Proceedings of the 4th
International Conference on Robotics and Mechatronics
October 26-28, 2016, Tehran, Iran

Experimental Study on Optimal Motion Planning of Wheeled Mobile Robot Using Convex Optimization and Receding Horizon Concept

Mojtaba Zarei
Human and Robot Interaction Laboratory
Faculty of New Sciences and Technologies
University of Tehran, Tehran, Iran
Email: mojtaba.zarei@ut.ac.ir

Roya Sabbagh Novin
Department of Mechanical Engineering, IEEE student member
University of Utah, Salt Lake City, UT, USA
Email: roya.sabbaghnovin@utah.edu

Mehdi Tale Masouleh
School of Electrical and Computer Engineering
, Human and Robot Interaction Laboratory
University of Tehran, Tehran, Iran
Email: m.t.masouleh@ut.ac.ir

***Abstract*—In this paper, a novel algorithm for collision-free motion planning of two wheeled mobile robots is presented. The proposed approach stands on discrete motion planning, convex optimization and model-based control. It was employed for motion planning and control of E-puck mobile robot to pass through an unknown environment both in simulation software in MATLAB and real setting. In this regard, CVX package benefited by the Gurobi solver is employed to solve the optimization problem. Obtained results show the reliability of this algorithm for safe collision avoidance. The reported results reveal this fact that by considering the maximum velocity of the E-puck, obtained computational time is less than 0.2 seconds in each stage which is fast enough for robot motion planning tasks.**

I. Introduction

Mobile robots like flying, swimming, wheeled and legged robots, are gaining widespread applications in different aspect of the Human life. Given the nature of their applications, safe and collision free motion planning is of crucial importance. For this purpose, several algorithms have been developed so far. Path planning with obstacle avoidance in dynamic environments is one of the most challenging criteria in robotics for autonomous vehicles. The motion planning problem can be defined as a given drivable vehicle which travels from the starting point to a fix or moving target without obstacles collision while satisfying some specific constraints. These constraints can be kinematic and dynamic constraints. Also, some optimal criterion such as minimum traveling length, consuming time or energy can be defined as objective function for motion planning problems. The existing algorithms can be divided into the two main groups, global and local path planner [1]. Unlike the global path planners, the local path planners are mainly based on sensory data collected during operations which make them computationally efficient and hence suitable for real-time applications. Moreover, in the literatures, some researches are conducted for the static environments [2, 3]. But, in the dynamical environments, when the vehicle encounters to dynamical constraints, any tracking capability is de facto limited. In order to solve the latter problem, recently some progresses have been made [4–6]. In this field, Receding Horizon Control (RHC) has been employed in motion planning in order to improve the planning performance [7–9].

The main contribution of this paper consist in a comprehensive algorithm for collision-free motion planning control of mobile robots. Due to innate capability of constrained RHC to generate feasible and optimal trajectories, it has been considered as well-known methodology for the obstacle avoidance motion planning problems. It's worth to mention that the proposed algorithm has been implemented in [10, 11] for parallel and serial manipulators, respectively. Predicated on this fact, in this paper, constrained RHC strategy in conjunction with convex optimization concept is employed to enhance the motion planning performance. In the first section, a brief introduction about mobile robot kinematics, motion planning and proposed structure of controller is presented. In the subsequent section, utilizing the obtained motion planning algorithm is simulated for two different scenarios with two separate cost functions. Moreover, after simulation, the proposed algorithm is implemented practically on E-puck mobile robot and the results are demonstrated.

II. Main components

In this section, fundamental components of the proposed system such as the governing equations on the kinematics of the mobile robot and constituted elements of the controller are discussed.

A. Kinematics equations

The E-puck mobile robot actuators are able to receive velocity commands. Also, precise model of system is one of

978-1-5090-3222-8/16/$31.00 ©2016 IEEE

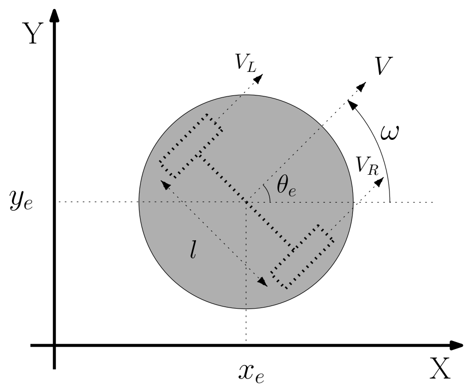


Fig. 1: Kinematics of E-puck mobile robot.

