Proceeding of the IEEE
International Conference on Robotics and Biomimetics
Dali, China, December 2019

Collaborative Object Transportation by Multiple Robots with Onboard Object Localization Algorithm

Zhixian Hu[2], Zhixiang Zhao[2], Lianxin Zhang[2], Hengli Liu[2,3], Ning Ding[1,2], Zhenglong Sun[1,2]
Tin Lun Lam[1,2], Huihuan Qian[1,2,†]

***Abstract*— Collaborative object transportation has become a popular study trend with its remarkable application foreground. In previous relevant studies, localization of the transported object has always been accomplished by additional devices rather than robot onboard equipments. This paper presents a generalized multi-robot leader-follower system for collaborative object transportation and an onboard object localization algorithm for trajectory tracking of the target object. In this system, the mobile robots can directly push a cubic object without extra gripping devices, when tracking the reference trajectory. During the control process, the object is regarded as an virtual leader, whose localization information is utilized as the feedback, while the mobile robots are considered as the followers. In absence of external localization systems, the proposed onboard localization algorithm provides the real-time position information of the object using scan data from lidars equipped on the robots. A performed measurement accuracy test shows high precision of this algorithm. Finally, a lane-changing experiment of object transportation is conducted, and it verifies this multi-robot leader-follower system.**

***Index Terms*— collaborative transportation, leader-follower, trajectory-tracking, object localization algorithm.**

I. Introduction

When ants go out foraging and find a large and heavy piece of food, they would cooperate and transport the food together, as shown in Fig. 1. This idea inspired researchers to put forward the concept of collaborative transportation. Collaborative transportation, as an archetype of multi-agent cooperation problems, has drawn considerable attention in recent years due to its tremendous application foreground in automatic parking, warehouse management and other various circumstances [1]. Multi-agent collaborative transportation has potential advantages over single-agent object transportation, because multi-agent has more flexibility, better fault-tolerance, greater robustness, and higher feasibility [2].

*This work is supported by Project U1613226 and U1813217 supported by NSFC, China, Project 2019-INT009 from the Shenzhen Institute of Artificial Intelligence and Robotics for Society, the Shenzhen Science and Technology Innovation Commission, fundamental research grant KQJSCX20180330165912672, R&D Project of Ministry of housing and urban construction (Grant No.2018-K8-034).

[1]Shenzhen Institute of Artificial Intelligence and Robotics for Society
[2]The Chinese University of Hong Kong, Shenzhen
[3]University of Science and Technology of China
Zhixian Hu and Zhixiang Zhao contributed equally to this work.
[†]Corresponding author is Huihuan Qian hhqian@cuhk.edu.cn

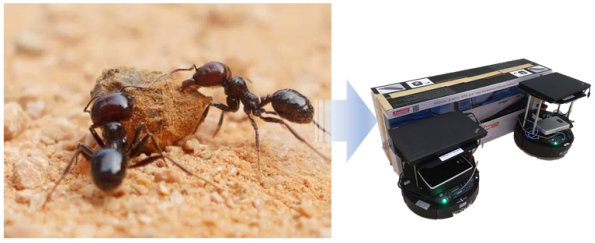

Fig. 1. Ants manipulation vs. Robots manipulation. Multi-robot cooperation control theory originates from group behaviour in biology (e.g., insect foraging, handling behaviour, etc.).

Control strategy for multi-robot transportation has developed for decades. Verginis et al. [3] and Nikou et al. [4] presented a decentralized nonlinear model predictive control scheme to transport an object rigidly grasped by robots with collision avoidance in a bounded workspace. Besides, the fuzzy logic controller is applied in [5] to virtually link two robots. Also, Chen et al. [6] proposed a strategy using a number of robots to transport a object which is large and over the payload of one robot. Furthermore, Wang et al. [7] designed a distributed force and torque controller to collectively transport the object without explicit communication. Among these strategies, decentralized leader-follower multi-robot system is extensively used, due to its scalability and robustness. Usually, in a leader-follower system, only the leader knows the trajectory and destination of the object. The followers track the leader with a certain formation and align themselves with the object [7–10]. To make the mobile manipulator more suitable for formation control, a modified end-effector is proposed in [11]. Yufka and Ozkan [2] take the transported object as the virtual leader and the robots as followers. Three types of leaders, including an autonomous robot, a robot teleoperated by a human and a human, are tested in [12] to demonstrate the ability of followers during the transport process.

