Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

A Novel Coordination Scheme Applied to Nonholonomic Mobile Robots

Nusrettin Gulec and Mustafa Unel

Abstract— **The coordinated motion of a group of autonomous mobile robots for the achievement of a goal has been of high interest in the last decade. Previous research has revealed that one of the essential problems in the area is to avoid collision of the robots with obstacles and each other while they still achieve the goal. In this work, we develop a novel coordination scheme for a group of mobile robots and a new online collision avoidance algorithm. Several coordinated tasks have been presented and the proposed scheme is verified by simulations.**

I. INTRODUCTION

There has been a growing interest in modeling groups of autonomous mobile robots engaged in coordinated behavior [1] - [7].

Coordinated task manipulation by a group of mobile robots is the accomplishment of a specified task together in certain formations. The necessary formation may vary from scenario to scenario [8]. A rectangular formation could be better to carry a heavy rectangular object by a group of autonomous robots whereas circular formations might be better to provide security in surveillance areas by capturing and enclosing the target or the invader [9].

Achievement of a goal by a group of autonomous robots acting in a coordinated manner is an example of decentralized systems. Failure of one of the components leads to system failure in centralized systems, whereas this is not necessarily the case in decentralized systems. Economic cost of a centralized system is usually higher than a decentralized system that can carry out the same task [10]. A huge single robot, no matter how powerful it is, is spatially limited while the same task could be accomplished by smaller robots more efficiently. Natural groupings such as schooling fish, flocking birds or bees building a honeycomb in the beehive are all decentralized systems where each member works in coordination with the others in the group. Decentralized schemes outclass centralized schemes in situations where the goal is the exploration of an area for some task such as a search and rescue action [11].

In literature, it is common to define the coordinated behavior on the basis of mutual forces among the autonomous robots. In 1995, Vicsek proposed the simple "neighbors" method to model the coordinated motion of bacteria [12]. It has recently been shown that the group will break into subgroups after some finite time in the absence of other factors [13]. Yamaguchi proposed a coordination scheme by defining "formation vectors" for the achievement of certain formations and forces to avoid collisions and approach the target for a group of mobile holonomic [9] and nonholonomic [14] robots to enclose a target by a circular formation.

There has been a steadily growing literature towards vision-based control algorithms for mobile robots [15] - [21]. It has recently been shown that visual sensing is advantageous over other sensing strategies in [16].

The organization of this paper is as follows. In section II, we describe a coordinated task scenario. In section III, we present a novel coordination method by introducing a "*virtual reference model*", which in turn implies online collision-free reference trajectories for autonomous mobile robots. Section IV is on the feedback control of nonholonomic robots. In section V, the simulation results are presented to verify the framework we introduce. In section VI, we conclude with some remarks and suggest some future work.

II. A COORDINATED TASK SCENARIO

We consider n autonomous nonholonomic mobile robots and an object that will serve as a target for the group. We assume all of the robots are in the same plane with the the target. We will denote i^{th} robot in the group by R_i and the target object by T.

The coordinated task in our scenario is to enclose T by n nonholonomic mobile robots, namely $(R_1, R_2, \ldots, R_i, \ldots, R_{n-1}, R_n)$, by means of a circular formation around it. The robots form the circle in such a way that each R_i keeps its distance from the target at some predefined value d_{target}. In the desired formation, the robots will be uniformly distributed on the circle so that each R_i keeps its distance from its closest neighbors at some predefined value $d_{formation}$ and orients itself towards the target.

Aforementioned formation is a general one that would be the basis for a sequence of coordinated tasks. Manipulation of a heavy object, T, by a mobile robot group could be one example. Once they achieve the desired formation described above, they can grasp and move the object in a coordinated fashion. Another example could be enclosing and catching a prisoner, T, in a surveillance area by decreasing the distances d_{target} and $d_{formation}$ after the explained formation is achieved. Depending on the nature of the coordinated task, R_i might need to check if the remaining $(n-1)$ robots have accomplished the current task before it goes on with the next.

During each phase of the coordinated motion, autonomous robots must avoid collisions with others and obstacles. Collision is one of the central problems in the context of coordinated motion as pointed out in [20].

Nusrettin Gulec is with the Faculty of Engineering and Natural Sciences, Sabanci University, Orhanli-Tuzla 34956, Istanbul, Turkey nusrettin@su.sabanciuniv.edu

Mustafa Unel is with the Faculty of Engineering and Natural Sciences, Sabanci University, Orhanli-Tuzla 34956, Istanbul, Turkey munel@sabanciuniv.edu

0-7803-9568-9/05/$20.00 ©2005 IEEE

In this paper, we are assuming a stationary target, T, the pose information of which is given to all robots beforehand. We are also assuming that each robot, R_i, can sense at least two of the other robots at each instant; i.e. the underlying sensing graph is connected.