the prerequisite of RHC. As shown in Fig. 1, robots location in the main coordinate system can be expressed as bellow:

$$X = \begin{bmatrix} x_e & y_e & \theta_e \end{bmatrix}^T \tag{1}$$

where θ_e is the angle between robot's main axis (parallel with the wheels) and the horizontal axis. Time derivative of robot's location is:

$$\dot{X} = \begin{bmatrix} \dot{x}_e & \dot{y}_e & \dot{\theta}_e \end{bmatrix}^T \tag{2}$$

Let's consider linear and angular velocities as control inputs and robot's velocity along the X and Y axis as outputs. The control equation can be expressed as:

$$\begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} \frac{(v_r + v_l)\cos\theta_e}{2} \\ \frac{(v_r + v_l)\sin\theta_e}{2} \\ \frac{(v_r - v_l)}{l} \end{bmatrix} \tag{3}$$

where v_r and v_l are linear velocities of right and left wheels respectively, and l is the distance between them.

B. Path planning and control of a mobile robot

In order to further clarification of the motion planning optimization problem, the constituting elements like cost function and constraint related to obstacles and the robot are examined and the main problem is rewritten as a single convex optimization problem.

1) Cost function: For the cost function different functions can be considered which are determined based on the purpose of the application. The purpose of the application can be regarded as passing through the path as fast as possible, passing through the shortest path or using the least energy. The only limiting condition in determining the objective function is its convexity. It is also plausible to consider a combination of the aforementioned objectives as a multi-objective function in which weighted coefficients are needed for each criterion. These coefficients are determined according to the problem priority and applications.
In this research, the shortest time and feasible path as the objective functions are considered. Considering other combinations of these objectives, leads to another forms of objective functions. Since discrete programming is employed, the path length could be calculated by summing the distance between several points within the path. Put differently, for a mobile robot (which is considered as a single point), setting the control horizon as h, the cost function for the shortest path could be written as:

$$\min \quad \sum_{k=1}^{h} ||\mathbf{z}_e(k+1) - \mathbf{z}_e(k)||_2^2. \tag{4}$$

where $\mathbf{z}_e(k) \in \mathbb{R}^2$ is the location of mobile robot in the stage k. The time taken for completing the path is obtained by summing each stage's time as:

$$\min \quad \sum_{k=1}^{h} dt(k). \tag{5}$$

Moreover, since the concept of receding horizon is used here, an ancillary cost function is necessary in order to ensure moving toward the final point. For this purpose, the distance between the final point and the last point of the control horizon is considered as:

$$\min \quad ||\mathbf{z}_e(h) - \mathbf{z}_g||_2^2. \tag{6}$$

where $\mathbf{z}_g$ is the location of the goal point.

2) Obstacle avoidance: For the sake of keeping the convex form of the optimization problem, all of the obstacles, even those which are not convex, are made equivalent as convex forms fully surrounded them. Therefore, each obstacle is estimated by a polygon shape. This is a more conservative approach and provides safer results. Polygon shape is defined as the intersection of a series of half spaces:

$$\mathscr{O} = \{\xi | \mathbf{A}\xi < \mathbf{b}\}. \tag{7}$$

In the above equations, $\mathbf{A}$ is a matrix with $\mathbf{A}_k$ arrays in which $k \in \{1, ..., s\}$ and $\mathbf{b} = (b_1, ..., b_s)^T$ is the corresponding vector to the half-spaces where s is the number of dimensions. The point ξ is defined to be outside of the shape $\mathscr{O}$ if at least one of the $\mathbf{A}\xi < \mathbf{b}$ inequalities is satisfied. In other words:

$$\mathbf{A}\xi \geq \mathbf{b} + (\mathbf{v} - \mathbf{1})M, \tag{8}$$

$$\sum_{i=1}^{s} v_i \geq 1. \tag{9}$$

where $\mathbf{v} = (v_1, ..., v_s)^T$ is a vector of binary variables $v_i \in \{0, 1\}$ and M is a constant value used in the Big-M method [12]. Equation (8) ensures that at least one element of the vector $\mathbf{v}$ equals to 1 so the point ξ would be out of the polygon shape.