All the above strategies can be mainly divided into three categories according to how robots handle the object, namely grasping, caging, and pushing-only [13]. In grasping strategy, the object is held using grippers or mechanical joints and can be pulled, pushed, or lifted [2–5] [11] [14]. In caging strategy, a "closure", which traps the object, is maintained during the transport process to prevent the object escaping from the cage [8] [15–16]. Compared to grasping and caging, pushing-only

978-1-7281-6321-5/19//$31.00 ©2019 IEEE

strategy is a relatively general method because the robots do not need to grasp the object or to know the object shape and size [10] [17]. However, how to avoid the object-losing situation, which is caused by the unbalanced pushing force and friction between robots and the object, is an important research direction. To solve this problem, the real-time state of the object, including position and heading angle, is needed for the robots to dynamically adjust their relative positions to the object. In [18], a watcher robot is added to provide the object posture to the pusher robots. Some devices, such as cameras [19] and reflection markers [17], can also be used to obtain the actual state of the object.

In this paper, a generalized multi-robot leader-follower system is proposed to do collaborative object transportation tasks, and an onboard localization algorithm is designed to track the object trajectory. Mobile robots of this system can transport the object by directly pushing, without an assist from special gripping equipments. In the transportation process, the object is considered as an virtual leader, which is the control target of this system, and two mobile robots are regarded as followers. Meanwhile, the presented localization algorithm obtains location information of the object with scan data from onboard robot lidars. To show the precision of this localization algorithm, a measurement accuracy test is carried out using 3D Motion Capture as the benchmark. Finally, a lane-changing experiment is conducted to demonstrate the validity of the proposed multi-robot leader-follower system on collaborative object transportation, where a box is pushed by a pair of mobile robots and follows a reference trajectory.

The structure of this paper is organized as follows. The multi-robot leader-follower control system is stated in section 2, and section 3 presents the object localization algorithm. Experiment validation is put in section 4. Section 5 is conclusion.

II. Control System Statement

This section presents the kinematic model of the object transportation problem with one object and two robots, then briefly describes how to apply a trajectory tracking controller to reduce tracking errors.

A. Kinematics

In this object transportation problem, a leader-follower system with one object as an virtual leader and two robots as followers is taken into consideration. The global states, including position and heading angle, $S_i = \begin{bmatrix} x_i & y_i & \theta_i \end{bmatrix}^T$ of two robots and the object are represented by the robots' central points and the middle point of the object's pushed side, respectively, as shown in Fig. 2. The subscript $i = 0, 1, 2$, represents the object, the left and right robots, respectively. For the sake of convenience, we only consider the case that the object is pushed upon, which means $\theta_0 \in (0, \pi)$. As for the case $\theta_0 \in (\pi, 2\pi)$, it can be treated in the same way because of symmetry. Given the state of the object, the reference states of two robots are given by

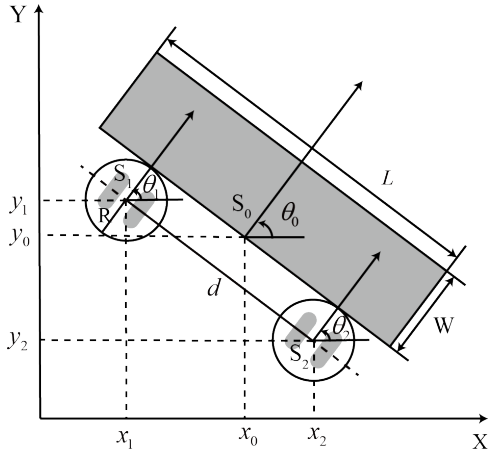

Fig. 2. The geometric relationship of object and two follower robots in the global coordinate system. L and W are the length and width of the object, respectively.

$$\theta_{r,1} = \theta_{r,2} = \theta_0, \tag{1}$$

$$\begin{bmatrix} x_{r,1} \\ y_{r,1} \end{bmatrix} = \begin{bmatrix} sin\theta_0 & cos\theta_0 \\ -cos\theta_0 & sin\theta_0 \end{bmatrix} \begin{bmatrix} -\frac{d}{2} \\ -R \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}, \tag{2}$$

$$\begin{bmatrix} x_{r,2} \\ y_{r,2} \end{bmatrix} = \begin{bmatrix} sin\theta_0 & cos\theta_0 \\ -cos\theta_0 & sin\theta_0 \end{bmatrix} \begin{bmatrix} \frac{d}{2} \\ -R \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}, \tag{3}$$

where d is the reference distance between two robots, and R is the radius of both robots.

