Formation-Based Cooperative Transportation by a Group of Non-holonomic Mobile Robots

Alpaslan Yufka and Osman Parlaktuna
Department of Electrical and Electronics Engineering
Eskisehir Osmangazi University
Eskisehir, Turkey
ayufka@gmail.com, oparlaktuna@ogu.edu.tr

Metin Ozkan
Department of Computer Engineering
Eskisehir Osmangazi University
Eskisehir, Turkey
meozkan@ogu.edu.tr

***Abstract*—In this study, motion planning and control scheme for a cooperative transportation system, which consists of a single object and multiple autonomous non-holonomic mobile robots, is proposed. Virtual leader-follower formation control strategy is used for the cooperative transportation system. The object is assumed as the virtual leader of the system and the robots carrying the object are considered as follower robots. A smooth path is generated by considering the constraints of the virtual robot. The origin of the coordinate system attached to the center of gravity of the object tracks the generated path. A path for each follower robot is generated to keep the formation structure. The follower robots track their paths. A communication framework is used for the messaging between robots, and asymptotically stable tracking control is used for trajectory tracking. The proposed method is verified with real applications and simulations using Pioneer P3-DX mobile robots and a single object.**

***Keywords*—cooperative transportation, motion planning, tracking control, formation, multiple robots**

I. INTRODUCTION

In daily life, when a person tries to carry a large and heavy object, he/she may need a help from others to cooperatively transport the object. This idea may be adapted to robots for cooperative transportation tasks. The cooperative transportation has become a long-standing issue in cooperative robotics whose aim is defined as multiple robots perform a given task together. It is believed that cooperative robots have potential advantages over a single robot, and there is an increase in the total utility of the system [1].

Many cooperative transportation methods for multiple robots have been introduced in order to evaluate cooperative transportation tasks, such as, leader-follower type control, constrain and move strategy, chain system, cooperative towing, and formation control. In [1], a leader-follower type decentralized control system is proposed to transport a single object with non-holonomic mobile robots. A particular end-effector linked to the robot through a steering joint is used to hold and grasp the object. In [2], based on constrain and move strategy, a decentralized method and its analysis are introduced for transporting objects. In [3], a system, which includes two car-like mobile robots coupled together via a carrier, was modeled in two-chain and single generator chained form for cooperative transportation. In [4], cooperative towing of payloads by multiple mobile robots is presented. In [5], authors propose a method for multiple robots working together to transport an object whose mass is not negligible, and the transportation task is viewed as a kind of formation control problem.

The control of cooperative multiple robots may be performed by one of the two approaches: centralized and decentralized. Decentralized control is more robust and scalable compared to centralized control [6]. A leader-follower control is applicable for decentralized control. This control approach includes one leader robot that tracks a desired trajectory, and follower robots predict the trajectory of the leader and track it [7].

In this study, motion planning and control scheme for a cooperative transportation system is proposed. The system includes a single object and multiple autonomous non-holonomic mobile robots. A virtual leader-follower formation control strategy is used for the cooperative transportation system. The object to be carried is considered as the virtual leader robot. A reference coordinate system of the virtual robot is attached to the center of the gravity of the object. A path is generated for the virtual robot and the origin of the reference coordinate system tracks this trajectory. The follower robots track their paths which are generated to keep the formation. The shape of the formation depends on the shape of the object and grasping points of the object by the robots. Additionally, a forklift-type constraint is used for the relative orientation between the virtual leader and each follower robot. This constraint is used in the generation of the trajectory. For trajectory tracking, an asymptotically stable tracking control proposed by Kanayama [8] is used. If the robots are able to track the trajectories with small tracking errors, the proposed method would not need dynamic model of the multi-body system and there is no need to determine the force distribution which should be applied by each follower.

The paper is organized as follows: Section 2 presents the proposed cooperative transportation system including motion planning and the asymptotically stable tracking control approach. In Section 3, conducted transportation experiments in both real and simulation environments are given. The conclusion is given in Section 4.

II. COOPERATIVE TRANSPORTATION

The cooperative transportation system organizes the mobile robots to transfer the object from an initial location to a desired one based on the virtual leader-follower formation control strategy. The object is considered the virtual leader (VL), and the robots carrying the object are considered followers (F_i, i=1,2,…,n) as illustrated in Fig. 1. The center of gravity of the object follows the path generated by considering the constraints of the robots.

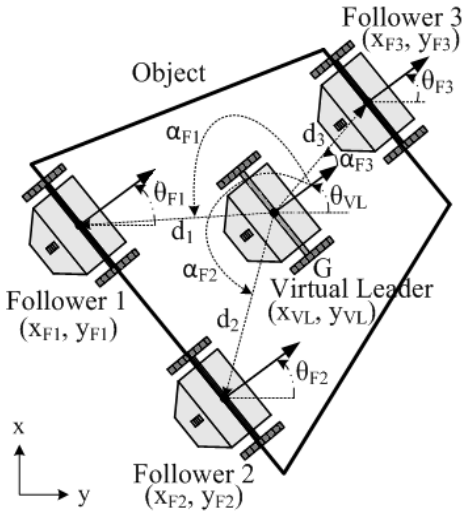

Figure 1. Virtual leader, followers and the object.

