Coordination Control of Wheeled Mobile Robot Using MPC

Muhammad Farhan Manzoor*, Qinghe Wu
School of Automation
Beijing Institute of Technology
Beijing, China
farhan_manzoor@hotmail.com

Rana Javed Masood
College of Automation
Nanjing University of Aeronautics & Astronautics
Nanjing, China

***Abstract*—In the last decades Model predictive control (MPC) is the leading paradigm for high-performance and cost-effective control of complex systems. MPC contains powerful technique for optimizing the performance of constrained systems on state and control variables. For this reason, we use a distributed controller to coordinate a fleet of moving robots with unicycle kinematics. The proposed algorithm is based on the concept of robust MPC control with tube based strategy (i.e. it imposes low complexity and conservativeness). Using this approach we are able to guarantee that the real controlled system, on which a bounded disturbance is supposed to act, remains constrained in a neighborhood of the trajectory of the ideal system, which is defined by neglecting the effect of uncertainties. Obstacle avoidance function introduced for prevent collision between robot coordination. Matlab implementation and real robot results are used to show the efficacy of the proposed control strategy.**

Keywords—model predictive control; distributed; kinematics; robust; obstacle avoidance;

I. INTRODUCTION

At the present time, robotics applications are mounting at a rapid tempo. In fact, the development in technology allows for the progress and the implementation of advanced control algorithms. The interest has shifted to the use of a large number of simpler and more inexpensive agents, capable with cooperation capabilities. Distributed control solutions have been studied as a result of these needs due to their fault tolerance features and to the lower computational requirements. [1]

Model Predictive Control (MPC) has been shown to be useful for developing distributed multi-agent coordination algorithms, in view of the fact that predictions of the state trajectory are available at each time step. MPC is a popular control technique which is able to handle constraints in a systematic way [2–4]. These constraints might be imposed on any element of the system signals, such as inputs, outputs, states and mainly significantly actuator control signals. A fundamental question about MPC is its robustness to model uncertainty and noise. There are three main approaches to robust Model Predictive Control that tackle uncertainty, Min–Max MPC [5, 6] Constraint Tightening MPC[7, 8] and an efficient theoretical concept called ‘‘tube-based MPC’’, [9, 10]. Tube-based MPC for linear system has been verified to be an efficient control method under different types of uncertainty [11, 12]. The objective of this research is to achieve desired behaviors with low computational cost implementations easily programmed locally on agents [13]. Building upon concepts of [14] this paper extends the single-vehicle optimal control problem to a multi-vehicle, multi-obstacle environment where the vehicles cooperatively navigate a path in an optimal fashion. A good control scheme would allow the vehicles to follow their driver's commands while cooperatively traversing the environment without colliding with obstacles or with each other. Distributed cooperative control can be achieved within a unique control interacting with all vehicles [15] or computed by each vehicle [16].

II. MOBILE ROBOT KINEMATICS

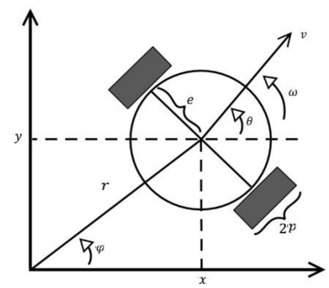

Figure 1. Unicycle Model

The unicycle model is presented in Figure 1 and kinematic equations that govern the motion of the robot is simply as given below

$$\begin{cases} \dot{x} = v\,cos\theta \\ \dot{y} = v\,sin\theta \\ \theta = \omega \end{cases} \tag{1}$$

Slipping prevention in the cross direction with existing constraint, we are able to obtain

$$\dot{y}\cos\theta - \dot{x}\sin\theta = 0 \tag{2}$$

Considering the accelerations along two axes in matrix form we have

$$\begin{pmatrix} a_x \\ a_y \end{pmatrix} = \begin{bmatrix} \cos\theta & -v\sin\theta \\ \sin\theta & v\cos\theta \end{bmatrix} \begin{pmatrix} \dot{v} \\ \omega \end{pmatrix}$$

If $v \neq 0$ the transformation is invertible we obtain the following first order dynamic linearization controller

$$\begin{cases} \dot{v} = a_x \cos\theta + a_y \sin\theta \\ \omega = \dfrac{-a_x \sin\theta + a_y \cos\theta}{v} \end{cases} \tag{3}$$

978-1-4673-7016-5/15 $31.00 © 2015 IEEE
DOI 10.1109/CICSyN.2015.49

Authorized licensed use limited to: Chalmers University of Technology Sweden. Downloaded on December 14,2023 at 10:13:34 UTC from IEEE Xplore. Restrictions apply.

By applying this controller to unicycle the feedback system is

$$\begin{cases} \ddot{x} = a_x \\ \ddot{y} = a_y \end{cases} \tag{4}$$

That is, as desired, and linear. The general control scheme is presented in Figure 2.

