Cooperative Carrying Task Control based on Receding Horizon Control for Mobile Robots

Kou Nakamura and Tohru Kawabe

***Abstract*—** A new control method for a cooperative carrying task control problem by two mobile robots is proposed. In the problem, a leading robot is assumed to be controlled by a human directory or remotely. On the other hand, a following robot must be run autonomously anytime without dropping a carrying object. The proposed method based on the receding horizon control (RHC) generates optimal left and right wheel motor torques of the following robot at each sampling time to hold constraint condition of relative position. Numerical examples are shown to demonstrate the effectiveness of the method.

***Keywords*–** Cooperative caring task, Tracking, Mobile robot, Receding horizon control, Relative position constraint

I. Introduction

RECENTLY, according to the progress of control theories and robot technologies, robots are expected to work for human support, or instead of human, under the dangerous condition. For example, rescue robots[1], which save the people life at the disaster e.g., big earthquake and so on, has been researched and developed in the world-wide. Also, many working robot in factories[2] or in construction work[3] are developed.

Under such situations, if two or more mobile robots can work cooperatively and autonomously, we can expect that the work efficiency is improved and the work plan is able to be flexible. From such a viewpoint, there have been many researches about the control problems with multiple mobile robots for various tasks. For example, the formation control using information of the relative distance and angle between leading robot and each following robot has been developed by [4]. The tracking control considering collision avoidance among followers by [5] is also targeted multiple robots. Anyway, powerful and effective control method is need for these problems [6], [7].

In last few decades, RHC (Receding Horizon Control) has been widely accepted in the industries[8], [9], since it's practical and effective control method for various systems. In RHC algorithm, manipulated variables are obtained by solving on-line optimization problem at each sampling time. Conventionally, RHC has been used the systems with slow-moving dynamics, for example, chemical process and so on, due to the computation time. But, recent dramatic improvement of computer performance has made it possible to control the systems with fast-moving dynamics (mechanics, electronics and so on). Hence, the digital RHC method has been effective for various kinds of continuous-time objects. In addition to this, RHC is one of the most effective controls against constrained systems. Actual controlled object has often constraints conditions: performance limitation of actuator, the cost for control, and safety margin and so on. If the controller ignores such constraints when to decide the manipulated values, it is possible that the system should be breakdown in the worst case. In contrast, RHC controller can calculate manipulated values not to violate the constraints, thus RHC is very suitable for practical use.

In this paper, therefore, new control method based on RHC for the cooperative carrying task problem of two two-wheeled mobile robots. In this problem, the most important constraint is that the following robot must be controlled anytime to hold the condition of relative position with given margin. The following robot is controlled by using only relative position information (without using absolute position information). The relative position must be within the restricted range in any situation to prevent a carried thing from dropping. The RHC seems to be a best way to improve the control performance meeting with such severe requirements. Numerical examples are given to demonstrate the effectiveness of the proposed method.

This paper is organized as follows. In section 2, an overview of RHC is introduced. In section 3, the robot model and problem formulation of the cooperative carrying task is defined. In section 4, the proposed method is shown as main contribution and in section 5, numerical simulation results are given. Finally, in section 6, concluding remarks and future works are stated.

II. Preliminaries: RHC

RHC algorithm is to decide the optimal manipulated values which converge the controlled values to reference values by iteration of optimizing a cost function under constraints[10]. To take advantage of the modern control theory, RHC mainly use the state space model to describe the controlled object[11]. At current time-step k controlled variables $x(k)$ is measured, and RHC controller predict the behavior of the controlled variables sequence from $\hat{x}(k+1)$ to $\hat{x}(k+H_p)$ by the dynamic model of the controlled object described as eq. (1). $\hat{}$ denotes a predictive value at k.

K. Nakamura is a Ph.D student of Department of Computer Science, Graduate School of Systems and Information Engineering, University of Tsukuba, Tsukuba 305-8573, Japan (e-mail: *nakamura@acs.cs.tsukuba.ac.jp*).,

T. Kawabe is with Department of Computer Science, Graduate School of Systems and Information Engineering, University of Tsukuba, Tsukuba 305-8573, Japan (e-mail: *kawabe@cs.tsukuba.ac.jp*).

$$x(k+1) = Ax(k) + Bu(k) \tag{1}$$

The behavior of system depends on future manipulated variables sequence from $\hat{u}(k)$ to $\hat{u}(k+Hp-1)$, that is why RHC controller calculate the sequence $U = [u(k),\ u(k+1),\ \cdots,\ u(k+H_P-1)]$ which makes desired behavior from perspective of cost-minimizing. After calculating, only $\hat{u}(k)$ is inputted to controlled object as current actual input, then, at the next time-step the plant state is sampled again and the prediction and the calculation are repeated. The outline of RHC algorithm is shown as fig.1.

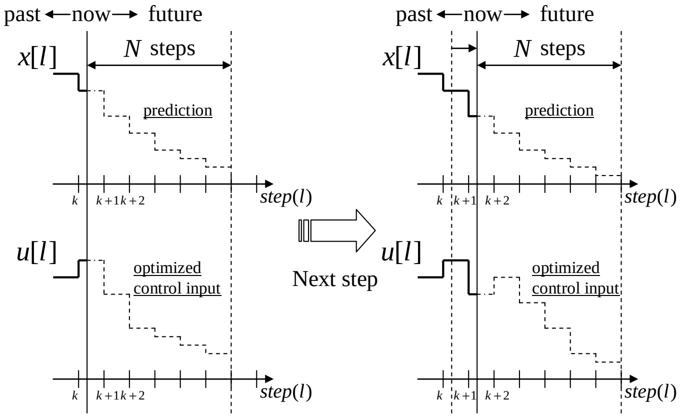

Fig. 1. The outline of RHC

The cost function $J(k)$ at current time-step is given by

$$J(k) = \sum_{i=0}^{H_p-1} \left\{ \| \hat{x}(k+i+1) - x^d \|_Q^2 + \| \hat{u}(k+i) \|_R^2 \right\}. \tag{2}$$