The center of the gravity of the object, G, is located at the midpoint, (x_{VL}, y_{VL}), of the VL's differential wheels. The VL has the same orientation, θ_{VL}, as the object. Initially, a motion plan is generated for the VL, and for synchronization purposes, the followers communicate with each other via Open Agent Architecture (OAA) [9]. Once the robots are synchronized, they behave in a distributed manner and keep a formation with respect to the VL with the distance, d_i, the distance between i^{th} follower and the VL with an angle of α_{Fi} (i=1,2,…). The center of gravity of each follower is at (x_{Fi}, y_{Fi}), and has an orientation θ_{Fi}, with respect to the global coordinates, x-y.

The location of the object is considered as a reference input for followers. Thus, due to the rigid body constraints of the single object, a natural formation is formed as the followers track their trajectories.

A. Motion planning

It is well known that the differential-drive mobile robot is a non-holonomic system because there are differential constraints that cannot be completely integrated [10]. To generate a smooth path for the wheeled mobile robot (WMR), first of all the kinematic model of the differential-drive mobile robot needs to be obtained.

1) Kinematic model of the WMR: In this study, Pioneer P3-DX mobile robot platform is used (Fig. 2a). It has two main wheels attached to its motors, and a caster wheel placed in the rear.

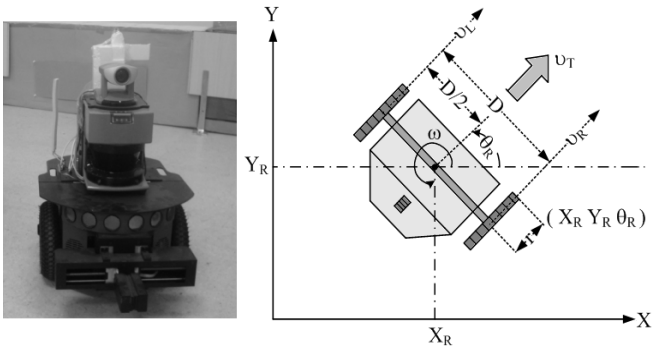

Figure 2. **a)** Pioneer P3-DX mobile robot **b)** the WMR and its parameters.

The kinematic model equations of the WMR are given in Equation (1). Here, D is the distance between left and right wheels and r is the radius of the wheels. υ_x and υ_y are translational velocities which are combined in υ_T as shown in Fig. 2(b). υ_R and υ_L specify the right and left wheel velocities, and ω is the angular velocity. υ_T and υ_Ψ are interpreted as action variables corresponding to "translate" and "rotate". The vector of $[x_R, y_R, \theta_R]^T$ is the posture of the robot's body.

$$\left.\begin{aligned} \upsilon_x &= r\frac{(\upsilon_R+\upsilon_L)}{2}\cos\theta_R \\ \upsilon_y &= r\frac{(\upsilon_R+\upsilon_L)}{2}\sin\theta_R \\ \omega &= \frac{r}{D}(\upsilon_R-\upsilon_L) \\ \upsilon_T &= \frac{(\upsilon_R+\upsilon_L)}{2} \\ \upsilon_\psi &= (\upsilon_R-\upsilon_L) \end{aligned}\right\} \quad (1)$$

Using Equation (1), the kinematic model of the differential drive WMR can be expressed as:

$$\begin{bmatrix} \upsilon_x \\ \upsilon_y \\ \omega \end{bmatrix} = \begin{bmatrix} r\cos\theta_R & 0 \\ r\sin\theta_R & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \upsilon_T \\ \omega \end{bmatrix} \quad (2)$$

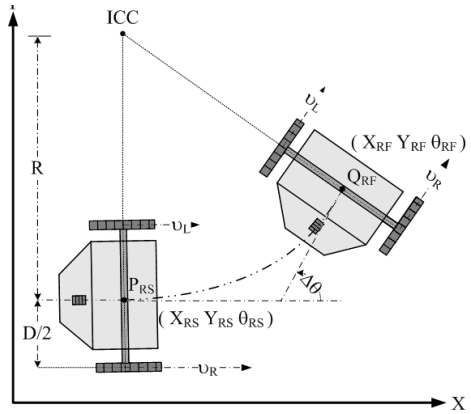

Figure 3. A curvature trajectory.

Based on the kinematic model, it is obvious that the location and the orientation of the WMR body is changed according to differential drive by means of the applied voltages to the right and left wheels. If υ_R is different from υ_L, the robot turns in the sense of clockwise or counter-clockwise direction depending on the relative velocities of two wheels. In this way, a curvature is generated whose center is at ICC (Instantaneous Center for Curvature) as shown in Fig. 3. R is the radius of the curved path which is tracked by the mobile robot. The relationship between the radius of the curvature and two wheel velocities is expressed as follows:

$$\left.\begin{aligned} R &= r\frac{\upsilon_T}{\omega} \\ R &= \frac{D}{2}\left(\frac{\upsilon_R+\upsilon_L}{\upsilon_R-\upsilon_L}\right) \end{aligned}\right\} \quad (3)$$

2) Smooth path generation: The path for the transportation of an object consists of critical nodes as in Fig. 4. These nodes can be connected using straight lines. However, due to constraints, the VL and followers cannot follow a path with sharp turns. Therefore, the path should be smoothed by considering non-holonomic constraints.