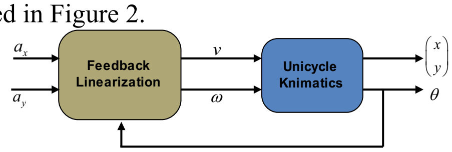

Figure 2. Control Scheme

III. PROBLEM FORMULATION

A. Robust Tube Based MPC

Consider a linear system

$$x(k+1) = Ax(k) + Bu(k) + w(k), \;\; w(k) \in W \tag{5}$$

Where $w(k)$ is a disturbance acting on the system, and W is a set that is compact and containing the origin. The objective is therefore to find a control law that guarantees convergence optimality and with respect to the constraints, regardless of the value assumed by disturbance in W. This disturbance represents the issues such as measurement errors and uncertainties on the model. Consider the nominal system associated with Eq. 5

$$\hat{x}(k+1) = A\hat{x}(k) + B\hat{u}(k) \tag{6}$$

The system in Eq. 6 use for the MPC technique, in order to obtain the optimum value of the input equation $\hat{u}(t:t+N-1|t)$ for the design of a controller that neglects the uncertainty. It is clearly necessary to define all constraints appropriately, that have to be applied to the new state variables and control. In particular, the robust input applied to the real system is obtained by adding a feedback between the nominal state and the real state

$$u(k) = \hat{u}(k) + K(x(k) - \hat{x}(k)) \tag{7}$$

Error variable is defined as $z(k) = x(k) - \hat{x}(k)$. The error dynamics of the real state with respect to the nominal one is

$$z(k+1) = (A+BK)z(k) + w(k) \tag{8}$$

If $(A+BK)$ is asymptotically stable, then there exists a robustly positively invariant set Z such that

$$z(t) \in Z, w(k) \in W, \forall k \geq t \Rightarrow z(t+i) \in Z, \forall i \geq 0 \tag{9}$$

The system error can always be constrained within the set as long as they are at the initial $(A+BK)\mathrm{Z} \oplus W \subseteq \mathrm{Z}$. Where the symbol $\oplus$ denotes the Minkowski sum operator between two sets. Now consider the constraints on the state and input of the real system

$$x(k) \in \mathrm{X}, u(k) \in U \tag{10}$$

On solving the optimization problem of the nominal system, it is necessary to transform the constraints into the relevant state and input variables. The new constraints with respect to Eq. 10 are assumed regardless of the value, by disorder $w(k)$

$$\hat{x}(k) \in \hat{\mathrm{X}} = \mathrm{X}\Theta\mathrm{Z} \quad u(k) \in \hat{U} = U\Theta K\mathrm{Z}w(t) \tag{11}$$

In particular we define the invariant set $\hat{\mathrm{X}}^f \subseteq \hat{\mathrm{X}}$ for each set $x(k) \in \mathrm{X}^f$ to the concept of the terminal set applied to the auxiliary law $\hat{u}(k) = K\hat{x}(k)$. Each nominal state evolves according to the closed loop law $\hat{x}(k+1) = (A+BK)\hat{x}(k)$ and the relation holds

$$\begin{cases} x(k+i) = (A+BK)\hat{x}(k) \in \hat{\mathrm{X}}^f \\ \hat{u}(k) = K\hat{x}(k+i) \in \hat{U} \end{cases} \forall i \geq 0$$

At this point we define the optimization problem by means of a quadratic cost function subject to the constraints

$$\begin{aligned} &\min_{\hat{u}(t:t+N-1)} \sum_{k=t}^{t+N-1} \|\hat{x}(k+i)\|_Q^2 + \|\hat{u}(k+i)\|_R^2 + \|\hat{x}(t+N)\|_P^2 \\ &\hat{x}(k+1) = A\hat{x}(k) + B\hat{u}(k) \\ &\hat{x}(k+i) \in \hat{\mathrm{X}}, \forall k = t:t+N-1 \\ &\hat{u}(k+i) \in \hat{U}, \forall k = t:t+N-1 \\ &\hat{x}(t+N) \in \hat{\mathrm{X}}^f \end{aligned} \tag{12}$$

Considering the choice of initial state of the system, $x(k) - \hat{x}(k) \in \mathrm{Z}$ is sufficient to obtain a trajectory of the real state such that $x(k+i) \in \hat{x}(k+i) \oplus \mathrm{Z}$, thus making this initial state a new optimization variable by imposing the constraint. Then the robust MPC problem is reformulated as

$$\begin{aligned} &\min_{\hat{x}(t)\hat{u}(t:t+N-1)} \sum_{k=t}^{t+N-1} \|\hat{x}(k+i)\|_Q^2 + \|\hat{u}(k+i)\|_R^2 + \|\hat{x}(t+N)\|_P^2 \\ &\hat{x}(k+1) = A\hat{x}(k) + B\hat{u}(k) \\ &\hat{x}(k+i) \in \hat{\mathrm{X}}, \forall k = t:t+N-1 \\ &\hat{u}(k+i) \in \hat{U}, \forall k = t:t+N-1 \\ &\hat{x}(t+N) \in \hat{\mathrm{X}}^f \\ &x(k) - \hat{x}(k) \in \mathrm{Z} \end{aligned} \tag{13}$$