The optimization problem with constraints is given by

$$\min_U J(k) \tag{3}$$

subject to

$$\begin{aligned} x_{min} &\leq \hat{x}(k+i) \leq x_{max} \\ u_{min} &\leq \hat{u}(k+i) \leq u_{max} \\ & \qquad i = 0, 1, \cdots, Hp. \end{aligned} \tag{4}$$

We assume the controlled object is a multi-input multi-output system, thus $x(k)$ and $u(k)$ are vectors with adequate dimensions. $\| x \|_Q^2$ denotes the quadratic form $x^T Qx$, and x^d reference value and where Q and R are weighting matrices.

III. Problem formulation

A. Model of two-wheeled robot

The two-wheeled robot has two motors which rotate independently. Although there are many control methods using velocities and angular velocities as manipulated variables[12], the dynamic model of the robot is used in this paper. Therefore, motor torques are set as manipulated variables[13], then the robot is torque-controlled and has two independent inputs. We assume the center of gravity (C.G.) of the robot corresponds to center of the two wheels, and let the position of C.G.set $(x,\ y)$, and θ denotes robot's direction (see fig. 2). The dynamic model of robot can be described following state space model eq. (5)[14]. Controlled variable v and ω are the velocity of C.G. and angular velocity respectively, u_r and u_l is right and left motors torques. The definition of parameters is shown in table. I.

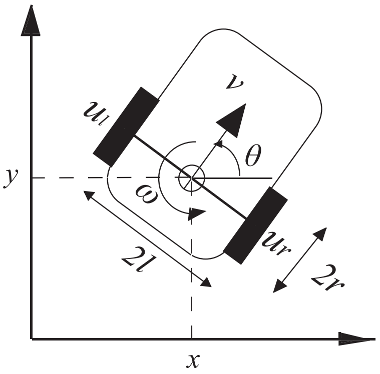

Fig. 2. Two-wheeled robot

TABLE I
DEFINITION OF ROBOT PARAMETERS

I_w	Inertia moment $[Nms^2/rad]$
M	Weight $[kg]$
I_v	Inertia moment about rotation center $[Nms^2/rad]$
l	Distance between wheel and rotation center $[m]$
c	Viscosity coefficient of friction $[Nms/rad]$
r	Wheel radius $[m]$
ϕ_r/ϕ_l	Rotation angle of left/right wheel $[rad]$

$$\begin{bmatrix} \dot{v} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} a_1 & 0 \\ 0 & a_2 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} + \begin{bmatrix} b_1 & b_1 \\ b_2 & -b_2 \end{bmatrix} \begin{bmatrix} u_r \\ u_l \end{bmatrix} \tag{5}$$

where

$$a_1 = \frac{-2c}{Mr^2 + 2I_\omega},\ a_2 = \frac{-2cl^2}{I_v r^2 + 2I_\omega l^2},$$
$$b_1 = \frac{r}{Mr^2 + 2I_\omega},\ b_2 = \frac{rl}{I_v r^2 + 2I_\omega l^2}$$

The relation between $(v,\ \omega)$ and $(x,\ y,\ \theta)$ is described in eq. (6).

$$\dot{x} = v\cos\theta,\ \dot{y} = v\sin\theta,\ \dot{\theta} = \omega \tag{6}$$

Input torques u_r and u_l change v, and ω according to eq. (5), v and ω change x, y, and θ according to eq. (6), too. Thus, we need to get proper torques which lead the robot to a desired position.

B. Control problem of the cooperative carrying task

Fig. 3 shows the cooperative carrying task. We assume that working environment is horizontal flat surface and there is no obstacle. A mobile robot's moving path is determined by the leading robot , which is assumed to be controlled by human, called "*Leader*", and the following robot is called "*Follower*".

Follower calculates its relative position $(x,\ y)$, direction θ and velocities $(v,\ \omega)$ from the communication with *leader* (by using *leader*'s position and direction). The control purpose is to generate *follower*'s input torques u_{fr} and u_{fl} which converge relative distance D to be desired distance D^d without violation of the position constraint described eq. (7) wherever *leader* is.

$$D_{min} \ \leq \ D \ \leq \ D_{max} \tag{7}$$

Leader's and *follower*'s behavior are summarized as follows.

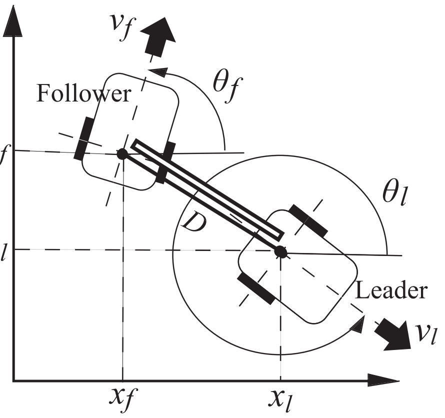

Fig. 3. Cooperative carrying task by Leader and Follower

Leader : runs arbitrarily (driven by a human directly or remotely), and position, direction and velocities at each sampling time is expressed by $(x_l,\ y_l,\ \theta_l)$, $(v_l,\ \omega_l)$ respectively.

Follower : calculate own input torques u_{fr} and u_{fl} using relative position, direction and velocities expressed by $(x_f,\ y_f,\ \theta_f)$, $(v_f,\ \omega_f)$ respectively from the communication with the *leader*.

