

This is the peer reviewed version of the following article:

A Quadratic Programming approach for coordinating multi-AGV systems / Digani, Valerio; Hsieh, M. Ani; Sabattini, Lorenzo; Secchi, Cristian. - 2015-:(2015), pp. 600-605. (11th IEEE International Conference on Automation Science and Engineering, CASE 2015 Gothenburg, Sweden 24-28 August 2015) [10.1109/CoASE.2015.7294144].

IEEE Computer Society
Terms of use:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

13/01/2026 03:23

A Quadratic Programming Approach for Coordinating Multi-AGV Systems

Valerio Digani¹, M. Ani Hsieh², Lorenzo Sabattini¹, and Cristian Secchi¹

Abstract—This paper presents an optimization strategy to coordinate multiple Autonomous Guided Vehicles (AGVs) on ad-hoc pre-defined roadmaps used in logistic operations in industrial applications. Specifically, the objective is to maximize traffic throughput of AGVs navigating in an automated warehouse by minimizing the time AGVs spend negotiating complex traffic patterns to avoid collisions with other AGVs. In this work, the coordination problem is posed as a Quadratic Programming (QP) problem where the optimization is performed in a centralized manner. The optimality of the coordination strategy is established and the feasibility of the strategy is validated in simulation for different scenarios and for real industrial environments. The performance of the proposed strategy is then compared with a decentralized coordination strategy which relies on local negotiations for shared resources. The results show that the proposed coordination strategy successfully maximizes vehicle throughput and significantly minimizes the time vehicles spend negotiating traffic under different scenarios.

I. INTRODUCTION

Recent years have seen the increased popularity of automated warehouses and the coordination of a fleet of Autonomous Guided Vehicles (AGVs) for increased production efficiency and flexibility [1]. Multi-vehicle coordination has been largely studied in existing literature (see, for instance, the recent papers [2], [3] and references therein) and in particular in the context of Autonomous Guided Vehicles [4]. However common problems, such as deadlock and collision avoidance, arise naturally when managing and coordinating a fleet of AGVs.

In general, a team of robots can be coordinated using either centralized [5]–[7] or decentralized approaches [8]. While centralized approaches have the advantage of providing theoretically optimal solutions [9], they often scale poorly as the fleet size increases and often come at a significant computational cost. In contrast, decentralized strategies (see [10]–[12] and references therein) are mostly concerned with the scalability and complexity of coordinating large numbers of autonomous vehicles operating in a common workspace. Decentralized techniques are generally faster than centralized ones, but they can fail in finding valid paths and thus result in deadlocks [13]. For this reason, centralized approaches are often preferred in many industrial applications despite being

known to scale poorly with respect to the number of agents. The reliance on centralized approaches is also further backed by the ever decreasing cost of fast high-end computing and the potential loss in revenue caused by deadlocks.

More recently, mixed strategies that exploit the benefits of both centralized and decentralized coordination have been proposed. For instance, in [14] a decentralized local coordination is performed based on a centralized sharing of information. In this work, the environment is divided into several sectors in which a local and decentralized negotiation scheme for AGVs is used to allocate shared resources (crossing and segments of the road) to individual agents. Since the structure of the roadmap affects the overall performance of the coordination strategy, their interplay has to be considered. In [15], an algorithm to generate a roadmap that matches with the coordination policy discussed in [14] is presented. A drawback of these previous approaches is the amount of time spent locally coordinating the fleet, specifically the amount of time vehicles spent negotiating to cross various intersections in the workspace.

In this work, a new local coordination strategy that relies on a centralized optimization approach is presented. The proposed method considers the partitioning of an indoor environment into several sectors. Within each sector, the coordination strategy seeks to maximize the throughput of AGVs through each sector by minimizing the number of interactions between AGVs. The idea is to minimize the amount of time vehicles spend negotiating complex traffic patterns within each sector as they navigate in the workspace while avoid collisions with one another. The contribution of this paper lies in the formulation of the coordination problem as a convex optimization problem applied to realistic scenarios where roadmaps can be automatically generated in the given workspace. In particular, the coordination problem is posed as a quadratic program (QP) where the crossing time for the vehicles, *i.e.*, the time it takes for the vehicles to enter and leave a given sector, is minimized with respect to the vehicle velocities.