The kinematic model of the nonholonomic robots can be described by

$$\begin{cases} \dot{x}_i(t) = v_i(t) cos\theta_i(t), \\ \dot{y}_i(t) = v_i(t) cos\theta_i(t), \\ \dot{\theta}_i(t) = w_i(t), \end{cases} \tag{4}$$

where $v_i(t)$ and $w_i(t)$ are the linear velocity and angular velocity of the robots, respectively.

In the ideal case, given the linear and angular velocity of the box, the velocities of the two robots can be derived using the following equations.

$$\begin{cases} w_1(t) = w_2(t) = w_0(t), \\ v_1(t) = v_0(t) - \dfrac{d}{2} w_0(t), \\ v_2(t) = v_0(t) + \dfrac{d}{2} w_0(t). \end{cases} \tag{5}$$

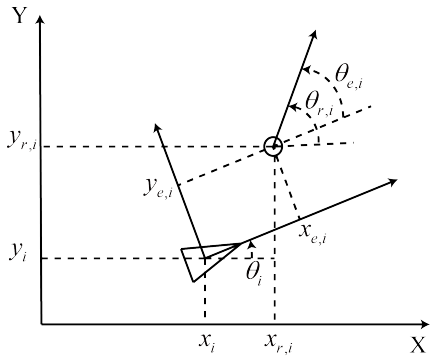

Fig. 3. Tracking configuration of the object and the two robots. The triangle represents the experimental position of the object and the circle represents the reference position [20].

B. Problem Formulation

The trajectory tracking errors of the object and the robots in their body-fixed frames are given by (see Fig. 3)

$$S_{e,i}(t) = \begin{bmatrix} x_{e,i}(t) \\ y_{e,i}(t) \\ \theta_{e,i}(t) \end{bmatrix} = \begin{bmatrix} cos\theta_i(t) & sin\theta_i(t) & 0 \\ -sin\theta_i(t) & cos\theta_i(t) & 0 \\ 0 & 0 & 1 \end{bmatrix} [S_{r,i}(t) - S_i(t)], \tag{6}$$

Thus, the trajectory tracking problem is equivalent to find a controller which ensures the error of the whole system is bounded and converged. The controller is

$$\lim_{t \to +\infty} S_{e,i}(t) = \begin{bmatrix} 0 \\ 0 \\ 2K_i\pi \end{bmatrix}, \tag{7}$$

where K_i is a constant integer.

C. Control Strategy

The whole control strategy is shown in Fig. 4. To decrease the trajectory tracking errors of the object and two robots, a controller [20], which takes velocity constraint and stability into account, is applied,

$$\begin{cases} v_i &= v_{r,i} + \dfrac{c_1 x_{e,i}}{1 + x_{e,i}^2 + y_{e,i}^2}, \\ w_i &= w_{r,i} + \dfrac{c_2 v_{r,i} \left[y_{e,i} cos\left(\dfrac{\theta_{e,i}}{2}\right) - x_{e,i} sin\left(\dfrac{\theta_{e,i}}{2}\right) \right]}{1 + x_{e,i}^2 + y_{e,i}^2} \\ & \quad + c_3 sin\left(\dfrac{\theta_{e,i}}{2}\right), \end{cases} \tag{8}$$

where c_1, c_2, c_3 are positive parameters that can be tuned subject to velocity constraints. The reference states and velocities of two robotss can be derived from the object's experimental state using (1–3) and (5). The convergence and stability of this controller have been proved in [20].

However, due to the frictions between robots and object, tracking error of robots along Y-axis, $y_{e,i}$ is hard to decrease and will be accumulated during the transportation process, such that the robots may not contact with the object finally. To solve this, when the robots can not touch the object, the transportation process is stopped and the two robots begin to adjust their states to the reference states. Not until the adjustment is finished will the transportation continue.

