Journal of Optimization in Industrial Engineering 22 (2017) 93-101
DOI: 10.22094/joie.2017.280

Direct Optimal Motion Planning for Omni-directional Mobile Robots under Limitation on Velocity and Acceleration

Naser Azim Mosheni[a], Ahmad Fakharian[b,*]

[a] *MSc, Department of Electrical, Biomedical, and Mechatronics Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran*

[b] *Associate Professor, Department of Electrical, Biomedical, and Mechatronics Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran*

Received 28 July 2016; Revised 26 December 2016; Accepted 13 January 2017

Abstract

This paper describes a low computational direct approach for optimal motion planning and obstacle avoidance of Omni-directional mobile robots within velocity and acceleration constraints on the robot motion. The main purpose of this problem is the minimization of a quadratic cost function while limitations on velocity and acceleration of robot are considered, and collision with any obstacle in the robot workspace is avoided. This problem can be formulated as a constrained nonlinear optimal control problem. To solve this problem, a direct method is utilized which employs polynomials functions for parameterization of trajectories. By this transforming, the main optimal control problem can be rewritten as a nonlinear programming problem (NLP) with lower complexity. To solve the resulted NLP and obtain optimal trajectories, a new approach is used with very small run time. Finally, the performance and effectiveness of the proposed method are tested in simulations and some performance indexes are computed for better assessment. Furthermore, a comparison between the proposed method and another direct method is done to verify the low computational cost and better performance of the proposed method.

Keywords: Direct trajectory planning, Obstacle avoidance, Motion constraints, Omni-directional mobile robots.

1. Introduction

Omni-directional mobile robots are becoming increasingly popular in mobile robot applications since they have some distinguishing advantages in comparison with nonholonomic mobile robots. The ability to move in any direction, irrespective of the orientation of the vehicle, makes it an attractive robot. The small-sized league of the annual RobuCup competition is an example where Omni-directional mobile robots are employed. Motion planning is one of the most important issues in these competitions. Indeed, the robot must be able to move in its workspace while avoiding obstacles to reach desired destination position in an optimal trajectory.

In a common form, the motion planning problem can be formulated as an optimal control problem (OCP). Since the resulted optimal control problem is a complex and nonlinear problem, analytic solutions are hard or even non-existent. Thus, numerical methods are needed for the solution of these problems. The numerical solutions of nonlinear OCPs are generally classified as indirect and direct methods (Rao, 2009; Betts, 1998; Von Stryk and principle or vibrational principle (Bryson, 1975; Hartl, 1995). Then, the problem is converted to a two-point boundary value problem which would be solved numerically. The main disadvantage of the indirect methods is that these methods cannot handle discontinuities in the problem which arise from high nonlinearities in the constraints of the problem.

Thus, direct methods come to replace indirect ones to solve more complex optimal control problems. The basic principle of the direct methods is parameterization of the OCP and solving it using appropriate nonlinear programming (NLP) techniques (Rao, 2009). The direct methods can be employed by parameterization of the state variables (Jaddu, 1998; Jaddu, 2002; Jaddu & Vlach, 2002), control variables (Goh & Teo, 1988) or both the state and control variables (Vlassenbroeck and Van Dooren, 1988; Frick and Stech, 1993).

Direct shooting methods (Kirches and Wirsching (2012); Assassa and Marquardt (2014)) and direct collocation methods (Kameswaran and Biegler (2008); Garg et al., (2011); Tohidi & Nik (2015); Von Stryk (1993)) are two basic methods of direct methods which operate by parameterization of control variable and both of state and control variables, respectively.

* Corresponding author Email address: ahmad.fakharian@qiau.ac.ir

In recent studies on trajectory planning, the following papers can be consulted. Duan et al. (2016) presented a trajectory planning method for a glass-handing robot based on execution time, acceleration and jerk. Richter et al. (2016) proposed an algorithm to generate trajectories for a differential flat quadrotor model in cluttered indoor environments. Li et al. (2015) obtained time-optimal trajectory for tractor-trailer vehicles. Lou et al. (2015) worked on trajectory planning of industrial robotic manipulators based on energy minimization.

This paper presents an optimal motion planning and obstacle avoidance for Omni-directional mobile robots under robot motion constraints. To obtain an optimal trajectory, a quadratic cost function is considered to be minimized. First, the optimal motion planning problem is rewritten as an optimal control problem. Since the resulted OCP has nonlinear constraints, the indirect method results in a complex problem with high computational problem. Thus, the direct method is utilized for solving the nonlinear constraints of optimal control problem. The employed direct method operates using state parameterization by polynomial functions.

Mohseni and Fakharian (2015) presented a solution method for solving the resulted NLP. In this method, the NLP is solved in pre-defined points in the whole time interval. Then, the main solution is obtained by collecting these individual solutions.

In this paper, a technique is employed in which the solution of NLP is obtained in one step. This method decreases the simulation time of solving the motion planning problem sharply.

