Proceedings of the 6th RSI
International Conference on Robotics and Mechatronics (IcRoM 2018)
October 23-25, 2018, Tehran, Iran

Laguerre based model predictive control for trajectory tracking of nonholonomic mobile robots

Massoud Hemmasian Ettefagh
Department of Mechanical Engineering
Amirkabir University of Technology
Tehran, Iran
Hemmasian@aut.ac.ir

Mahyar Naraghi
Department of Mechanical Engineering
Amirkabir University of Technology
Tehran, Iran
Naraghi@aut.ac.ir

Farzad Towhidkhah
Department of Biomedical Engineering
Amirkabir University of Technology
Tehran, Iran
Towhidkhah@aut.ac.ir

Hosein Izi
Department of Mechanical Engineering
Amirkabir University of Technology
Tehran, Iran
Hosein_Izi@aut.ac.ir

***Abstract*—This paper presents a Laguerre parametrization approach to employ Model Predictive Control (MPC) for the trajectory tracking problem of a non-holonomic mobile robot with input and state constraints. A time-varying error model is obtained for the trajectory tracking of the mobile robot. Then, a Laguerre based MPC (LMPC) for time-varying systems is designed and tuned to ensure asymptotic stability of the system. The proposed algorithm considers input and states, including velocity and acceleration, constraints to provide stability. It is shown that the proposed method is able to reduce the computation times. In order to confirm the effectiveness of the proposed method, extensive simulations results are provided.**

***Index Terms*—Mobile robot, Laguerre, Model predictive control, Trajectory tracking**

I. Introduction

Thanks to the multi-tasking ability and the capability to work in diverse conditions, mobile robots are increasingly in demands of surveillance, transportation, industry, household, agriculture [1] and even space [2] projects.

In addition to nonholonomic dynamics, mobile robots are subject to input and state constraints, i.e. physical limitations in actuator and workspace. These constraints render the trajectory tracking of mobile robots as one of the active research directions. Several controllers were offered for the mobile robots, such as sliding mode approach in [3] that achieves asymptotic stability in tracking errors in position and heading direction with employing two controllers. In [4], the authors used an adaptive-backstepping scheme to deal with the unknown parameters of the mobile robot. In a similar manner, [5] employed adaptive fuzzy logic controller to estimate system's uncertainties as well as to establish Lyapunov asymptotical stability. The authors of [6] used high-order dynamic model for the situation when the absolute velocity of the robot is not measurable. Despite the ability to estimate absolute velocity and other uncertainties, all of the aforementioned methods lack the ability to cope with unequal constraints including input and state constraints [7].

Due to the intrinsic ability of Model Predictive Control (MPC) to consider and satisfy unequal constraints, this method has obtained vast attention from the control community. Gregor et al. [7] used MPC for the tracking problem of a deferentially drive mobile robot. In order to guarantee stability and convergence, [8] employed state contractive technique in model predictive control. Gonzalez et al. [9] designed a robust tube-based MPC for a mobile robot to ensure robust stability in the presence of uncertainties such as slip. Later, in [10], [11], the same authors experimentally validate robust tube-based MPC for mobile robots where the uncertainties of the robust design were modelled as additive disturbance. Bi et al. [12] presented an MPC approach based on recurrent neural networks that was able to work in real time experiments. A disturbance rejection MPC algorithm is suggested in [13] to deal with unknown disturbances and uncertain parameters while the constraints are satisfied.

Laguerre based MPC (LMPC) belongs to the larger family of orthonormal parametrization of input vector for model predictive control [14]. Originally developed by L. Wang in [14]–[16], the method proved its capability to reduce total number of decision variables for Linear Time Invariant (LTI) systems. This reduction eventually leads to the reduction of computation times [16]. In [17], the authors expanded the framework of Laguerre parametrization to MPC for Linear Time-Varying (LTV) systems subject to input and state constraints. It is shown in [18] that under mild conditions, stability and convergence of the Laguerre parametrization method is guaranteed for constrained LTV systems which enables the method to be applied to a large family of systems.

In this paper, a Laguerre based MPC is used for the trajectory tracking problem of a nonholonomic mobile robot subjects to input and state constraints. Performance and stability of the designed controller is compared to the ones of a traditional MPC and a conventional controller.