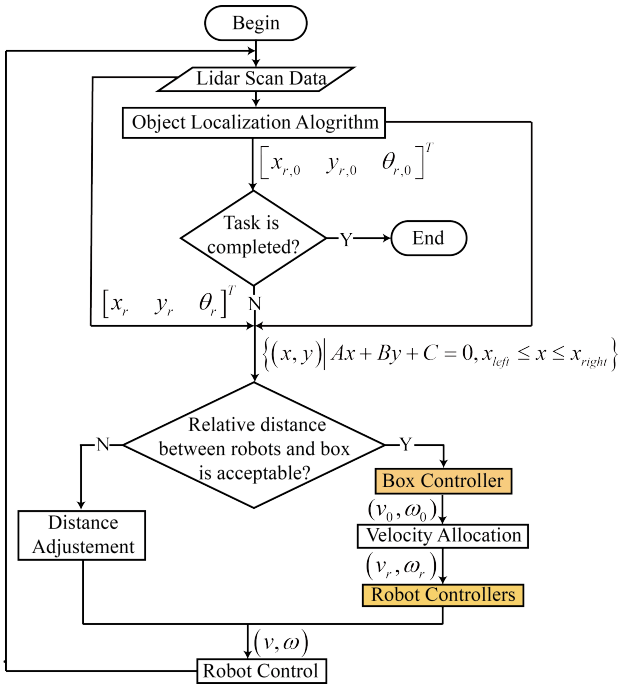

Fig. 4. Control strategy for the multi-robot system. Panel (a) is the flow chart representing the control system. Panel (b) is the controller for the object and robots, and (8) is used in the 'Velocity Controller' part.

III. Object Localization Algorithm

To obtain the experimental state of the object, an object localization algorithm is designed based on the lidar scan data which can be converted to a set of points representing the relative positions between obstacles and lidar. The point set is defined as μ. Because the points in the point set are in the body-fixed frame of robots, a conversion is needed to transform them into the global frame. For any point (x_{lidar}, y_{lidar}) in the body-fixed frame of robots, its global coordinate (x, y) is given by:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} sin\theta_i & cos\theta_i \\ -cos\theta_i & sin\theta_i \end{bmatrix} \begin{bmatrix} x_{lidar} \\ y_{lidar} \end{bmatrix} + \begin{bmatrix} x_i \\ y_i \end{bmatrix}. \quad (9)$$

The object can be represented by a line segment with the length L in the body-fixed frame of robots. The expression of the line segment is

$$\begin{cases} Ax + By + C = 0, \\ x_{left} \leq x \leq x_{right}, \end{cases} \quad (10)$$

where A, B, and C are corresponding parameters of the line segment, and x_{left} and x_{right} are X-axis positions of the extreme left point and the extreme right point, respectively.

The experimental state S_0 of the object can be got by

$$\begin{cases} x_0 = \dfrac{x_{left} + x_{right}}{2}, \\ y_0 = \dfrac{y_{left} + y_{right}}{2}, \\ \theta_0 = tan^{-1}(-\frac{A}{B}) + \dfrac{\pi}{2}, \end{cases} \quad (11)$$

where $tan^{-1}(-\frac{A}{B}) \in (-\frac{\pi}{2}, \frac{\pi}{2})$. Thus, the experimental state of the object can be expressed by the line segment. The pseudo code for this algorithm is shown below.

Algorithm 1: Object Localization Algorithm

Input: μ_j, d_0, d_1 α, A_{j-1}, B_{j-1}, C_{j-1}, $x_{left,j-1}$ and $x_{right,j-1}$
Output: S_0, A_j, B_j, C_j, $x_{left,j}$ and $x_{right,j}$
for *point* (x, y) *in* μ_j **do**
  Calculate the distance d_2 from (x, y) to origin;
  if $d_2 \geq d_0$ **then**
    delete (x, y);
if $A_{j-1}! = None$ **then**
  for *point* (x, y) *in* μ_j **do**
    Calculate the distance d_3 from (x, y) to (10);
    if $d_3 \geq d_1$ **then**
      delete (x, y);
for *point* (x, y) *in* μ_j **do**
  Compute the point density α_1 of the region which contains this point (x, y);
  if $\alpha_1 \leq \alpha$ **then**
    delete (x, y);
Apply the least square method using μ_j to get A_j, B_j, C_j, $x_{left,j}$ and $x_{right,j}$;
Get S_0 using (11);