II. PROBLEM STATEMENT

The paper considers the coordination of a fleet of agents/vehicles on a fixed roadmap. In particular the roadmap is generated using the method proposed in [15]. Fig. 2 shows an example of a roadmap in a real warehouse environment and the partitioning of the workspace.

Definition 1 Sector A sector S is an area, or a region, of the roadmap which can be distinguished from the other ones based on topological, logistical and geometrical aspects.

¹Valerio Digani, Lorenzo Sabattini and Cristian Secchi are with the Department of Sciences and Methods for Engineering, University of Modena and Reggio Emilia, Reggio Emilia, Italy {valerio.digani,lorenzo.sabattini,cristian.secchi}@unimore.it

²M. Ani Hsieh is with the SAS Laboratory, Mechanical Engineering & Mechanics Department, Drexel University, Philadelphia, PA 19104, USA {dnk32,mhsieh1}@drexel.edu

Within a single sector, an intersection area is the bounded region of a sector where the coordination problem occurs. The sector partitioning affects the performance of the proposed approach, but these considerations will be analyzed in future works. Let us define the q -th sector as S_q which contains Z_q path, that is $S_q = \{\pi^1, \dots, \pi^{Z_q}\}$. The i -th path within S_q is split into M_i entities called segments, namely: $\pi^i = \{\pi_1^i, \dots, \pi_{M_i}^i\}$. Each segment is assumed to be unidirectional.

Each vehicle is modeled as 2D circle with a radius of $\frac{\delta}{2}$. Let $\mathcal{V}(\pi_h^i)$ be the portion of space occupied by a vehicle moving along the segment π_h^i , that is the trace of the vehicle along that segment. A segment collides with another segment when their traces are intersecting. An intersection area \mathcal{A}_q of S_q is then formally defined as:

Definition 2 Intersection Area $\mathcal{A}_q = \{\pi_h^l \mid \pi^l \in S_q, \exists j = 1, \dots, Z_q, j \neq l, \text{ and } \exists r = 1, \dots, M_j \text{ such that } \mathcal{V}(\pi_h^l) \cap \mathcal{V}(\pi_r^j) \neq \emptyset\}$

An intersection area is then the set of the colliding segments of different paths in the sector and, in general, each sector contains at least one intersection area.

As explained in [14], coordination is only required within each sector, and in particular within an intersection area, and the number of interactions among the agents corresponds to the number of local negotiations among them. A shared resource, *i.e.*, a road segment, is then allocated to the winner of each negotiation round, while the losers of the negotiation round must wait until the shared resource becomes free. It is worth noting that the results of these negotiations are not known a priori and that they depend on the path and the priorities of each AGV. Thus, the system is not deterministic and the traffic control and management is not trivial. In particular, the negotiation processes affect the total time a vehicle spends traversing a given sector. As such, planning minimum time paths between sectors is challenging since the actual delays generated by the negotiations are difficult to account for.

The proposed methodology seeks to overcome this issue by avoiding the negotiations entirely. This is achieved by choosing the best speed for each AGV to traverse its path that minimizes its total crossing time while avoiding conflicts with other vehicles, *i.e.*, collisions.

The problem can be formally stated as:

Problem 1 Multi AGV Coordination *Given:*

- a fleet of \mathcal{N} AGVs,
- a roadmap partitioned in sectors, and
- the initial and final positions for all the AGVs on the roadmap,

define a coordination strategy such that each AGV is able to move from its initial position to its assigned final position while minimizing the total crossing time and avoiding conflicts with other AGVs within the same sector.

The following **Assumptions** are needed:

- A1 No unforeseen events, such as the presence of dynamic obstacles (manual forklift, people, etc.) are considered.

- A2 An arbitrary velocity along a path v_i is assigned to the AGV i such that $V_{min} < v_i < V_{max}$. Where V_{min} and V_{max} are the same for all the vehicles.

- A3 The velocity along a segment is constant.

- A4 Each AGV has a different pair of initial and final positions.

Since the coordination is only required within each sector, hereafter a single sector scenario is used to describe the proposed methodology. In fact, it is worth noting that the coordination in each sector is independent with respect to the other sectors. In other words, the Problem 1 can be split among the sectors where it can be locally solved.