III. VIRTUAL REFERENCE MODEL FOR TRAJECTORY GENERATION

Coordination in a group of mobile robots can be defined on the basis of forces between the robots and the target. However, the control of nonholonomic mobile robot is based on its kinematic model. To achieve coordinated motion using forces, we introduce a virtual reference model for each robot that moves under robot-robot coordination forces and target-robot attraction force. Actual robot is forced to track the reference trajectory generated by the virtual reference model by implementing a robust low level controller.

There are several possibilities to model the virtual reference system. For example, one could use *electrostatic forces* to model coordination forces between *electrical charges* or *gravitational forces* between *masses*.

A. Virtual Masses

The reference model we propose consists of *virtual masses* that are interconnected via virtual springs and dampers; i.e. *virtual bonds*. Our virtual mass-spring-damper model is analogous to the molecules formed by chemical bonds between atoms. The forces generated on the *virtual springs and dampers* are responsible for the coordinated motion of the virtual masses. Position and velocity of these virtual masses are then input as references to the low level controller.

Virtual masses can be either point particles or finite size shells as discussed below.

1) Point Particles: Point particles of finite masses that are interconnected via the virtual bonds described above can be used to generate a reference trajectory for the nonholonomic mobile robots. Since a point particle is holonomic, this approach relaxes the nonholonomic constraint. Fig. 1(a) depicts a virtual reference system for 3 robots.

2) Shells: Geometric shells, i.e. circular or elliptical in 2D and spherical or ellipsoidal in 3D, of finite size and mass can also be used as references for the robots. This approach is depicted in Fig. 1(b).

In this work, we use point masses to create virtual reference models for the robots. Since there are n robots, n virtual masses, $(m_1, m_2, ..., m_{n-1}, m_n)$, are generated on the computer.

B. Virtual Forces

We consider a biologically inspired coordination scheme where we take two closest neighbors into account as shown in Fig. 2.

The coordination force exerted on m_i by its two closest neighbors, m_{c1} and m_{c2}, is defined as:

$$\begin{array}{r}\mathbf{F}_{coord} = -k_{coord}\left((d_{c1} - d_{coord})\,\mathbf{n}_{c1} + (d_{c2} - d_{coord})\,\mathbf{n}_{c2}\right) \\ -c_{coord}\left(\left(\left(\dot{\mathbf{X}}_i - \dot{\mathbf{X}}_{c1}\right)\bullet\mathbf{n}_{c1}\right)\mathbf{n}_{c1} + \left(\left(\dot{\mathbf{X}}_i - \dot{\mathbf{X}}_{c2}\right)\bullet\mathbf{n}_{c2}\right)\mathbf{n}_{c2}\right)\end{array}, \tag{1}$$

where $\bullet$ denotes vector dot product, k_{coord} and c_{coord} are spring and damping coefficients, respectively, $\dot{\mathbf{X}}_i = \left[\begin{array}{cc} \dot{x}_i & \dot{y}_i \end{array}\right]^t$ is the velocity vector of virtual mass m_i, d_{c1} is the signed distance between m_i and m_{c1}, $\mathbf{n}_{c1}$ is the unit vector from m_{c1} to m_i, $\dot{\mathbf{X}}_{c1} = \left[\begin{array}{cc} \dot{x}_{c1} & \dot{y}_{c1} \end{array}\right]^t$ is the velocity vector of virtual mass m_{c1}, d_{c2} is the signed distance between m_i and m_{c2}, $\mathbf{n}_{c2}$ is the unit vector from m_{c2} to m_i, $\dot{\mathbf{X}}_{c2} = \left[\begin{array}{cc} \dot{x}_{c2} & \dot{y}_{c2} \end{array}\right]^t$ is the velocity vector of virtual mass m_{c2} with d_{coord} being the distance to be maintained among the neighbors for coordination.

Similarly, we are forming virtual bonds between the target and the virtual masses. The force exerted by T on m_i is defined as:

$$\mathbf{F}_{target} = -k_{target}\,(d_T - d_{target})\,\mathbf{n}_T - c_{target}\left(\dot{\mathbf{X}}_i \bullet \mathbf{n}_T\right)\mathbf{n}_T\ , \tag{2}$$

where d_T is the signed distance between m_i and T, $\mathbf{n}_T$ is the unit vector from T to m_i with d_{target} is the distance to be maintained from the target in the desired final formation.