3) Maximum velocity: The only dynamic constraint considered in this paper is the velocity limit denoting by $v_{\max}$. Thus, one has:

$$|\frac{\mathbf{z}_e(k+1) - \mathbf{z}_e(k)}{\Delta t}| \leq v_{\max}, \quad k = 1, ..., (h-1), \tag{10}$$

where Δt is the period between two points which could be either constant (in the case of considering minimum length for cost function) or variable (in the case of considering minimum time for cost function). Considering the aforementioned explanations for each section, the problem of motion planning and control of a mobile robot could be formulated as following optimization problem:

$$
\begin{aligned}
\min \quad & w_1 \times T + w_2 \times \sum_{k=1}^{h} ||\mathbf{z}_e(k+1) - \mathbf{z}_e(k)||_2^2 && \text{(11a)}\\
& + w_3 \times ||\mathbf{z}_e(h) - \mathbf{z}_g||_2^2, \\
\text{s.t.} \quad & \mathbf{A}_{\mathscr{O}}(k)\mathbf{z}_e(k) \geq \mathbf{b}_{\mathscr{O}}(k) + (\mathbf{v}(k) - \mathbf{1})M, && \text{(11b)}\\
& \sum_{i=1}^{s} v_i(k) \geq 1, && \text{(11c)}\\
& \mathbf{z}_e(1) = \mathbf{z}_s, && \text{(11d)}\\
& \mathbf{z}_e(h) = \mathbf{z}_g, && \text{(11e)}\\
& |\frac{\mathbf{z}_e(k+1) - \mathbf{z}_e(k)}{\Delta t}| \leq v_{\max} \qquad \forall k \in \{1, ..., h\}. && \text{(11f)}
\end{aligned}
$$

As mentioned previously, in the aforementioned equations, $\mathbf{z}_e(k) \in \mathbb{R}^2$ and $\mathbf{z}_g$ stand for the location of the mobile robot in k^{th} stage and the final point, respectively. T denotes the period taken to go entirely through the control horizon:

$$T = \Delta t \times h,$$

where h is the length of the control horizon. It is worth to note that for this problem, a weighted combination of minimum time and length objectives are considered for formulating the cost function. In addition, Eqs. (11b) and (11c) ensure that the mobile robot will not collide with the obstacles ($\mathscr{O} \cap \mathscr{Z} = \emptyset$). Equations (11d) and (11e) determine the initial and final points, respectively, while Eq. (11f) limits the maximum velocity. In the next sections, this optimization problem is used for motion planning and control of a mobile robot and by determining a suitable algorithm, the obtained results are simulated and implemented for a two-wheel robot.

III. Implementation of the Proposed Algorithm and Simulation Results

In this section, first, a general algorithm is determined for motion planning of a mobile robot and then, the results are reported for two different cost functions.

A. *Path planning simulation algorithm for a two-wheel mobile robot*

The employed algorithm for this purpose is shown in Algorithm 1. In this algorithm, first, the required constants including the length of control horizon (h), the radius of the circle embracing the perimeter of the robot (r_e), the maximum allowed velocity ($v_{\max}$), number of the dimensions used in making the obstacles convex(s), the proper parameter for the Big-M method (M), desired accuracy in approaching the final point (ϵ) and the obstacle-related parameters ($x_{\mathscr{O}}, y_{\mathscr{O}}, r_{\mathscr{O}}$) are determined. After this, using the following equations, the required parameters for the half-spaces forming polygon shapes, which are employed in the optimization problem, are found.

Algorithm 1 Proposed Algorithm for motion planning of the mobile robot.

Initialize: h, $v_{\max}$, s, M, r_e, $r_{\mathscr{O}}$, $x_{\mathscr{O}}$, $y_{\mathscr{O}}$, ε;
% Find the circumscribing polygon parameters for obstacles
Eqs. (5.16-5.20);
while $dz_{\text{final}} \geq \varepsilon$ **do**
  % Solve the optimization problem by Gurobi
  $(\mathbf{z}_e, \mathbf{z}_f, dt, \mathbf{dx}_e, \mathbf{dy}_e) =$ `Path_Optimizer`$(h, \mathbf{z}_s, \mathbf{z}_g, v_{\max}, \mathbf{m}_{\mathscr{O}}, \mathbf{b}_{\mathscr{O}})$;
  if $\mathbf{dx}_e(2) \geq 0$ **then** *% Find the angel of movement*
    $\theta_e = \arctan(\frac{\mathbf{dy}_e(2)}{\mathbf{dx}_e(2)})$;
  else
    $\theta_e = \arctan(\frac{\mathbf{dy}_e(2)}{\mathbf{dx}_e(2)}) + \pi$;
  end if
  $v = \mathbf{dy}_e(2)/(\sin(\theta_e).dt)$; *% Update the linear velocity*
  $\omega = (\theta_e - \theta_{e0})/dt$; *% Update the angular velocity*
  % Update the position of the robot using kinematic model
  $x_e = x_e + v.\cos(\theta_e).dt$;
  $y_e = y_e + v.\sin(\theta_e).dt$;
  $\theta_{e_0} = \theta_{e_0} + \omega.dt$;
  $\mathbf{z}_s = [x_e, y_e]$;
  $dz_{\text{final}} = ||\mathbf{z}_s - \mathbf{z}_g||_2$; *% Find the distance to the goal point*
  $T_{\text{total}} = T_{\text{total}} + dt$; *% Update the total time*
end while
Go to the goal position and stop;