Three filters are designed to filter out invalid points in μ from an onboard lidar. First, during the transportation process, because the object is next to the robot, the point whose distance to the robot is larger than a threshold d_0 can be filterd out. Second, as the robot moves at a relatively low velocity and the lidar scans at a high frequency, the change of the object's position between two continuous lidar scans is limited. Hence, the point whose distance to the line segment obtained in the $(j-1)_{th}$ scan is larger than the threshold d_1 can be filtered out. However, this filter is not used for the first scan because of no historical data. Besides, we also add a point density filter to filter out points in the regions of point densities which are lower than the threshold α. After filtering by the above three filters, we can assume that the remaining points are mainly from the side of the object. Then we can easily get the expression of object position using the least square method in the j_{th} scan.

During the locating process, there may be some noise that affects scan data of the lidar, and some jitters displayed on the track of the object in the pre-test. To improve the performance of this localization algorithm, an extended Kalman filter is applied, which has been extensively used to estimate unknown variables given continuous observation measurement data and is famous for its stability and intuition [21].

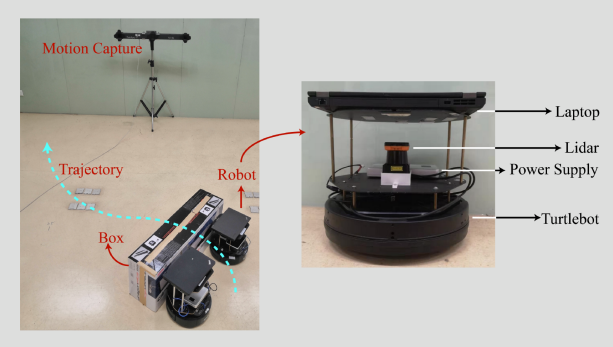

Fig. 5. Experiment Platform.

IV. Experiment Validation

A. Experiment Platform

To test and validate the proposed control system, this paper utilized an experiment platform shown in Fig. 5. Both Turtlebots behind are followers. Laser radar Hokuyo is fixed on each Turtlebot to provide robot position and to detect the object state. Phoenix Technologies 3D Motion Capture is an equipment that can record object movements, and it is used to test the validity and measurement accuracy of the onboard object localization algorithm. Here, Turtlebot has a maximum translational velocity of $0.70m/s$ and a maximum rotational velocity of $110°/s$; Hokuyo has a measuring distance of 0.1-$10m$ and the accuracy is $\pm 30mm$; the measurement accuracy of the Motion Capture is $\pm$1mm. Because the Motion Capture has a rather high measurement accuracy, it is served as the benchmark in measurement accuracy test on object localization algorithm. A box with length $1.05m$, width $0.50m$, and height $0.30m$, is employed as the transported object.

Other constituents equipped in Turtlebot include a control

laptop and a power supply battery for lidar. The laptop on Turtlebot can receive lidar data, process the data to get the box location using object localization algorithm, implement the control strategy to get control instructions, and output the control commands to drive Turtlebot. The laptop on the left Turtlebot is responsible for detecting the box location and send the location information to the laptop on the right Turtlebot through TCP/IP.

B. Measurement Accuracy Test

To test the measurement accuracy of the onboard object localization algorithm and to prove its validity, a measurement accuracy test is conducted which applies object localization algorithm during a box-pushing process and uses the measurements from the Motion Capture as the benchmark.

We set the threshold d_0, d_1 and α of the three filters in object localization algorithm to $1.5m$, $0.05m$ and 3 points per square decimetre, respectively. Two robots push the box while running the algorithm to detect the box position. Then the results are compared with the measurements by the Motion Capture, which serves as the benchmark. The results and corresponding error distributions obtained by the algorithm are displayed in Fig. 6 and Fig. 7.