The paper is organized as follows. The model of the Omni-directional mobile robot is presented is section 2. In section 3, necessity and importance of motion planning in real environments are stated. Section 4 describes the main aims of optimal motion planning problem; by some assumption, the final optimal control formulation is written. In section 5, the proposed direct method is presented. First, the optimal control problem is converted to a nonlinear programming problem, and then the new solution method for solving the resulted NLP is proposed. The proposed method is tested under simulation in section 6 and is compared with another direct trajectory planning method in section 7. Finally, the conclusion of the paper is presented in section 8.

2. The Omni-directional Mobile Robot Model

In this section, the model of Omni-directional mobile robot is presented. In trajectory and motion planning problems, often the dynamics of the robot are ignored and only translations and rotations required to move are considered (LaValle, 2006).

The motion equations of a four-wheeled Omni-directional mobile robot by Purwin and D'Andrea (2005) is presented as follows:

$$\ddot{x}(t) = q_x(t) \tag{1}$$

$$\ddot{y}(t) = q_y(t) \tag{2}$$

where $q_x(t)$ and $q_y(t)$ are the control inputs to the robot in x and y directions, respectively. First, the state variables are considered as follows:

$$x_1(t) = x, \quad x_2(t) = y, \quad x_3(t) = \dot{x}, \quad x_4(t) = \dot{y} \tag{3}$$

where x, y, $\dot{x}$, and $\dot{y}$ are the position in x direction, position in y direction, velocity in x direction, and velocity in y direction, respectively.

Then, the control inputs are denoted as follows:

$$U = [u_1 \quad u_2]^T, \quad u_1(t) \triangleq q_x(t), \quad u_2(t) \triangleq q_y(t) \tag{4}$$

Finally, state space model of the robot can be written as follows:

$$\dot{X}(t) = AX(t) + Bu(t) \tag{5}$$

where

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6}$$

3. Applications of the Motion Planning for Mobile Robots in Real Environments

Mobile robots have various applications in different fields. One of the important issue for these automatic systems is finding appropriate motions to move the robot in desired trajectories. Motion planning is a term used in robotics for the process of breaking down the desired movement task into discrete motions that satisfy movement constraints and possibly optimize some aspect of the movement. Motion planning problem would take a description of these tasks as inputs and produce the speed and turning commands sent to the robot's wheels.

For instance, consider navigating a rescue robot inside a building. It should perform its tasks while avoiding collision with walls and other objects.

Another example is a mobile robot which works in a storehouse. It should be able to move from the first point to destination point in an appropriate path and avoid the collision with other racks and products.

By the increase and evolution of robotic systems, Robotic studies have become a popular field in scientific research studies.

RoboCup is a competition domain designed to advance robotics and AI research through a friendly competition. The small-sized league of the annual RobuCup competition is an example where Omni-directional mobile robots are employed. A small-sized robot soccer game takes places between two teams of six robots each.

The robot should be able to move in an optimal trajectory while avoiding the collision from other soccer player robots. Also, there is a limitation on maximum velocity

and acceleration of these robots which should be satisfied in motion planning problem.
One image of these competitions is shown in Fig. 1.

Fig. 1. Small-sized Robocop competitions

4. Problem Statement, Formulation, and Assumption

In this section, the main purpose of motion planning problem is described and formulated as a nonlinear optimal control problem.

- A quadratic cost function be minimized.

$$J = \frac{1}{2}\int_{t_0}^{t_f} (\mathrm{x}(t)^T Qx(t) + u(t)^T Ru(t))\,\mathrm{dt} \tag{7}$$

- Collision with obstacles be avoided.

$$\bigcup_{i=1}^{k}\left[S_i(\mathrm{X(t),t}) \geq 0\right] \tag{8}$$

- The maximum velocity and acceleration on robot be satisfied.

$$\sqrt{x_3^2(\mathrm{t}) + x_4^2(\mathrm{t})} \leq v_{\max}, \tag{9}$$

$$\sqrt{u_1^2(\mathrm{t}) + u_2^2(\mathrm{t})} \leq a_{\max} \tag{10}$$

- The robot must move from initial point to goal point.

$$\begin{aligned} &x_1(t_0) = x_{1_0},\ x_2(t_0) = x_{2_0}, \\ &x_3(t_0) = x_{3_0},\ x_4(t_0) = x_{4_0}, \end{aligned} \tag{11}$$

$$\begin{aligned} &x_1(t_f) = x_{1_f},\ x_2(t_f) = x_{2_f}, \\ &x_3(t_f) = x_{3_f},\ x_4(t_f) = x_{4_f}, \end{aligned} \tag{12}$$

- The model of the Omni-directional robot be considered.

$$\begin{cases} \dot{x}_1(\mathrm{t}) = x_3(\mathrm{t}) = v_x(\mathrm{t}) \\ \dot{x}_2(\mathrm{t}) = x_4(\mathrm{t}) = v_y(\mathrm{t}) \\ \dot{x}_3(\mathrm{t}) = u_1(\mathrm{t}) = a_x(\mathrm{t}) \\ \dot{x}_4(\mathrm{t}) = u_2(\mathrm{t}) = a_y(\mathrm{t}) \end{cases} \tag{13}$$

In which Q is a symmetric positive semi-definite matrix, and R is a symmetric positive definite matrix; x_{j_0} and x_{j_f}, $(j = 1,...,4)$ are the initial and final positions and velocities of robot, respectively; $S_i(\mathrm{X(t),t})$ represents the time-varying boundaries of the static obstacles, $v_{\max}$ and $a_{\max}$ are the maximum velocity and acceleration of robot, respectively.

