Cooperative controls for car–like robot coordination

A. Marigo B. Piccoli D. Vergni

Istituto per le Applicazioni del Calcolo "Mauro Picone"
Consiglio Nazionale delle Ricerche,
Viale del Policlinico 137, 00161 ROMA, Italy
marigo,piccoli,vergni@iac.rm.cnr.it

***Abstract*— In this paper we consider the problem of coordination of large number of robots subject to low information constraint. We solve the problem of safe–minimum time control for robots, that are car–like modeled and have different goal configurations, by designing two different controls: the safe and the time–optimal ones, and by tuning them suitably. We can see the tuning as a cooperation among the two controls. Simulation results show that the use of cooperative controls efficiently solves the problem of robot coordination when the information available by one of each other is reduced to the knowledge of their position when the distance among them is smaller than a relatively small bound.**

I. INTRODUCTION

The coordination for large number of robots is attracting increasing attention, see [1], [2], [3], [4], [5], [6]. This is due both to the fact that technological developments will render soon possible the production of multi–vehicle networks and to the consequent new challenges of control algorithms design for such robot groups. In the typical framework, each robot dynamic is independent but the presence of objective functions such as robot formations, that depend on the properties of the whole network, forces to implement coordinated motions. It is clear that communication issues play a central role in such control problems since perfect robot coordination is based on complete information. However limitations on communication channels bandwidth and other constraints impose to keep low the level of communication [1], [3]. Thus the link between information and control seems to be crucial as in many other control applications [7], [8], [9].

Our point of view is to consider simplified car–like robot models but keeping to a minimum the information needed to coordinate the robot group. Control algorithms are designed that rely only on partial detection of robot positions and are thus robust for failures in one or more single robot behavior and measurement and communication errors.

For given initial position of each robot and final desired configuration, i.e. a final network formation or final destination for each single vehicle, we consider two problems:

P.1 The motion of a single robot with final destination and N moving obstacles in the environment.

P.2 The motion of N robots with distinct initial position and final destinations moving at the same time in the environment.

The first problem is a key step to solve the second. Indeed, considering all other vehicles just as moving obstacles permits to construct a completely decentralized control algorithm that can be seen as single robot behavoir in the worst case.

The main technique is the use of cooperative controls. In a general setting of our definition, a cooperative control systems is just a system on which many controls of different nature (open–loop, feedback, quantized, etc.) act with non conflicting purposes, opposed to the model of game theory. Cooperative controls seem to be natural not only for coordination problems but also in case of competitors as in economic problems. In fact the possibility of coordination, for better common resources usage, let competitors move toward configurations that are not Nash equilibria as expected by game theoretical predictions, see [10]. The term cooperative control is sometimes used to indicate distributed controls with common cost function and our definition is the mathematical counterpart of such idea, the single controls being possibly distributed or not. Other possible applications of this methodology include air traffic management, see [11].

Both problems P.1 and P.2 can be formulated as cooperative control problems. Considering the Dubins' car–like robot model, in the first case we design a minimum time feedback to final destination and a time–varying feedback for moving obstacle avoidance. The velocity of the robot is kept constant and the robot reacts only to the presence of close obstacles. The main mathematical problems are related to the definition of trajectories for the resulting system because both feedbacks are discontinuous. We overcome this difficulty first introducing the natural definition of Krasowskii solutions to the corresponding multifunction and then selecting the *good* ones imposing robustness with respect to initial point perturbations. The methodology relies on geometrical properties of each feedback collected in the definition of Regular Stratified Feedback.

Simulation tests show good behaviour both of total time to final destination and minimum distance to obstacles for large statistics. The key parameter is a radius R: the robot reacts only if an obstacle is present at distance less than or equal to R.

The second problem P.2 is also solved via cooperative control. Each robot acts with a control algorithm similar to that constructed to P.1. The use of the very same control algorithm leads to dangerous geometrical configurations in which two or more robots enter a loop behavior. Such loops are stable and correspond to Nash type equilibria. However, if we let each vehicle slightly adjust its velocity, depending on the position of robots in its forward neighborhood, the resulting motion avoids more efficiently the previously encountered dangerous configurations and simulations confirm it with a good final time statistics.