The total force acting on m_i is obtained as the sum of $\mathbf{F}_{coord}$ in (1) and $\mathbf{F}_{target}$ in (2). Consequently, the equation of motion for the virtual mass, m_i - the position values, x_i and y_i, obtained by the solution of which will be used as the reference signals for the robot, R_i - will be as follows:

$$m_i \left[\begin{array}{c} \ddot{x}_i \\ \ddot{y}_i \end{array}\right] = \mathbf{F}_{coord} + \mathbf{F}_{target}\ . \tag{3}$$

To achieve coordinated tasks by formations, adaptable springs and dampers between virtual masses might be required. For example, the coefficient of the spring between masses can be changed in a smooth manner. In our scenario, we split the coordinated task into two main phases: (1) approaching T; and (2) achieving a circular formation around T with radius d_{target}.

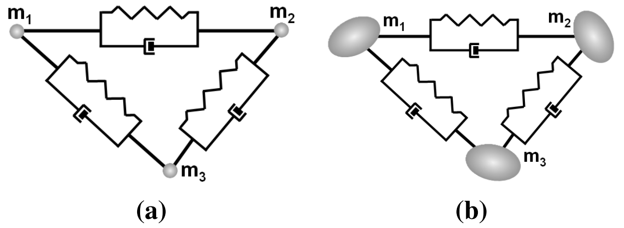

Fig. 1. Possible virtual masses: (a)Point particles (b)Geometric shells

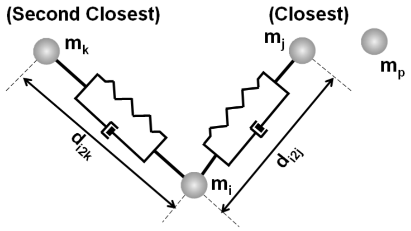

Fig. 2. Closest two neighbors for definition of coordination force

C. Adaptable Model Parameters

In phase (1), i.e. the virtual masses approach the target from some initial formation, the priority is given to coordinated motion. $\mathbf{F}_{target}$ acts on the virtual masses and attracts them towards T. However, $\mathbf{F}_{coord}$ in this phase is dominant so they move together. As long as the distance between m_i and T is greater than a predefined value d_{break}, k_{coord} is set to be $k_{approach}$, which is higher than k_{target}. c_{coord} has some finite value in this phase so that the virtual masses are forced to move to the same direction.

In phase (2), the distance between the virtual mass m_i and T is lower than d_{break}, constants of coordination forces are decreased; hence $\mathbf{F}_{target}$ becomes dominant. Since the main property of the final formation is to keep m_i at distance d_{target} from T for all virtual masses, the constraint dictated by $\mathbf{F}_{coord}$ should be relaxed. For example, c_{coord} is reduced to zero so that m_i need not move in the same direction with its neighbors. The equilibrium length of the springs between the masses is changed from the initially specified value of $d_{approach}$ to $d_{formation}$. It follows from *Law of Cosines* that, for a uniform placement of n masses on a circle of radius d_{target}, $d_{formation}$ is given as:

$$d_{formation} = d_{target}\sqrt{2(1-\cos(2\pi/n))}\,. \tag{4}$$

When m_i is closer than d_{break} to T, it performs circular motion because $\mathbf{F}_{target}$ is acting on it in the radial direction. k_{coord} is reduced to some finite value, $k_{formation}$, rather than zero so that the virtual masses are forced to distribute on the formation circle uniformly. Otherwise, m_i would stop when it achieves its distance from T equal to d_{target}. For $\mathbf{F}_{target}$ to be the dominant force, $k_{formation}$ must be less than k_{target}.

When the virtual masses and therefore robots are uniformly distributed around T on a circle of radius d_{target} based on the framework given above, the last maneuver that robots should take is to head towards the target.

k_{coord} is reduced as a function of distance from T. This gives a smooth change in $\mathbf{F}_{coord}$ and avoids discontinuity. Otherwise the discontinuity in $\mathbf{F}_{coord}$ might yield a big jump in the velocity of m_i, and R_i may not be able to follow such a change due to the nonholonomic constraint. k_{coord} is modeled by the sigmoid function defined by

$$k_{coord} = k_{formation} + \frac{k_{approach} - k_{formation}}{1+\exp(\mu(d_{break} - d_T + \phi))}, \tag{5}$$

where μ and ϕ are the constants of the sigmoid model, and d_T is the signed distance between m_i and T. Switching of k_{coord} is depicted in Fig. 3.

D. Velocity Update of Virtual Masses to Avoid Collisions

Collision avoidance is the last factor contributing to the generation of reference trajectories for the robots. In this work, we are not considering any static obstacles. We are also not considering T as an obstacle because $\mathbf{F}_{target}$ keeps the virtual masses at distance d_{target} from T.