Since the desired trajectory will be used for the motion planning of Omni-directional mobile robots in small-sized league, the following assumptions are considered.

Assumption 1:

The weights of all states and inputs in the cost function are considered to be equal.

$$J = \frac{1}{2}\int_{0}^{t_f} (\mathrm{x}_1^2 + \mathrm{x}_2^2 + \mathrm{x}_3^2 + \mathrm{x}_4^2 + u_1^2 + u_2^2)dt, \tag{14}$$

Assumption 2:

Obstacles are presented by circles with radius r_i which are centered at $X_{c_i} = \begin{bmatrix} x_{c_i} & y_{c_i} \end{bmatrix}^T$.

$$\bigcup_{i=1}^{k}\left[(\mathrm{x}_1(\mathrm{t}) - \mathrm{x}_{c_i})^2 + (\mathrm{x}_2(\mathrm{t}) - y_{c_i})^2 \geq r_i^2\right] \tag{15}$$

Assumption 3:

The initial time and position of the robot are considered at the origin.

$$x_1(0) = 0,\ x_2(0) = 0 \tag{16}$$

Assumption 4:

The initial and final velocity of the robot should be zero.

$$x_3(0) = 0,\ x_4(0) = 0,\ x_3(t_f) = 0,\ x_4(t_f) = 0 \tag{17}$$

Finally, the resulted nonlinear OCP can be presented as follows:

$$Min \quad J = \frac{1}{2}\int_{0}^{t_f} (\mathrm{x}_1^2 + \mathrm{x}_2^2 + \mathrm{x}_3^2 + \mathrm{x}_4^2 + u_1^2 + u_2^2)dt, \tag{18}$$

s.t:

$$\begin{cases} \dot{x}_1(\mathrm{t}) = x_3(\mathrm{t}) = v_x(\mathrm{t}) \\ \dot{x}_2(\mathrm{t}) = x_4(\mathrm{t}) = v_y(\mathrm{t}) \\ \dot{x}_3(\mathrm{t}) = u_1(\mathrm{t}) = a_x(\mathrm{t}) \\ \dot{x}_4(\mathrm{t}) = u_2(\mathrm{t}) = a_y(\mathrm{t}) \end{cases} \tag{19}$$

$$\bigcup_{i=1}^{k}\left[(\mathrm{x}_1(\mathrm{t})-\mathrm{x}_{c_i})^2+(\mathrm{x}_2(\mathrm{t})-y_{c_i})^2 \geq r_i^2\right] \tag{20}$$

$$\sqrt{x_3^2(\mathrm{t})+x_4^2(\mathrm{t})} \leq v_{\max}, \tag{21}$$

$$\sqrt{u_1^2(\mathrm{t})+u_2^2(\mathrm{t})} \leq a_{\max} \tag{22}$$

$$\begin{gathered} x_1(0)=0, x_2(0)=0, x_1(t_f)=x_{1_f}, x_2(t_f)=x_{2_f} \\ x_3(0)=0, x_4(0)=0, x_3(t_f)=0, x_4(t_f)=0 \end{gathered} \tag{23}$$

5. The Proposed Optimal Motion Planning Method

5.1. Direct method to convert Nonlinear OCP to NLP

In this subsection, 4-order polynomial functions are employed to parameterize the position state variables. By this transformation, the Nonlinear OCP is converted to a NLP.

First, the position state variables are parameterized as follows:

$$x = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4, \tag{24}$$

$$y = b_0 + b_1 t + b_2 t^2 + b_3 t^3 + b_4 t^4 \tag{25}$$

Also, according to (13), velocity and control inputs are obtained as follows:

$$v_x(\mathrm{t}) = a_1 + 2a_2 t + 3a_3 t^2 + 4a_4 t^3 \tag{26}$$

$$v_y(\mathrm{t}) = b_1 + 2b_2 t + 3b_3 t^2 + 4b_4 t^3 \tag{27}$$

$$u_1(\mathrm{t}) = 2a_2 + 6a_3 t + 12a_4 t^2 \tag{28}$$

$$u_2(\mathrm{t}) = 2b_2 + 6b_3 t + 12b_4 t^2 \tag{29}$$

Finally, by substituting (24) and (25) into (18)-(23), the resulted NLP is obtained as follows:

$$\min \hat{J}(\mathrm{a}_n, \mathrm{b}_n) \tag{30}$$

subject to:

$$\bigcup_{i=1}^{n}\left[L_i(\mathrm{a}_n, \mathrm{b}_n, \mathrm{t}) \geq 0\right], \tag{31}$$

$$P(\mathrm{a}_n, \mathrm{b}_n, \mathrm{t}) \leq 0, \tag{32}$$

$$\mathrm{Q}(\mathrm{a}_n, \mathrm{b}_n, \mathrm{t}) \leq 0 \tag{33}$$

In which $\hat{J}$, L_i, P, and Q are new polynomial functions associated with cost function, obstacle constraints, maximum velocity constraint, and maximum acceleration constraint, respectively.