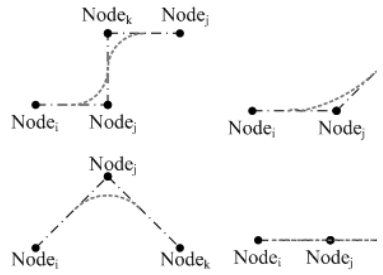

Figure 4. Possible straight lines in the cooperative transportation.

The trajectory defined by the straight lines can be transformed into a curved trajectory (Fig. 5). This trajectory is divided into three sections; before, during and after the rotation. $P_S = [X_S\ Y_S\ \theta_S]^t$ is the initial point where the WMR may slow down and wait (if necessary) for the beginning of the rotation. P_{RS} is the point where the WMR starts to rotate around ICC, Q_{RF} is the point where the robot finishes its rotation, and $Q_F = [X_F\ Y_F\ \theta_F]^t$ is the final point where the WMR completes its tracking on the curved trajectory.

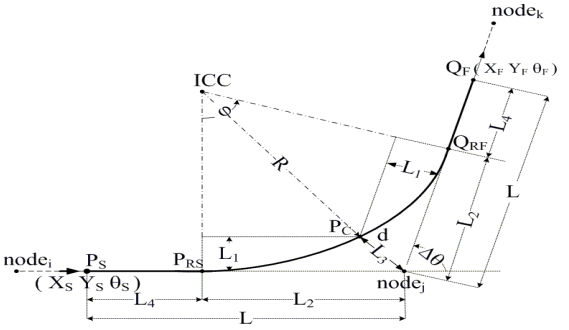

Figure 5. The symmetric curved-path.

When the WMR is rotating from P_{RS} to Q_{RF}, the difference between the initial and the final orientation of the robot is $\Delta\theta$ expressed as follows;

$$\Delta\theta = \theta_F - \theta_S \quad or \quad \Delta\theta = \theta_{RF} - \theta_{RS} \quad (4)$$

Since there are constraints imposed by walls and obstacles, the bound of path-deviation L_1 from the corner at node$_j$ (L_2) is defined in Equation (5) as follows:

$$\left.\begin{aligned} L_1 &= L_3\cos(\Delta\theta/2) \\ L_2 &= R\tan(\Delta\theta/2) \\ L_3 &= R\left[\frac{1}{\cos(\Delta\theta/2)}-1\right] \\ L &= L_2 + L_4 \end{aligned}\right\} \quad (5)$$

d is the total distance traveled by the WMR during the rotation with the rotation angle φ. During the rotation, the robot frequently updates its location and orientation, and controls the translational speed υ_T, rotational speed ω, the heading angle θ_R.

3) Path discretization: The WMR's position and heading are computed using output of incremental encoders and the predictive model shown in Fig. 6.

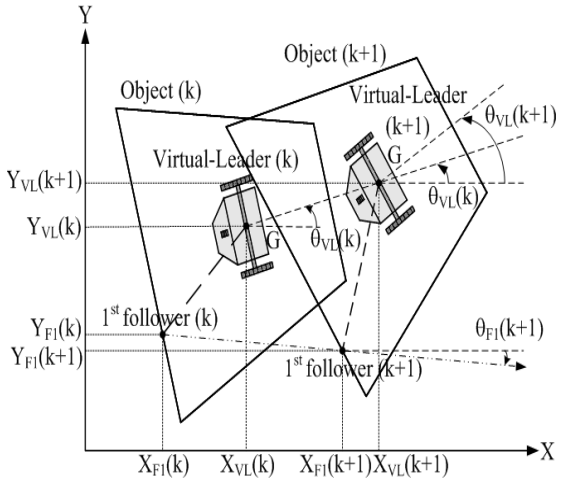

Figure 6. Predictive model.

At any instant, the center of the VL [x_{VL}, y_{VL}], changes due to its translational and rotational speed. The kinematic model of the VL assumes no slippage. Thus, its motion can be defined by the simple kinematic of rigid bodies.

The posture of the VL can be estimated from integration of equations given in Equation (1) and Equation (2). This integration is also implemented by the following numeric algorithm given in Equation (6).

$$
\begin{aligned}
&if \quad \omega \neq 0 \\
&x_{\mathrm{VL}}(k+1) = x_{VL}(k) + \frac{v_{VL}}{\omega_{VL}}[\sin(\theta_{VL}(k+1)) - \sin(\theta_{VL}(k))] \\
&y_{\mathrm{VL}}(k+1) = y_{VL}(k) - \frac{v_{VL}}{\omega_{VL}}[\cos(\theta_{VL}(k+1)) - \cos(\theta_{VL}(k))] \\
&\theta_{VL}(k+1) = \theta_{VL}(k) + \omega_{VL} t_s \\
&if \quad \omega = 0 \\
&x_{\mathrm{VL}}(k+1) = x_{VL}(k) + v_{VL} t_s \cos(\theta_{VL}(k)) \\
&y_{\mathrm{VL}}(k+1) = y_{VL}(k) + v_{VL} t_s \sin(\theta_{VL}(k)) \\
&\theta_{VL}(k+1) = \theta_{VL}(k)
\end{aligned}
\qquad (6)
$$

The reference posture vector, $[x_{VL}(k+1), y_{VL}(k+1), \theta_{VL}(k+1)]^t$, for VL is obtained from Equation (4). Furthermore, the reference velocity vector, $[\upsilon_{VL}(k+1), \omega_{VL}(k+1)]^t$, for VL is determined by the desired translational and angular velocities specified by the user.