The developed algorithm uses sensory information coming from the robots to predict collisions ahead of time, updates the velocities of the virtual masses to avoid the possible collisions *online*. For each R_i, we define a *virtual collision prediction region (VCDA)*, ψ_i, given by a circular arc of radius r_{VCDA} and angle θ_{VCDA}, symmetric with respect to its velocity as depicted in Fig. 4(a).

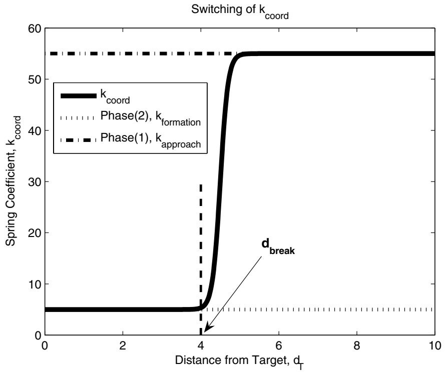

Fig. 3. Adaptive spring coefficient, k_{coord} vs distance from T, d_T

R_i, detects a collision risk when any of the other members of the group touches its virtual region, ψ_i. In case of such a collision risk detection, the velocity of m_i is updated to safely avoid the collision. After the velocity of m_i is updated, it moves with that constant velocity for a short period of time. The forces described above start to act on m_i after this time delay.

The final velocity of m_i after the velocity update is designed based on the relative velocity of the sensed robot, R_j, with respect to R_i. The coordinate frame attached to R_i and two parts, R_ψ and L_ψ, as shown in Fig. 4(b) are generated for the collision avoidance algorithm. R_i checks if any of the sensed robots touches ψ_i; i.e. if there's a collision risk, at each computational step. If a collision risk is detected, the following algorithm is run:

- Calculate the component of the velocity of R_j projected on the axis y_i.
- Rotate counter-clockwise if that component is positive.
- Rotate clockwise if that component is negative.
- If that component is zero:
    - Rotate counter-clockwise if R_j touches R_ψ.
    - Rotate clockwise if R_j touches L_ψ.

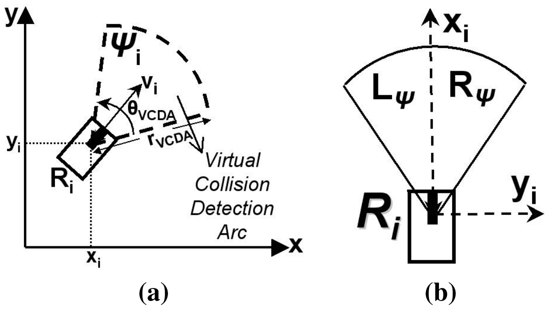

Fig. 4. (a)*VCDA*, ψ_i, for the robot, R_i (b)R_i's coordinate frame and parts of *VCDA*

IV. NONHOLONOMIC MOBILE ROBOTS: MODELING AND CONTROL

The low level controller runs on R_i to track the position and velocity of the computer-generated reference m_i. We are assuming two-wheeled nonholonomic mobile robots, i.e. unicycle robots. The kinematic model for these robots is given by [22]

$$\dot{x} = u_1 \cos\theta \ , \quad \dot{y} = u_1 \sin\theta \ , \quad \dot{\theta} = u_2 \ , \tag{6}$$

where x and y are the Cartesian coordinates of the center of mass of the robot, and θ is its orientation with respect to the horizontal axis.

The controls u_1 and u_2 in (6) are related to the velocities of the centers of right and left wheels, u_R and u_L respectively, with

$$u_1 = \tfrac{1}{2}\left(u_R + u_L\right) \ , \quad u_2 = \tfrac{1}{2\lambda}\left(u_R - u_L\right) \ , \tag{7}$$

where λ is the half length of the wheel axis as shown in Fig. 5.

For time varying reference trajectory tracking, the reference trajectory must be selected to satisfy the nonholonomic constraint. The reference trajectory is hence generated using a *virtual reference robot* [23] which moves according to the model

$$\dot{x}_r = u_{1r} \cos\theta_r \ , \quad \dot{y}_r = u_{1r} \sin\theta_r \ , \quad \dot{\theta}_r = u_{2r} \ , \tag{8}$$

where $\left[\begin{array}{ccc} x_r & y_r & \theta_r \end{array}\right]^t$ is the reference pose obtained from the virtual masses. We are assuming that x_r, y_r and θ_r are continuously differentiable and bounded in as $t \longrightarrow \infty$. Under this assumption, one can easily show that

$$u_{1r} = \dot{x}_r \cos\theta_r + \dot{y}_r \sin\theta_r \ , \qquad u_{2r} = \frac{\ddot{y}_r \dot{x}_r - \ddot{x}_r \dot{y}_r}{\dot{x}_r^2 + \dot{y}_r^2} \ , \qquad \theta_r = \arctan\left(\frac{\dot{y}_r}{\dot{x}_r}\right) \ . \tag{9}$$