5.2. The new approach for solving resulted NLP

In this subsection, a low computational approach for solving the resulted NLP is presented. Unlike the previous NLP solution method (Mohseni and Fakharian (2015)) in which the nonlinear inequality constraints are considered in some pre-defined points, in the new proposed approach, the nonlinear inequality constraints are considered only once. The main idea for this approach is the division of the nonlinear constraints into non-positive and non-negative constraints. Then, it is guaranteed that minimum value of non-negative constraints is non-negative, and the maximum value of non-positive constraints is non-positive.

Thus, an optimization problem is created as follows:

$$\min \hat{J}(\mathrm{a}_4, \mathrm{b}_4) \tag{34}$$

subject to:

$$\min\left(\bigcup_{i=1}^{k}\left[L_i(\mathrm{a}_4, \mathrm{b}_4, \mathrm{t})\right]\right) \geq 0, \tag{35}$$

$$\max\left(P(\mathrm{a}_4, \mathrm{b}_4, \mathrm{t})\right) \leq 0, \tag{36}$$

$$\max\left(\mathrm{Q}(\mathrm{a}_4, \mathrm{b}_4, \mathrm{t})\right) \leq 0 \tag{37}$$

6. Simulation Results

To demonstrate the computational efficiency of the proposed method, simulations are performed. Also, simulation results are compared with the result obtained by Mohseni and Fakharian (2015).
Table 1 shows the simulation data in SI units for scenario#1 and scenario#2.

Table 1
Simulation data

Parameters	Scenario#1	Scenario#2
X_f	$(2,2)$	$(2,2)$
t_f	3.5	3.5
X_{c_1}, r_1	$(0.3,0.4), 0.11$	$(0.3,0.4), 0.11$
X_{c_2}, r_2	$(0.6,0.5), 0.1$	$(0.6,0.5), 0.1$
X_{c_3}, r_3	$(1,0.9), 0.16$	$(1,0.9), 0.16$
X_{c_4}, r_4	$(1.3,1.4), 0.18$	$(1.3,1.4), 0.18$
$v_{\max}$	–	$\sqrt{1.5}$
$a_{\max}$	–	$\sqrt{4}$

Polynomials trajectories for scenario#1 are obtained as follows:

$$x_1(t) = 0.3832t^2 - 0.0324t^3 - 0.0087t^4, \tag{38}$$

$$x_2(t) = -5.28t^2 + 3.2037t^3 - 0.471t^4 \tag{39}$$

$$x_3(t) = 0.7664t - 0.0972t^2 - 0.0348t^3, \tag{40}$$

$$x_4(t) = -10.5599t + 9.6111t^2 - 1.884t^3, \tag{41}$$

$$u_1(t) = 7.6644 - 0.0194t - 0.1044t^2, \tag{42}$$

$$u_2(t) = -10.5599 + 19.2222t - 5.652t^2 \tag{43}$$

Fig. 2 shows the position, velocity, and acceleration of robot using two NLP solution methods in scenario#1.
The optimal trajectories using two NLP solution methods for scenario#1 are illustrated in Fig. 3.
Also, to study the impact of maximum limitation on the velocity and acceleration of robot, the diagrams of the square of total velocity, and the acceleration of robot using two solution methods in scenario#1 is shown in Fig. 4.

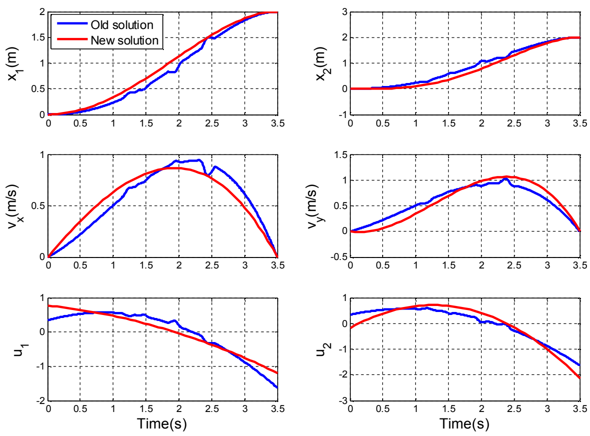

Fig. 2. Position, velocity, and acceleration of robot using two NLP solution methods in scenario#1