The remainder of the paper is organized as follows. Nonlinear model of a mobile robot and its time-varying linearized

978-1-7281-0127-9/18/$31.00 ©2018 IEEE

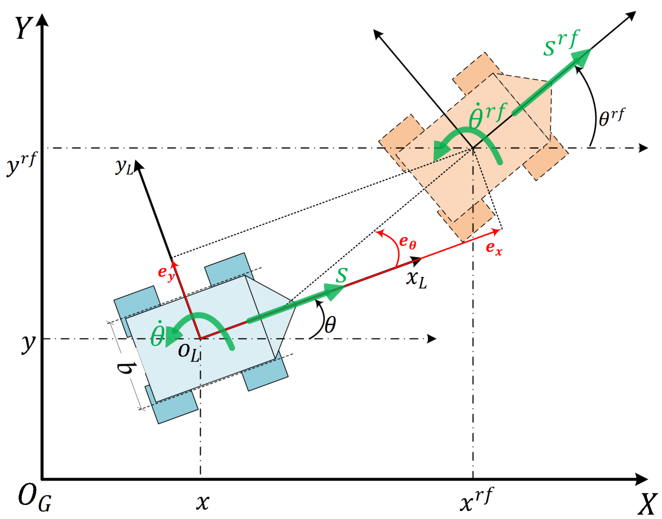

Fig. 1. robot coordinate reference and reference robot

model around a reference trajectory are discussed in section II. A brief outline of Laguerre based MPC for LTV systems is given in section III. Section IV is devoted to design the LMPC controller and evaluate its characteristics via simulations. Conclusion and future works are given in section V.

II. Modeling of a mobile robot

A. modeling

In this section, a skid-steer mobile robot is modelled to be used for the trajectory tracking problem. For mobile robots with linear speed less than 2 $[m/s]$, the dynamical effects are negligible and can be neglected [19]. In order to derive the governing equations of motions, it is supposed in Fig.1 that the robot is following a reference or a virtual mobile robot that represents the desired positions and velocities. The objective of the designed controller is to converge the error between the actual and the desired position and orientation to zero.

Slip occurs between the wheels of the mobile robots and the ground surface. Therefore, it is defined as the difference between the theoretical velocity of the center of the wheel in pure rolling and the actual forward velocity, or

$$i = 1 - \frac{s}{r\phi}, \tag{1}$$

wherein $0 \leq i \leq 1$, r, s and φ denote slip, wheel radius, actual forward velocity and the wheel angular velocity, respectively.

Assuming $s \leq 2$, the lateral slip is negligible [19]. Hence, according to Fig.1 the relation between the robot's velocity and its left and right wheels' velocity are

$$\dot{x}(t) = \frac{s_R(t) + s_L(t)}{2}\cos(\theta(t)) \tag{2}$$

$$\dot{y}(t) = \frac{s_R(t) + s_L(t)}{2}\sin(\theta(t)) \tag{3}$$

$$\dot{\theta}(t) = \frac{s_R(t) - s_L(t)}{b} \tag{4}$$

in which $[x \quad y \quad \theta]^T$ is the position vector and $b \in \mathbb{R}$ is the distance between mobile robot's right and left wheels.

Considering slip effect (1) for the left and right wheels as i_l and i_r, the relation between angular velocity of the left and right wheels with the velocities of the mobile robot is governed by

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\cos(\theta) & 0\\ \sin(\theta) & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}\frac{1-i_L}{2} & \frac{1-i_R}{2}\\ -\frac{1-i_L}{b} & \frac{1-i_R}{b}\end{bmatrix}\begin{bmatrix}r\varphi_L\\ r\varphi_R\end{bmatrix} \tag{5}$$

According to Fig.1, it is considered that the mobile robot with the governing dynamics (5) is following a reference mobile robot with position $\begin{bmatrix}x^{rf} & y^{rf} & \theta^{rf}\end{bmatrix}^T$ and similar dynamics. Therefore, the error vector from the reference robot in the coordinate system O_L attached to the actual robot is:

$$\begin{bmatrix}e_x\\ e_y\\ e_\theta\end{bmatrix} = \begin{bmatrix}\cos(\theta) & \sin(\theta) & 0\\ -sin(\theta) & \cos(\theta) & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}x^{rf} - x\\ y^{rf} - y\\ \theta^{rf} - \theta\end{bmatrix} \tag{6}$$

Differentiating (6) with respect to time and doing some algebraic manipulation, one can easily obtain the nonlinear model of the error dynamics as [19]

$$\dot{e}_x = \dot{\theta}e_y + \cos(e_\theta)s^{rf} - s \tag{7}$$

$$\dot{e}_y = -\dot{\theta}e_x + \sin(e_\theta)s^{rf} \tag{8}$$

$$\dot{e}_\theta = \dot{\theta}^{rf} - \dot{\theta} \tag{9}$$

in which s^{rf} and θ^{rf} are linear and angular velocities of the reference mobile. These expressions are defined explicitly with time, i.e. $s^{rf} = s^{rf}(t)$ and $\theta^{rf} = \theta^{rf}(t)$, and a high-level trajectory planning algorithm provides them for the trajectory tracking controller.