$$\phi(i) = \frac{2\pi i}{s}, \quad \forall i = 1, ..., s, \tag{12}$$

$$m_{\mathscr{O}}(i) = \tan(\frac{2\pi i}{s} + \frac{\pi}{2} - \frac{\pi}{s}), \quad \forall i = 1, ..., s, \tag{13}$$

$$x_{\text{aux}}(i) = r_{\mathscr{O}} \times \cos(\phi(i)) + x_{\mathscr{O}}, \quad \forall i = 1, ..., s, \tag{14}$$

$$y_{\text{aux}}(i) = r_{\mathscr{O}} \times \sin(\phi(i)) + y_{\mathscr{O}}, \quad \forall i = 1, ..., s, \tag{15}$$

$$b_{\mathscr{O}}(i) = y_{\text{aux}}(i) - m_{\mathscr{O}}(i) \times x_{\text{aux}}(i), \quad \forall i = 1, ..., s. \tag{16}$$

Once all the aforementioned parameters are determined, the optimization problem is solved at each stage and the corresponding outputs are applied to the robot. In this section, the second array of the obtained results is considered as the reference value for the control horizon. The motion angle and the control variables, which are actually the linear and angular velocity of the robot, respectively, could be determined by the following equations:

$$\theta_e = \arctan(\frac{\mathbf{dy}_e(2)}{\mathbf{dx}_e(2)}), \tag{17}$$

$$v = \mathbf{dy}_e(2)/(\sin(\theta_e).dt), \tag{18}$$

$$\omega = (\theta_e - \theta_{e0})/dt. \tag{19}$$

where dt is the time period, $\mathbf{dx}_e(2)$ and $\mathbf{dy}_e(2)$ are the position variations between the current and next stage locations, θ_e is the motion angle and finally v and ω are the linear and angular velocities, respectively.

Applying these control parameters to the kinematic model of the robot, described in the previous sections, the value of the state variable of the system are renewd which are actually the inputs of the optimization problem in the next stage. This process continues until the robot reaches close enough to the final point. At last, the control parameters for reaching the final point are calculated and the robot moves straight toward

TABLE I: The input values for initial and final points as well as the obstacle for implementation of motion planning for a two-wheel mobile robot considering static obstacle.

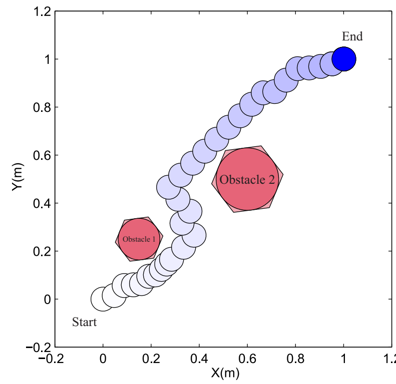
	x (m)	y (m)	R (m)
Starting point	0	0	
Final point	1	1	
Static obstacle 1	0.15	0.25	0.1
Static obstacle 2	0.6	0.5	0.15

it. In the next sections, the obtained results of the simulation and the implementation of this algorithm are reported.