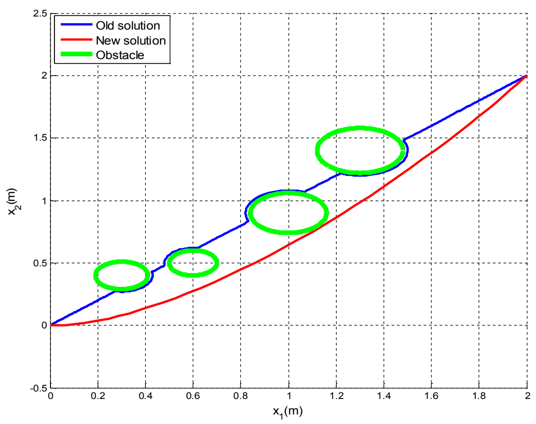

Fig. 3. Obtained optimal trajectories using two NLP solution methods in scenario#1

As shown in Fig. 4, the maximum of square of total velocity and acceleration of robot by two NLP solution methods are specified in the diagrams. With respect to these values, we get:

$$\begin{aligned}
&\max(v_x^2(t) + v_y^2(t))\Big|_{new\ solution\ method} = 1.805m/s, \\
&\max(v_x^2(t) + v_y^2(t))\Big|_{old\ solution\ method} = 1.889m/s, \\
&\max(a_x^2(t) + a_y^2(t))\Big|_{new\ solution\ method} = 5.969m/s^2, \\
&\max(a_x^2(t) + a_y^2(t))\Big|_{old\ solution\ method} = 5.261m/s^2
\end{aligned} \tag{44}$$

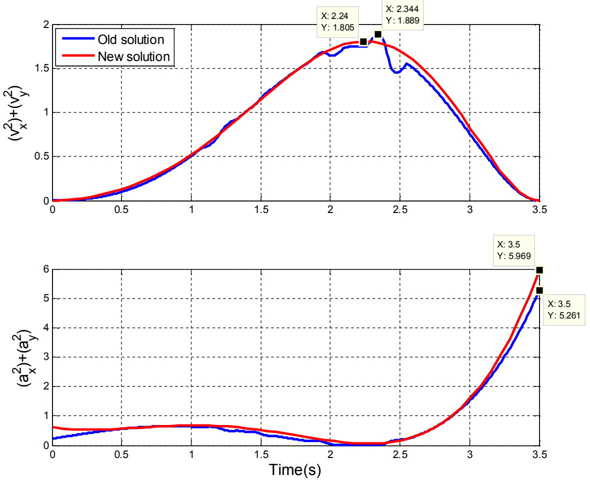

Fig. 4. Square of total velocity and acceleration of robot using two NLP solution methods in scenario#1

Thus, the maximum limitation on velocity and acceleration of robot in scenario#2 is considered as follows:

$$v_{max} = \sqrt{1.5} \simeq 1.22m/s,\ a_{max} = \sqrt{4} = 2m/s^2 \tag{45}$$

So, we have two constraints as follows:

$$\sqrt{x_3^2(t) + x_4^2(t)} \le 1.22, \tag{46}$$

$$\sqrt{u_1^2(t) + u_2^2(t)} \le 2 \tag{47}$$

Polynomials trajectories for scenario#2 are obtained as follows:

$$x_1(t) = 1.8128t^2 - 0.8493t^3 + 0.108t^4, \tag{48}$$

$$x_2(t) = 0.1652t^2 + 0.0922t^3 - 0.0265t^4 \tag{49}$$

$$x_3(t) = 3.6256t - 2.5479t^2 + 0.432t^3, \tag{50}$$

$$x_4(\mathrm{t}) = 0.3303t + 0.2766t^2 - 0.106t^3, \tag{51}$$

$$u_1(\mathrm{t}) = 3.6256 - 5.0958t + 1.296t^2, \tag{52}$$

$$u_2(\mathrm{t}) = 0.3303 + 0.5532t - 0.318t^2 \tag{53}$$

Fig. 5 shows the position, velocity, and acceleration of robot using two NLP solution methods in scenario#2.

Also, the optimal trajectories by two NLP solution methods in scenario#2 are illustrated in Fig. 6.

Square of total velocity and acceleration of robot using two solution methods in scenario#2 is depicted in Fig. 7.

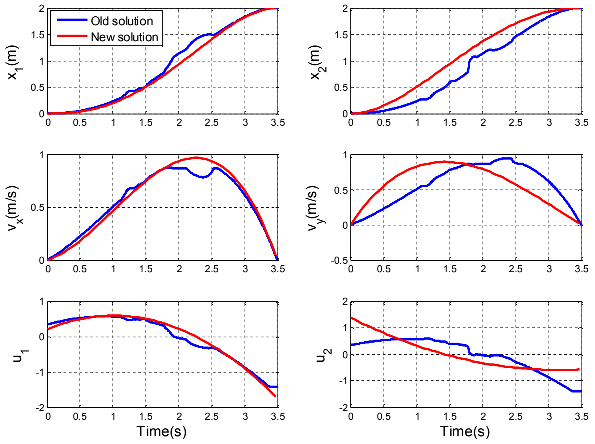

Fig. 5. Position, velocity, and acceleration of robot using two NLP solution methods in scenario#2