IV. RHC Method for Cooperative Carrying Task

A. Generation of reference control values

As mentioned above, the two-wheeled robot usually can not control by the continuous feed-back law. About this point, Astolfi has been proposed the discontinuous feedback law which can solve the normal tracking control problem of non-holonomic mobile robot[15] as follows.

Astolfi's control law

The robot's position and direction $(x,\ y,\ \theta)$ are converted three variables $(\rho,\ \alpha,\ \phi)$ according to eq. (8) (see fig. 4).

$$\begin{aligned} \rho &= \sqrt{x^2+y^2} \\ \alpha &= -\theta + \arctan\frac{-y}{-x} \\ \phi &= \frac{\pi}{2} - \theta \end{aligned} \tag{8}$$

Then we can get v and ω which converge $(x_f,\ y_f,\ \theta_f)$ to $(0,\ 0,\ \pi/2)$ according to eqs. (9) and (10). K_ρ, K_α and K_ϕ are the control parameters.

$$v = K_\rho \rho \tag{9}$$

$$\omega = K_\alpha \alpha + K_\phi \phi \tag{10}$$

Astolfi's control law leads the robot to the origin wherever the robot is. Although this method is a superior method for the normal tracking control problem of one mobile robot, it's not applicable to the cooperative carrying task problem with two robots as it is.

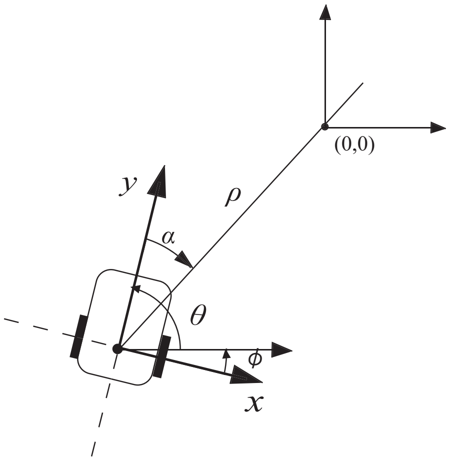

Fig. 4. Polar coordinate model

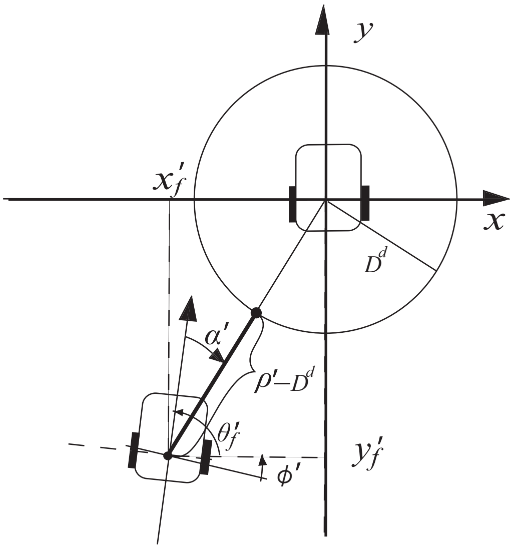

Fig. 5. Coordinate after conversion

To overcome the fault of Astolfi's method, firstly, coordinate conversion which changes $(x_l,\ y_l,\ \theta_l)$ to origin $(0,\ 0,\ \pi/2)$ is done. After this conversion to $(x_f,\ y_f,\ \theta_f)$, it changes to $(x'_f,\ y'_f,\ \theta'_f)$ as shown in fig. 4. Then, $(x'_f,\ y'_f,\ \theta'_f)$ is converted to (ρ', α', ϕ') according to eq. (11).

$$\begin{aligned}\rho' &= \sqrt{(x'_f)^2+(y'_f)^2}\\ \alpha' &= -\theta'_f+\arctan\frac{-y'_f}{-x'_f}\\ \phi' &= \frac{\pi}{2}-\theta'_f\end{aligned}\tag{11}$$

ρ' is equal to D. Our purpose is to converge ρ' to D^d. If we apply Astolfi's control law without any modification, it converges ρ' to 0 (*follower* collides *leader*). To avoid the collision we should converge $\rho'-D^d$ (see fig. 5) to 0. Moreover, we should take account in the Leader's velocity. From these points, we modify eq. (9) to eq. (12). The first term of right side in eq. (12) works to converge ρ' to D^d, and the second term compensate the effect of v_l.

$$v_f^d = K_\rho\left(\rho'-D^d\right)+v_l \tag{12}$$

We do not modify eq. (10) because changing ω affect to follow Leader too much. Therefore, we use eq. (13), which is same eq. (10).

$$\omega_f^d = K_\alpha\alpha' + K_\phi\phi' \tag{13}$$

B. Generation of constraints

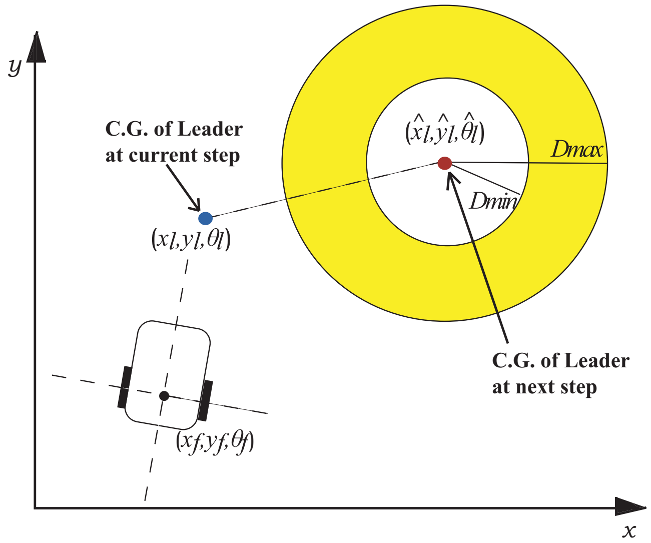

Fig. 6. The feasible region of *follower*

Follower must follow *leader* satisfying relative position condition at all time. In other words, *follower* must be inside of the yellow region, namely inside of circle $C2$ and outside of the circle $C1$ as shown in fig. 6, if any physical constraints due to the motor performance do not exist. The center of the $C1$ and $C2$ is predictive position and direction of *leader* $(\hat{x}_l,\hat{y}_l,\hat{\theta}_l)$ at next time-step, according to eq. (14)(T_s denotes sampling interval). The radius of the circle $C1$ is D_{min} and $C2$ is D_{max}.