The control variable for the real system is again constructed according to the robust approach

$$u(t) = \hat{u}(t|t + K(x(t) - x(t|t))$$

For the optimal input sequence $\hat{u}(t:t+N-1)$. The resulting scheme of control, the effect of a closed loop, and the nominal system used to solve MPC is dependent on the state variables of the real system as shown in Figure 3.

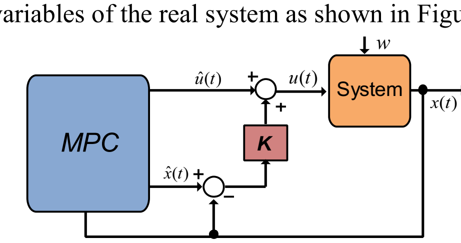

Figure 3. Tune Based strategy

Authorized licensed use limited to: Chalmers University of Technology Sweden. Downloaded on December 14,2023 at 10:13:34 UTC from IEEE Xplore. Restrictions apply.

B. Coordination Control Problem Formulation

Suppose the system that is to perform coordination is composed of N agents and each of them described by a linear discrete time dynamic equation given as

$$x^{[i]}(k+1) = Ax^{[i]}(t) + Bu^{[i]}(t) \quad (14)$$

And output variables associated with the equation is

$$z^{[i]}(t) = Cx^{[i]}(t) \quad (15)$$

The adaptation of feedback linearization control describes the dynamics of single robot as a linear and time-invariant system for double Integrator dynamics. For system let us define a vector of reference position along the prediction horizon N as $\tilde{y}^{[i]}_{[t:t+N-1]} = \{\tilde{y}_k, k = t \ldots t+N-1\}$. From here the equivalent references vector for the state and input derived for the control, the difference between the reference position and the equivalent predicted is shown by observer

$$\begin{aligned}\tilde{x}^{[i]}(k+1) &= A\tilde{x}^{[i]}(k) + B\tilde{u}^{[i]}(k) + G^x\left(\tilde{y}^{[i]}(k+1) - C\tilde{x}^{[i]}(k)\right)\\ \tilde{u}^{[i]}(k+1) &= \tilde{u}^{[i]}(k)G^u\left(\tilde{y}^{[i]}(k+1) - C\tilde{x}^{[i]}(k)\right)\end{aligned} \quad (16)$$

The matrix that governs the new system will be asymptotically stable

$$\tilde{A} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \quad \tilde{C} = \begin{bmatrix} C & 0 \end{bmatrix} \quad \tilde{G} = \begin{bmatrix} G^x \\ G^u \end{bmatrix}$$

It is necessary that the gain $\tilde{G}$ is selected for the closed loop system that has as a dynamic matrix $\tilde{A} - \tilde{G}\tilde{C}$ due to the discrete-time case, and all eigenvalues lie within a unit circle. In this way the guaranteed convergence of the two references is asymptotically stable. In order to be consistent with the idea of robust control and to ensure appropriate boundedness properties of trajectory tracking error, we define a set of 2ε which lie within the difference between two successive references and this is indicated with $B_\varepsilon(0)$

$$\tilde{y}^{[i]}_{t+1} \in \tilde{y}^{[i]}_t \oplus B_\varepsilon(0) \quad (17)$$

Starting from this condition by using the equation of speed to define the stationary condition this corresponds to the output $\tilde{y}^{[i]}_t$. From the above we obtain the following expression that is used to define the set Δ_i^{SS}

$$\Delta_i^{SS} = \underbrace{\begin{bmatrix} x^{[i]\,SS}_{t+1} & x^{[i]\,SS}_t \\ u^{[i]\,SS}_{t+1} & u^{[i]\,SS}_t \end{bmatrix}}_{\Psi} \in \begin{bmatrix} I-A & -B \\ -C & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ I \end{bmatrix} B_\varepsilon(0)$$

In regard to references x and u we can then write

$$\Psi = \left(\tilde{A} - \tilde{G}\begin{bmatrix} C & 0 \end{bmatrix}\right)\begin{bmatrix} x^{[i]}_t - x^{[i]\,SS}_t \\ u^{[i]}_t - u^{[i]\,SS}_t \end{bmatrix} + \left(\tilde{A} - \tilde{G}\begin{bmatrix} C & 0 \end{bmatrix}\right)\begin{bmatrix} x^{[i]\,SS}_t - x^{[i]\,SS}_{t+1} \\ u^{[i]\,SS}_t - u^{[i]\,SS}_{t+1} \end{bmatrix}$$