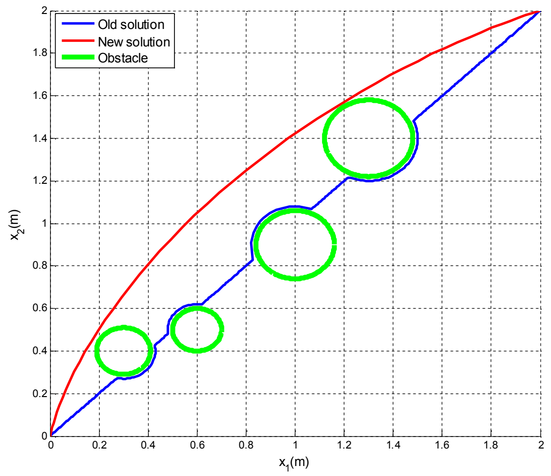

Fig. 6. Obtained optimal trajectories using two NLP solution methods in scenario#2

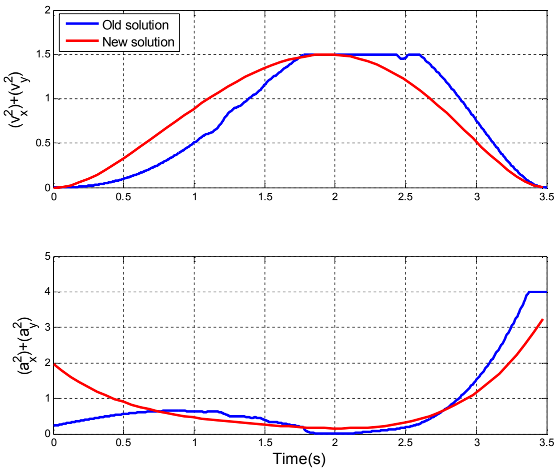

Fig. 7. Square of total velocity and acceleration of robot using two NLP solution methods in scenario#2

We can see in Figs. 3 and 6 that the robot without collision with obstacles obtains the desired goal positions in two scenarios. Also, it can be seen in Figs. 2 and 5 that the initial and final conditions on the velocity of the robot are satisfied in two scenarios.

As shown in Fig. 7, the maximum of the square of total velocity and acceleration of robot by two solution methods is limited on $1.5m/s$ and $4m/s^2$, respectively.

Table 2 presents simulation times and the cost function values by two NLP solution methods.

where $t_{run_{new}}$ and $t_{run_{old}}$ are the simulation times of the proposed method and the previous method (Mohseni and Fakharian (2015)), respectively. Also, J_{new} and J_{old} are the performance index values using the proposed method and the previous method (Mohseni and Fakharian (2015)), respectively.

Table 2
Simulation times and performance index values

Parameters	Obtained values in scenario#1	Obtained values in scenario#2
J_{old}	7.24	7.27
J_{new}	7.48	7.69
$t_{run_{old}}$ (s)	42.66	59.52
$t_{run_{new}}$ (s)	0.12	0.1

As shown in Table 2, the simulation times have decreased dramatically using new NLP solution method in comparison with the old NLP solution method. Also, the performance index value using the new NLP solution method has increased slightly in comparison with the old NLP solution method.

7. Comparison with another Direct Method

In this section, the proposed method is compared with another direct method called *Gaussian pseudo spectral method (GPM)* to illustrate its effectiveness.
The GPM is a direct method which formulates the optimal control problem directly into a NLP. In this method, the state and control variables are approximated using orthogonal polynomials based on interpolation at collocation points.
Recently, this method has become increasingly popular and is considered as a powerful direct computational method for complex problems (Garg and Hager, 2011; Garg et al., 2010; Meng et al., 2014; Wang et al., 2013).
Table 3 presents the simulation data in SI units for this simulation.
Simulation results are given in Fig. 8-Fig. 10.

Table 3
Simulation data

Parameters	Values
X_f	$(1,1)$
t_f	3
X_{c_1}, r_1	$(0.6,0.5), 0.085$
X_{c_2}, r_2	$(0.3,0.4), 0.085$
X_{c_3}, r_3	$(0.7,0.75), 0.085$
$v_{\max}$	3
$a_{\max}$	4