This algorithm is also expanded for followers according to the VL as given in Equation (7) where d_{Fi} is the Euclidean distance between i^{th} follower and the VL with an angle of αF_i (i=1,2,3,..,n).

$$
\begin{aligned}
&x_{Fi}(k+1) = x_{\mathrm{VL}}(k+1) + d_{Fi}\cos(\alpha_{Fi} + \theta_{VL}(k)) \\
&y_{Fi}(k+1) = y_{\mathrm{VL}}(k+1) + d_{Fi}\sin(\alpha_{Fi} + \theta_{VL}(k)) \\
&if \quad \omega \neq 0 \\
&\theta_{Fi}(k+1) = \tan^{-1}\left(\frac{y_{\mathrm{Fi}}(k+1) - y_{\mathrm{Fi}}(k)}{x_{\mathrm{Fi}}(k+1) - x_{\mathrm{Fi}}(k)}\right) \\
&if \quad \omega = 0 \\
&\theta_{Fi}(k+1) = \theta_{VL}(k+1)
\end{aligned}
\qquad (7)
$$

The posture given in Equation (7) is the reference input of location and orientation expressed as $[x_{Fi}(k+1), y_{Fi}(k+1), \theta_{Fi}(k+1)]^t$ for followers. The reference linear and angular velocities for robot i, $[\upsilon_{Fi}(k+1), \omega_{Fi}(k+1)]^t$, are different than the velocities of the VL. The velocities of followers are given as follows:

$$
\begin{aligned}
&\upsilon_{x_{Fi}}(k+1) = [x_{Fi}(k+1) - x_{Fi}(k)] / t_s \\
&\upsilon_{y_{Fi}}(k+1) = [y_{Fi}(k+1) - y_{Fi}(k)] / t_s \\
&\upsilon_{Fi}(k+1) = \sqrt{{\upsilon_{x_{Fi}}}^2(k+1) + {\upsilon_{y_{Fi}}}^2(k+1)} \\
&-\omega_{VL-\max} \le \omega_{Fi} \le \omega_{VL-\max}
\end{aligned}
\qquad (8)
$$

In the Section II-B, how the robots use these references in their tracking control will be discussed.

4) Motion Constraints for WMR: To generate a feasible path for VL, the kinematic constraint of the VL, $A(q)\dot{q} = 0$, is considered. This Pfaffian constraint matrix for WMR is expressed as follows;

$$
\begin{bmatrix} -\sin\theta & \cos\theta & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = 0 \qquad (9)
$$

Moreover, since the VL tracks the smooth trajectory considering the kinematic constraint, a relative orientation angle deviation occurs between i^{th} follower and the VL. The maximum absolute allowable deviation is specified by the user as $\theta_{desired}$. This is similar to the constraint for forklift-type robots and expressed as:

$$
\left|\theta_{VL}(k+1) - \theta_{Fi}(k+1)\right| \le \theta_{desired} \qquad (10)
$$

Using Equation (6), (7), and (10), a feasible path for the VL is generated and followers follow their references within range of $[-\theta_{desired}, \theta_{desired}]$.

B. Control of a non-holonomic WMR

This section discusses asymptotically stable tracking control proposed by Kanayama [8]. The controller is designed using Lyapunov stability theory.

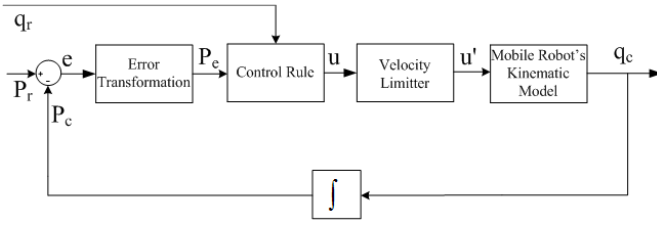

Figure 7. Block diagram of the tracking controller.

Fig. 7 shows the block diagram of a tracking control system for the non-holonomic mobile robot where the reference posture $p_r=[x_r,y_r,\theta_r]^t$, and reference velocities, $q_r=[\upsilon_r,\omega_r]^t$, are inputs from the discrete path information. This control structure is composed of an error block, a control rule, and a velocity limiter.

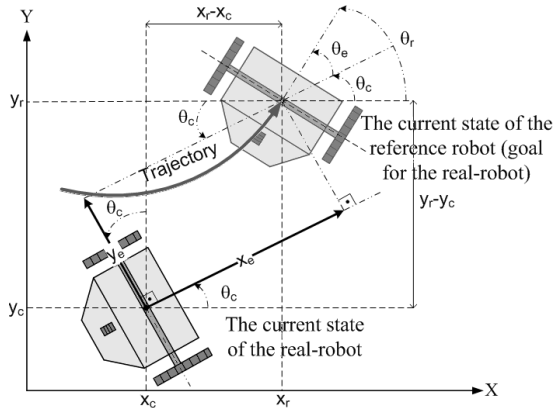

Figure 8. Error components.