II. BASIC DEFINITIONS

Given a set $A \subset \mathbb{R}^m$ we denote by by $\mathrm{Int}(A)$ its interior, by $\mathrm{cl}(A)$ the closure of A and by $\mathrm{co}(A)$ (respectively $\overline{\mathrm{co}}(A)$) the convex hull (respectively closed convex hull). For compact sets $A, A' \subset \mathbb{R}^m$ we indicate by $d(A, A')$ their Hausdorff distance.
Assume that E is a multifunction from $\mathbb{R}^{m'}$ to $\mathbb{R}^m$. We say that E is continuous if it is continuous for the Hausdorff distance and that it is upper semicontinuous at x if for every $\epsilon > 0$ there exists $\delta > 0$ such that $E(y) \subset E(x) + \epsilon B_m$ for every $|y - x| < \delta$. Given a multifunction E from $\mathbb{R}^m$ to $\mathbb{R}^m$, we consider the corresponding differential inclusion that is

$$\dot{x} \in E(x). \tag{1}$$

A classical result, see [12], states

Proposition 1: If E is a multifunction from $\mathbb{R}^m$ to $\mathbb{R}^m$ that is upper semicontinuous, with nonempty compact convex values, then (1) admits local (in time) solutions for every initial data.

III. COOPERATIVE CONTROLS

Consider the system:

$$\dot{x} = F(x) + hG_1(x)s + (1-h)G_2(x)v \tag{2}$$

where $x \in \mathbb{R}^n$, $F : \mathbb{R}^n \to \mathbb{R}^n$, $G_i(x) \in \mathbb{R}^{n \times p_i}$, $i = 1, 2$, $h \in [0,1]$, $s \in \mathbb{R}^{p_1}$ and $v \in \mathbb{R}^{p_2}$. We call this system a cooperative control system, since we interpret s and v as two controls of different nature (or chosen by different controllers) but with non conflicting goals. In the model of robot coordination v represents a minimum time control to final destination, while s a *safety* control that prevents from crashes with other objects (robots or obstacles) moving in the environment. We assume

A.1 F, G_1, G_2 are smooth functions,
A.2 there exists $C_i > 0$, $i = 0, 1, 2$, such that $\|F(x)\| \leq C_0(1 + \|x\|)$, $\|G_i(x)\| \leq C_i(1 + \|x\|)$,
A.3 $s \in U_s \subset \mathbb{R}^{p_1}$, $v \in U_v \subset \mathbb{R}^{p_2}$, where U_s and U_v are compact.

The safety control s should act ($h \neq 0$) only when necessary, that is when the system is close to a bad or dangerous region $\mathcal{B}(t)$ of the configuration space. The aim of s is to keep the system in a safe region $\mathcal{S}(t)$ in the most efficient way, that is maximizing the distance from $\mathcal{B}(t)$ and possibly minimizing some running cost. For robot coordination, keeping the system in a safe region means avoiding the routes of other agents.
In order to obtain this behaviour automatically we shall design h to be a smooth feedback which is equal to 1 on the bad region $\mathcal{B}(t)$ and 0 on the safe region $\mathcal{S}(t)$ and $s(t, \cdot)$ an optimal feedback that in general may be not continuous. On the other hand, v is a feedback that minimizes time, or a more complicate cost, and leads to a fixed final destination. Also v is in general discontinuous. We thus obtain a system:

$$\dot{x} = F(x) + h(t,x)G_1(x)s(t,x) + (1 - h(t,x))G_2(x)v(x) \tag{3}$$

which is comprised of an ODE with a right–hand side discontinuous both in t and x. When h is not constantly equal to 0 or 1 we immediately face the problem of defining a solution, see for example [13], even if both $\dot{x} = s(t,x)$ (t fixed) and $\dot{x} = v(x)$ admit solutions.