This is feasible position area of *follower* at next step include physically impossible area. However, *follower* has constraint conditions due to motor performance actually. They are expressed with max values of velocity and angular velocity respectively. Therefore, the actual position is restricted smaller than this yellow region as follows.

Firstly, the *leader*'s position and direction at next step are predicted as

$$\begin{aligned}\hat{x}_l &= x_l + v_l\ \cos\theta_l\ T_s\\ \hat{y}_l &= y_l + v_l\ \sin\theta_l\ T_s\\ \hat{\theta}_l &= \theta_l + \omega_l\ T_s.\end{aligned}\tag{14}$$

Then, we derive the velocity constraints due to motor performance described as follows.

$$(v_{fmin}\ \leq)\ v_f\ \leq\ v_{fmax} \tag{15}$$

In fig. 7, we first consider the line segment from (x_f,y_f) to $(\hat{x}_l,\hat{y}_l)$, and we set $P1$ the intersection point of the segment with $C1$. $P1$ uses to obtain v_{fmax}, and is obtained by the smaller solution of following simultaneous equations eq. (16).

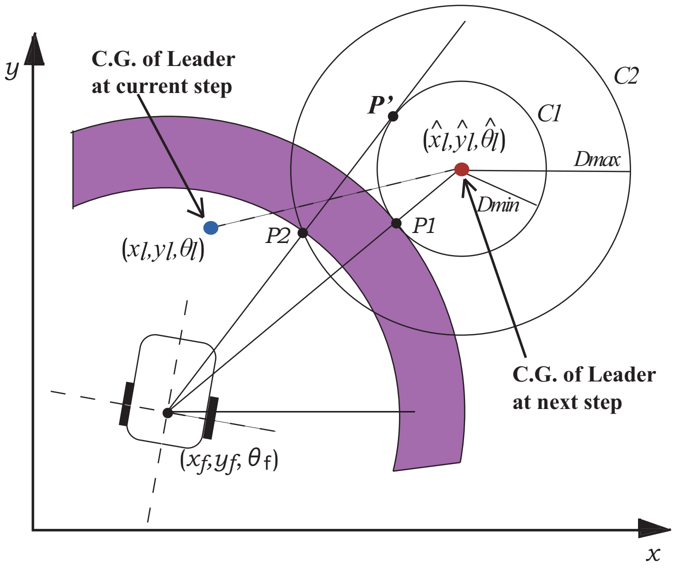

Fig. 7. The region restricted by velocity constraint

$$\begin{aligned}\sqrt{(\hat{x}_l-x_f)^2+(\hat{y}_l-y_f)^2} &= D\\ (x_{P1}-\hat{x}_l)^2+(y_{P1}-\hat{y}_l)^2 &= D_{min}^2\\ (x_{P1}-x_f)^2+(y_{P1}-y_f)^2 &= (D-D_{min})^2\end{aligned}\tag{16}$$

Then we consider tangent line from (x_f,y_f) to point of tangency P', and we set $P2$ the intersection point of the line with $C2$. $P2$ uses to obtain v_{min}. Now, P' is needed to obtain $P2$, and is obtained by one of the solution of following simultaneous equations eq. (17).

$$\begin{aligned}(x_{P'}-\hat{x}_l)^2+(y_{P'}-\hat{y}_l)^2 &= D_{min}^2\\ (x_{P'}-\hat{x}_l)(x_f-\hat{x}_l)+(y_{P'}-\hat{y}_l)(y_f-\hat{y}_l) &= D_{min}^2\end{aligned}\tag{17}$$

Then, $P2$ is obtained by one of the solution of following simultaneous equations eq. (eq:p2).

$$\begin{aligned}(x_{P2}-\hat{x}_l)^2+(y_{P2}-\hat{y}_l)^2 &= D_{max}^2\\ y_{P2}-y_f &= \tan\theta_1(x_{P2}-x_f)\\ \tan\theta_1 &= \frac{y_{P'}-y_f}{x_{P'}-x_f}\end{aligned}\tag{18}$$

Finally, we calculate the velocity constraint as shown in eq. (19). Since v_{fmax} is obtained by dividing the distance between $(x_f,\ y_f)$ and $(\hat{x}_l,\ \hat{y}_l)$ in the time T_s. If v_f is not larger than v_{fmax}, *follower* does not approach than D_{min} at next step (after T_s seconds). On the other hand, if v_f is not smaller than v_{fmin}, *follower* does not part from D_{max} at next step.

$$\begin{aligned}v_{fmax} &= \frac{\sqrt{(x_{P1}-x_f)^2+(y_{P1}-y_f)^2}}{T_s}\\ v_{fmin} &= \frac{\sqrt{(x_{P2}-x_f)^2+(y_{P2}-y_f)^2}}{T_s}\end{aligned}\tag{19}$$

Next, the angular velocity constraint is also described as eq. (20).