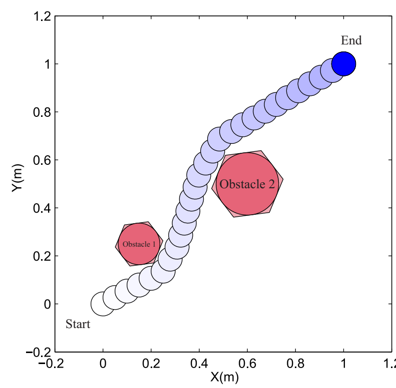
1) Results of the motion planning simulation of a two-wheel mobile robot with static obstacles using the minimum time as the cost function criterion: For all simulations performed in this section, a computer featuring as Core(TM)2 Duo CPU with 2.40 GHz Clock rate is used and since MATLAB is not able to employ multiple core simultaneously, all the problems are solved taking advantage of a single-core. Also, the optimization problem solves by CVX package in MATLAB benefited by Grubi optimization solver. Considering the input variables as shown in Table I, the obtained results from motion planning simulation are shown in Fig. (2a). In this case, the number of dimensions is $s = 6$ and the length of the control horizon is set as $h = 10$. The maximum allowed velocity is $v_{\max} = 0.05$m/s as well. In this case, total time taken for completing the path is 26.2 seconds and the traveled length is 1.68 meters. It is also worth to mention that the required time for solving the optimization problem is less than 0.1 seconds for each stage.

2) Results of the motion planning simulation of a two-wheel mobile robot with static obstacles using the minimum length as the cost function criterion: The optimization problem is solved considering inputs and constraint mentioned in the previous section for the cost function with minimum length criterion. The obtained results are shown in Fig. (2b). The obtained results show that the length and the time taken to complete the path are 1.49 meters and 28 seconds, respectively. The required time for solving the optimization problem is approximately 0.2 seconds, because the cost function is more complex in this case. However, this could be regarded as real time computation for the under study robot.

3) Results of the motion planning simulation of a two-wheel mobile robot with moving obstacles using the minimum time as the cost function criterion: For a moving obstacle, in each moment, the new location should be determined using the received signals and with regard to this new location, the optimum path should be specified. Since the methods used for identifying the obstacles are not in the scope of this research, it has been assumed in each moment, the new position of the obstacle is known. A circular motion is used for simulation which, as shown in Table II, is considered as a function of time and for each moment, by substituting in the corresponding function, the required values are obtained which resemble to receiving data from the sensors. The results for motion planning simulation of the two-wheel mobile robot are shown in Fig. (3a). In this case, the number of dimensions is $s = 6$



(a) The cost function considering minimum time. The length and time taken in this case are 1.68 meters and 26.2 seconds respectively.



(b) The cost function considering minimum length. The length and time taken in this case are 1.49 meters and 28 seconds respectively.

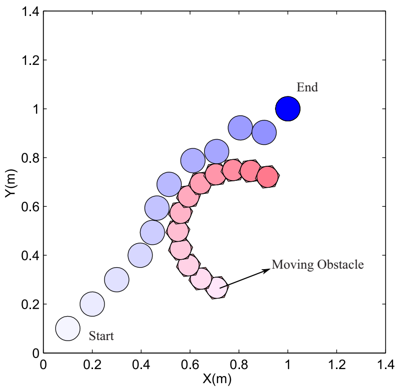
Fig. 2: The results of the motion planning simulation for a two-wheel mobile robot considering two static obstacles.

TABLE II: The input values for initial and final points as well as the obstacle for simulating a two-wheel mobile robot with a moving obstacle.

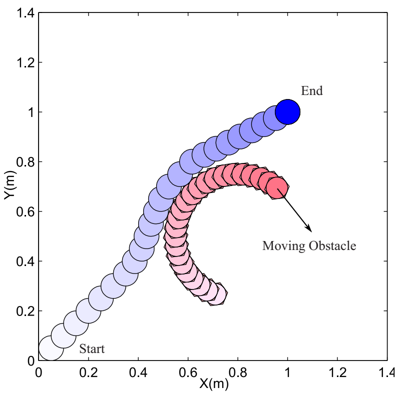
	Starting point	Final point	Moving obstacle
x (m)	0	1	$-0.25 \times \cos(0.15 \times t - 2) + 0.7$
y (m)	0	1	$0.25 \times \sin(0.15 \times t - 2) + 0.4$
R (m)	-	-	0.05

and the length of the control horizon is set as $h = 7$. The maximum allowed velocity is $v_{\max} = 0.05$m/s as well. Total time taken for completing the path is 23.6 seconds and the traveled length is 1.50 meters. Moreover, the required time for solving the optimization problem is less than 0.06 seconds for each stage.