IV. ROBOT COORDINATION

In both situations we consider a simple model for a car–like robot usually called Dubins' car. The position of the car is identified by the coordinates (x_1, x_2) of its baricenter and the angle θ with the positive x–axis. The kinematic motion is modeled as:

$$\begin{aligned} \dot{x}_1 &= u_1 \cos(\theta) \\ \dot{x}_2 &= u_1 \sin(\theta) \\ \dot{\theta} &= u_2 \end{aligned} \tag{4}$$

where u_1 is the velocity control and u_2 is the steering control. Both velocities are physically bounded hence we assume $0 \leq u_1 \leq 1, |u_2| \leq 1$.

A. Problem P.1

The first problem P.1 consists in designing two cooperative controls v and s. The first control v should lead the robot to final destination in minimum time, thus we aim at designing v as a (discontinuous) optimal feedback depending on the position of the robot and the final destination. Clearly v corresponds to maximal velocity that is $|u_1| = 1$.
The control s is a safety control to avoid moving obstacles, therefore it must be a time–varying feedback. Also in this case we want to move at maximum velocity in the direction opposite to the obstacle thus similarly the control $s(t, x_1, x_2, \theta)$ is discontinuous in the last variables for every value of t. However these discontinuities have good geometrical properties as we see later.

B. Problem P.2

We then treat the second problem P.2 using the information from P.1. In this case we have N robots whose positions are identified by N elements $(x_1^i, x_2^i, \theta_i) \in \mathbb{R}^2 \times S^1$. Each robot evolves according to equations (4).
Since we aim at a decentralized algorithm, we let each robot act without communicating with other robots, but only detecting the position of nearby robots. Now we can not choose a fixed velocity u_1 due to the possibility of *resonant configurations* when the robots are in opposition to their final

destinations and enter a loop behaviour observed also via simulations. We thus design v and s adjusting the velocity u_1 depending on the number and position of nearby robots.

V. Good Definition of Solution for a Cooperative Control System

Given a function $e : \mathbb{R}^{m'} \to \mathbb{R}^m$ and a subspace $\Lambda \subset \mathbb{R}^{m'}$, we associate the multifunction $E^\Lambda(y) = \cap_{\delta>0} E^{\Lambda,\delta}(y) = \cap_{\delta>0}\overline{\mathrm{co}}\ e(y+\delta B_\Lambda)$. A Krasowskii solution for a discontinuous ODE

$$\dot{x} = e(x), \tag{5}$$

on $\mathbb{R}^m$, is a solution to the corresponding multifunction $E = E^{\mathbb{R}^m}$. Given bounded measurable controls $s(t,x)$ and $v(x)$, a Krasowskii admissible solution to equation (3) is a solution to the differential inclusion

$$\dot{x}(t) \in F(x)+h(t,x)G_1(x)S^x(t,x)+(1-h(t,x))G_2(x)V(x) \tag{6}$$

where S^x indicates the multifunction obtained from s taking $\Lambda = \{(0,x) : x \in \mathbb{R}^n\}$ and V the multifunction obtained from v choosing Λ to be $\mathbb{R}^n$. Even if the differential inclusion (6) admits solutions, we need to select the "good" ones. Indeed some Krasowskii solutions may be not optimal or not even reach the desired target, see [13], [14], [15].

Definition 1: A Krasowskii solution $x : [0,T] \to \mathbb{R}^m$ to (5) is called Krasowskii Cone Robust if there exist a constant $a > 0$ and a multifunction $\mathcal{K}$ from $[0,T]$ to $\mathbb{R}^m$, such that the following holds.
1. $\mathcal{K}(t)$ is a closed convex cone with $\mathrm{Int}(\mathcal{K}(t)) \neq \emptyset$,
2. the multifunction $t \mapsto \mathcal{K}(t) \cap \mathrm{cl}(B_m)$ is continuous with respect to the Hausdorff distance,
3. for every $t \in [0,T]$ and every sequence $y_n = x(t) + \epsilon_n w + o(\epsilon_n)$, with $\epsilon_n \to 0$ and $w \in \mathcal{K}(t)$, if $x_n(\cdot)$ are Krasowskii solutions with $x_n(t) = y_n$, then $x_n(\cdot)$ converge to x uniformly on $[t, \min\{t+a,T\}]$.
ϵ Krasowskii Weakly Robust Solution are defined similarly asking $\mathcal{K}(t)$ to be only a set of positive measure at $x(t)$ and the conclusion **3.** to hold for at least one sequence x_n (not all sequences). Moreover, similar definitions are given for a differential inclusion.