III. METHODOLOGY

The objective is to solve the Multi AGV Coordination (Problem 1) to obtain a set of velocities that guarantee a safe minimum distance between all agents. In general the problem can be solved considering each sector independently. In this section we show how the coordination problem within a single sector can be formulated as a convex optimization problem with feasible solutions. The coordination on a map composed by several sectors is performed by means of a dedicated process for each sector. Thus each process solves the optimization problem whenever a new agent enters or leaves the specific sector.

A. Objective Function

Considering \mathcal{N} AGVs, the path of the i -th AGV in a sector is composed of M_i segments of length d_i^k , $k = 1, \dots, M_i$. Let us define $d_i = \sum_{k=1}^{M_i} d_i^k$ as the length of the path of the i -th AGV. The velocity on the k -th segment for the i -th AGV is v_i^k . Then the crossing time for the i -th AGV is

$$\Delta t_i = \sum_{k=1}^{M_i} \frac{d_i^k}{v_i^k}.$$

The total crossing time of a sector is then provided by the maximum time taken to clear the intersection among all the AGVs. It is formally given by:

$$\Delta T = \max_i \Delta t_i = \max_i \sum_{k=1}^{M_i} \frac{d_i^k}{v_i^k} \quad (1)$$

The objective is to minimize ΔT on a given sector. It is worth noting that minimizing the total crossing time (the time of the slowest AGV) leads to increasing the number of AGVs in the same time window. Namely the throughput is maximized on a given time window (fixed time). Let us define $\bar{\mathcal{M}} = \sum_{i=1}^{\mathcal{N}} M_i$, then $\mathbf{v} = [v_1^1, \dots, v_i^{M_i}, \dots, v_{\mathcal{N}}^{M_{\mathcal{N}}}]^T \in \mathbb{R}^{\bar{\mathcal{M}}}$ is a vector containing all the vehicle velocities, then Eq. (1) is non-linear with respect to the parameters \mathbf{v} . We will now show how the same behavior can be described by using a linear function of \mathbf{v} . Since a path is already assigned to each vehicle, the distance to travel is constant for each vehicle. Then the only parameters to be chosen in Eq. (1) are the velocities v_i^k . It is then possible to state that minimizing the total crossing time is equivalent to maximizing the velocity

on a fixed distance, or equivalently to minimizing its negative. In particular the objective function can be formulated as follows:

$$f(\mathbf{v}) = - \sum_{i=1}^{\mathcal{N}} \sum_{k=1}^{\mathcal{M}_i} v_i^k. \quad (2)$$

It is straightforward to show that this objective function is both linear and convex with respect to the parameters \mathbf{v} .

B. Constraints

The parameters \mathbf{v} are lower and upper bounded due to the properties of the vehicles (Assumption A2). The linear constraints are formalized as:

$$V_{min} < v_i < V_{max}, \quad \forall i = 1, \dots, \bar{\mathcal{M}}$$

Each AGV is centered at the point $P_i = [x_i, y_i]^T$. Collision avoidance is guaranteed by ensuring that the relative distance, $\gamma_{i,j}$, between any two AGVs is greater than δ , namely:

$$\gamma_{i,j} = \|P_i - P_j\| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} > \delta. \quad (3)$$

Eq. (3) describes the non-linear constraints for the pairwise distances between AGVs. It is worth noting that the distance has to be computed for each pair of AGVs, (P_i, P_j) , along their vehicle paths. This process is inefficient and increases the complexity of the optimization.

Consider an intersection area \mathcal{A}_q and \mathcal{N} vehicles moving on \mathcal{N} different set of segments $\mathcal{K}_1^q, \dots, \mathcal{K}_{\mathcal{N}}^q$, that is $\mathcal{K}_i^q = \{\pi_h^i \in \mathcal{A}_q\}$ is the set of segments in the case of vehicle i . Then the sets of segments $\mathcal{K}_1^q, \dots, \mathcal{K}_{\mathcal{N}}^q$ are parameterized by $s_1 \in [0, d_1], \dots, s_{\mathcal{N}} \in [0, d_{\mathcal{N}}]$, where s_i can be any scalar parameter used to denote an AGV's position along its trajectory, namely along the set of segments.