$$(\omega_{fmin}\ \leq)\ \omega_f\ \leq\ \omega_{fmax}\tag{20}$$

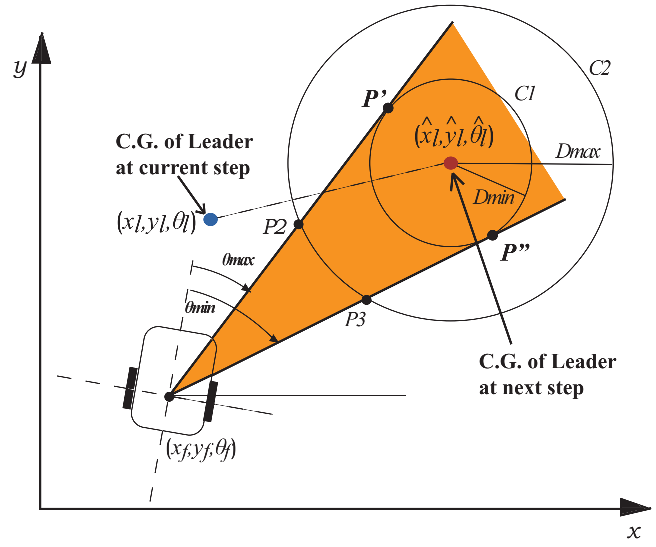

Fig. 8. The region restricted by angular velocity constraint

In fig. 8, let's consider $P3$ which is obtained as well as $P2$ using the other solution P'' of eq. (17). The relative angles θ_{max} and θ_{min} are set as shown in fig. 8, respectively. Then, ω_{fmax} and ω_{fmin} are calculated by following eq. (21). Since ω_{fmax} is obtained by θ_{max}/T_s, *follower*'s direction turns between P2 and P3 at next step if v_f satisfy the constraint.

$$\begin{aligned}\omega_{fmax} &= \frac{\theta_{max}}{T_s} = \frac{\arctan\dfrac{(y_{P2}-y_f)}{(x_{P2}-x_f)}-\theta_f}{T_s}\\ \omega_{fmin} &= \frac{\theta_{min}}{T_s} = \frac{\arctan\dfrac{(y_{P3}-y_f)}{(x_{P3}-x_f)}-\theta_f}{T_s}\end{aligned}\tag{21}$$

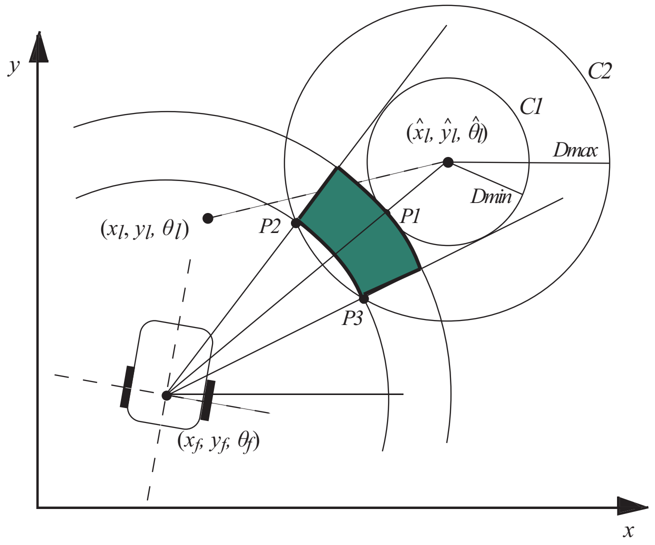

Fig. 9. The actual region restricted by all constraints

Finally, *follower* 's actual reachable position at next step without dropping the carrying object is restricted by velocity and angular velocity constraints due to physical motor performance into the green region in fig. 9. Namely, *follower* must be controlled to be in this green region at every next step.

C. RHC method

The proposed method based on RHC algorithm which calculates *follower* motor torques is summarized as follows. Fig. 10 illustrates the algorithm flow.

(Step1) Set initial values to all parameters.
(Step2) The controller get *leader*'s information $(x_l,\ y_l,\ \theta_l)$, $(v_l,\ \omega_l)$, and *follower*'s information $(x_f,\ y_f,\ \theta_f)$, $(v_f,\ \omega_f)$ at current time-step k.
(Step3) The controller calculates *follower*'s reference velocities according to eqs. (12) and (13), and conditions according to eqs. (19) and (21).
(Step4) The controller calculates input torques u_{fr} and u_{fl}.
(Step5) *Follower* runs by the input torque in Step 4. Return to Step 2 after T_s seconds.

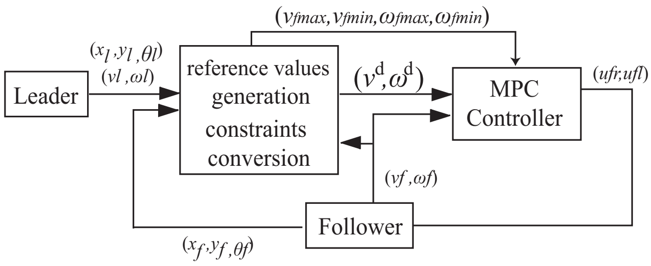

Fig. 10. A flow of control

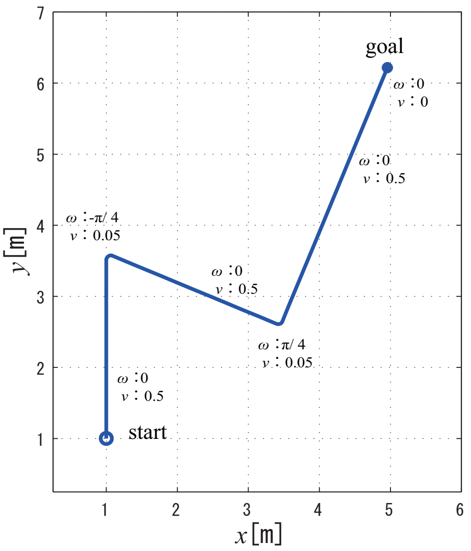

Fig. 11. Straight path

At Step 3-2, RHC controller solves the following optimization problems. Eq. (22) describe discretized dynamic model by sampling time T_s.