B. Linearization and Discretization of equation

In order to apply Laguerre based MPC on the mobile robot, Expressions (7)–(9) must be linearized around the equilibrium point ($\dot{e}_x = \dot{e}_y = \dot{e}_\theta = 0$). Omitting high order terms and with the abuse of notation, (7)–(9) simplifies to the linearized equations

$$\begin{bmatrix}\dot{e}_x\\ \dot{e}_y\\ \dot{e}_\theta\end{bmatrix} = \begin{bmatrix}0 & \theta^{rf}(t) & 0\\ -\theta^{rf}(t) & 0 & s^{rf}(t)\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}e_x\\ e_y\\ e_\theta\end{bmatrix} + \begin{bmatrix}0 & 1\\ 0 & 0\\ -1 & 0\end{bmatrix}\begin{bmatrix}\dot{\theta}^{FB}\\ s^{FB}\end{bmatrix}, \tag{10}$$

where $\dot{\theta}^{FB}$ and s^{FB} are defined as $\dot{\theta}^{FB} = \dot{\theta}^{rf} - \dot{\theta}$ and $s^{FB} = s^{rf}$, respectively.

Denoting φ_L^{FB} and φ_R^{FB} as the input for the left and right wheel in (1) and (5), the error dynamics of the mobile robot is

$$\begin{bmatrix}\dot{e}_x\\ \dot{e}_y\\ \dot{e}_\theta\end{bmatrix} = \begin{bmatrix}0 & \theta^{rf}(t) & 0\\ -\theta^{rf}(t) & 0 & s^{rf}(t)\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}e_x\\ e_y\\ e_\theta\end{bmatrix} + \begin{bmatrix}\frac{1-i_L}{2} & \frac{1-i_R}{2}\\ 0 & 0\\ \frac{1-i_L}{b} & -\frac{1-i_R}{b}\end{bmatrix}\begin{bmatrix}r\varphi_L^{FB}\\ r\varphi_R^{FB}\end{bmatrix} \tag{11}$$

As discussed in [18], the LMPC method works with discrete time systems. Discretization of (11) with sampling time T_s leads to

$$\begin{bmatrix} e_x(k+1) \\ e_y(k+1) \\ e_\theta(k+1) \end{bmatrix} = \begin{bmatrix} 1 & T_s\theta^{rf}(k) & 0 \\ -T_s\theta^{rf}(k) & 1 & T_s s^{rf}(k) \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} e_x(k) \\ e_y(k) \\ e_\theta(k) \end{bmatrix} + \begin{bmatrix} T_s\frac{1-i_L}{2} & T_s\frac{1-i_R}{2} \\ 0 & 0 \\ T_s\frac{1-i_L}{b} & -T_s\frac{1-i_R}{b} \end{bmatrix} \begin{bmatrix} r\varphi_L^{FB}(k) \\ r\varphi_R^{FB}(k) \end{bmatrix} \quad (12)$$

which has the standard form of

$$e(k+1) = A(k)e(k) + B(k)u(k) \quad (13)$$

and can be used in the LMPC framework [17], [18].

III. Laguerre based MPC for time-varying systems

In MPC, a sequence of control actions

$$\mathbf{U} = \begin{bmatrix} u_0^T & u_1^T & \cdots & u_{N_p-1}^T \end{bmatrix}^T \quad (14)$$

over the prediction horizon N_p are computed based on the information from the current states, given model, prediction of future states and future control actions over a finite interval of time. Then the first element of the control sequence, i.e. u_0, is given to the system. By obtaining a new measurement at the next sample time, the predicted states and control actions update continuously. This consists the principle of receding horizon in MPC.

The total number of decision variables in traditional MPC is equal to mN_p where m is the dimension of the input matrix B in (13). Laguerre based MPC captures the control action sequence by smaller number of decision variables, which results in a reduction in computation times [18]. This method uses transformation $\mathbf{U} = \Gamma\eta$ where Γ and η are the Laguerre network matrix and the new decision variable vector, respectively, and their values are given in the appendix.