The matrix $\left(\tilde{A} - \tilde{G}\begin{bmatrix} C & 0 \end{bmatrix}\right) = \tilde{F}$ is hypothetically stable. Implementation of a robust control MPC requires a nominal system with optimization. In particular, the considerations made so far, select the structure for the observer

$$\hat{x}^{[i]}_{t+1} = A\hat{x}^{[i]}_t + B\hat{u}^{[i]}_t + G^x\left(\tilde{y}^{[i]}_{t+1} - C\tilde{x}^{[i]}_t\right)$$

The actual value of the control variable to be applied to the system is obtained by a linear term, based on an auxiliary law K

$$u^{[i]}_t = \hat{u}^{[i]}_t + K\varepsilon^{[i]}_t \quad (18)$$

For the problem of coordination, we define certainty set around the reference trajectory within which we are guaranteed to find the real trajectory and all the information we need to prevent collisions between multiple agents. Adding constraint to each MPC

$$C\left(\hat{x}^{[i]}_k - \tilde{x}^{[i]}_{t+1}\right) \in \Delta_i^z \quad k = t \ldots t+N-1$$

With the set Δ_i^z is completely arbitrary

$$C\tilde{x}^{[i]}_k \oplus Z_i = Cx^{[i]}_k \in C\tilde{x}^{[i]}_k \oplus \varepsilon_i \oplus \Delta_i^z$$

Then value function toward the set point of navigation is

$$\begin{aligned}V_i^N = &\sum_{k=t}^{t+N-1} \left\|\hat{x}^{[i]}_k - \tilde{x}^{[i]}_k\right\|^2_Q + \left\|\hat{u}^{[i]}_k - \tilde{u}^{[i]}_k\right\|^2_R + \left\|\hat{x}^{[i]}_{t+N} - \overline{x}^{[i]}_{t+N}\left(\overline{y}^{[i]}_{t+N}\right)\right\|^2_P \\ &+ \left\|\overline{y}^{[i]}_{t+N} - \overline{y}^{[i]}_{t+N-1}\right\|^2 + \left\|\overline{y}^{[i]}_{t+N} - \overline{y}^{[i]}_{setpoin}\right\|^2_T + V_i^{coll}\end{aligned} \quad (19)$$

That allows us to implement a penalty, on the tracking error of the trajectory and control variable for reference along the predictive horizon relative to the final state of the system. This provides a term that guarantees the proper evolution of the reference position and finally the component that penalizes due to the obstacle avoidance. The term V_i^{coll} indeed indicates that part of the figure of merit used to ensure that avoidance of collisions between subsystems occurs. The variables $\hat{x}$ and $\hat{u}$ formed by equivalent sets related to the real system, i.e.

$$\hat{x}^{[i]}_k \in \hat{X}_i \quad \hat{u}^{[i]}_k \in \hat{U}_i \quad \forall k = t \ldots t+N-1$$

Impose terminal constraints, on the basis of the auxiliary control law of stabilization

$$\overline{x}^{[i]}_{t+N}\left(\overline{y}^{[i]}_{t+N}\right) = A\tilde{x}^{[i]}_{t+N-1} + B\tilde{u}^{[i]}_{t+N-1} + G^x\left(\overline{y}^{[i]}_{t+N} - C\tilde{x}^{[i]}_{t+N-1}\right)$$

The set of obstacles defined online, possibility by choosing for the set of i^{th} agent is

$$O = \left\{obst: \quad obst \in C\tilde{x}^{[j]}_k \quad k = t \ldots t+N-1 \;\; j \neq i\right\}$$

C. Cost Function

The Index $[i]$ is omitted for the sake of simplicity in later sections. The figure of cost function is defined as

$$J = \sum_{k=t}^{t+N-1} \left\|\hat{x}_t - \tilde{x}_t\right\|^2_Q + \left\|\hat{u}_t - \tilde{u}_t\right\|^2_R + \left\|\hat{x}_{t+N} - \tilde{x}_{t+N}\right\|^2_P \quad (20)$$

Authorized licensed use limited to: Chalmers University of Technology Sweden. Downloaded on December 14,2023 at 10:13:34 UTC from IEEE Xplore. Restrictions apply.