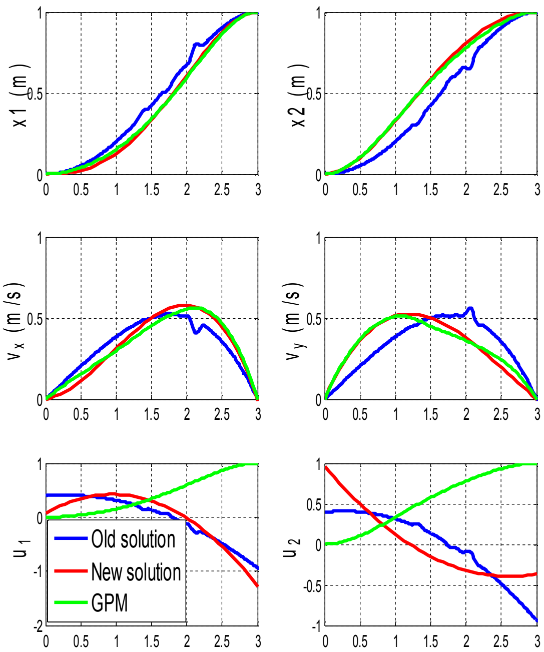

Fig. 8. Position, velocity, and acceleration of robot using three direct methods

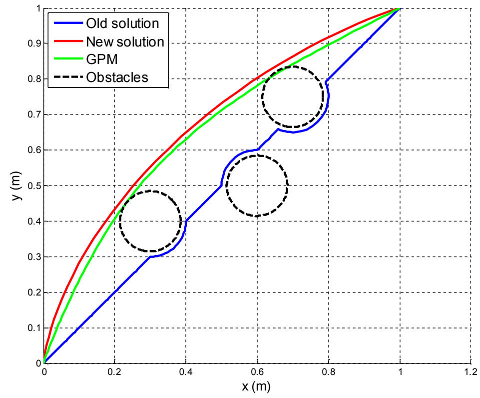

Fig. 9. Optimal trajectories using three direct methods

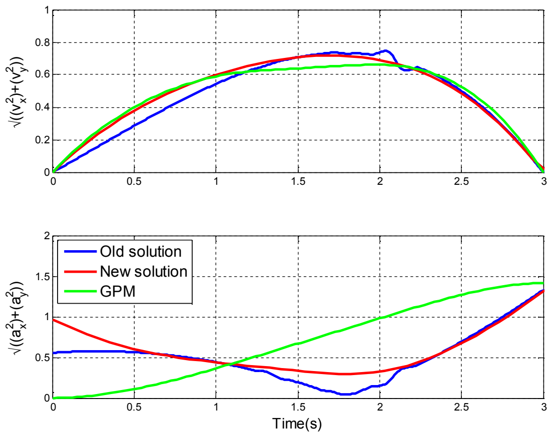

Fig. 10. Total velocity and acceleration of robot using three direct methods

As seen in Fig. 9, the robot passes obstacles and reaches the desired goal positions in all three methods. Also, it can be seen in Fig. 8 that the initial and final conditions on the velocity of the robot are satisfied using these methods.
As shown in Fig. 10, the maximum of total velocity and acceleration of robot by all three methods are satisfied.
Table 4 compares simulation times and the cost function values by three direct methods.

Table 4
Simulation results

Parameters	Obtained values
J_{old}	1.9
J_{new}	2.08
J_{GPM}	3.63
$t_{run_{old}}$ (s)	20.74
$t_{run_{new}}$ (s)	0.81
$t_{run_{GPM}}$ (s)	3.4

As shown in Table 4, the simulation times have decreased dramatically using new NLP solution method in comparison with the previous NLP solution method (Mohseni and Fakharian (2015)) and the Gaussian pseudo pectral method (GPM). Also, the cost function values in the presented two polynomial methods are obtained less than the Gaussian pseudo spectral method (GPM). Thus, the proposed method can be employed as an appropriate and low computational cost method in optimal motion planning of Omni-directional mobile robots in presence of obstacles and under motion constraints.

8. Discussion and Conclusion

In this paper, a direct method has been presented for optimal trajectory planning and obstacle avoidance of Omni-directional mobile robot under velocity and acceleration constraints. The main idea was to convert the main optimal control problem to a NLP using polynomial functions for parameterization of the trajectories. Also, the paper proposed a new approach for solving the resulted NLP which greatly reduces the required computational cost for obtaining the desired optimal trajectories.
According to the obtained graphs of simulations using the proposed method, the following results were observed in all scenarios and situations:
- The robot moved in obtained optimal trajectory from initial position to desired destination position.
- The robot was stopped at the destination point.
- The limitations on maximum velocity and acceleration of robot were satisfied.
- The robot was transferred in desired trajectories without any collision with obstacles.

Also, by comparison of obtained values in the presented tables, the following results can be stated:
- The elapsed time in simulation using the new proposed method was decreased sharply in comparison with the other mentioned direct methods.
- The cost function values in two direct polynomial methods were obtained less than GPM direct method.

So, the comparison of results obtained by the proposed method with the previous polynomial method and Gaussian pseudo spectral method (GMP) demonstrates the effectiveness and viability of the proposed method as a suitable direct method for optimal trajectory planning and obstacle avoidance of Omni-directional mobile robots.

References

Assassa, F., & Marquardt, W. (2014). Dynamic optimization using adaptive direct multiple shooting. *Computers & Chemical Engineering*, *60*, 242-259.

Betts, J. T. (1998). Survey of numerical methods for trajectory optimization.*Journal of guidance, control, and dynamics*, *21*(2), 193-207.

Bryson, A. E. (1975). *Applied optimal control: optimization, estimation and control*. CRC Press.

Duan, H., Zhang, R., Yu, F., Gao, J., & Chen, Y. (2016). Optimal trajectory planning for glass-handing robot based on execution time acceleration and jerk. *Journal of Robotics*.

Frick, P. A., & Stech, D. J. (1993). Epsilon-Ritz method for solving optimal control problems: Useful parallel solution method. *Journal of optimization theory and applications*, *79*(1), 31-58.