We can state assumptions on discontinuous optimal feedbacks under which Krasowskii Cone Robust Solutions coincide with the optimal ones. First one introduces the concept of Regular Stratified Feedback, based on that of Regular Synthesis given by Boltyanskii and Brunovsky. Then conditions are given on the corresponding cells, that are submanifolds on which the feedback is smooth, to ensure the conclusion. Regular Stratified Feedbacks that satisfy such conditions are called Krasowskii admissible. The whole construction is developed in [15].

Theorem 1: For a Krasowskii admissible Regular Stratified Feedback, Krasowskii Cone Robust Solutions coincide with optimal trajectories.

VI. P.1: Cooperative Controls for One Robot with Moving Obstacles

As explained in section IV for P.1 we can set the velocity $u_1 \equiv 1$. Thus for a single robot, a cooperative control model can be written as

$$\begin{aligned} \dot{x}_1 &= \cos(\theta) \\ \dot{x}_2 &= \sin(\theta) \\ \dot{\theta} &= hs + (1-h)v \end{aligned} \tag{7}$$

that is of type (2) with $x = (x_1, x_2, \theta)$, $F(x) = (\cos(\theta), \sin(\theta), 0)$, $G_1 = G_2 = (0,0,1)$ and $-1 \le s, v \le 1$. Conditions A.1–A.3 are thus verified. In the following, for any point $a = (a_1, a_2, \varphi)$ we denote by $\tilde{a} = (a_1, a_2)$ the component of a in $\mathbb{R}^2$, and by $<,>$ the scalar product on $\mathbb{R}^2$.
Our aim is to let the robot move to a final destination avoiding moving objects. Thus v is the minimum time control to a target point $\bar{x} = (\bar{x}_1, \bar{x}_2)$. Various results about minimum time are available (see [13], [14]). We are only interested to a region far from the final destination, hence we give a precise description of v outside a ball $B(\bar{x}, 2)$ centered at $\bar{x}$ and of radius 2, while some part of $B(\bar{x}, 2)$ is not treated. It is easy to see that we can chose $v = 0$ if the robot is already oriented in the direction toward the target point, that is when $\tilde{F}(x)$ is parallel to $\bar{x} - \tilde{x}$ and with the same versus. If $\tilde{F}(x)$ and $\bar{x} - \tilde{x}$ are not parallel then v can be chosen as

$$v(x_1, x_2, \theta) = \mathrm{sign}(<\tilde{F}(x), (\bar{x}_2 - \tilde{x}_2, \tilde{x}_1 - \bar{x}_1)>).$$

Finally if $\tilde{F}(x)$ and $\bar{x} - \tilde{x}$ are parallel but with opposite versus, then we can chose either $v = 1$ or $v = -1$.

Proposition 2: The feedback v generates a Boltyanskii–Brunovsky synthesis, that is v is a Regular Stratified Feedback.