Figure 6 demonstrates both measurement results on box trajectory from the lidar and the Motion Capture. From this figure, the whole track detected by lidar stays close to the track monitored by the Motion Capture, which shows that the object localization algorithm performs well. Figure 7 displays the measurement error distribution on x, y, θ between the results from lidar and those from the Motion Capture. From Fig. 7, all error distributions are concentrated at zero and follow Gaussian distribution in the rough. $x_{e,0}$ is highly centralized at 0; $y_{e,0}$ is mainly concentrated between $[-0.05, 0.05](m)$ and is centralized at about $-0.01m$; $\theta_{e,0}(rad)$ is roughly within $[-0.05, 0.05]$ and centralized at zero; overall the errors are acceptable compared to the size of the box. Figure 7 further indicates that the performance of the object localization algorithm is reliable. Therefore, combined with the results from Fig. 6 and Fig. 7, based on lidar data, the onboard object localization algorithm is creditable to find the real-time location of the box with a torrent range of errors.

C. Lane-changing Experiment

Lane-changing experiment is derived from driving behaviours where drivers need to change from one lane to another lane while driving. Here, this experiment is conducted because changing lane is a common strategy in obstacle avoidance and path planning, and it can test the applicability of the generalized multi-robot leader-follower system by containing both moving straightly part and turning directions part. The lane-changing experiment is divided into three stages. Stage I is the stage where the system moves forward straightly. In Stage II, the system does the lane-changing task by following a designed curve. The three-order

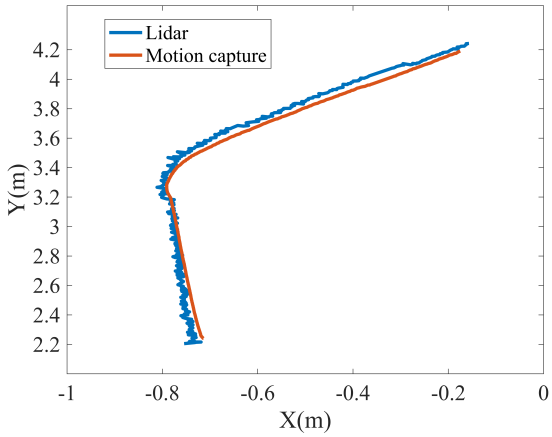

Fig. 6. Measurement results obtained by lidar and Motion Capture. The blue line represents the box trajectory obtained by the onboard object localization algorithm, and the red line displays the box trajectory monitored by the Motion Capture.

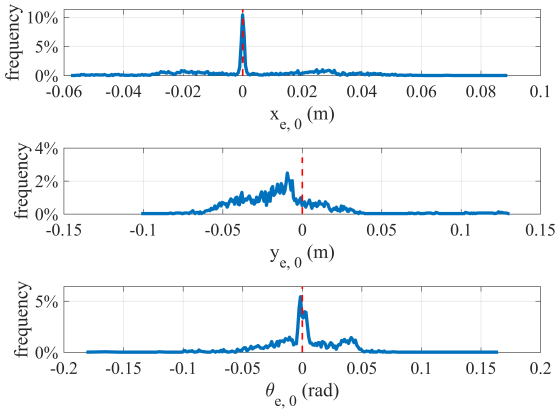

Fig. 7. Measurement error distribution of the state vector $\begin{bmatrix} x & y & \theta \end{bmatrix}^T$. The blue line represents the frequency of error and the red line calibrates zero point.

polynomial curve is adopted here, as it is easily computed and meets the requirement that the system can change lane smoothly. In Stage III, the system goes forward straightly after finishing lane changing. According to the start point and end point of the lane-changing path, the fitted polynomial curve is obtained, and then the path is determined. This reference trajectory of the box in the lane-changing part can be expressed as

$$\begin{cases} x_r = -0.11y_r^3 + 1.73y_r^2 - 4.88y_r + 6.68, \\ v_r = 0.05m/s, \\ w_r = 0. \end{cases} \tag{12}$$

To be noted here, because angular velocity is relatively small, the angular velocity of the reference trajectory is set to zero and the controller is served to adjust real angular velocity.

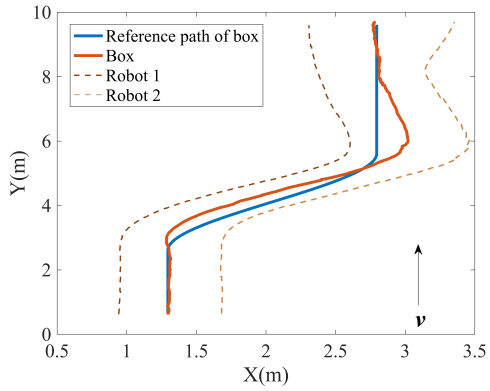

Fig. 8. Tracking result of the box and the two robots. The red line represents box experimental trajectory, the blue line represents reference trajectory, and two dashed line represent robot experimental trajectories.