Let $\mathcal{P}^i(s_i) : \mathbb{R}^+ \rightarrow \mathbb{R}^2$ be a function which maps the curvilinear abscissa s_i to the Cartesian position P_i of the vehicle on its path. Let $\Delta X_{i,j}$ be the portion of i -th path where vehicle i collides with vehicle j on the i -th path, i.e., $\gamma_{i,j} < \delta$. Here $\Delta X_{i,j}$ is a collision region for the i -th and j -th AGVs and can be formally defined as:

$$\Delta X_{i,j} = \{s_i \mid \exists s_j \text{ such that } \|\mathcal{P}^i(s_i) - \mathcal{P}^j(s_j)\| \leq \delta\} \quad (4)$$

It is worth noting that the collision regions can be computed totally off-line since they are based only on geometric information. The computational complexity of Eq. (4) can be formulated as $O(\bar{\mathcal{M}}^2)$.

Let $X_{i,j}^{min}$ and $X_{i,j}^{max}$ be the values of the curvilinear abscissa s_i corresponding to the start and final point respectively of the collision region $\Delta X_{i,j}$. Let us introduce $\omega_{i,j}^{min}$ and $\omega_{i,j}^{max}$ as the time that the vehicle i would take to reach the position $\mathcal{P}^i(X_{i,j}^{min})$ and $\mathcal{P}^i(X_{i,j}^{max})$ respectively. Furthermore the projection of $\Delta X_{i,j}$ on the time axis provides the set of collision time $\Omega_{i,j}$. Collision avoidance between the i -th and j -th AGVs is then satisfied when the following condition occurs:

$$\Omega_{i,j} \cap \Omega_{j,i} = \emptyset. \quad (5)$$

Fig. 1 pictorially depicts the described problem: with abuse of notation the ordinate axis represents both the parameter s_i and s_j . The condition (5) implies that a set of points between the two sets $\Omega_{i,j}, \Omega_{j,i}$, denoted by the red and blue line segments on the time axis in Fig. 1, exist in order to avoid a conflict.

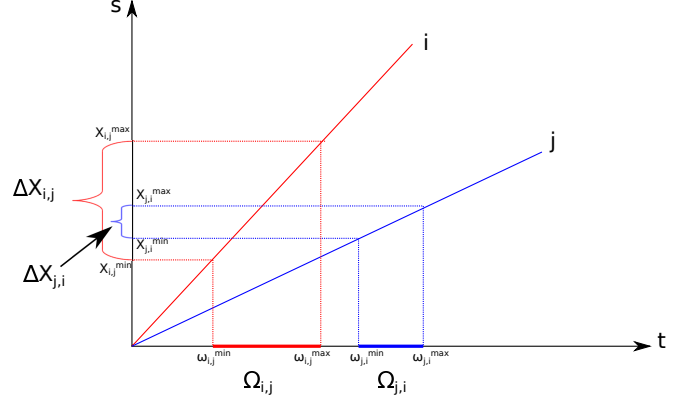


Fig. 1: Coordination space in the case of two agents: the lines represents the velocities of the AGVs.

This constraint can be reformulated in terms of the midpoints and the total time defined by each set $\Omega_{*,*}$. In particular the distance between two mid-points has to be greater than the sum of the distances between the mid-point and one extreme point of both the sets. We introduce the following notation:

- $\alpha_{i,j} = \omega_{i,j}^{max} + \omega_{i,j}^{min}$
- $\beta_{i,j} = \omega_{i,j}^{max} - \omega_{i,j}^{min}$

The condition described by Eq. (5) can then be formalized as follows:

$$\left| \frac{\alpha_{i,j}}{2} - \frac{\alpha_{j,i}}{2} \right| > \frac{\beta_{i,j}}{2} + \frac{\beta_{j,i}}{2} \quad (6)$$

which simplifies to

$$|\alpha_{i,j} - \alpha_{j,i}| > \beta_{i,j} + \beta_{j,i} \quad (7)$$

Once Eq. (7) is squared, the constraint becomes quadratic:

$$(\alpha_{i,j} - \alpha_{j,i})^2 > (\beta_{i,j} + \beta_{j,i})^2. \quad (8)$$

C. Quadratic Constraints Linear Programming

The coordination within a single sector is modeled as an optimization problem using a linear objective function, and a set of quadratic constraints with linear constraints on the boundary. The optimization problem is given by:

$$\text{minimize} \quad - \sum_{i=1}^{\mathcal{N}} \sum_{k=1}^{\mathcal{M}_i} v_i^k \quad (9a)$$

$$\text{subject to} \quad V_{min} \leq v_i \leq V_{max}, \quad \forall v_i \in \mathbf{v} \quad (9b)$$

$$\text{and} \quad v_i^2 e_{ij} + v_j^2 e_{ij} + v_i v_j f_{ij} < 0, \quad \forall i, j, \quad i \neq j \quad (9c)$$

where $e_{ij} = c_{ig} - a_{ij}$ and $f_{ij} = d_{ij} + b_{ij}$ are scalar constant values. Then the expression can be more compactly written

in the following standard form [16].