Proof: Recall that a Boltyanskii–Brunovsky synthesis presents cells of two types: I on which a feedback is defined and II from which trajectories exit immediately. In this case there are two type I cells of dimension 3 of $\mathbb{R}^2 \times S^1$, on which $v \equiv \pm 1$. These are formed exactly by points at which $\tilde{F}(x)$ is not parallel to $\bar{x} - \tilde{x}$. These two cells are separated by other two cells of dimension 2 which are helices as in Figure 1. The first helix, which we call P_I, is a cell of type I on which the feedback v is constantly equal to zero. Trajectories that enter this cell remain inside it reaching $\bar{x}$ without turning. The second helix, called P_{II} is a cell of type II and the stratified trajectories leave this cell entering one of the two cells of dimension 3, depending on the initial choice of v to be equal to 1 or -1. ■

Proposition 3: The feedback v is a Krasowskii admissible Regular Stratified Feedback, hence Krasowskii Cone Robust Solutions coincide with Stratified Solutions.
We assume to have N moving obstacles that represent the other robots (or fixed obstacles) met along the trajectory to final destination. Hence the *safety* control $s(t,x)$ depends on some parameters $y_1 = (y_1^1, y_2^1), \ldots, y_N = (y_1^N, y_2^N)$

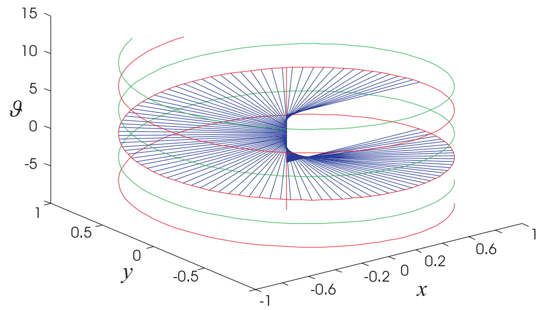

Fig. 1.

that represent the positions on the plane of the N moving obstacles. By denoting $d_i = (y_1^i - x_1, y_2^i - x_2)$ and $k = \sum_{i=1}^{N} d_i/\|d_i\|^2$, we define the control s as

$$s(t,x) = \mathrm{sign}(<(\cos(\theta), \sin(\theta)), (-k_2, k_1)>) \quad (8)$$

where k_1, k_2 are the components of k. Hence the control s forces the robot to escape from nearby obstacles, with priority given to the nearest ones. Reasoning as for v we get the following:

Proposition 4: The feedback s is a Krasowskii admissible Regular Stratified Feedback, hence Krasowskii Cone Robust Solutions coincide with Stratified Solutions.

We assume that the robot is able to avoid obstacles if the distance remains big enough, i.e. if obstacles remain outside a region described by a ball centered at $\tilde{x}$ and of radius R_0. Moreover, we define an *operating* region described by a ball centered at $\tilde{x}$ and with radius R with $R_0 < R$. This operating region may correspond either to limitation in the possibility of detecting far objects or simply to an arbitrary choice for better performance. Obstacles are not considered if outside the ball of radius R, thus we let formally $d_i = +\infty$ in the definition of s if $d_i > R$.
Regarding h, let $d_{\min} = \min_i \|d_i\|$ and $h = h(d_{\min})$ be any smooth function that is equal to 1 on a ball of center $\tilde{x}$ and radius $\tilde{R}$, with $R_0 < \tilde{R} < R$ and equal to 0 outside the ball of center $\tilde{x}$ and radius R.
The obtained system (7) can be treated by the theory developed in previous sections for systems of type (2).

Proposition 5: For the feedbacks s and v, there exists $C > 0$ such that for each ϵ Krasowskii Weakly Robust Solution γ to (7) there exists a Stratified Solution $\bar{\gamma}$ such that $\|\gamma - \bar{\gamma}\|_{C^0} \leq C\epsilon$.

In particular the above Proposition implies that ϵ Krasowskii Weakly Robust Solutions are almost optimal with an error proportional to ϵ.