Using (29)–(33) in (13), one can obtain

$$e(k+1) = A(k)e(k) + B(k)\gamma(k)\eta \quad (15)$$

for the system's dynamics in which $\gamma(k)$ is the appropriate row of Γ in (29) of Appendix associated with time k. Expanding (15) for the prediction horizon N_p leads to

$$X = S_X e(0) + S_U\Gamma\eta \quad (16)$$

in which X, S_X and S_U are defined as

$$X^T = \begin{bmatrix} e(0)^T & e(1)^T & e(2)^T & \ldots & e(N_p)^T \end{bmatrix}^T \quad (17)$$

$$S_X = \begin{bmatrix} I & \Phi(1,0)^T & \Phi(2,0)^T & \ldots & \Phi(N_p,0)^T \end{bmatrix}^T \quad (18)$$

$$S_U(N_p;k_0) = \begin{bmatrix} 0 & 0 & 0 & \ldots \\ \Phi(1,1)B(0) & 0 & 0 & \ldots \\ \Phi(2,1)B(0) & \Phi(2,2)B(1) & 0 & \ldots \\ \vdots & \vdots & \vdots & \ddots \\ \Phi(N_p,1)B(0) & \Phi(N_p,2)B(1) & \ldots & \Phi(N_p,N_p)B(N_p-1) \end{bmatrix} \quad (19)$$

$$\Phi(k,k_0) = A(k)A(k-1)\ldots A(k_0) \quad (20)$$
$$\Phi(k_0,k_0) = I$$

The Laguerre based MPC employs prediction model (16) in the following constrained optimization problem

$$J(e_0) = \|e(N_p)\|_P^2 + \sum_{k=0}^{N_p-1} \|e(k)\|_{Q(k)}^2 + \|u(k)\|_{R(k)}^2 \quad (21a)$$

$$J^*(e_0) = \min_{\mathbf{U}} J(e_0) \quad (21b)$$

$$\mathbf{U}^* = \arg\min_{\mathbf{U}} J(e_0) \quad (21c)$$

$$\text{subject to :}$$

$$e(k+1) = A(k)e(k) + B(k)u(k), \quad (21d)$$

$$\underline{u(k)} \le u(k) \le \overline{u(k)}, \quad k = 0,1,\cdots,N_p-1, \quad (21e)$$

$$\underline{e(k)} \le e(k) \le \overline{e(k)}, \quad k = 0,1,\cdots,N_p-1 \quad (21f)$$

where $\|e\|_Q^2 = e^TQe$ and matrices $Q(k), R(k)$ and P are symmetric positive definite. Using (14), (16)–(20), the optimization problem (21) simplifies to the standard quadratic programming problem [18]

$$J(e_0) = X^T\overline{Q}X + \eta^T\Gamma^T\overline{R}\Gamma\eta \quad (22a)$$

$$J^*(e_0) = \min_{\eta} J(e_0) \quad (22b)$$

$$\eta^* = \arg\min_{\eta} J(e_0) \quad (22c)$$

$$\text{subject to :}$$

$$X = S_X e(0) + S_U\Gamma\eta, \quad (22d)$$

$$G\eta \le g \quad (22e)$$

for appropriate matrices $\overline{Q}, \overline{R}, \overline{G}$ and g. The decision variable vector η is the solution of the optimization problem (22). Then, the input vector $\mathbf{U}$ and its first element u_0 are determined by employing $\mathbf{U} = \Gamma\eta$. Stability and convergence of the solution of (22) for system (13) is illustrated in [18].

IV. Design of MPC controller for mobile robot and simulation

A. Controller Design

The overall Control structure for the trajectory tracking problem of mobile robot involves three layers and is shown in Fig. 2. The first layer generates the reference trajectory by considering the user's command, data from a given map and current position of the robot. The Second layer is responsible for maintaining the robot on the desired trajectory. The third layer consists of low level controllers for regulation of motor speeds. The second layer consists of two parts: a feedforward part to follow the desired trajectory in advanced and a feedback part to eliminate any mismatch between the desired position and the actual position. Our LMPC algorithm locates in the feedback part. It is assumed in this paper that the desired trajectory is always available for the second layer. The dynamics of the low-level controllers in the third layer is fast and negligible with respect to the rest of the system.