First, the error posture, $p_e=[x_e,y_e,\theta_e]^t$, is a transformation of the difference between the reference posture, $[x_r, y_r, \theta_r]^t$, and the current posture, $[x_c,y_c,\theta_c]^t$. Equation (11) shows the error posture seen in Fig. 8.

$$p_e = \begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} \cos(\theta_c) & \sin(\theta_c) & 0 \\ -\sin(\theta_c) & \cos(\theta_c) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r - x_c \\ y_r - y_c \\ \theta_r - \theta_c \end{bmatrix} \quad (11)$$

The tracking control input is calculated by feed-forward and feedback actions as:

$$u = \begin{bmatrix} v(e,q_r) \\ \omega(e,q_r) \end{bmatrix} = \begin{bmatrix} v_r \cos(\theta_e) + K_x x_e \\ \omega_r + v_r (K_y y_e + K_\theta \cos(\theta_e)) \end{bmatrix} \quad (12)$$

The control rule part is developed using a Lyapunov function defined in Equation (13), and is given in Equation (12) [8]. Moreover, the gains K_x, K_y, and K_θ of the state feedback can be found by trial and error, and velocity limiter is used to limit the translational and rotational velocities which are harmful for hardware of the WMR. To prove the stability of the control system, the following Lyapunov function is used [8].

$$V = \frac{1}{2}({x_e}^2 + {y_e}^2) + \frac{(1-\cos(\theta_e))}{K_y} \quad (13)$$

III. EXPERIMENTS

The proposed cooperative transportation system for multiple autonomous non-holonomic mobile robots is coded in C++, and tested using P3-DX robots both in MobileSim simulation environment and in real-world applications. In real-world application, ARNL [11], that implements the Monte Carlo Localization Algorithm to accurately place the robot within a given map by using the information derived from laser-ranging and robot odometry, is used.

In the experiments, an obstacle-free environment whose sizes are 9000mm x 9000mm in simulations and 7260mm x 6655mm in real-world applications is used. A trajectory illustrated in Fig. 9 is used and generated for the VL using nodes (600mm, 600mm), (3700mm, 600mm), (3700mm, 5600mm), (6400mm, 5600mm) in real-world applications, and nodes (1000mm, 1000mm), (4000mm, 1000mm), (4000mm, 7000mm), (8000mm, 7000mm) in simulations.

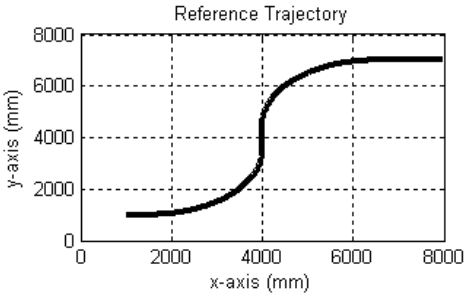

Figure 9. The path followed by the transportation system.

Robots are synchronized initially via communication framework. A smooth trajectory for the VL is generated using the nodes given above. The followers track the reference inputs discussed in previous sections in order to keep formation with the VL. The VL has a 50 mm/s translational speed and a 5 degrees/s rotational speed both in experiments and applications. Additionally, the value of $\theta_{desired}$ is chosen as 20 degrees.

A. Simulation With Two Robots

First, a simulation experiment where a rectangular object is carried by two robots is conducted. Results of this experiment are given in Fig. 10-12.

In Fig. 10, the path followed by the transportation system is shown. The point located at the center of gravity of the object tracks the reference path. The relative orientation of the followers with respect to the object is given in Fig. 11. As seen in the figure, the absolute relative orientation angle is less than 17 degrees. It is obvious that this angle is in the range of $[-\theta_{desired}, \theta_{desired}]$.The tracking errors of this point in x and y directions and the 2-norm error are given in Fig. 12. The errors are smaller than 25 mm.

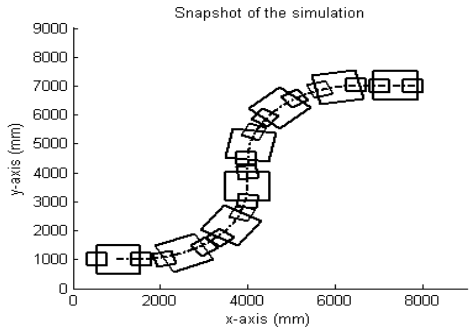

Figure 10. The path followed by the transportation system.

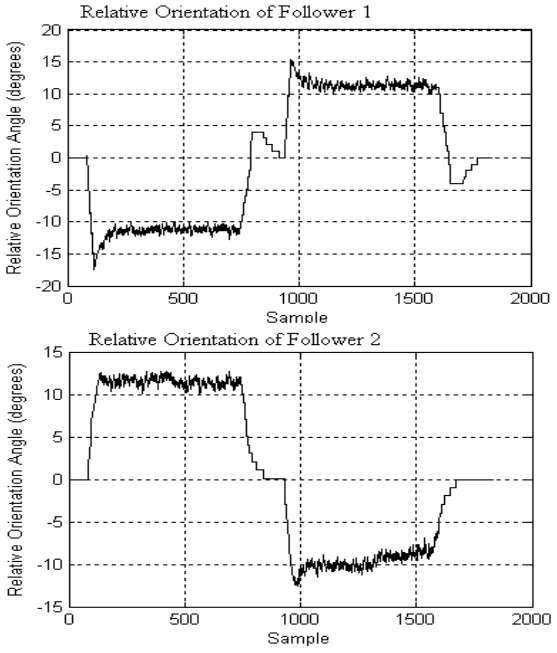

Figure 11. Relative orientation of followers with respect to the object.