It is shown in [23] that for time-variant reference trajectories the following control regulates the error to zero

$$\left[\begin{array}{c} u_1 \\ u_2 \end{array}\right] = \left[\begin{array}{c} -k_1 e_1 + u_{1r} \cos e_3 \\ -u_{1r} \frac{\sin e_3}{e_3} - k_2 e_3 + u_{2r} \end{array}\right] \ , \tag{10}$$

where $k_1 > 0$ and $k_2 > 0$ are positive and constant control gains, e_1, e_2 and e_3 are transformed errors in terms of tracking errors $\tilde{x} = x - x_r$, $\tilde{y} = y - y_r$ and $\tilde{\theta} = \theta - \theta_r$ whereas u_{1r} and u_{2r} are the control inputs of the reference robot given in (8).

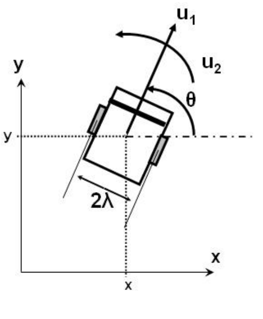

Fig. 5. A unicycle robot and its variables of interest

Parking the robot at a fixed reference point is yet another control problem and the control given in (10) doesn't regulate the error to zero in this case. The following controls were proved to stabilize the system for fixed point references [23].

$$\left[\begin{array}{c} u_1 \\ u_2 \end{array}\right] = \left[\begin{array}{c} -k_1 e_1 \\ -k_2 e_3 + e_2^2 \sin(t) \end{array}\right] \ , \tag{11}$$

where $k_1 > 0$ and $k_2 > 0$ are positive and constant control gains, e_1, e_2 and e_3 are the transformed errors.

Since we are using the trajectory of the virtual masses as references for the robots, we are implementing the control given by (10) until the equilibrium on the formation circle is reached. The control given by (11) then enables robots to park on the formation circle, headed towards T.

V. SIMULATION RESULTS

The virtual masses used in simulations have $m_i = 1.0kg$. The maximum linear velocity of the masses were set to $(0.5)m/sec$, while the angular velocity was saturated by $(\pi)rad/sec$. The virtual collision detection arc has radius $r_{VCDA} = (0.45)m$ and angle $\theta_{VCDA} = (\pi/2)rad$. The constants in the control laws were set as: $k_1 = k_2 = 20$ in (10), and $k_1 = 23$, $k_2 = 16$ in (11).

A. Simulation of Collision Avoidance

The simulations were run with $\mathbf{F}_{target} = \mathbf{F}_{coord} = 0$ to verify the performance of the collision avoidance algorithm.

In the first scenario, the robots are moving towards each other with constant velocities as seen in Fig. 6(a). Fig. 6(b) shows the moment of collision prediction. Their new situation is given in Fig. 6(c) after they avoid the collision by taking the necessary action. Since both robots touch the other's *VCDA*, they both change orientations. The result proves the success of the algorithm for head-to-head collisions.

Fig. 7(a) shows the robots before any collision is predicted for this scenario where one of the robots is approaching the other from its right hand side. Only one of the robots, A, detects the risk of a collision, as shown in Fig. 7(b). The new headings of the robots, after A changes its orientation, are depicted in Fig. 7(c). The result proves the success of the collision avoidance algorithm in this situation as well.