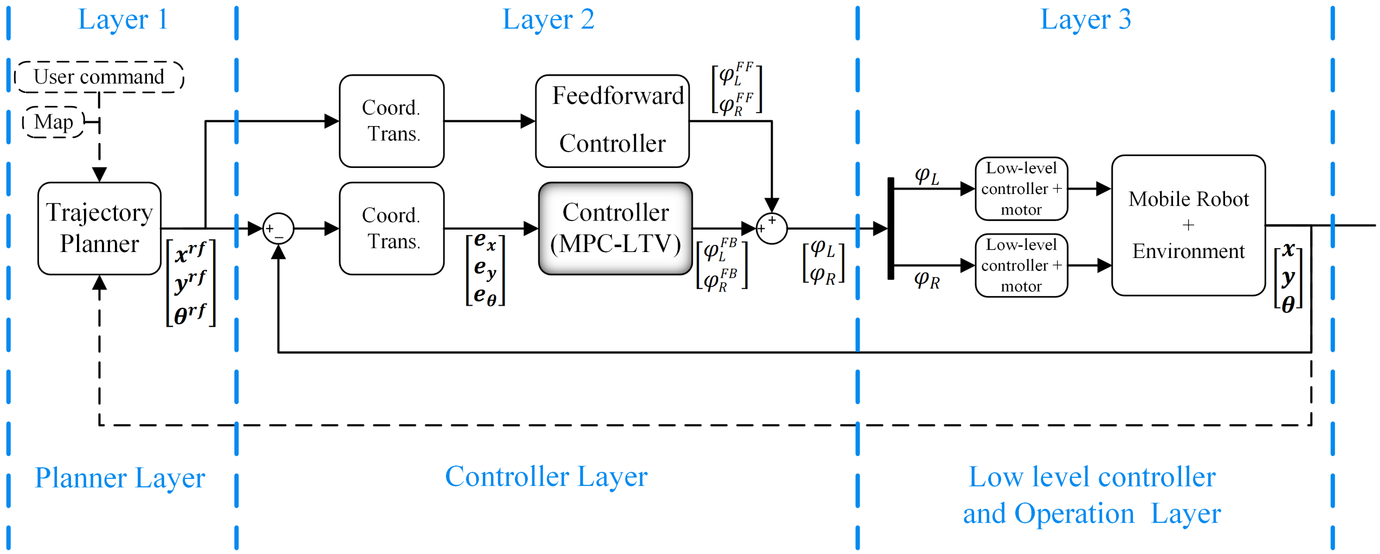

Fig. 2. Block diagram of control. Block diagram consist of three layers: 1- planner layer, 2- controller layer, and 3- low level controller and operation layer.

According to Fig. 2, the applied angular velocity for the each motor consists of a feedforward part $\varphi^{FF}(k)$ and a feedback part $\varphi^{FB}(k)$. Therefore, for each motor we have

$$\begin{cases} \varphi_L(k) = \varphi_L^{FF}(k) + \varphi_L^{FB}(k) \\ \varphi_R(k) = \varphi_R^{FF}(k) + \varphi_R^{FB}(k) \end{cases} \tag{23}$$

The feedforward part of (23) is determined by the first layer of the control architecture and its value at each time is out of the reach of the second layer to modify. Similarly, the angular velocities of the motors are restricted between a lower and an upper bounds, i.e. $\underline{\varphi} \leq \varphi_L, \varphi_R \leq \overline{\varphi}$. Thus, the input constraint for the LMPC problem (21) is time-varying and can be expressed as

$$\begin{cases} \underline{\varphi} - \varphi_L^{FF}(k) \leq \varphi_L^{FB}(k) \leq \overline{\varphi} - \varphi_L^{FF}(k) \\ \underline{\varphi} - \varphi_R^{FF}(k) \leq \varphi_R^{FB}(k) \leq \overline{\varphi} - \varphi_L^{FF}(k) \end{cases} \tag{24}$$

where $\overline{\varphi} = -\underline{\varphi} = 15\ [rad/s]$. State constraints are imposed to assure the performance of the mobile robot. The imposed value for state constraints are

$$\begin{cases} -0.3[m] \leq e_x, e_y \leq 0.3[m] \\ -20[deg] \leq e_\theta \leq 20[deg] \end{cases} \tag{25}$$

The rest of the parameters are: $T_s = 0.1[s], b = 0.4[m], r = 0.11[m]$. The controller's design parameters are tuned to be

$$Q = P = \begin{bmatrix} 750 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 0.25 \end{bmatrix}, \quad R = \begin{bmatrix} 0.25 & 0 \\ 0 & 0.25 \end{bmatrix} \tag{26}$$

$$N_p = 25, \quad N = 3, \quad a = 0.9 \tag{27}$$

B. Simulations

Performance of the proposed algorithms is studied here via comparison with the traditional MPC described in [7] and a Discrete time Linear Quadratic Regulator (DLQR) controller with the same weighing matrices.