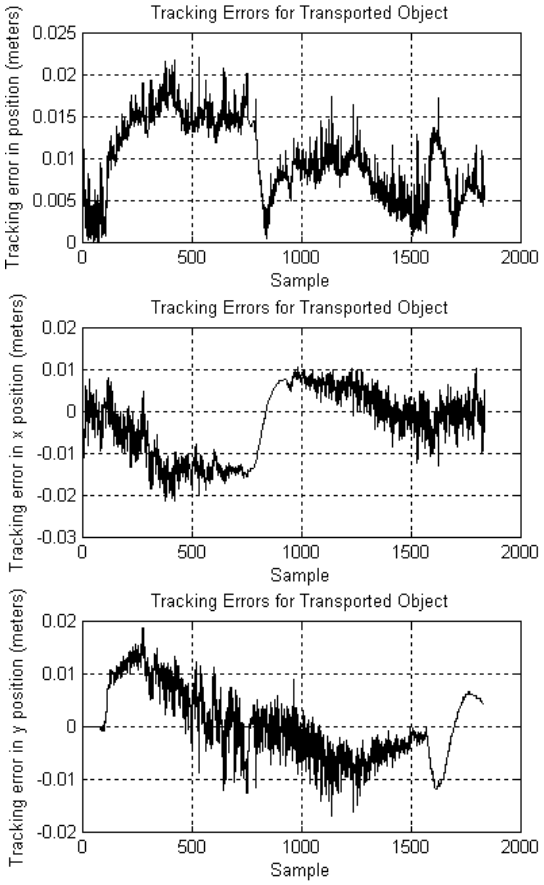

Figure 12. Tracking errors (meters).

B. Real-World Application With Two Robots

The previous simulation experiment is carried out with real robots. The rectangular object is attached to the robots by using the following attachment mechanism.

In the attachment, a T-shaped mechanism with a pin connection at the top is used to connect the object to the robots. Fig. 13 illustrates a schematic view of the attachment which consists of a follower and T-shaped bed.

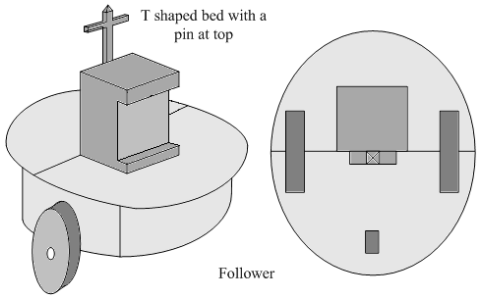

Figure 13. Schematic view of the handling mechanism.

In Fig. 14, there are snapshots of the real world application of the transportation system. The relative orientation of the followers with respect to the object is given in Fig. 15. As seen in Fig. 15, the orientation angle is less than 20 degrees. The tracking errors of the transported object in x and y directions and the deviation of the center of the object from the actual center are given in Fig.16. The errors are smaller than 35mm. A video of this application is recorded and can be reached in [12].

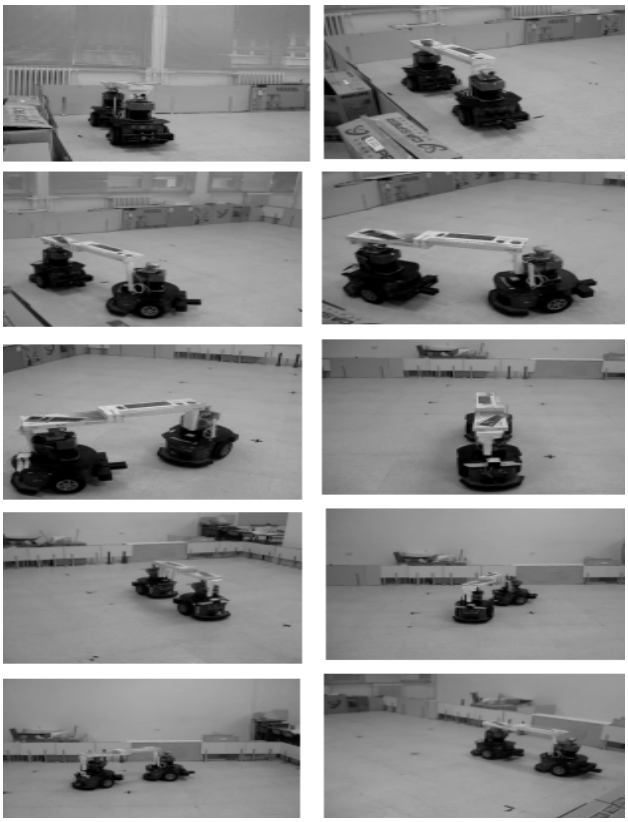

Figure 14. Snapshot of real-world application for two robots and an object (left to right).

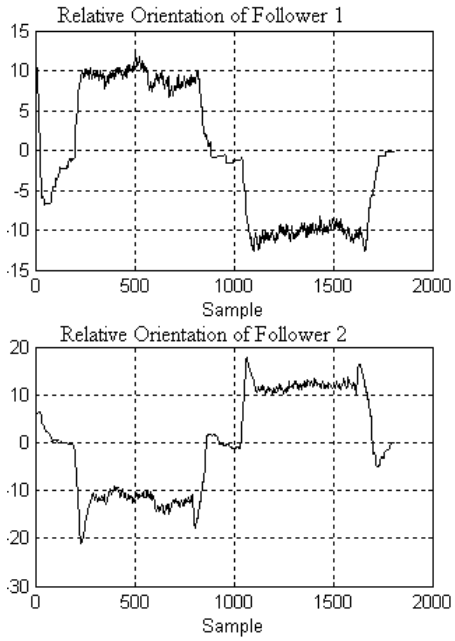

Figure 15. Relative orientation angle of followers with respect to the object.