$$\text{minimize } f(\mathbf{v}) \quad (10a)$$

$$\text{subject to } A\mathbf{v} \leq b \quad (10b)$$

$$\text{and } 1/2\mathbf{v}^T H^{ij} \mathbf{v} < 0 \quad \forall i, j = 1, \dots, \bar{\mathcal{M}}, \text{ with } j > i \quad (10c)$$

where $A \in \mathbb{R}^{2\bar{\mathcal{M}} \times \bar{\mathcal{M}}}$ and $b \in \mathbb{R}^{2\bar{\mathcal{M}}}$ encode velocity bounds in Eq. (9b). The matrix $H^{ij} \in \mathbb{R}^{\bar{\mathcal{M}} \times \bar{\mathcal{M}}}$ represents the inequality constraint in Eq. (9c) for the pair (i, j) and it is defined such that the element (l, p) is:

$$H_{l,p}^{ij} = \begin{cases} 2e_{ji} & \text{if } l = p = i \\ 2e_{ij} & \text{if } l = p = j \\ f_{ij} & \text{if } (l = i \text{ and } p = j) \text{ or } (p = i \text{ and } l = j) \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

IV. ANALYSIS

First, some concepts from convex optimization theory are introduced.

Lemma 1 [16, Section 4.2.2] *Given a solution to the optimization problem, the solution is guaranteed to be optimal if the problem is convex.*

Lemma 2 [16, Section 3.1.4] *A twice differentiable function f is convex if and only if $\text{dom } f$ is convex and its Hessian is positive semi-definite.*

Now we can show the optimality of the proposed coordination problem.

Proposition 1 *If a solution exists to the optimization problem given by Eq. (9), the solution is optimal.*

Proof: Given a solution it is sufficient to prove the convexity of the problem. Let us define $f(\mathbf{v})$ the objective function in Eq. (9a), then $\text{dom } f = \{v \in \mathbb{R}\}$ is a convex set. Thus the objective function is convex since it consists of the sum of a collection of linear, and thus convex, functions. The linear constraints given by Eq. (9b) are affine and thus convex and essentially define a bounded convex set in the solution space. Considering the constraint given by Eq. (10c), the Hessian is $\nabla^2(1/2\mathbf{v}^T H^{ij} \mathbf{v}) = H^{ij}$, then since $H^{ij} \succeq 0$ the function is convex according to Lemma 2. Thus the optimization problem is convex and any solution is optimal according to Lemma 1. ■

While this work has not shown the existence of a feasible solution under all possible scenarios, empirical results presented in the following sections strongly suggest the existence of solutions to the optimization problem under most realistic conditions. Theoretical analysis of the conditions which guarantee the existence of a solution is a direction of future work.

V. SIMULATION RESULTS

The proposed optimized coordination is compared to the coordination strategy first presented in [14] which relies on local negotiations. Hereafter we refer to the current proposed

methodology as the *Optimized Strategy*, and to the other one as *Negotiated Strategy*. A decoupled optimal priority scheme [17] is applied to the *Negotiated Strategy* in order to obtain a better comparison.

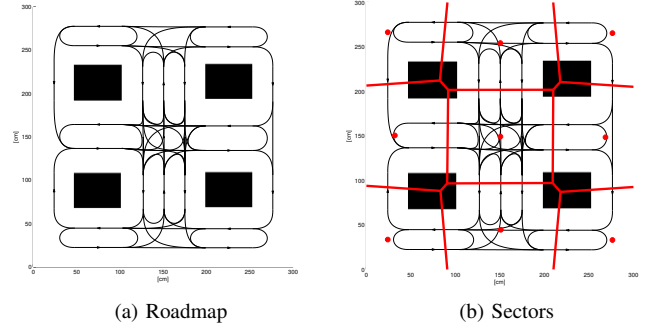


Fig. 2: Real map used in the simulations

The proposed methodology is first validated by means of simulations on different single intersections shown in Fig. 3 and then on a realistic environment. In particular the simulations are performed on realistic scenarios of real industrial warehouses (see Fig. 2a), available thanks to the close cooperation with industrial partners. A roadmap is built on that environment using the algorithm described in [15].