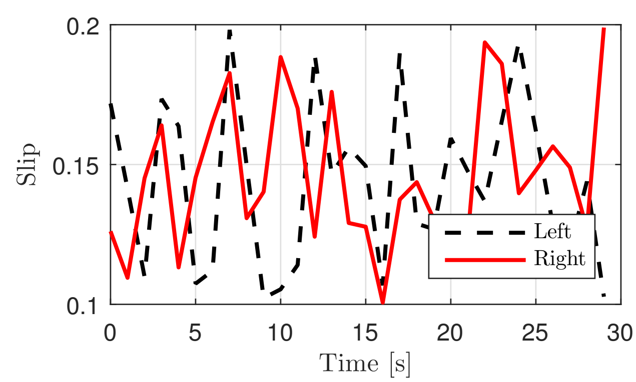

Fig. 3. Slip value for wheels of the mobile robot

Even though a number of different trajectories have been tested, we confine the results to an ∞-shape and an S-shape trajectories. To make the simulations more realistic we add unknown slip to both wheels. The slip value for each wheel is shown in Fig. 3 where the values are bounded to be $0.1 \leq i_L, i_R \leq 0.2$. As it can be seen in Figs. 4 and 5, the initial location and orientation of the mobile robot is different from the reference one.

The reference trajectory for the ∞-shape is defined by

$$\begin{cases} x_r(k) = 1.1 + 3\sin(2\pi k/30) \\ y_r(k) = 0.9 + 3\sin(4\pi k/30) \end{cases} \tag{28}$$

and it can be seen in Fig. 4 as the green-dashed line. The initial error is $e(0) = [-0.1 \quad -0.2 \quad -0.2]^T$. The results for different control algorithms are also presented in Fig. 4. Due to the effect of the unknown slip all methods show some deviation from the reference trajectory, especially at the beginning of it. However, it is observed that the error of the traditional MPC and the Laguerre based MPC are less than

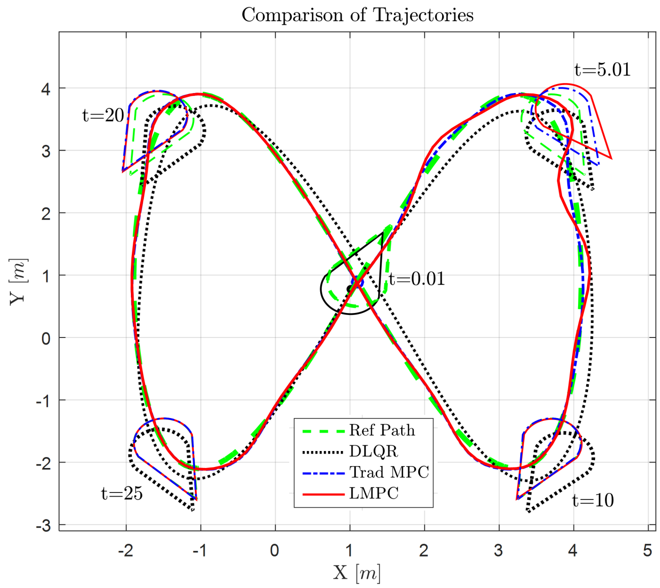

Fig. 4. Trajectory tracking of ∞-shape path for different control algorithms: Reference path $(--)$, DLQR method $(\ldots)$, Traditional MPC $(-\cdot-)$, Laguerre based MPC $(-)$.

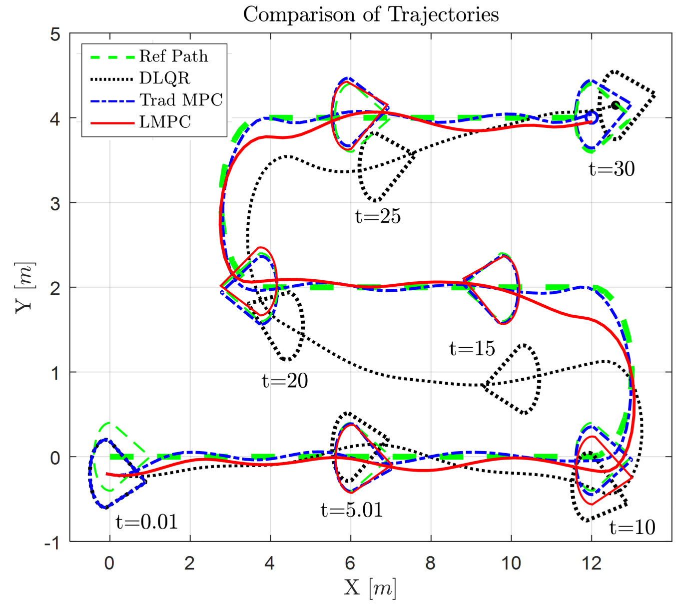

Fig. 5. Trajectory tracking of ∞-shape path for different control algorithms: Reference path $(--)$, DLQR method $(\ldots)$, Traditional MPC $(-\cdot-)$, Laguerre based MPC $(-)$.

the error of the DLQR method. Traditional MPC provides a better result at the cost of demanding more decision variables and heavier computations. The decision variable vector for the Traditional MPC has 50 elements ($N_p = 25, m = 2$), while this vector has only 6 elements in the LMPC method. Consequently, as pointed out in [18], the reduction in the total number of decision variables results in a decrease of the required computation times.