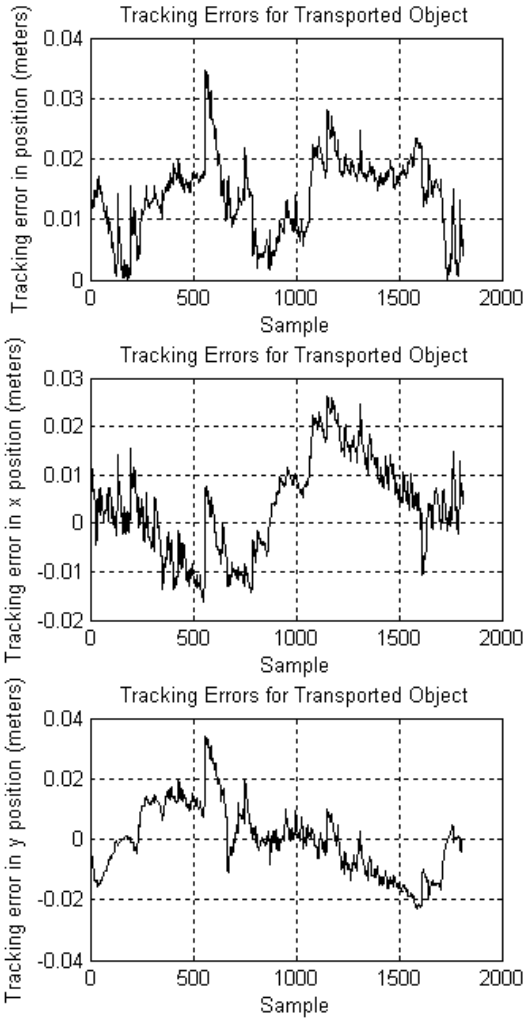

Figure 16. Tracking errors (meters).

C. *Simulation With Three Robots*

To validate the proposed cooperative transportation with more robots, an experiment where a greater rectangular object is carried by three robots is conducted. The simulation results are shown in Fig. 17-19.

In Fig. 17 the path followed by the transportation system is shown. The point located at the center of gravity of the object tracks the reference path. The tracking errors of this point in x and y directions and the 2-norm error are given in Fig. 18. The errors are smaller than 20 mm. The relative orientation of the followers with respect to the object is given in Fig. 19. As seen in the figure, the absolute relative orientation angle is less than 18 degrees.

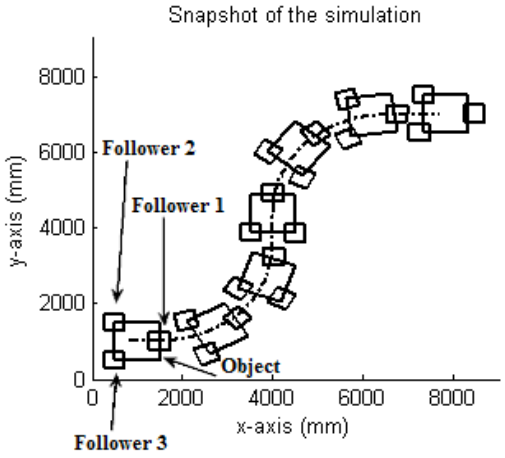

Figure 17. Snapshot of simulation for three robots and a single object.

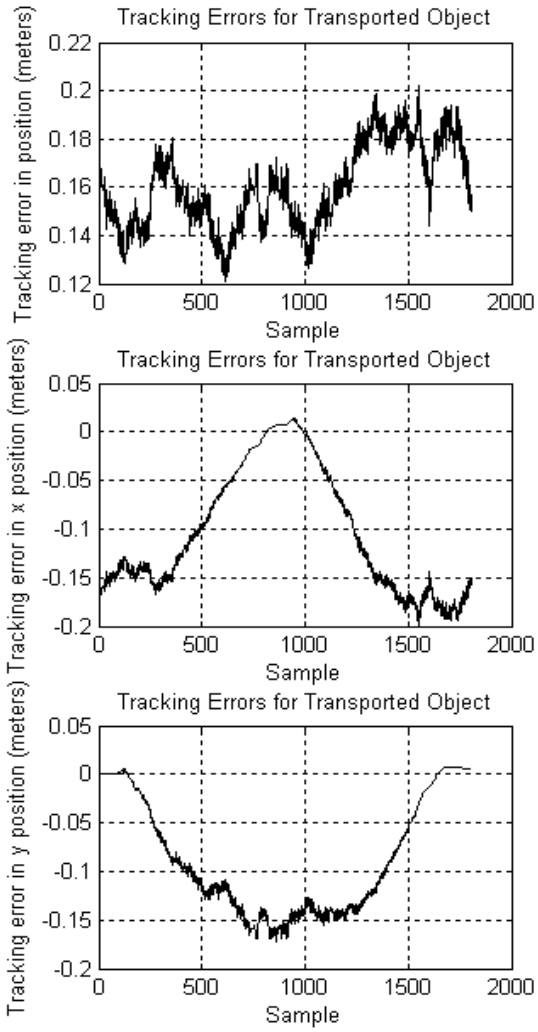

Figure 18. Tracking errors (meters).