$$z_f(k+1) = A_d z_f(k) + B_d u_f(k) \tag{22}$$

where

$$z_f(k) = \begin{bmatrix} v_f(k) \\ \omega_f(k) \end{bmatrix},\ u_f(k) = \begin{bmatrix} u_{fr}(k) \\ u_{fl}(k) \end{bmatrix}$$

Thus, the input torques are obtained by the optimal solutions of following equations.

$$\min_{\hat{u}_f} \sum_{i=0}^{Hp-1} \| \hat{z}_f(k+i+1) - z^d(k) \|_Q^2 + \| \hat{u}_f(k+i) \|_R^2 \tag{23}$$

subject to

$$\begin{aligned} &\hat{z}_f(k+i+1) = A_d \hat{z}_f(k+i) + B_d \hat{u}_f(k+i) \\ &z_{fmin}(k) \leq \hat{z}_f(k+i) \leq z_{fmax}(k) \\ &\qquad\qquad i = 0, 1, \cdots, Hp-1 \end{aligned} \tag{24}$$

where

$$\hat{z}_f(k) = z_f(k),\ z_f^d(k) = \begin{bmatrix} v_f^d(k) \\ \omega_f^d(k) \end{bmatrix},$$

$$z_{fmin}(k) = \begin{bmatrix} v_{fmin}(k) \\ \omega_{fmin}(k) \end{bmatrix},\ z_{fmax}(k) = \begin{bmatrix} v_{fmax}(k) \\ \omega_{fmax}(k). \end{bmatrix}$$

V. Numerical examples

This section shows the results of the computer simulations to confirm the effectiveness of the proposed method. The parameters in table I are $I_w = 0.005$, $I_v = 0.05$, $l = 0.1$, $c = 0.05$, and $r = 0.05$. The control parameters set $K_\phi = -0.3$, $K_\rho = 1.5$, and $K_\alpha = 1.5$ initially. The weights of the cost function in eq. (23) are $Q = \begin{bmatrix} 100 & 0 \\ 0 & 10 \end{bmatrix}$ and $R = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$. The initial positions and directions of *leader* and *follower* are $(x_l,\ y_l,\ \theta_l) = \left(1,\ 1,\ \frac{\pi}{2}\right)$ and $(x_f,\ y_f,\ \theta_f) = \left(1,\ 0.5,\ \frac{\pi}{2}\right)$ respectively.

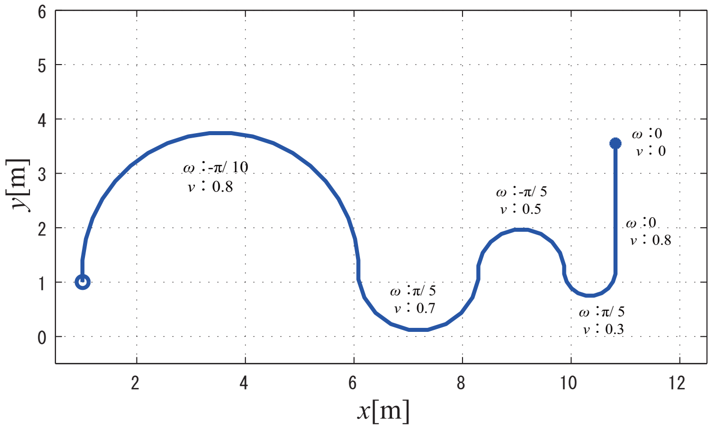

Fig. 12. Circular path

Let's desired relative distance $D^d = 0.5$, and assume two migration path called "Straight path" and "Circular path". These paths are shown in figs. 11 and 12. *Leader* changes own velocities on the path. For example, the velocity of C.G. is $v_l = 0.5$ in straight line, then v_l is down to 0.05 and an angular velocity occurs $\omega_l = \pm\pi/4$ at the corner of the straight path.

A. Result of straight path

Let the relative distance constraints be $D_{max} = 0.6$[m] and $D_{min} = 0.4$[m]. The result of *follower*'s path and the time fluctuation of D are shown in figs. 13 and 14. Fig. 13 indicates that *follower* can follow *leader* from start to goal. In addition to this, fig. 14 indicates though *follower* is almost close to D_{min} at $t = 8.5[s]$, there is no violation of the constraints while *follower* runs behind *leader*. These result show the two robots can carry the object.

In first condition, maximum value of D is 0.554[m], minimum is 0.401[m]. Then, we impose more severer constraints $D_{max} = 0.55$[m] and $D_{min} = 0.45$[m] which the result of the first condition do not satisfy. In this case, *follower* can also follow *leader* shown in fig. 17. Fig. 18 shows there is no violation of constraints of D. Compare fig. 14 with fig. 18, maximum of D is changed to $0.545[m]$ and minimum of D is changed to $0.450[m]$ in order to satisfy each constraints of D. This result indicates the effectiveness of the proposed method.

B. Result of circular path

Firstly, assume the relative distance constraints $D_{max} = 0.6$[m] and $D_{min} = 0.4$[m]. The result of *follower*'s path and the time fluctuation of D are shown in figs. 15 and 16. These results indicate carrying task is well done as well as the straight path.

Then, we also impose more severer constraints $D_{max} = 0.55$[m] and $D_{min} = 0.45$[m]. The results with the more severer constraint are shown in figs. 19 and 20. There is no violation of the relative distance constraint, but, we can see D does not converge to $D^d = 0.5$[m] when *follower* runs a first semicircle. This phenomenon seems to be attributed to ω^d because the phenomenon does not appear the straight path.