A. Single Intersection

It is worth noting that the topological complexity of the roadmaps is different in each scenario which depends on the number of possible paths, the number of possible interactions, and physical size of the environment. Repeated tests have been conducted under the following conditions:

- 4 topology of intersection
- number of AGVs $\in [2, 11]$,
- the simulation stops when the intersection area is cleared,
- the priorities [14] for the *Negotiated Strategy* are optimized each time,
- the same paths assigned to the AGVs are considered for the comparison, and
- the paths are assigned randomly.

Ten simulation runs were performed for each configuration.

To compare, we consider the time needed for all AGVs to clear the sector or the maximum of the crossing time ($t_{clearing}$). We also considered the worst waiting time (t_{wait}), the average waiting time (\bar{t}_{wait}), and the computational time needed to obtain a solution (t_{calc}). The waiting time is the time that the AGVs have to wait in the same position, i.e., $v_i = 0$, when yielding to other robots with higher priorities at an intersection in the roadmap. The worst waiting time is the maximum waiting time for all the AGVs in the sector. The computational time is the actual time to compute a solution to the centralized optimization problem. The worst waiting time and the average waiting time are computed for the *Negotiated Strategy*, while the computational time is computed for the *Optimized Strategy*.

The simulations were performed in Matlab with the standard optimization solver. In order to simulate the *Negotiated*

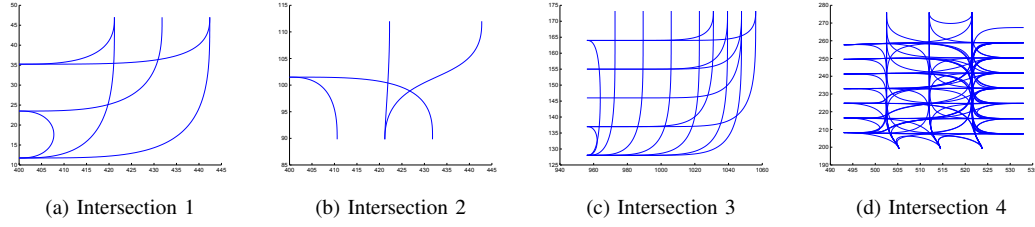


Fig. 3: Different intersections used during the simulations

Strategy in a decentralized way, the algorithm is executed in a parallel manner by implementing one single dedicated process per AGV. The results are summarized in Figures 4, 5, and 6.

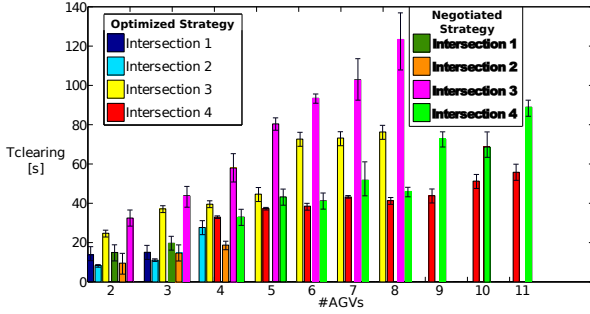


Fig. 4: Average of the Clearing Time versus number of AGVs in the different single intersections on 10 runs.

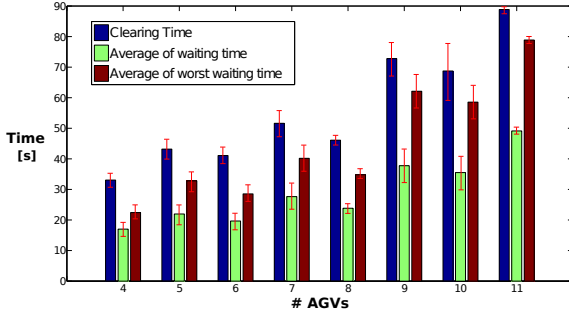


Fig. 5: Waiting time versus clearing time for the *Negotiated Strategy*. Data presented is from Intersection 4.