VII. P.2: COOPERATIVE CONTROLS FOR N MOVING ROBOTS

We now consider N moving robots whose cooperative dynamics are given by

$$\begin{aligned} \dot{x}_1^i &= u_i \cos(\theta_i) \\ \dot{x}_2^i &= u_i \sin(\theta_i) \\ \dot{\theta}_i &= h_i s_i + (1-h_i) v_i. \end{aligned} \quad (9)$$

For minimum information constraint we let each u_i, v_i, s_i and h_i depend only on the position of nearby robots, that is on (x_1^j, x_2^j) such that $d((x_1^i, x_2^i), (x_1^j, x_2^j)) < R$ (d is the Euclidean distance of $\mathbb{R}^2$). Thus from now on we fix an index i and describe the cooperative control for the i–th robot. We implement two algorithms:
A1 We define v_i, s_i and h_i as done for the problem P.1, considering the other robots as moving obstacles, and let $u_i \equiv 1$.
A1 We define v_i, s_i and h_i as for P.1 and let u_i depend on robot seen forward.
The algorithm A1 is immediately obtained by the description given for P.1. Let us illustrate A2: we have only to specify u_i. We let s_i depend only on the position of robot that are on the forward half ball, that is, indicating by d the distance on $\mathbb{R}^2$:

$$\begin{aligned} B_f((x_1^i, x_2^i), R) = \{&(x_1, x_2) : d((x_1^i, x_2^i), (x_1, x_2)) < R, \\ &\langle(\cos(\theta_i), sin(\theta_i)), (x_1 - x_1^i, x_2 - x_2^i)\rangle \geq 0\}, \end{aligned}$$

Recalling the definition of k and equation (8), this is done replacing, in the definition of s, the vector k by that obtained counting the robot j only if its plane–position (x_1^j, x_2^j) is in the ball $B_f((x_1^i, x_2^i), R)$.
Let us now describe the function u_i. Define for each robot j the quantities: $D_{ij} = d((x_1^i, x_2^i), (x_1^j, x_2^j))$ that is the distance between robot i and robot j, and

$$c_{ij} = \frac{\langle(\cos(\theta_i), sin(\theta_i)), (x_1^j - x_1^i, x_2^j - x_2^i)\rangle}{\left((x_1^j)^2 + (x_2^j)^2\right)^{\frac{1}{2}}}$$

that is the cosine of the angle formed by the orientation of the i–th robot and the oriented distance between the two robots.
Let $J(i) = \{j : (x_1^j, x_2^j) \in B_f((x_1^i, x_2^i), R)\}$ and denote by $\mathrm{Card}(J)$ its cardinality. We set

$$u_i = 1 - \alpha \frac{\sum_{j \in J} h(D_{ij}) c_{ij}}{\mathrm{Card}(J)}$$

where $0 \leq \alpha < 1$. Thus robot i decreases its velocity depending on the number of robots in its forward ball and their positions with respect to its orientation: robots that are more in front count more. The parameter α measures the decrease rate in the velocity u_1.
Since the structure of singularities for these new cooperative controls are the same as those described for problem P.1, the set of Krasowskii Cone Robust solutions coincide with optimal ones.

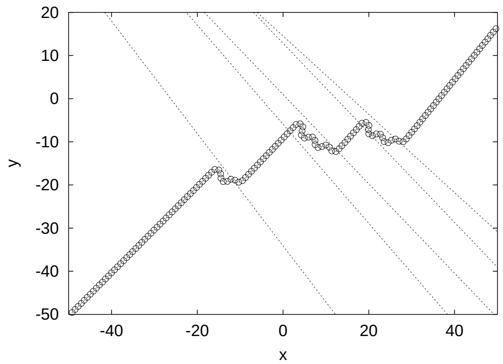

Fig. 2. Robot trajectory.

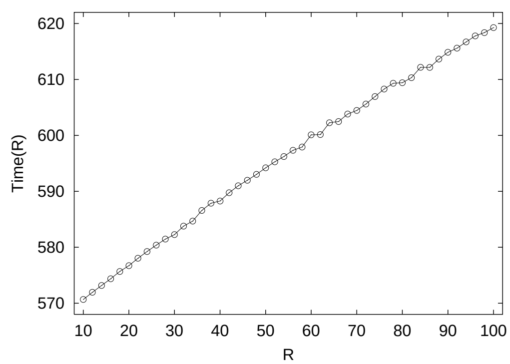

Fig. 3. Time to destination for single robot.