We can rewrite the dynamic equations along the prediction horizon in the case of the general MPC

$$\hat{x}_t - \tilde{x}_t = I\left(\hat{x}_t - \tilde{x}_t\right) \tag{21}$$

And consequently in matrix form

$$\hat{X}_t - \tilde{X}_t = \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} \hat{x}_t - \tilde{x}_t \\ \hat{U}_t - \tilde{U}_t \end{bmatrix} = \begin{bmatrix} A & B \end{bmatrix} \xi_t \tag{22}$$

Cost is rewritten by using Eq.21 & 22, in the matrix form

$$J = \left\| \hat{X}_t - \tilde{X}_t \right\|_Q^2 + \left\| \hat{U}_t - \tilde{U}_t \right\|_R^2 + \left\| \begin{bmatrix} I \\ I \end{bmatrix} \bar{y}_{t+N} - \begin{bmatrix} \tilde{y}_{t+N-1} \\ \bar{y}_{setpont} \end{bmatrix} \right\|_{\begin{bmatrix} \gamma I & 0 \\ 0 & T \end{bmatrix}}^2 \tag{23}$$

Defining the vector

$$\xi_t = \begin{pmatrix} \hat{x}_t - \tilde{x}_t \\ \hat{U}_t - \tilde{U}_t \\ \bar{y}_{t+N} \end{pmatrix}$$

The matrices of weight $Q = diag\left(Q \ldots Q, P\right)$ in which Q is a diagonal matrix block with $N+1$ and $R = diag(R \ldots R)$ in which the R matrix block is with N. The figure of merit in relation to the ξ_t (ignoring the penalty due to obstacles) is given by

$$J = \left\| \begin{bmatrix} A & B \end{bmatrix} \xi_t \right\|_Q^2 + \left\| \begin{bmatrix} 0 & I \end{bmatrix} \xi_t \right\|_R^2 = \left\| \mathbb{A} \xi_t \right\|_Q^2 \tag{24}$$

Also note that the vector ξ_t is partly known for input and state references. In particular we have

$$\xi_t = \hat{\xi}_t - \tilde{\xi}_t$$

Where $\tilde{\xi}_t$ is completely defined at instant t, while $\hat{\xi}_t$ is the vector of optimization. ξ_t is completely known at each instant of time. The minimization problem in the absence of the component of collision avoidance becomes

$$\min_{\hat{\xi}_t} \left\| \mathbb{A}\left(\hat{\xi}_t - \tilde{\xi}_t\right) \right\|_Q^2 = \hat{\xi}_t^T \mathbb{A}^T \mathbb{Q} \mathbb{A} \hat{\xi}_t^T - 2 \tilde{\xi}_t^T \mathbb{A}^T \mathbb{Q} \mathbb{A} \hat{\xi}_t^T + \tilde{\xi}_t^T \mathbb{A}^T \mathbb{Q} \mathbb{A} \tilde{\xi}_t^T \tag{25}$$

This quadratic form resolved in the presence of constraints.

D. *Constraints Formulization*

The constraints to be imposed in optimization process must be expressed with respect to the vector $\hat{\xi}_t$ of the linear quadratic problem. It is necessary to impose linear constraints that are characterized by a system of linear inequalities set in the form

$$\mathrm{X} = \left\{ x : Hx \le L \right\} \quad \varepsilon = \left\{ x : H_\varepsilon x \le L_\varepsilon \right\} \tag{26}$$

Therefore the constraint to impose is

$$H_\varepsilon \left(x_t - \hat{x}_t \right) \le L_\varepsilon \Rightarrow H_\varepsilon x_t - H_\varepsilon \begin{bmatrix} I & 0 \end{bmatrix} \hat{\xi}_t \le L_\varepsilon \tag{27}$$

This constraint varies at each sampling moment as a function of the measured value of the real state x_k. Keeping the matrix H_ε constant and varying the term L_ε, the constraint obtained is

$$-H_\varepsilon \begin{bmatrix} I & 0 \end{bmatrix} \hat{\xi}_t \le L_\varepsilon - H_\varepsilon x_t \Rightarrow H_1 \hat{\xi}_t \le L_{1,t} \tag{28}$$

By choosing the polyhedral set of X, for the real state, we obtain a set for nominal status in the form

$$\hat{\mathbb{X}} = \left\{ x : H_{\hat{\mathrm{X}}} x \le L_{\hat{\mathrm{X}}} \right\}$$

The vector of optimization with respect to $\hat{\xi}_t$ is

$$\bar{H}_{\hat{\mathrm{X}}} \left(\bar{A} \begin{bmatrix} I & 0 \end{bmatrix} + \bar{B} \begin{bmatrix} 0 & I \end{bmatrix} \right) \hat{\xi}_t \le \bar{L}_{\hat{\mathrm{X}}} \Rightarrow H_2 \hat{\xi}_t \le L_2 \tag{29}$$

Choosing a set $\mathbb{U}$ polyhedral, $\hat{\mathbb{U}}$ can be expressed as

$$\hat{\mathbb{U}} = \left\{ x : H_{\hat{\mathrm{U}}} x \le L_{\hat{\mathrm{U}}} \right\}$$

That expressed with respect to the vector of optimization becomes

$$\bar{H}_{\hat{\mathrm{U}}} \begin{bmatrix} 0 & I \end{bmatrix} \hat{\xi}_t \le L_{\hat{\mathrm{U}}} \Rightarrow H_3 \hat{\xi}_t \le L_3 \tag{30}$$