B. Complete Scenario

The proposed method is also validated on real scenarios where the roadmap is divided into several sectors. The simulation are conducted under the same conditions and the same assumptions of the previous section except for the number of AGVs, that is: 3, 5, 10, 15. The main difference compared to the single intersection simulation is that the coordination of the fleet is based on the hierarchical control architecture explained in [14]. The proposed method (*Optimized Strategy*) is still compared to the *Negotiated Strategy* with respect to the time needed for all AGVs to reach their final destination, that is t_{total} . Fig. 7 shows the trend of the t_{total} with respect the number of AGVs on both of the strategies.

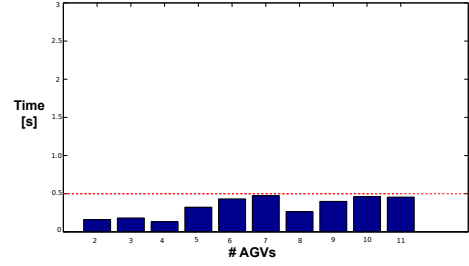


Fig. 6: The computational time for the optimized strategy is lower than 0.5 s in all the configurations.

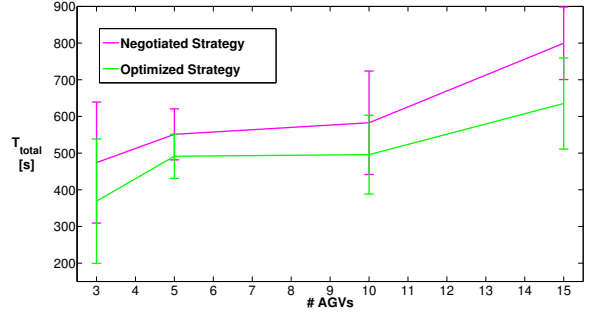


Fig. 7: Total time versus number of agents in the complete map with standard deviation

VI. DISCUSSION

For the single intersection scenario, the results show that the average clearing time (or the maximum crossing time) is always less using the *Optimized Strategy* rather than the *Negotiated Strategy* and clearly shows that the proposed methodology performs better than the previous one. Fig. 5 shows the relationship between the waiting time and the time a vehicle is actually moving, i.e., $v_i > 0$, in the sector for the *Negotiated Strategy*. Overall, the time a vehicle spends waiting or idling is considerable when the *Negotiated Strategy* is used. In particular, the results show that the average waiting time is almost 50% of the total time a vehicle spends within the sector and the worst waiting time is up to 90% of the total time. One can conclude that under the *Negotiated Strategy* the time an AGV spends at an intersection is mostly spent on waiting for its turn to cross. It is important to note that by construction the waiting time for each vehicle under the *Optimized Strategy* is always null. Here, coordination is achieved by managing the relative velocities of the agents to ensure collision avoidance while constraining the velocities to be always higher than

zero. When considering the average per vehicle waiting time resulting from the *Negotiated Strategy*, the computational time of the *Optimized Strategy* is essentially insignificant (see Fig. 6) despite being implemented in a centralized manner. This claim is supported by the fact that the strategy is designed to work with specific roadmaps and thus always tuned to the worst case scenario (Fig. 3d). Furthermore, the required computation burden still stays low when the number of segments increases. The *Optimized Strategy* is concerned only about the number of vehicles and the collision regions are computed totally off-line. Fig. 7 shows a comparison between the scalability of the *Negotiated Strategy*, which is almost linear, and the *Optimized Strategy*. We note that in the *Optimized Strategy* the maximum crossing time does not linearly increase with the number of agents but rather its trend is piecewise linear. This suggests that the *Optimized Strategy* can potentially be further exploited when the system is complex. This is further supported by the fact that the waiting time for the *Negotiated Strategy* is two order of magnitude higher than the computational time for the *Optimized Strategy*. For the complete scenario, the results are very similar to the previous simulation. In particular t_{total} in the *Optimized Strategy* is always less than in the *Negotiated Strategy*. This is another prove to validate the proposed method: also on a complete and real map the *Optimized Strategy* works better in term of total time. It is worth noting that the coordination and the performances within a sector are, in general, independent of the other sectors. Thus the complete behavior of the proposed methodology on the full map is approximately a linear combination of the local behavior within each sector. It is worth noting that an increasing of the number of intersections does not correspond to an increasing of the computation burden. Furthermore, as shown in Fig. 7, the displacement obtained from the results of the two strategies, that is the distance between the lines, increases as the number of AGVs gets bigger. It is worth noting that the more agents there are in the map, the higher the difference between the total time of the two strategies. Of course the complexity of the scenario affects the total time, however the *Optimized Strategy* consistently performs better in the presence of more agents.