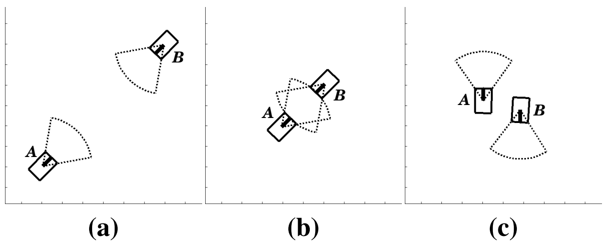

Fig. 6. Head-to-Head Collision Avoidance: (a)Before (b)Prediction (c)After

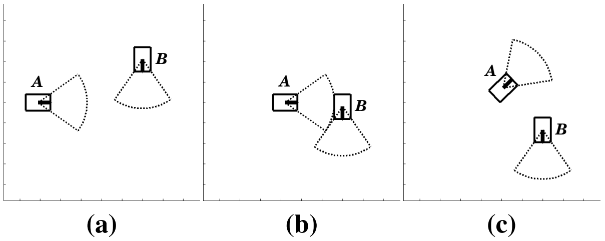

Fig. 7. Single-Robot Collision Avoidance: (a)Before (b)Prediction (c)After

B. Simulation of Coordinated Motion

The coordinated motion method was simulated on the basis of the specified coordinated task. The presented method was simulated for two distinct initial configurations for three robots and a single initial configuration for four robots. The constants were set as: $c_{coord} = 10Ns/m$ and $d_{approach} = 2.0m$ in (1), and $k_{target} = 15.0N/m$, $c_{target} = 10.0Ns/m$ in (2); and $\mu = 10$, $\phi = 0.5$ and $d_{break} = 1.3d_{target}$ in (5).

1) Three Autonomous Mobile Robots: Since virtual bonds are constructed for each m_i with its closest two neighbors, mutual coordination forces among each binary combination of the robots exist in these simulations.

The initial configuration in the first scenario is such that the robots and the target are placed on opposite corners of the room as depicted in Fig. 8(a). They detect and avoid the collisions and move in a coordinated fashion in the form of an equilateral triangle with sides $d_{approach}$. The robots start circular motion since the virtual bonds are relaxed and $\mathbf{F}_{target}$ becomes dominant in phase (2). Finally, they take the form of an equilateral triangle with sides equal to $d_{formation}$. Snapshots from the animation are given in Fig. 8.

The initial setting of the second scenario with three robots is depicted in Fig. 9(a). In this case, the robots first move towards each other and then approach T as a group. This is because $k_{coord} = k_{approach} > k_{target}$ initially. After the robots form the triangle seen in Fig. 9(b), they move in a coordinated manner and achieve the desired formation.

2) Four Autonomous Mobile Robots: Since virtual bonds are constructed for each m_i with its closest two neighbors, the farthest robot's virtual mass with respect to m_i doesn't apply any force on m_i in cases where number of robots is higher than three such as these simulations.

In the third scenario, the robots and the target are placed on opposite corners of the room as in Fig. 10(a). Fig. 10(b) shows their motion in the form of a parallelogram with sides equal to $d_{approach}$. Fig. 10(c) is a snapshot depicting the circular motion of the robots around T under the dominant effect of $\mathbf{F}_{target}$. Finally, they take the form of a square with sides $d_{formation}$ as seen in Fig. 10(d).

The results of the simulations for the proposed coordination scheme with three and four robots are all satisfactory in the sense that the robots move in a coordinated manner, detect and avoid any possible collisions and form a uniform polygon with sides equal to $d_{formation}$, with each robot heading towards the target, where the center of the peripheral circle of that polygon is the target.

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

We have developed a decentralized coordinated motion scheme for a group of nonholonomic autonomous mobile robots by introducing a virtual reference model which updates its reference trajectories based on feedback from sensory

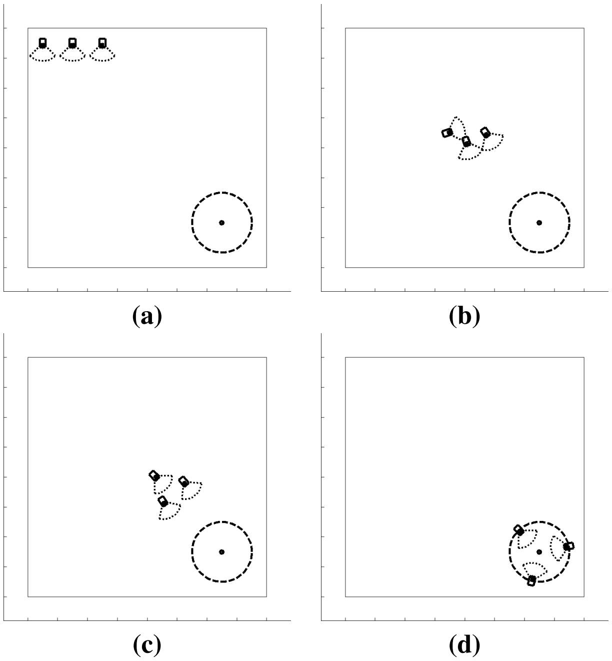

Fig. 8. Scenario-1: (a)Initial configuration (b)Collision avoidance (c)Coordinated motion (d)Desired formation achieved