Garg, D., Hager, W. W., & Rao, A. V. (2011). Pseudospectral methods for solving infinite-horizon optimal control problems. *Automatica*, *47*(4), 829-837.

Garg, D., Patterson, M. A., Francolin, C., Darby, C. L., Huntington, G. T., Hager, W. W., & Rao, A. V. (2011). Direct trajectory optimization and costate estimation of finite-horizon and infinite-horizon optimal control problems using a Radau pseudospectral method. *Computational Optimization and Applications*, *49*(2), 335-358.

Garg, D., Patterson, M., Hager, W. W., Rao, A. V., Benson, D. A., & Huntington, G. T. (2010). A unified framework for the numerical solution of optimal control problems using pseudospectral methods. *Automatica*, *46*(11), 1843-1851.

Goh, C. J., & Teo, K. L. (1988). Control parametrization: a unified approach to optimal control problems with general constraints. *Automatica*, *24*(1), 3-18.

Hartl, R. F., Sethi, S. P., & Vickson, R. G. (1995). A survey of the maximum principles for optimal control problems with state constraints. *SIAM review*,*37*(2), 181-218.

Jaddu, H. (2002). Direct solution of nonlinear optimal control problems using quasilinearization and Chebyshev polynomials. *Journal of the Franklin Institute*, *339*(4), 479-498.

Jaddu, H. M. H. (1998). Numerical Methods for solving optimal control problems using chebyshev polynomials.

Jaddu, H., & Vlach, M. (2002). Successive approximation method for non-linear optimal control problems with application to a container crane problem.*Optimal Control Applications and Methods*, *23*(5), 275-288.

Kameswaran, S., & Biegler, L. T. (2008). Convergence rates for direct transcription of optimal control problems using collocation at Radau points. *Computational Optimization and Applications*, *41*(1), 81-126.

Kirches, C., Wirsching, L., Bock, H. G., & Schlöder, J. P. (2012). Efficient direct multiple shooting for nonlinear model predictive control on long horizons. *Journal of Process Control*, *22*(3), 540-550.

LaValle, S. M. (2006). Planning algorithms. *Cambridge university press*.

Li, B., Wang, K., & Shao, Z. (2015, September). Time-optimal trajectory planning for tractor-trailer vehicles via simultaneous dynamic optimization. *International Conference on Intelligent Robots and Systems (IROS), IEEE/RSJ Press,* 3844-3849.

Luo, L. P., Yuan, C., Yan, R. J., Yuan, Q., Wu, J., Shin, K. S., & Han, C. S. (2015). Trajectory planning for energy minimization of industry robotic manipulators using the Lagrange interpolation method. *International Journal of Precision Engineering and Manufacturing*, *16*(5), 911-917.

Meng, S., Xiang, J., Luo, Z., Ren, Y., & Zhuang, N. (2014). A novel trajectory planning strategy for aircraft emergency landing using Gauss pseudospectral method. *Control Theory and Technology*, *12*(4), 393-401.

Mohseni, N. A., & Fakharian, A. (2015). Optimal trajectory planning for an omni-directional mobile robot with static obstacles: a polynomial based approach. In *AI & Robotics (IRANOPEN),* IEEE Press, 1-6.

Purwin, O., & D'Andrea, R. (2005, June). Trajectory generation for four wheeled omnidirectional vehicles. *Proceedings of the 2005, American Control Conference*, IEEE Press, 4979-4984.

Rao, A. V. (2009). A survey of numerical methods for optimal control. *Advances in the Astronautical Sciences*, *135*(1), 497-528.

Richter, C., Bry, A., & Roy, N. (2016). Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments. In *Robotics Research, Springer International Publishing*, 649-666.

Tohidi, E., & Nik, H. S. (2015). A Bessel collocation method for solving fractional optimal control problems. *Applied Mathematical Modelling*, *39*(2), 455-465.

Vlassenbroeck, J., & Van Dooren, R. (1988). A Chebyshev technique for solving nonlinear optimal control problems. *IEEE transactions on automatic control*, *33*(4), 333-340.

Von Stryk, D. M. O. (1993). Numerical solution of optimal control problems by direct collocation. *In Optimal Control,* 129-143. *Birkhäuser Basel*.

Von Stryk, O., & Bulirsch, R. (1992). Direct and indirect methods for trajectory optimization. *Annals of operations research*, *37*(1), 357-373.

Wang, Y., De Schutter, B., van den Boom, T. J., & Ning, B. (2013). Optimal trajectory planning for trains–A pseudospectral method and a mixed integer linear programming approach. *Transportation Research Part C: Emerging Technologies*, *29*, 97-114.

This article can be cited: Azim Mosheni, N. & Fakharian, A. (2017). Direct Optimal Motion Planning for Omni-directional Mobile Robots under Limitation on Velocity and Acceleration. *Journal of Optimization in Industrial Engineering.*10 (22), 93-101.

URL: http://qjie.ir/article_280.html

DOI: 10.22094/joie.2017.280