4) Results of the motion planning simulation of a two-wheel mobile robot with moving obstacles using the minimum length as the cost function criterion: Following the same reasoning as the previous sections, the optimization problem is solved for a cost function considering minimum length and applying the same inputs and constraints as before. The results are shown in Fig. (3b). Total time taken for completing the path is 24

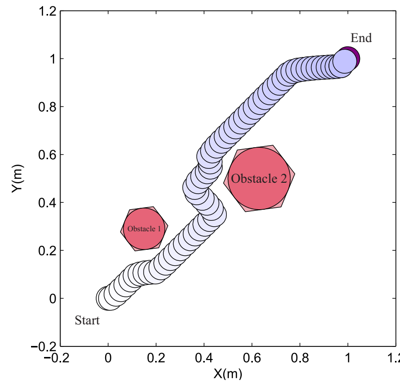


(a) The cost function considering minimum time. The length and time taken in this case are 1.50 meters and 23.6 seconds, respectively.

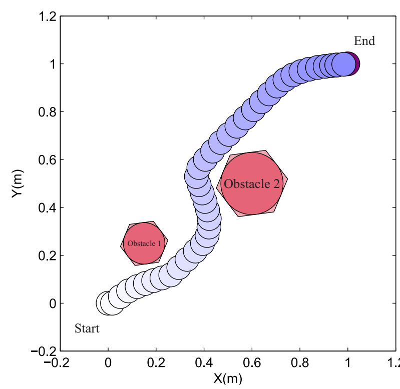


(b) The cost function considering minimum length. The length and time taken in this case are 1.46 meters and 24 seconds, respectively.

Fig. 3: The results of the motion planning simulation for a two-wheel mobile robot considering moving obstacles.



(a) The cost function considering minimum time. The length and time taken in this case are 1.72 meters and 64 seconds, respectively.



(b) The cost function considering minimum length. The length and time taken in this case are 1.64 meters and 41 seconds, respectively.

Fig. 4: The results of the practical motion planning implementation for E-puck mobile robot considering two static obstacles
.

seconds and the traveled length is 1.46 meters. Moreover, the required time for solving the optimization problem is approximately 0.08 seconds.

B. Implementing motion planning algorithm and proposed controller on E-puck

In this section, due to the favorable results obtained from simulation, the proposed algorithm is implemented in the E-puck mobile robot and the results are reported. Also, some frames of the implementation video is presented in Fig. 6.

1) Path planning algorithm implementation for E-puck robot considering two static obstacles: The results for cost function considering minimum time and length objectives are illustrated in Fig. (4a) and (4b), respectively. As it can be observed from the later results, the implementation results with the simulation ones. It should be noted that the constant parameters such as number of dimensions, time periods between points and the maximum allowed velocity have significant impacts on the traveled path and the time required for solving the equations. Furthermore, the required time for solving the problem will extend drastically by increasing the control horizon. Considering values from Table I as inputs, length of control horizon as $h = 10$ and the maximum allowed velocity as $v_{\max} = 0.05$m/s, in case of considering minimum time as cost function criterion, total time taken for traveling through the path is 64 seconds and the path length is 1.72 meters. While considering minimum length as the criterion for cost function, by assuming the time period between points as $dt = 1$, the path length and time are 1.64 meters and 41 seconds, respectively. The maximum time required for solving the problem in each stage is 0.2 seconds.

2) Path planning algorithm implementation for E-puck robot considering moving obstacles: In this case, because of some limitations of E-puck mobile robot, implementation for minimum time cost function was not feasible and the implementation process could be conducted only in the minimum length cost function case. The obtained results are shown in Fig. 5. In this case, in which the inputs are from Table III, the length of the control horizon is set as $h = 10$ and the maximum allowed velocity is $v_{\max} = 0.05$m/s, the length and

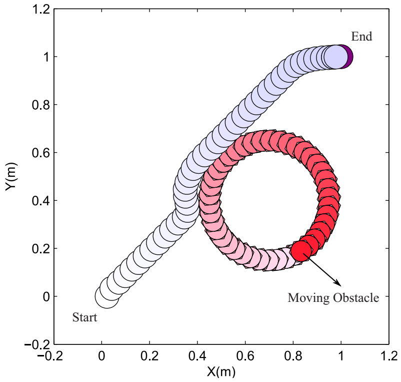


Fig. 5: The results of the practical motion planning implementation for E-puck mobile robot considering a moving obstacle with minimum length cost function. The length and time taken in this case are 1.56 meters and 41 seconds, respectively.