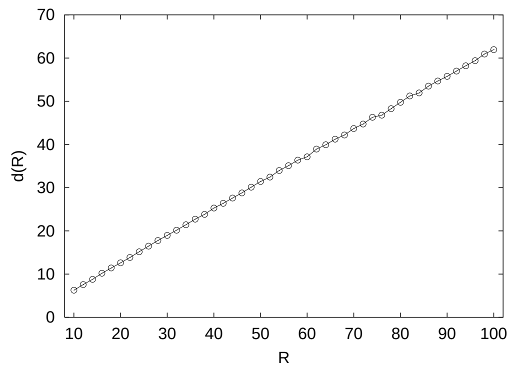

Fig. 4. Minimal distance for single robot.

Remark 1: Notice that the number of operations to compute controls u_i and v_i is quite limited. Moreover, the complexity increases linearly with the number of robots.

VIII. SIMULATIONS

A. Cooperative controls for P.1.

We fix the maneuver radius $R_0 = 4$, set $\tilde{R} = R - 4$ and consider the trajectory of a robot in presence of 10 moving obstacles with velocity around $1/2$ of that of the robot, whose trajectories are represented in Figure 2. The thickest one is the trajectory of the robot while the others are trajectories of some obstacles met.

The outputs, we are interested in, are the effective time to final destination and the minimal distance to an obstacle reached during the execution of the trajectory. Statistics is developed with random initial and final positions of moving obstacles for ten thousands realizations and obstacles trajectories almost perpendicular to that of the robot. Figure 3 represents time to destination as function of the parameter R that measures the operating region. The time to destination without obstacles is around 565, notice that the time to destination behaves linearly with respect to R. The same happens for the minimal distance and Figure 4 shows the minimal distance as function of R.

B. Cooperative controls for P.2.

We consider both algorithms A.1 and A.2 for the problem of five robots that start at random initial positions on the boundary of a circular region of radius 100 and with final positions on points symmetric with respect to the center. Thus robots enter in conflict near the center.
First we show the presence of resonant configurations: some robots are in opposition to their final destination and enter loops. In Figure 5 we show an example with two robots for algorithm A.1. Initial and final configurations for one robot are circles and squares for the other.

Then we pass to compare algorithms A.1 and A.2 for statistics with ten thousands realizations. We measure two key outputs: the time to final destination for the first arriving robot and for the last arriving robot. The difference between the two algorithms is striking. We represent the comparison for first arriving robot time in Figure 6 and for last arriving robot time in Figure 7 again as functions of the operating radius R.

IX. CONCLUSIONS

In this paper, two cases of robot coordination are illustrated: that of a single robot with N moving obstacles and N moving robots with given initial positions and final destinations. We developed cooperative control algorithms in case of minimum communication: each agent can only detect positions of nearby agents. For both cases it is considered the model of Dubins' car–like robot. The mathematical problems are related to the definition of solutions in presence of discontinuous feedbacks and are solved via the geometric theory of stratified feedbacks.
For a single robot a satisfactory behaviour is reached with fixed velocity, while for N robots a variable velocity is introduced to avoid resonant configurations. Simulation tests confirm theoretical predictions.
The dependence of performance from communication should be subject of future investigations.

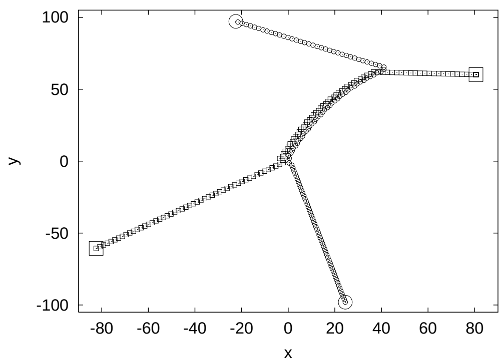

Fig. 5. Resonant configurations for A.1.