Once again, this is imposed on the nominal state throughout the horizon predictive. By choosing a set, ΔZ can be expressed in the form

$$\Delta^z = \left(y : H_z \le L_z \right)$$

Constraint expressed in the function of the vector of optimization, this becomes

$$\bar{H}_z \begin{bmatrix} A & B \end{bmatrix} \hat{\xi}_t \le L_z + \bar{H}_z \begin{bmatrix} A & B \end{bmatrix} \tilde{\xi}_t \Rightarrow H_{45} \hat{\xi} \le L_{45,t} \tag{31}$$

At this point merge all the constraints in a single matrix by

$$\mathrm{H} \hat{\xi}_t \le \mathrm{L}_k \tag{32}$$

IV. EXPERIMENTAL SETUP & PARAMETER CHOICE

In the experimental tests the E-puck robot is used, which was developed at the "Swiss Federal Institute of Technology Lausanne" with a simple design and nonlinear model. The algorithms related to image processing and control will be implemented on a single computer, which communicates with the agents through a Bluetooth channel. Matlab software is implemented, for managing the camera and the wireless communication to each robot. The experimental setup is shown in Figure 4.

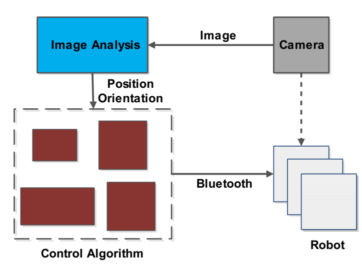

Figure 4. Experimental setup

The parameters of choice for the robust control law to be defined by the gain K of the Eq. 18 and the cost matrices Q, R and P for the state and the input, related to the optimization problem. As a result of various experimental tests weights Q and R are chosen, respectively, on the state and the input, equal to $Q = I_4 \quad R = 20 I_2$.

Authorized licensed use limited to: Chalmers University of Technology Sweden. Downloaded on December 14,2023 at 10:13:34 UTC from IEEE Xplore. Restrictions apply.

Through the Matlab function $[\mathrm{K}, \mathrm{P}] = \mathrm{dlqr}(\mathrm{A}, \mathrm{B}, \mathrm{Q}, \mathrm{R})$ in disturbance $w(k) \in W$ eexperimental tests showed measurement errors always less than 1cm and is therefore chosen to use a set of polyhedral type

$$\mathbb{W} = \left\{(x_1, x_2) : -1 \le x_1 \le 1, -1 \le x_2 \le 1\right\}$$

Considering the constraints

$$\mathbb{X} = \begin{cases} x_{\min,1} \le x_1 \le x_{\max,1} \\ x_{\min,3} \le x_3 \le x_{\max,3} \\ v_{\min} \le x_2, x_4 \le v_{\max} \end{cases}$$

$$\mathbb{U} = \left\{(a_x, a_y) : a_{\min} \le a_x, a_y \le a_{\max}\right\}$$

$$\Delta^z = \left\{(y_1, y_2) : -1 \le \mathrm{y}_1 \le 1, -1 \le \mathrm{y}_2 \le 1\right\}$$

V. Simulation Study

The implementation of the entire control algorithm is summarized and illustrated in Figure 5.

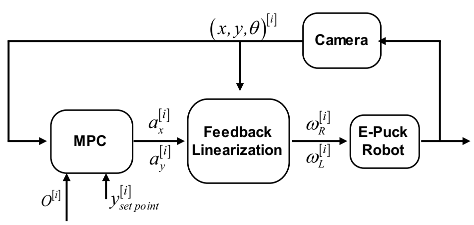

Figure 5. Control algorithm implementation

In the controller developed [14] that has the ability to maintain the robot away from obstacles and avoid the possible collision between robots. Finally a technique has built capable of solving the problem of coordination of multiple robots extending through the use of distributed implementation robust MPC. The optimization algorithm simulates by efficient Solver in TOMLAB package which provides the results in lesser computational times with the ability to easily specify cost function. By considerations made so far, the final implementation was chosen to use for testing, and to be applied to each of the robots

$$\min_{\hat{\xi}_t} \hat{\xi}_t^T \mathbb{A}^T \mathbb{Q} \mathbb{A} \hat{\xi}_t - 2\left(\mathbb{A}\tilde{\xi}_t\right)^T Q \hat{\xi}_t + \left(\mathbb{A}\tilde{\xi}_t\right)^T \mathbb{Q} \mathbb{A} \tilde{\xi}_t + V_i^{coll} \quad (33)$$