VI. Conclusion

In this paper, the RHC based control method for cooperative carrying task problem of two mobile robots has been proposed. The method get the optimal torques of *follower* robot under constraint conditions. Moreover, reference values of velocities and angular velocities that *follower* can keep up with *leader* which runs arbitrarily paths has been derived. This is realized by extending the Astolfi's control law, and converts geometrically a relative distance constraint into velocities' constraints which can easily deal with optimization problem in RHC algorithm.

Simulation results have been illustrated to indicate *follower* can runs effectively using the input torques calculated by the proposed method. Although severe conditions on same path are imposed, *follower* can run effectively satisfying relative position constraint.

To realize the more practical systems, there is a room to improve in the proposed method. One of the issues is to improve ω_f^d by using *leader*'s angular velocity for enhanced following performance when running in a circular path.

Other issue is to consider the effective method of the Digital-to-Analog (DA) conversion of control inputs. In the control problem of this paper, the continuous-time objects (robots) are controlled by a discrete-time controller (computer). In such system, the Analog-to-Digital(AD) and the DA conversions of signals are indispensable operations. In this paper, the conventional zero-order hold is assumed to be used for the DA conversion on the assumption that the analog signals in each sampling interval are considered as constant values. However, to improve the control performance, it's very important to take account of the behavior of systems in the sampling intervals. For this point, we have proposed the adaptive DA converter which switch the sampling functions optimally according to the system status [16]. Hence, we apply to this to the proposed method and verify the improvement of the performance as future.

Moreover, the robustness property of the proposed method with respect to the model uncertainties and disturbances should be added. One of the possible strategy for adding the robustness property is solving the so-called minimax optimization problem, namely minimization problem over the control input of the performance measure J maximized by plant model uncertainties or disturbances as follows. However, it is difficult to solve such problem, since the saddle point may not exist in general. Therefore, some effective solving method is need. Since we have proposed the new practical solving approach of minimax RHC [17], [18] problem recently, we will try to apply this approach to the proposed method as a future work.

References

[1] S. Tadokoro, "Robot Technologies for Assisting Search and Rescue in Disasters", *Journal of Society of Automotive Engineers of Japan*, Vol. 64(5), pp.48-51, 2010 (in Japanese).

[2] S. Sakakibara, "Robots Working at Factories : Industrial Robots, Now and the Future", *Journal of the Robotics Society of Japan*, Vol. 27(3), pp.263-264, 2009 (in Japanese).

[3] Y. Yanagihara, et al., "Development of the construction waste management RT system with the use of Next-generation Manipulator", *Journal of the Robotics Society of Japan*, Vol. 27(8), pp.842-850, 2009 (in Japanese).

[4] J.P. Desai, J. Ostrowski and V. Kumar, "Controlling formations of multiple mobile robots", *Proc. IEEE Intl. Conf. Robotics and Automation*, pp.2864-2869, 1998.

[5] K. Kon, H. Fukushima and F. Matsuno, "Model Predictive Formation Control with Multiple Mobile Robot Considering Collision Avoidance", *Trans. Society of Instrument and Control Engineers (Japan)* , Vol. 42(8), pp.877-883, 2006 (in Japanese).

[6] R.M. Murray, "Recent research in cooperative control of multi vehicle systems", *Journal of Dynamic Systems Measurement and Control-Transactions of The Asme*, Vol.129, pp.571-583, 2007

[7] T. Keviczkya, F. Borrelli, G. J. Balas "Decentralized receding horizon control for large scale dynamically decoupled systems", *Automatica*, Vol. 42(12), pp.2105-2115, 2006

[8] C.E. Garcia, D.M. Prett and M. Morari, "Model Predictive Contrl: Theory and Practice - a survey", *Automatica*, Vol. 25(3), pp.335-348, 1989.

[9] D.Q. Mayne, J.B. Rawlings, C.V. Rao and M. Scokaert, "Constrained model predictive control: Stability and optimality", *Automatica*, 36(6), pp.789-814, 2000.

[10] J.M. Maciejowski, *Predictive Control with Constraints*, (Trans. by S. Adachi and M. kanno), Tokyo Denki University Press, pp.44-126, 2005 (in Japanese).

[11] M. Ohshima and M. Ogawa, "Model Predictive Control-I- history and at present of model predictive control", *Systems, control and information*, Vol. 46(5), pp.286-293, 2002 (in Japanese).

[12] M. Sampei and M. Ishikawa, "State Estimation and Output Feedback Control of Nonholonomic Mobile Systems using Time-Scale Transformation ", *Trans. of the Institute of Systems, Control and Information Engineers (Japan)*, Vol. 13(3), pp.124-133, 2000 (in Japanese).

[13] R. Fierro, F.L. Lewis, "Control of a Nonholonomic Mobile Robot : Backstepping Kinematics into Dynamics", *Journal of Robotic Systems*, Vol. 14(3), pp.149-163, 1997.

[14] J. Tang, K. Watanabe, M. Nakamura and N. Koga, "Fuzzy Gaussian Neural Network Controller and Its Application to the Control of a Mobile Robot", *Trans. Japan Society of Mechanical Engineers, Series C*, Vol. 59(564), pp.2290-2297, 1993 (in Japanese).

[15] A. Astolfi, "Exponential stabilization of a wheeled mobile robot via discontinuous control", *ASME Journal of Dynamic Systems Measurement and Control*, **121**-1, pp.121-125, 1999.

[16] T. Motoyama, T. Kawabe, et al., "New Integrated Design Approach of RHC with Adaptive DA Converter", *WSEAS Transaction on Systems*, Issue 5, Vol. 5, pp.981-988, 2006.

