February 2004.

[2] T. D. Barfoot, C. M. Clark, and S. M. Rock. Kinematic path-planning for formations of mobile robots with a nonholonomic constraint. In *IEEE International Conference on Intelligent Robots and Systems*, October 2002.

[3] J. Cortes, S. Martinez, Karatas T., and F. Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, 20(2):243–255, 2004.

[4] A. De Luca, G. Oriolo, and C. Samson. Feedback control of a nonholonomic car-like robot. In J.-P. Laumond, editor, *Planning robot motion*. Springer-Verlag, 1998.

[5] M. Hess, M. Saska, and K. Schilling. Formation driving using particle swarm optimization and reactive obstacle avoidance. In *Proceedings First IFAC Workshop on Multivehicle Systems (MVS'06)*, Salvador, Brazil, 2006.

[6] M. Hess, M. Saska, and K. Schilling. Enhanced motion planning for dynamic formations of nonholonomic mobile robots. In *Proceedings of the 6th IFAC Symposium on Intelligent Autonomous Vehicles (IAV2007)*, September 2007.

[7] http://maps.google.com. *[cit. 2007.2.5]*. Google maps.

[8] B.J. Jorgen and G. Gutin. *Digraphs: Theory, Algorithms and Applications*. Elsevier North Holland, 1979.

[9] C.S. Kong, N.A. Peng, and I. Rekleitis. Distributed coverage with multi-robot system. In *Proc. of the Conf. on Robotics and Automation*, pages 2423– 2429, Orlando, Florida, 2006.

[10] D. Kurabayashi, J. Ota, T. Arai, S. Ichikawa, S. Koga, H. Asama, and I. Endo. Cooperative sweeping by multiple mobile robots with relocating portable obstacles. In *Proc. of the Conf. on Intelligent Robots and Systems*, pages 1472 – 1477, Osaka, Japan, 1996.

[11] D. Kurabayashi, J. Ota, T. Arai, and E. Yoshida. Cooperative sweeping by multiple mobile robots. In *Proc. of the Conf. on Robotics and Automation*, Minneapolis, Minnesota, 1996.

[12] W. Li and C. G. Cassandras. Distributed cooperative coverage control of sensor networks. In *44th IEEE Conference on Decision and Control and 2005 European Control Conference*, pages 2542–2547, December 2005.

[13] C. Luo and X. Yang. A real-time cooperative sweeping strategy for multiple cleaning robots. In *Proc. of the Symp. on Intelligent Control*, Vancouver, Canada, 2002.

[14] C. Luo, X. Yang, and D. Stacey. Real-time path planning with deadlock avoidance of multiple cleaning robots. In *Proc. of the Conf. on Robotics and Automation*, Taipei, Taiwan, 2003.

[15] T.W. Min and H.K. Yin. A decentralized approach for cooperative sweeping by multiple cleaning robots. In *Proc. of the Int. Conf. on Intelligent Robots and Systems*, Victoria, Canada, 1998.

[16] S. Mitrovic-Minic and R. Krishnamurti. m-tsptw: Bounds for the number of vehicles. In *Proc. of the INFORMS Annual Meeting*, November 2002.

[17] K. Schilling and Q. Meng. The merlin vehicles for outdoor applications. In G. R. Gerhart, C. M. Shoemaker, and D. W. Gage, editors, *Unmaned Ground Vehicle Technology IV, Proceedings of SPIE*, volume 4715, pages 43–49, 2002.