Where

$$\hat{\xi}_t = \begin{bmatrix} \hat{x}_t \\ \hat{u}_{[t:t+N-1]} \\ \overline{y}_{t+N} \end{bmatrix} \quad \tilde{\xi}_t = \begin{bmatrix} \tilde{x}_t \\ \tilde{u}_{[t:t+N-1]} \\ 0 \end{bmatrix}$$

$$\mathbb{A} = \begin{bmatrix} A & B & 0 \\ 0 & I & 0 \\ 0 & 0 & \begin{bmatrix} I \\ I \end{bmatrix} \end{bmatrix} \quad \mathbb{Q} = \begin{bmatrix} Q & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & \begin{bmatrix} \gamma I & 0 \\ 0 & T \end{bmatrix} \end{bmatrix}$$

The penalizing figure of merit, applied to the obstacles in order to ensure the collision avoidance, and choice of the polynomials is as follow

$$\rho = \left\|D\hat{\xi}_t - Cx_{obst}\right\|^2$$

$$f(\rho) = a_o + a_1\rho + a_2\rho^2 + a_3\rho^3 + a_4\rho^4 + a_5\rho^5$$

$$\sum_{\rho=0}^{M} \begin{cases} f\left(\left\|D\hat{\xi}_t - Cx_{obst,p}\right\|^2\right) & \left\|D\hat{\xi}_t - Cx_{obst,p}\right\|^2 \le d_{\lim} \\ 0 & \left\|D\hat{\xi}_t - Cx_{obst,p}\right\|^2 > d_{\lim} \end{cases} \quad (34)$$

All the M obstacles that the algorithm takes into account consists of all references $\tilde{y}_{t+N}$ calculated for the other robot interacting, i.e.

$$O^{[i]} = \left\{obst^{[i]} : \quad obst^{[i]} = \tilde{y}_{t+N-1}^{[j]} \quad \forall j \ne i\right\}$$

A. Tests positioning in simulation

The algorithm presented in this work is tested and verified for the approach. To evaluate the avoidance operation of the figure of merit require two robots are used to complete task i.e. follow trajectories that overlap. The ideal trajectory produced by the optimizer is presented in yellow and magenta and actual trajectory model of unicycle robot is presented in green and blue as shown in Figure 6. With regard to the management policy for the collision, we first choose to give priority to one of the two robots, forcing the other to escape the maneuver.

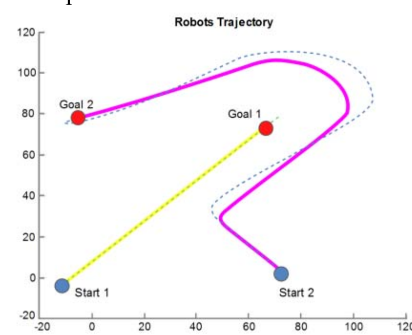

Figure 6. Coordination of two agents with possible collision

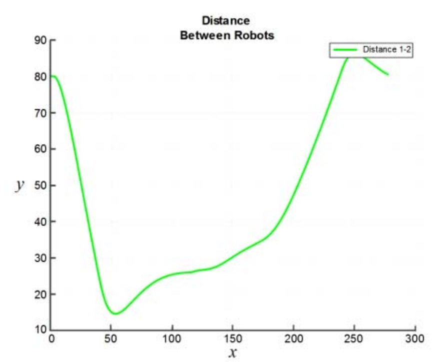

Figure 7. Distance between two robots

The trajectory of the distance between two robots is shown in Figure 7. We then consider a set of three robots interacting on the same workspace. Suppose we want to analyze the behavior of robots when each of the robots is given priority to start at the same time. The result of the simulation is shown in Figure 8. It is also interesting to observe the distance development between each of the robots, confirming the absence of collisions. The trajectories of the distance between three robots is shown in Figure 9.

Authorized licensed use limited to: Chalmers University of Technology Sweden. Downloaded on December 14,2023 at 10:13:34 UTC from IEEE Xplore. Restrictions apply.

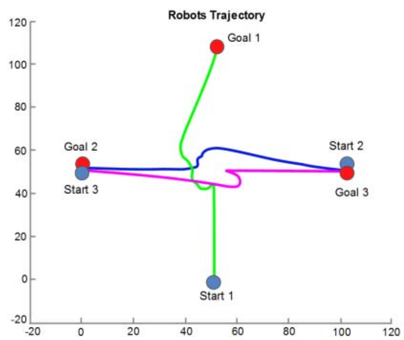

Figure 8. Coordination of three agents

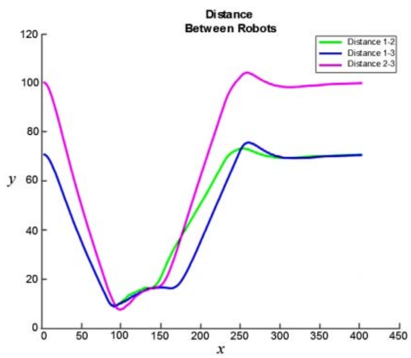

Figure 9. Distance between three robots

B. Position test in real

Based on what we have seen so far, we then tested the algorithm on real system; considering a control problem involving two agents. The results of this testing for the current positioning in the absence of collisions are illustrated in Figure 10. The pattern obtained confirmed the validity of the approach adopted with regard to the substantive component of the figure which has the responsibility to drive the robot toward the end goal.