TABLE III: The input values for initial and final points as well as the obstacle for implementation of motion planning for a two-wheel mobile robot considering a moving obstacle.

	Starting point	Final point	Moving obstacle
x (m)	0	1	$-0.25 \times \cos(0.15 \times t - 2) + 0.7$
y (m)	0	1	$0.25 \times \sin(0.15 \times t - 2) + 0.4$
R (m)	-	-	0.05

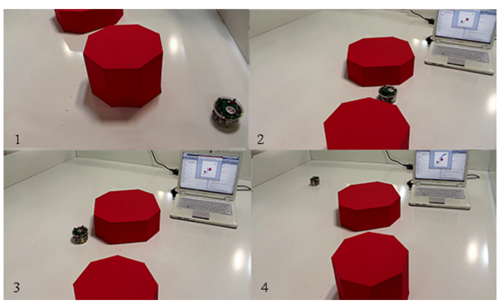


Fig. 6: Practical implementation of the proposed algorithm on E-puck mobile robot.

time are 1.56 meters and 41 seconds, respectively.

IV. Conclusion

In this paper, motion planning and control of the mobile robot using convex optimization and discrete planning structure and receding horizon concept were discussed. First, a brief introduction of mobile robot kinematics equations was presented. After that, the motion planning of the mobile robot as mathematical expression and Pseudocode algorithm were presented. Using the proposed algorithm and CVX package and Grubi optimization solver in MATLAB for the simulation of a two-wheeled mobile robot and obtaining the desired results, the algorithm was implemented on real robots. It is worth noting that in this paper, the two cost functions (the shortest time and shortest length of the path) for both fixed and moving obstacles were used. The results of this paper demonstrated that the solution time is less than 0.2 seconds at each stage. By considering the maximum velocity of this category of the mobile robots, it means the proposed algorithm is quite suitable for real-time applications in motion planning. Also, all safety conditions and no-obstacle collision was satisfied with the results.

References

[1] J.-C. Latombe, *Robot motion planning*. Springer Science & Business Media, 2012, vol. 124.

[2] M. L. Steven and L. Valle, "Planning algorithms," 2006.

[3] F. Fahimi, *Autonomous robots: modeling, path planning, and control*. Springer Science & Business Media, 2008, vol. 107.

[4] G. Chesi, "Visual servoing path planning via homogeneous forms and lmi optimizations," *Robotics, IEEE Transactions on*, vol. 25, no. 2, pp. 281–291, 2009.

[5] A. V. Savkin and C. Wang, "Seeking a path through the crowd: Robot navigation in unknown dynamic environments with moving obstacles based on an integrated environment representation," *Robotics and Autonomous Systems*, vol. 62, no. 10, pp. 1568–1580, 2014.

[6] A. Richards and J. P. How, "Aircraft trajectory planning with collision avoidance using mixed integer linear programming," in *American Control Conference, 2002. Proceedings of the 2002*, vol. 3. IEEE, 2002, pp. 1936–1941.

[7] C.-H. Hsieh and J.-S. Liu, "Nonlinear model predictive control for wheeled mobile robot in dynamic environment," in *Advanced Intelligent Mechatronics (AIM), 2012 IEEE/ASME International Conference on*. IEEE, 2012, pp. 363–368.

[8] T. M. Howard, C. J. Green, and A. Kelly, "Receding horizon model-predictive control for mobile robot navigation of intricate paths," in *Field and Service Robotics*. Springer, 2010, pp. 69–78.

[9] G. Franzè and W. Lucia, "An obstacle avoidance model predictive control scheme for mobile robots subject to nonholonomic constraints: A sum-of-squares approach," *Journal of the Franklin Institute*, vol. 352, no. 6, pp. 2358–2380, 2015.

[10] R. S. Novin, A. Karimi, M. Yazdani, and M. Tale Masouleh, "Optimal motion planning for parallel robots via convex optimization and receding horizon," *Advanced Robotics*, pp. 1–19, 2016.

[11] R. S. Novin, M. T. Masouleh, and M. Yazdani, "Optimal motion planning of redundant planar serial robots using a synergy-based approach of convex optimization, disjunctive programming and receding horizon," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, p. 0959651815617883, 2015.

[12] H. Ding, G. Reissig, D. Gross, and O. Stursberg, "Mixed-integer programming for optimal path planning of robotic manipulators," in *Automation Science and Engineering (CASE), 2011 IEEE Conference on*. IEEE, 2011, pp. 133–138.