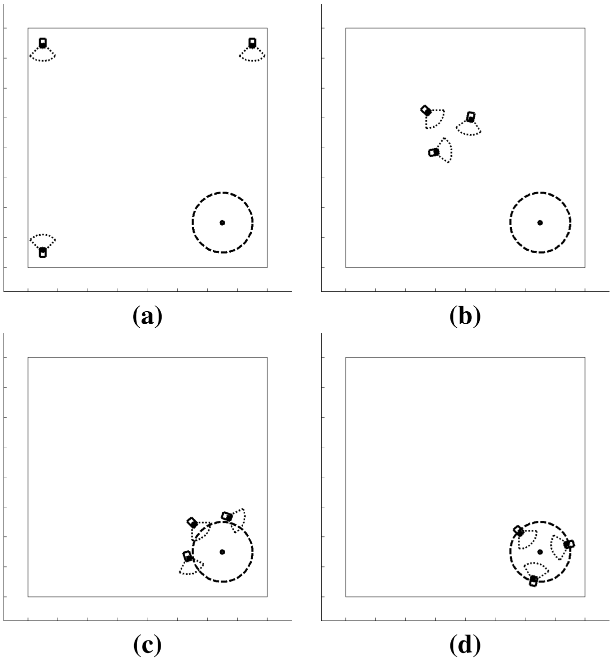

Fig. 9. Scenario-2: (a)Initial configuration (b)Coordination dominant (c)Virtual bonds relaxed (d)Desired formation achieved

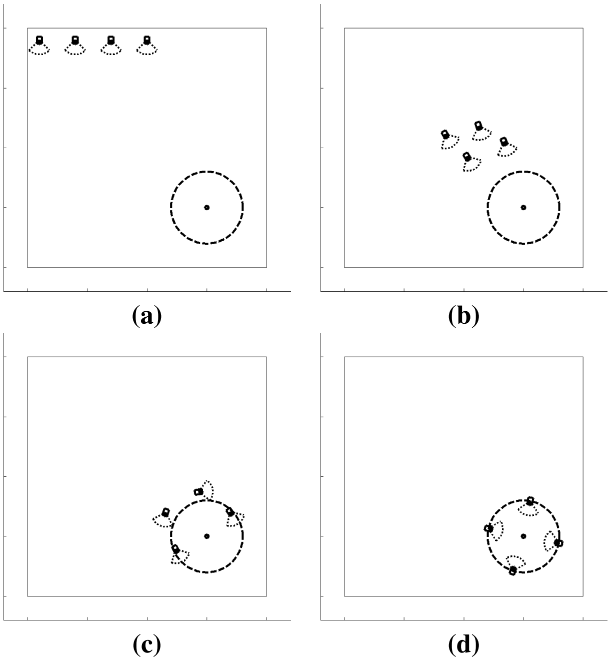

Fig. 10. Scenario-3: (a)Initial configuration (b)Coordinated motion (c)Virtual bonds relaxed (d)Desired formation achieved

inputs in the case of collisions. Collisions are detected online by virtual collision detection arcs. The position and velocity of the virtual point masses were then used as references for the robots.

Simulation results are promising. The proposed coordinated motion scheme with 3 or 4 robots are quite successful. Since the proposed algorithm is modular, the success of the system would not be affected by increasing the number of the robots in the group provided the robots can always sense and distinguish their closest two neighbors. The low-level controller proved to be robust enough since the robots follow the references although their velocities are saturated at certain limits.

B. Future Works

We are working on the physical implementation of the proposed coordination scheme with two-wheeled mobile robots equipped with vision sensors. We are also doing work with finite size shells mentioned in this paper but not considered in detail.

REFERENCES

[1] P. Seiler, A. Pant, K. Hedrick, "Analysis of Bird Formations", *Proceedings of the 41st IEEE Conference on Decision and Control*, Vol. 1, December 2002, 118-123.

[2] A. Borkowski, M. Gnatowski, J. Malec, "Mobile Robot Cooperation in Simple Environments", *Proceedings of the Second International Workshop on Robot Motion and Control*, October 2001, 109-114.

[3] S. Souissi, X. Defago, T. Katayama, "Decomposition of Fundamental Problems for Cooperative Autonomous Mobile Systems", *Proceedings of the 24th International Conference on Distributed Computing Systems*, March 2004, 554-560.

[4] A. Jadbabaie, J. Lin, A.S. Morse, "Coordination of Groups of Mobile Autonomous Agents Using Nearest Neighbor Rules", *Proceedings of the 41st IEEE Conference on Decision and Control*, Vol. 3, December 2002, 2953-2958.

[5] N. Sarkar, T.K. Podder, "Coordinated Motion Planning and Control of Autonomous Underwater Vehicle-Manipulator Systems Subject to Drag Optimization", *IEEE Journal of Oceanic Engineering*, Vol. 26, No. 2, April 2001, 228-239.