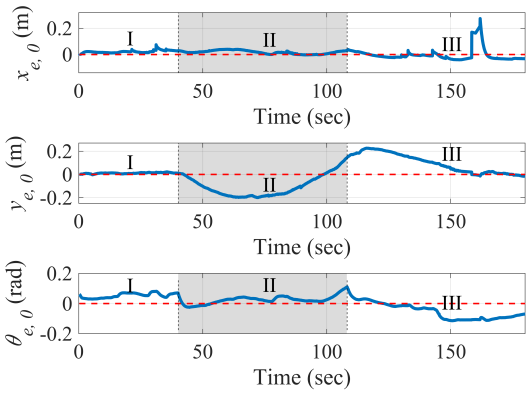

Fig. 9. Tracking errors of box. Stage I represents the first stage: straight-line movement; stage II represents the second stage: lane-changing movement with a three-order polynomial curve; stage III represents the third stage: straight-line movement after changing lane.

In this experiment, the distance d between two robots is $0.75m$, the radius R of both robots is $0.19m$, and the initial trajectory tracking errors of two robots are set to zero. The parameters of the controller are $c_1 = 1.3, c_2 = 1.6$ and $c_3 = 0.2$. Figure 8 and 9 present the result.

Figure 8 shows that the tracking result of the system. The system moves at the first lane of 2 meters. After that, it changes lane with the three-order polynomial curve. Then the system moves at the second lane of 4 meters. In Fig. 8, the box follows the trajectory in a torrent error range and lane-changing task is accomplished. Figure 9 shows the tracking error of box, and it reveals that the error is reducing within each stage. $x_{e,0}$ of the box roughly stays around zero. A sharp increase of $x_{e,0}$ in Stage III may be because Robot 2 wants to adjust its position and this adjustment influences the box, while this sharp increase of $x_{e,0}$ is fixed quickly after Robot 2 finishes its position adjustment. There are some fluctuations in $y_{e,0}$ and they start at the transitions of stages. This phenomenon may come from that the transition of stages depends on the reference trajectory of the box instead of the box current state, which casts delays in y of the box and makes $y_{e,0}$ increase at stage transitions. As $y_{e,0}$ is decreasing at each stage in Fig. 9, is suggests that the controller is trying to fix $y_{e,0}$ and achieves considerable results. As for $\theta_{e,0}$ of the box, it stays around zero and has little fluctuations when the stages are shifted, which is acceptable. Overall, errors are decreased by the control strategy gradually and the actual trajectory converges to the reference trajectory basically, which proves that the improved leader-follower control strategy functions well and validates the object localization algorithm.

V. Conclusion

Herein, we present a generalized multi-robot leader-follower system to solve collaborative object transportation problem, where the transported object is considered as an virtual leader. Besides, an onboard object localization algorithm is designed which uses lidar data to obtain the real-time position of the object during experiments. The measurement accuracy test shows that the proposed algorithm has high precision within $0.07m$ and $0.05rad$. The lane-changing experiment result proves that, based on the given system, the object can be transported along the reference trajectory within allowed error range.

In the future, we would like to extend our study by considering transporting objects with irregular shapes with an updated object localization algorithm, and plan to employ more robots to accomplish transportation tasks in dynamic environments.

References

[1] J. E. Inglett and E. J. Rodrguez-Seda, "Object transportation by cooperative robots", in *SoutheastCon 2017*, 2017, pp. 1-6: IEEE.

[2] A. Yufka and M. Ozkan, "Formation-based control scheme for cooperative transportation by multiple mobile robots," *International Journal of Advanced Robotic Systems*, vol. 12, no. 9, p. 120, 2015.

[3] C. K. Verginis, A. Nikou, and D. V. Dimarogonas, "Communication-based decentralized cooperative object transportation using nonlinear model predictive control," in *2018 European Control Conference (ECC)*, 2018, pp. 733-738: IEEE.