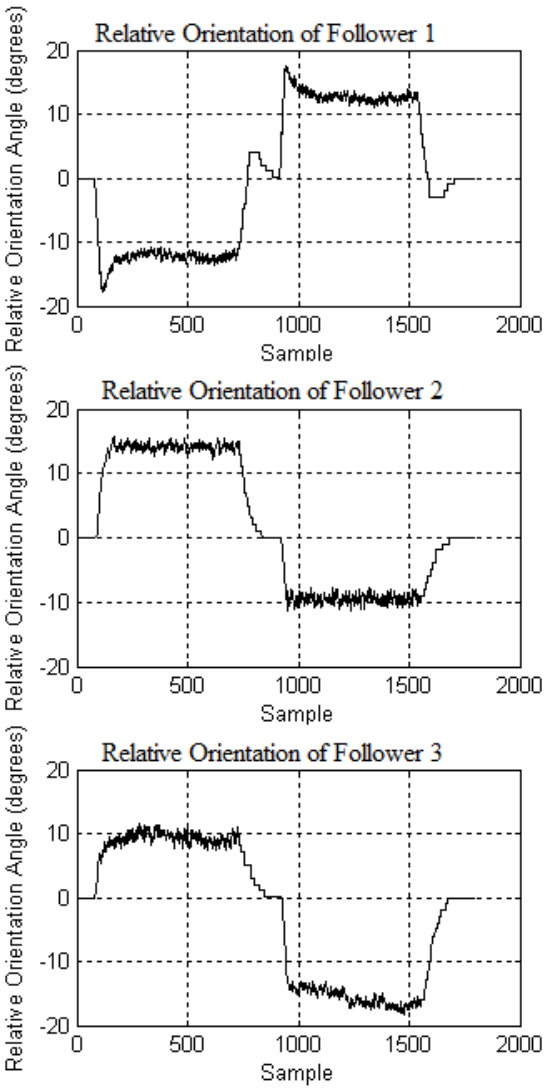

Figure 19. Relative orientation of followers with respect to object (degrees).

IV. CONCLUSION AND PROPOSALS

This study proposes a formation-based cooperative transportation approach by considering the object as a virtual leader robot. A reference trajectory is generated for the virtual leader and each follower robot generates its trajectory to satisfy the formation constraints. A controller designed by using Lyapunov stability theory is used by each follower robot. The results of simulations and experiments show that the proposed method is effective to carry an object by multiple robots.

Future study will include applications of cooperative transportation both in real-world and simulations for different transportation tasks and number of robots.

REFERENCES

[1] X. Yang, K. Watanabe, K. Izumi and K. Kiguchi, "A decentralized control system for cooperative transportation by multiple non-holonomic mobile robots", International Journal of Control, Vol. 77, No. 10, 2004, pp. 949–963.

[2] A. Zaerpoora, M.N. Ahmadabadia, M.R. Barunia and Z.D. Wang, "Distributed object transportation on a desired path based on Constrain and Move strategy", Robotics and Autonomous Systems, Vol.50, Issues 2-3, 2005, pp.115-128.

[3] H. Yamaguchi and T. Arai, "A Conversion of a Cooperative Transportation System with Two Car-Like Mobile Robots into Two-Chain, Single-Generator Chained Form and its Steering", Journal of Robotics and Mechatronics, Vol.21, No.1, 2009, pp. 57-65.

[4] P. Cheng, J. Fink, V. Kumar and J.S. Pang, "Cooperative Towing with Multiple Robots", Journal of mechanisms and robotics, vol. 1, no 1, 2009, note(s). 011008.1-011008.8.

[5] D. Zhaohui, W. Min and C. Xin, "Multi-robot cooperative transportation using formation control", 27th Chinese Control Conference (CCC'08), 2008, pp. 346-350.

[6] M. Udomkun and P. Tangamchit, "Cooperative Behavior-based Control of Decentralized Mobile Robots on an Overhead Box Carrying Task", Proceedings of ECTI-CON'08, 2008, pp. 633 – 636.

[7] M. Fujii, W. Inamura, H. Murakami, K. Tanaka, K. Kosuge, "Cooperative Control of Multiple Mobile Robots Transporting a Single Object with Loose Handling", IEEE International Conference on Robotics and Biometrics, Vols. 1-5, 2007, pp. 816-822.

[8] Y. Kanayama, Y. Kimura, F. Miyazaki and T. Noguchi, "A Stable Tracking Control Method for an Autonomous Mobile Robot", IEEE International Conference on Robotics and Automation, 1990, pp.384-389.

[9] D. Martin, A.J. Cheyer and D.B. Moran, "The Open Agent Architecture: A Framework for Building Distributed Software Systems", Applied Artificial Intelligence, vol. 13, 1999, pp. 91-128.

[10] S.M. LaValle, Planning Algorithms, Cambridge University Press, 2006.

[11] ARNL, Installation and Operations manual (2007). Version 2.1, Available from: http://robots.mobilerobots.com/, Accessed: 2009-05-15.

[12] AIRLAB, Artificial Intelligence & Robotics Laboratory, Eskisehir Osmangazi University, http://www.ai-robotlab.ogu.edu.tr.