[6] J.B. de Sousa, F.L. Pereira, "Specification and design of coordinated motions for autonomous vehicles", *Proceedings of the 41st IEEE Conference on Decision and Control*, Vol. 1, December 2002, 101-106.

[7] J. Shen, D.A. Schneider, A.M. Bloch, "Controllability and Motion Planning of Multibody Systems with Nonholonomic Constraints", *Proceedings of the 42nd IEEE Conference on Decision and Control*, Vol. 5, December 2003, 4369-4374.

[8] Y. Hur, R. Fierro, I. Lee, "Modeling Distributed Autonomous Robots using CHARON: Formation Control Case Study", *Sixth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing, ISORC '03*, May 2003, 93-96.

[9] H. Yamaguchi, "A Cooperative Hunting Behavior by Mobile-Robot Troops", *The International Journal of Robotics Research*, Vol. 18, No. 8, September 1999, 931-940.

[10] J. Spletzer, A.K. Das, R. Fierro, C.J. Taylor, V. Kumar, J.P. Ostrowski, "Cooperative Localization and Control for Multi-Robot Manipulation", *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 2, October 2001, 631-636.

[11] W. Sheng, Q. Yang, J. Tan, N. Xi, "Risk and Efficiency: A Distributed Bidding Algorithm for Multi-robot Coordination", *Proceedings of the Fifth World Congress on Intelligent Control and Automation, WCICA 2004*, Vol. 5, June 2004, 4671-4675.

[12] T. Vicsek, A. Czirk, E. Ben-Jacob, I. Cohen, O. Shochet, "Novel Type of Phase Transition in a System of Self-Driven Particles" *Phys. Rev. Lett.*, Vol. 75, No. 6, August 1995, 1226-1229.

[13] A.V. Savkin,"Coordinated Collective Motion of Groups of Autonomous Mobile Robots: Analysis of Vicsek's Model", *IEEE Transactions on Automatic Control*, Vol. 49, No. 6, June 2004, 981-982.

[14] H. Yamaguchi, "A Cooperative Hunting Behavior by Multiple Nonholonomic Mobile Robots", *1998 IEEE International Conference on Systems, Man, and Cybernetics*, Vol. 4, October 1998, 3347-3352.

[15] A.K. Das, R. Fierro, V. Kumar, J.P. Ostrowski, J. Spletzer, C.J. Taylor, "A Vision-Based Formation Control Framework", *IEEE Transactions on Robotics and Automation*, Vol. 18, No. 5, October 2002, 813-825.

[16] K.N. Kutulakos, C.R. Dyer, V.J. Lumelsky, "Provable Strategies for Vision-Guided Exploration in Three Dimensions", *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, Vol. 2, May 1994, 1365-1372.

[17] K. Han, M. Veloso, "Reactive Visual Control of Multiple Non-Holonomic Robotic Agents", *Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, Vol. 4, May 1998, 3510-3515.

[18] Z. Hai-bo, Y. Kui, L. Jin-dong, "A Fast and Robust Vision System for Autonomous Mobile Robots", *Proceedings of the 2003 IEEE International Conference on Robotics, Intelligent Systems and Signal Processing*, Vol. 1, October 2003, 60-65.

[19] A.K. Das, R. Fierro, V. Kumar, B. Southhall, J. Spletzer, C.J. Taylor, "Real-Time Vision-Based Control of a Nonholonomic Mobile Robot", *Proceedings of the 2001 IEEE International Conference on Robotics and Automation, ICRA '01*, Vol. 2, May 2001, 1714-1719.

[20] T. Rabie, A. Shalaby, B. Abdulhai, A. El-Rabbany, "Mobile Vision-based Vehicle Tracking and Traffic Control", *Proceedings of the Fifth IEEE International Conference on Intelligent Transportation Systems*, September 2002, Singapore, 13-18.

[21] H.S. Oh, C.W. Lee, I. Mitsuru, "Navigation Control of a Mobile Robot based on Active Vision", *1991 International Conference on Industrial Electronics, Control and Instrumentation, IECON '91*, Vol. 2, October 1991, 1122-1126.

[22] B. Kwolek, T. Kapuscinski, M. Wysocki, "Vision-based implementation of feedback control of unicycle robots", *Proceedings of the First Workshop on Robot Motion and Control, RoMoCo '99*, June 1999, 101-106.

[23] C. Samson, "Trajectory tracking for non-holonomic vehicles: overview and case study", *Proceedings of the Fourth International Workshop on Robot Motion and Control, RoMoCo'04*, June 2004, 139-153.