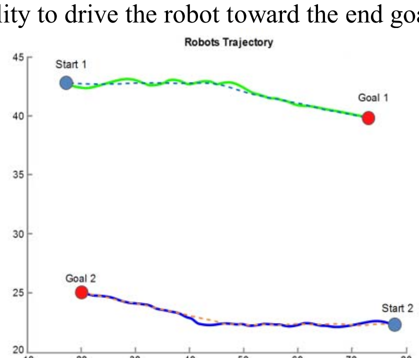

Figure 10. Real test for the coordination of two agents for the absense of collision

VI. CONCLUSION

We investigate a Robust MPC (tube based) control technique with distributed implementation to the problem of coordination of a set of mobile robots in planar movement. Using MPC to accomplish the task has to make available, at each sampling step, a prediction of the future horizon, calculated by the optimization algorithm that allows the anticipation and avoidance of collisions. The robots are in an environment characterized by obstacles, and the robots are able to communicate with each other by controllers in order to exchange information about their predicted trajectory. The proposed technique allows us to achieve the desired result to manage the placement of robots within the same environment and in the absence of collisions. Penalty function is used to solve the problem of coordination. The performance of the proposed approach is tested by simulation of the model of an available robot together with a ring of internal control for feedback linearization dynamics; such testing validated the results.

REFERENCES

[1] Scattolini, Riccardo. "Architectures for distributed and hierarchical model predictive control–a review." *Journal of Process Control,* vol.19, no. 5, pp. 723-731, 2009

[2] Mayne, J. Rawlings, C. Rao, P. Scokaert, "Constrained model predictive control: stability and optimality." *Automatica*, vol. 36, no. 6, pp. 789–814, 2000

[3] Camacho, E. F. & Alba, C. B. *Model predictive control*. Springer Science & Business Media, 2013.

[4] Bemporad, F. Borrelli, M. Morari, "Model predictive control based on linear programming-the explicit solution," *IEEE Transactions on Automatic Control*, vol. 47, no 12, pp. 1974–1985, 2002.

[5] Bemporad, Alberto, and Manfred Morari. "Robust model predictive control: A survey." *Robustness in identification and control*. Springer London, pp. 207-226, 1999.

[6] P. O. Scokaert and D. Q. Mayne, "Min-max feedback model predictive control for constrained linear systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 8, pp. 1136-1142, 1998.

[7] Y. Kuwata, A. Richards, and J. How, "Robust receding horizon control using generalized constraint tightening," *American Control Conference*, *ACC'07, New York, USA*, Jul. 2007, pp. 4482-4487.

[8] Richards and J. How, "Robust stable model predictive control with constraint tightening," *American Control Conference, Minneapolis, USA*, Jun. 2006, pp. 1557-1562.

[9] L. Chisci, J. Rossiter, G. Zappa, "Systems with persistent disturbances: predictive control with restricted constraints", *Automatica,* vol. 37, no 7, pp. 1019–1028, 2001.

[10] W. Langson, I. Chryssochoos, S. Rakovic, D. Mayne, "Robust model predictive control using tubes," *Automatica,* vol. 40, no 1, pp. 125–133, 2004.

[11] Q. Mayne, E. C. Kerrigan and P. Falugi, "Robust model Predictive control: Advantages and disadvantages of tube-based methods," *in World Congress*, pp. 191-196, 2011.

[12] M. Cannon, B. Kouvaritakis, S. V. Rakovic and Q. Cheng, "Stochastic tubes in model predictive control with probabilistic constraints," *Automatic Control, IEEE Transactions*, vol. 56, pp. 194-200, 2011.

[13] Maja J. Matari`c. "From Local Interactions to Collective Intelligence". *The Biology and Technology of Intelligent Autonomous Agents. A cura di Luc Steels. NATO ASI Series. Springer Berlin Heidelberg,* Vol. 144 pp. 275–295, 1995.

[14] M. F. Manzoor, Q. Wu. "Control and obstacle avoidance of wheeled mobile robot," unpublished. (CICSyN2015)

[15] W. Dunbar, R. Murray, "Receding Horizon Control of Multi-vehicle Formations: A Distributed Implementation," *Proceedings of the 43rd IEEE Conference on Decision and Control, Atlantis, Bahamas,* vol. 2, pp. 1995–2002, 2004.

[16] R. Scsttolini, "Architectures for Distributed and Hierarchical Model Predictive Control–a Review," *Journal of Process Control*, vol. 19, no. 5, pp.723–731, 2009.

Authorized licensed use limited to: Chalmers University of Technology Sweden. Downloaded on December 14,2023 at 10:13:34 UTC from IEEE Xplore. Restrictions apply.