[4] A. Nikou, C. Verginis, S. Heshmati-Alamdari, and D. V. Dimarogonas, "A nonlinear model predictive control scheme for cooperative manipulation with singularity and collision avoidance," in *2017 25th Mediterranean Conference on Control and Automation (MED)*, 2017, pp. 707-712: IEEE.

[5] M. A. Teixeira, H. B. Santos, A. S. de Oliveira, J. A. Fabro, L. V. R. de Arruda, and F. Neves-Jr, "Cooperative load transport based on fuzzy logic controllers," in *2016 12th IEEE International Conference on Industry Applications (INDUSCON)*, 2016, pp. 1-6: IEEE.

[6] J. Chen, M. Gauci, W. Li, A. Kolling, and R. Gro, "Occlusion-based cooperative transport with a swarm of miniature mobile robots," *IEEE Transactions on Robotics*, vol. 31, no. 2, pp. 307-321, 2015.

[7] Z. Wang, G. Yang, X. Su, and M. Schwager, "Ouijabots: Omnidirectional robots for cooperative object transport with rotation control using no communication," in *Distributed Autonomous Robotic Systems: Springer*, 2018, pp. 117-131.

[8] Y. Dai, D. Qian, and S. Lee, "Multiple robots motion control to transport an object", *Filomat*, vol. 32, no. 5, 2018.

[9] A. Tsiamis, C. P. Bechlioulis, G. C. Karras, and K. J. Kyriakopoulos, "Decentralized object transportation by two nonholonomic mobile robots exploiting only implicit communication," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 171-176: IEEE.

[10] M. Fujii, W. Inamura, H. Murakami, K. Tanaka, and K. Kosuge, "Cooperative control of multiple mobile robots transporting a single object with loose handling," in *2007 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2007, pp. 816-822: IEEE.

[11] X. Chen and Y. Li, "Cooperative transportation by multiple mobile manipulators using adaptive NN control," in *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, 2006, pp. 4193-4200: IEEE.

[12] Z. Wang and M. Schwager, "Kinematic multi-robot manipulation with no communication using force feedback," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 427-432: IEEE.

[13] E. Tuci, M. H. Alkilabi, and O. Akanyeti, "Cooperative object transport in multi-robot systems: A review of the state-of-the-art," *Frontiers in Robotics and AI*, vol. 5, p. 59, 2018.

[14] C. P. Tang, R. M. Bhatt, M. Abou-Samah, and V. Krovi, "Screw-theoretic analysis framework for cooperative payload transport by mobile manipulator collectives", *IEEE/ASME Transactions on Mechatronics*, vol. 11, no. 2, pp. 169-178, 2006.

[15] I. Mas and C. Kitts, "Object manipulation using cooperative mobile multi-robot systems", in *Proceedings of the World Congress on Engineering and Computer Science*, 2012, vol. 1.

[16] G. A. Pereira, M. F. Campos, and V. Kumar, "Decentralized algorithms for multi-robot manipulation via caging", *The International Journal of Robotics Research*, vol. 23, no. 7-8, pp. 783-795, 2004.

[17] S. Moon, D. Kwak, and H. J. Kim, "Cooperative control of differential wheeled mobile robots for object pushing problem", in *2012 12th International Conference on Control, Automation and Systems*, 2012, pp. 140-144: IEEE.

[18] B. P. Gerkey and M. J. Mataric, "Pusher-watcher: An approach to fault-tolerant tightly-coupled robot coordination," in *Proceedings 2002 IEEE International Conference on Robotics and Automation* (Cat. No. 02CH37292), 2002, vol. 1, pp. 464-469: IEEE.

[19] H. Sugie, Y. Inagaki, S. Ono, H. Aisu, and T. Unemi, "Placing objects with multiple mobile robots-mutual help using intention inference," in *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, 1995, vol. 2, pp. 2181-2186: IEEE.

[20] X. Yu, L. Liu, and G. Feng, "Trajectory tracking for nonholonomic vehicles with velocity constraints', *IFAC-PapersOnLine*, vol. 48, no. 11, pp. 918-923, 2015.

[21] Y. Kim and H. Bang, *Introduction and Implementations of the Kalman Filter*, IntechOpen, 2019, pp. 1-16.