[17] T. Kawabe and K. Hirata, "A New Approach to Finite Minimax Receding Horizon Control Problem with Constrained Conditions", *Trans. of The Japan Society of Mechanical Engineers* (C1), Vol. 70, No. 695, pp.1992-1998, 2004.

[18] T. Kawabe, "A practical Design Approach to Robust Receding Horizon Controller with Adaptive DA Converter", *Proc. of IASTED International Conference on Control and Applications 2009* , pp. 258-263, 2009.

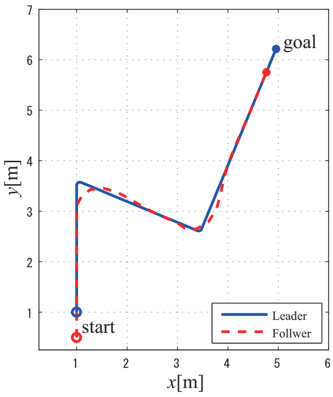

Fig. 13. Paths of Leader and Follower in the case of D_{max} = 0.6[m] and D_{min} = 0.4[m]

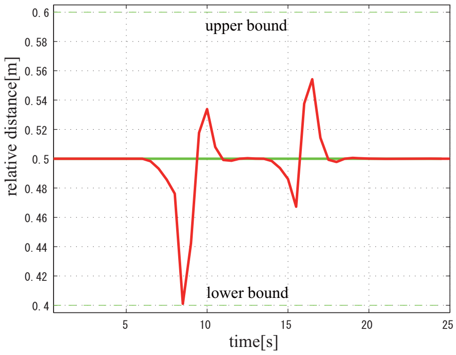

Fig. 14. Time fluctuation of relative distance in the case of D_{max} = 0.6[m] and D_{min} = 0.4[m]

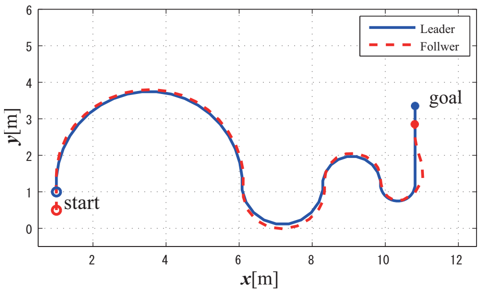

Fig. 15. Paths of Leader and Follower in the case of D_{max} = 0.6[m] and D_{min} = 0.4[m]

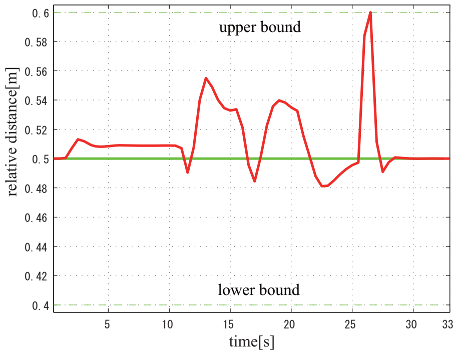

Fig. 16. Time fluctuation of relative distance in the case of D_{max} = 0.6[m] and D_{min} = 0.4[m]

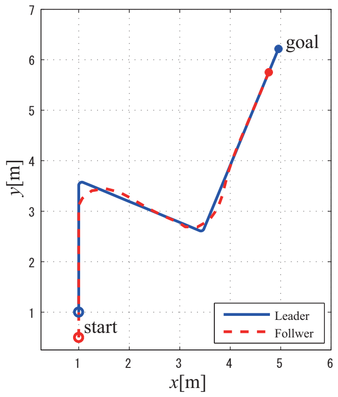

Fig. 17. Paths of Leader and Follower in the case of D_{max} = 0.55[m] and D_{min} = 0.45[m]

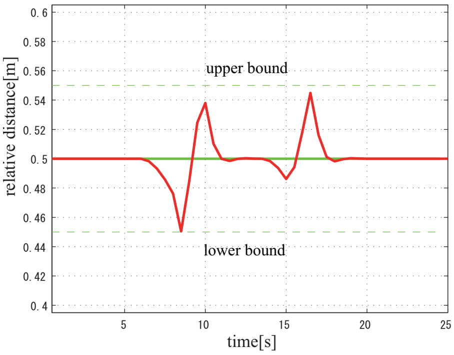

Fig. 18. Time fluctuation of relative distance in the case of D_{max} = 0.55[m] and D_{min} = 0.45[m]

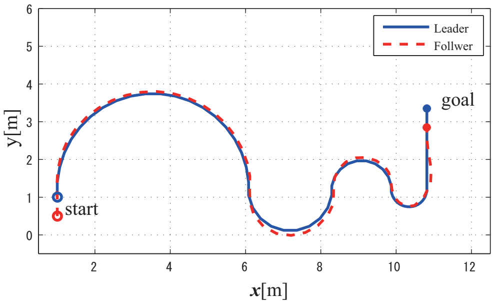

Fig. 19. Paths of Leader and Follower in the case of D_{max} = 0.55[m] and D_{min} = 0.45[m]

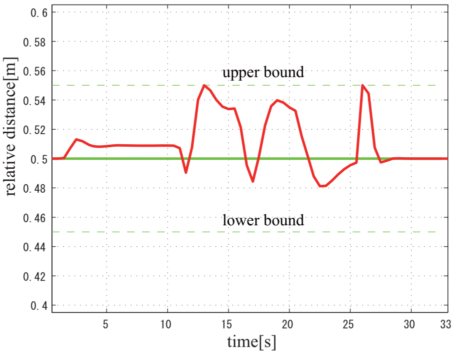

Fig. 20. Time fluctuation of relative distance in the case of D_{max} =0.55[m] and D_{min} = 0.45[m]