VII. CONCLUSION

This paper presents an optimized coordination strategy that minimizes the traversal time of a fleet of AGVs through different sectors of an indoor environment for industrial applications such as warehouse automation. The coordination problem is posed as an optimization problem where a procedure was used to transform a non-linear and non-convex optimization problem into an equivalent quadratic program. Coordination is then achieved by solving the optimization problem in a centralized fashion within each sector of the warehouse. The method was compared to a decentralized negotiated strategy developed in a previous work where the coordination was carried out by assigning different priorities to each AGV. Simulation results show that the proposed optimized strategy significantly outperforms the decentralized

negotiated strategy.

In conclusion, the proposed coordination strategy achieves an optimal coordination among the AGVs negotiating a complex traffic pattern in different sectors of an automated warehouse. Optimality of the resulting strategy is guaranteed by posing the coordination problem as a convex optimization problem. One direction for future work is to determine the conditions under which a feasible solution to the convex problem will always exist. Another improvement aims at validating this coordination strategy with real experiments by using real robots.

REFERENCES

- [1] L. Sabattini, V. Digani, C. Secchi, G. Cotena, D. Ronzoni, M. Foppoli, and F. Oleari, "Technological roadmap to boost the introduction of agvs in industrial applications," in *IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, 2013.
- [2] Y. Zhang and H. Mehrjerdi, "A survey on multiple unmanned vehicles formation control and coordination: Normal and fault situations," in *International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2013, pp. 1087–1096.
- [3] S. Hoshino and H. Seki, "Multi-robot coordination for jams in congested systems," *Robotics and Autonomous Systems*, 2013.
- [4] R. Olmi, C. Secchi, and C. Fantuzzi, "Coordination of industrial AGVs," *International Journal of Vehicle Autonomous Systems*, vol. 9, no. 1, pp. 5–25, 2011.
- [5] L. Kavraki, P. Svestka, J.-C. Latombe, and M. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, Aug 1996.
- [6] L. Makarem and D. Gillet, "Fluent coordination of autonomous vehicles at intersections," *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2012, pp. 2557–2562, Oct. 2012.
- [7] N. Hui, "Coordinated motion planning of multiple mobile robots using potential field method," in *2010 International Conference on Industrial Electronics, Control Robotics (IECR)*, 2010, pp. 6–11.
- [8] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press, 2006, available at <http://planning.cs.uiuc.edu/>.
- [9] S. LaValle and S. Hutchinson, "Optimal motion planning for multiple robots having independent goals," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 912–925, 1998.
- [10] P. Yang, R. Freeman, and K. Lynch, "Multi-agent coordination by decentralized estimation and control," *IEEE Transactions on Automatic Control*, vol. 53, no. 11, pp. 2480–2496, 2008.
- [11] S. Manca, A. Fagiolini, and L. Pallottino, "Decentralized coordination system for multiple agvs in a structured environment," in *18th World Congress of the International Federation of Automatic Control (IFAC 2011)*, vol. 18, no. 1, 2011, pp. 6005–6010.
- [12] D. Habib, H. Jamal, and S. A. Khan, "Employing multiple unmanned aerial vehicles for co-operative path planning," *International Journal of Advanced Robotic Systems*, vol. 10, 2013.
- [13] W. Zhang, M. Kamgarpour, D. Sun, and C. Tomlin, "A hierarchical flight planning framework for air traffic management," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 179–194, 2012.
- [14] V. Digani, L. Sabattini, C. Secchi, and C. Fantuzzi, "Hierarchical traffic control for partially decentralized coordination of multi agv systems in industrial environments," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.
- [15] V. Digani, L. Sabattini, C. Secchi, and C. Fantuzzi, "An automatic approach for the generation of the roadmap for multi-agv systems in an industrial environment," in *IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2014.
- [16] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2009.
- [17] M. Bennett, W. Burgard, and S. Thrun, "Constraint-based optimization of priority schemes for decoupled path planning techniques," in *KI 2001: Advances in Artificial Intelligence*. Springer, 2001, pp. 78–93.