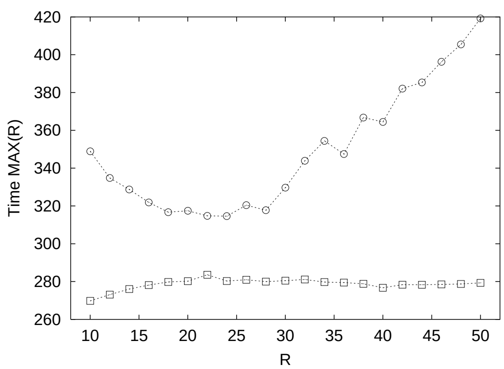

Fig. 6. A.1 vs A.2: first arriving robot. Circles for A.1 and squares for A.2.

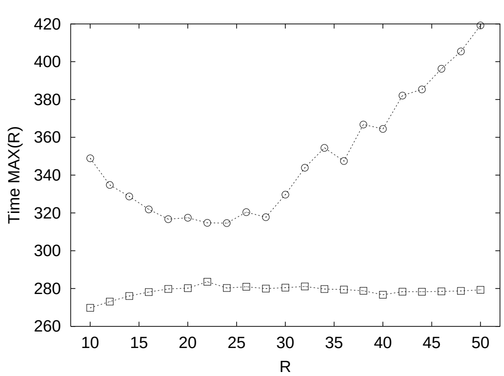

Fig. 7. A.1 vs A.2: last arriving robot. Circles for A.1 and squares for A.2.

X. REFERENCES

[1] J. Cortes and F. Bullo, "From geometric optimization and nonsmooth analysis to distributed coordination algorithm," in *Proc. IEEE Int. Conf. on Decision and Control*, 2003.

[2] M. Egerstedt and X. Hu, "Formation constrained multi agent control," *IEEE Trans. on Robotics and Automation*, vol. 17, no. 6, pp. 947–951, 2001.

[3] J. A. Fax and R. Murray, "Information flow and cooperative control of vehicle formatio," in *XV World Congress on Automatic Control b'02*, 2002.

[4] N. E. Leonard and E. Fiorelli, "Virtual leaders, artificial potentials and coordinated control of groups," in *Proc. IEEE Int. Conf. on Decision and Control*, 2001.

[5] C. Tomlin, G.J. Pappas, and S.Sastry, "Conflict resolution for air traffic management: A study in multiagent hybrid systems," *IEEE Trans. on Automatic Control*, vol. 43, no. 4, pp. 509–521, 1998.

[6] Y. Yamaguchi, T. Atai, and G. Beni, "A distributed control scheme for multiple robot vehicles to make group formation," *Robotics and autonomous systems*, vol. 36, pp. 125–147, 2001.

[7] A. Bicchi, A. Marigo, and B. Piccoli, "On the reachability of quantized control systems," *IEEE Trans. on Automatic Control*, vol. 47, no. 4, pp. 546–563, 2002.

[8] M. Egerstedt, "Motion description languages for multi–modal control in robotics," in *Proc. Int. Workshop on Hibrid Systems: Computation and Control*, 2003.

[9] S. K. Mitter, "Control with limited information," *European Journal of Control, Special issue on Fundamental issues in control*, vol. 7, pp. 122–131, 2001.

[10] H. I. Meinhardt, "Cooperative decision making in common pool situations," in *Lecture notes in economics and mathematical systems*, number 517. Springer Verlag, Berlin–Heidelberg, 2002.

[11] A. Marigo and B. Piccoli, "Cooperative control for air traffic management," *Communication in Pure and Applied Analysis*, 2003, in print.

[12] J.P. Aubin and A. Cellina, *Differential Inclusions*, Springer-Verlag, 1984.

[13] A. Marigo and B. Piccoli, "Safety controls and applications to the dubins' car," *Nonlinear Differential Equations and Applications*, 2002, in print.

[14] A. Marigo and B. Piccoli, "Safety driving for the dubins' car," in *XV World Congress on Automatic Control b'02*, 2002.

[15] A. Marigo and B. Piccoli, "Regular synthesis and solution to discontinuous ode," *ESAIM Control Optimisation and Calculus of Variations*, vol. 7, pp. 291–307, 2002.