Fig. 5 is devoted to the S-shape trajectory. The reference trajectory is borrowed from trajectories used in agricultural industry [10]. The initial error and slip values are similar to the ones of previous example. As in the previous simulation, it is observed that while the DLQR method poorly tracks the desired trajectory, both traditional and Laguerre based model predictive control methods are able to keep the robot close to the reference trajectory. Fig. 6 shows the error values of each method with respect to time. It can be seen that the tangential, normal and angular values of error in the traditional MPC is less than the ones of LMPC. The superiority of the Traditional MPC over the Laguerre based MPC is due to the fact that the number of decision variables in traditional MPC is much higher than the one of Laguerre based MPC. However, the Laguerre MPC has decreased the total number of decision variables and the computations. This enables the method to be applied to the systems with fast dynamics.

V. Conclusion and Future works

This paper presents the Laguerre based model predictive control algorithm for trajectory tracking of a constrained time-varying system with unknown parameters. The designed control scheme has been implemented on a mobile robot subject to input and state constraints as well as slip conditions. Performance of the proposed algorithm has been evaluated using simulations for two given trajectories. The results revealed that the Laguerre based MPC is able to produce satisfactory control input with a small number of decision variables. This can contribute to the reduction of online computation times in model predictive control of time-varying systems.

The successful result of the proposed algorithm motivates further works to implement the method on a nonholonomic mobile robot subject to slip conditions, sensor noise and other environmental uncertainties.

Appendix

Laguerre based MPC uses the transformation $u = \Gamma\eta$. The values of Γ is given here. Further details on this subject can be found in [17].

$$\eta^T = \begin{bmatrix} {\eta^{(1)}}^T & {\eta^{(2)}}^T & \ldots & {\eta^{(m)}}^T \end{bmatrix}, \tag{29}$$

$$\Gamma^T = \begin{bmatrix} L(0)^T & L(1)^T & \ldots & L(N_p-1)^T \end{bmatrix}, \tag{30}$$

$$L(k) = \begin{bmatrix} L^{(1)}(k) & 0 & \ldots & 0 \\ 0 & L^{(2)}(k) & \ldots & 0 \\ \vdots & 0 & \ddots & \vdots \\ 0 & \ldots & \ldots & L^{(m)}(k) \end{bmatrix}. \tag{31}$$

where $L(k)$ is defined as

$$L(k) = \begin{bmatrix} l_1(k) \\ l_2(k) \\ \vdots \\ l_N(k) \end{bmatrix} = \sqrt{b}\,\Psi^{k-1} \begin{bmatrix} 1 \\ -a \\ a^2 \\ -a^3 \\ \vdots \\ (-1)^{N-1}a^{N-1} \end{bmatrix}, \tag{32}$$

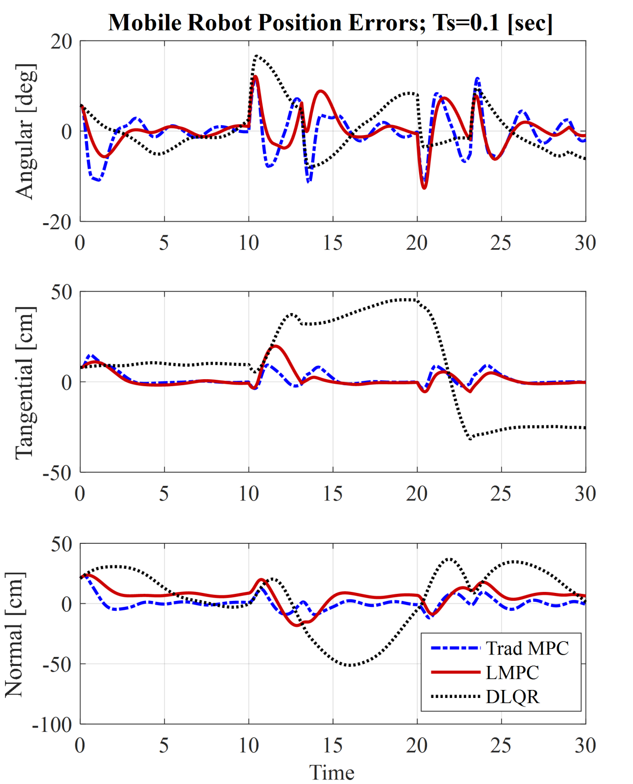

Fig. 6. Error value of S-shape path for for different control algorithms: DLQR method $(\ldots)$, Traditional MPC $(- \cdot -)$, Laguerre based MPC $(-)$.

in which $b = 1 - a^2$, and

$$\Psi = \begin{bmatrix} a & 0 & 0 & \ldots & 0 \\ b & a & 0 & \ldots & 0 \\ -ab & b & a & \ldots & 0 \\ a^2 b & -ab & b & \ddots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ (-1)^{N-2} a^{N-2} b & (-1)^{N-3} a^{N-3} b & \ldots & b & a \end{bmatrix}. \tag{33}$$

References

[1] M. M. G. Plessen and A. Bemporad, "Reference trajectory planning under constraints and path tracking using linear time-varying model predictive control for agricultural machines," *Biosystems Engineering*, vol. 153, pp. 28–41, 2017.

[2] M. Ono, T. J. Fuchs, A. Steffy, M. Maimone, and J. Yen, "Risk-aware planetary rover operation: Autonomous terrain classification and path planning," in *IEEE Aerospace Conference*. IEEE, 2015, pp. 1–10.

[3] D. Chwa, "Sliding-mode tracking control of nonholonomic wheeled mobile robots in polar coordinates," *IEEE transactions on control systems technology*, vol. 12, no. 4, pp. 637–644, 2004.

[4] T. Fukao, H. Nakagawa, and N. Adachi, "Adaptive tracking control of a nonholonomic mobile robot," *IEEE transactions on Robotics and Automation*, vol. 16, no. 5, pp. 609–615, 2000.

[5] T. Das and I. N. Kar, "Design and implementation of an adaptive fuzzy logic-based controller for wheeled mobile robots," *IEEE Transactions on Control Systems Technology*, vol. 14, no. 3, pp. 501–510, 2006.

[6] D. Wu, Y. Cheng, H. Du, W. Zhu, and M. Zhu, "Finite-time output feedback tracking control for a nonholonomic wheeled mobile robot," *Aerospace Science and Technology*, vol. 78, pp. 574–579, 2018.

[7] G. Klan čar and I. Škrjanc, "Tracking-error model-based predictive control for mobile robots in real time," *Robotics and Autonomous Systems*, vol. 55, no. 1, 2007.

[8] F. Xie and R. Fierro, "First-state contractive model predictive control of nonholonomic mobile robots," in *American Control Conference*. IEEE, 2008, pp. 3494–3499.

[9] R. Gonzalez, M. Fiacchini, J. L. Guzmán, and T. Alamo, "Robust tube-based mpc for constrained mobile robots under slip conditions," in *Proceedings of the 48th IEEE Conference on Decision and Control*. IEEE, 2009, pp. 5985–5990.

[10] R. González, M. Fiacchini, J. L. Guzmán, T. Álamo, and F. Rodríguez, "Online robust tube-based mpc for time-varying systems: a practical approach," *International Journal of Control*, vol. 84, no. 6, pp. 1157–1170, 2011.

[11] R. González, M. Fiacchini, J. L. Guzmán, T. Álamo, and F. Rodríguez, "Robust tube-based predictive control for mobile robots in off-road conditions," *Robotics and Autonomous Systems*, vol. 59, no. 10, pp. 711–726, 2011.

[12] S. Bi, G. Zhang, X. Xue, and Z. Yan, "Real-time robust model predictive control of mobile robots based on recurrent neural networks," in *International Conference on Neural Information Processing*. Springer, 2015, pp. 289–296.

[13] Z. Sun, Y. Xia, L. Dai, K. Liu, and D. Ma, "Disturbance rejection mpc for tracking of wheeled mobile robot," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 6, pp. 2576–2587, 2017.

[14] L. Wang, "Continuous time model predictive control design using orthonormal functions," *International Journal of Control*, vol. 74, no. 16, pp. 1588–1600, 2001.

[15] ——, "Discrete model predictive controller design using laguerre functions," *Journal of process control*, vol. 14, no. 2, pp. 131–142, 2004.

[16] ——, *Model predictive control system design and implementation using MATLAB®*. Springer Science & Business Media, 2009.

[17] M. Hemmasian Ettefagh, M. Naraghi, F. Towhidkhah, and J. De Doná, "Model predictive control of linear time varying systems using laguerre functions," in *2016 Australian Control Conference (AuCC)*. IEEE, 2016, pp. 120–125.

[18] M. Hemmasian Ettefagh, M. Naraghi, J. De Dona, and F. Towhidkhah, "Orthonormal function parametrisation of model-predictive control for linear time-varying systems," *International Journal of Systems Science*, vol. 49, no. 4, pp. 867–882, 2018.

[19] R. González, F. Rodríguez, and J. L. Guzmán, *Autonomous Tracked Robots in Planar Off-Road Conditions*. Springer, 2014.