Numerical method for attitude motion planning of one-legged hopping robot

Lili Yang

Zibo Vocational Institute
Zibo 255314, China

29065816@qq.com

ABSTRACT. *The attitude motion planning for one-legged hopping robot with nonholonomic constraint is studied. Firstly, the dynamic model of the robot is established by using the nonholonomic constraint characteristic. Secondly, the energy consumption of the robot is used as the optimization objective function. Lastly, a numerical algorithm is designed by combining curve fitting method and particle swarm optimization algorithm, which is used to realize the optimal trail of robot's attitude motion by optimizing the objective function. The designed algorithm makes use of curve fitting to approach the motion trail of the robot's drivable leg, and the coefficients of the fitting polynomial are taken as the optimization parameters which can be obtained by particle swarm optimization algorithm. The main advantage of this method lies in that the initial value and the final value of the optimal control input are all zero, which solves the problem that the initial value and the final value of the control input are not zero in the traditional method, making it convenient to control the motion of the drivable leg by the motor in engineering application. At the end of the study, the effectiveness of this method is proved by the results of the numerical simulation.*

RÉSUMÉ. *La planification du mouvement d'attitude d'un robot à sauts sur un pied avec contrainte non holonomique est étudiée. Premièrement, le modèle dynamique du robot est établi en utilisant la caractéristique de contrainte non holonomique. Deuxièmement, la consommation d'énergie du robot est utilisée comme fonction objectif d'optimisation. Enfin, un algorithme numérique est conçu en combinant une méthode d'ajustement de courbe et un algorithme d'optimisation par essaims particulaires, utilisés pour réaliser la trace optimale du mouvement d'attitude du robot en optimisant la fonction objectif. L'algorithme conçu utilise l'ajustement de courbe pour approcher la piste de mouvement du pied e du robot, et les coefficients du polynôme d'ajustement sont considérés comme les paramètres d'optimisation qui peuvent être obtenus par l'algorithme d'optimisation par essaims particulaires. Le principal avantage de cette méthode réside dans le fait que la valeur initiale et la valeur finale de l'entrée de contrôle optimale sont toutes égales à zéro, ce qui résout le problème selon lequel la valeur initiale et la valeur finale de l'entrée de contrôle ne sont pas nulles dans la méthode traditionnelle, ce qui rend il est pratique de contrôler le mouvement de la jambe pouvant être entraînée par le moteur dans les applications techniques. A la fin de l'étude, l'efficacité de cette méthode est prouvée par les résultats de la simulation numérique.*

KEYWORDS: *one-legged hopping robot, nonholonomic constraint, attitude motion planning, optimization.*

MOTS-CLÉS: robot sautant sur un pied, contrainte non holonomique, planification du mouvement d'attitude, optimisation.

DOI:10.3166/JESA.50.545-553

1. Introduction

Along with the acceleration of industrialization, robots have been widely applied in industrial field and the trend of "robot substituting human" is more and more obvious. For some operations on the industrial production line, especially repetitive and programmed work, the robot is an ideal substitute for human beings. The working efficiency of robot is usually 5-10 times of that of manual operation. At the same time, the robot can adapt to harsh working conditions. In particular, the robot is suitable for substituting human beings to work under extreme conditions such as industrial dust and mining, and can be widely used in space exploration, underground exploration, disaster relief and other fields. The robot control is a complex application technology with multi-disciplinary and multi-field cross-link, so it has been a hotspot and difficult problem in the field of robot research in recent years.

The one-legged hopping robot (Raibert and Tello, 2007), as a kind of typical humanoid robot, has been studied extensively by the scholars from these aspects, such as the stability of the robot system model (Naik *et al.*, 2006), jumping gait (Shabestari and Emami, 2016), dynamics characteristics (Hyon *et al.*, 2003) and control methods (Liu *et al.*, 2016). The author of this paper discussed the optimal attitude motion planning for one-legged hopping robot, but the results of numerical simulation show that both the initial value and the final value of the optimal control input law obtained by particle swarm optimization algorithm in the motion period of the robot system are not zero (Yang, 2013), so it is difficult to realize the motion control of the robot system directly with the motor in practical application.

In order to solve this problem, this study puts forward a numerical method combining curve fitting method and particle swarm optimization algorithm, which is used to realize the optimal trail of robot's attitude motion and solve the problem that the initial value and the final value of the control input are not zero. Firstly, the attitude motion equation of the robot system is derived by using multi-body dynamics. When there is no external force and external moment, the attitude motion planning is transformed into the nonholonomic motion planning by using the nonholonomic constraint characteristic of the robot system. The five-order curve fitting method is used to approximate the motion trail of the robot leg. By taking the coefficients of the fitting polynomial as the optimization parameters and combining with the particle swarm optimization algorithm, the optimal trail of the robot attitude motion is obtained. The optimal control input law of the system with zero initial value and zero final value is obtained. At the end of the paper, numerical simulation is carried out through examples and the simulation results indicate that the method is effective and feasible.

2. Robot model

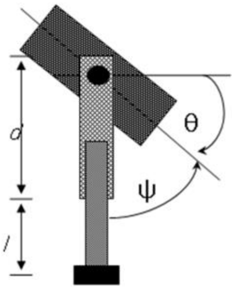

Figure 1. Model of one-legged hopping robot

Figure 1 shows the model of one-legged hopping robot system composed of a body and a drivable leg. The connecting joint between the body and the leg is a rotational joint so that the drivable leg can rotate around the body and stretch freely. The robot system has 2 degrees of freedom, considering the motion of the system in the plane. Using multi-body system dynamics, the total angular momentum of the system is obtained (Raibert and Tello, 2007) as:

$$I\dot{\theta} + m(l+d)^2(\dot{\theta} + \dot{\psi}) \tag{1}$$

Where, ψ, l, θ are the rotation angle of the drivable leg relative to the body, the expansion and contraction quantity of the drivable leg and the rotation angle of the body respectively, and the counterclockwise rotation is positive. I is the inertial matrix of the body, m is the mass of the leg. To simplify the calculation, the mass is assumed to be concentrated on the foot, d is the upper leg length of the robot, then the system configuration parameter can be expressed as q=(ψ, l, θ).

In the absence of external force and external moment, the "constraint" of the system is the conservation of angular momentum. Assuming that the initial angular momentum of the system is zero, then the following formula can be obtained through Formula (1):

$$I\dot{\theta} + m(l+d)^2(\dot{\theta} + \dot{\psi}) = 0 \tag{2}$$

Since the angle of rotation of the drivable leg relative to the body and the expansion and contraction quantity of the leg can be directly controlled by the motor, its speed is taken as the control input. $\dot{\psi} = u_1$, $\dot{l} = u_2$, then the two control inputs when the system is moving are the rotational angular speed and the linear speed of the robot, that is, three configuration parameters which can realize the attitude motion of the system are controlled by the two control input variables.

Substitute $\dot{\psi} = u_1$, $\dot{l} = u_2$, into Formula (2), then,

$$\dot{\boldsymbol{q}} = \boldsymbol{B}(\boldsymbol{q})\boldsymbol{u} \tag{3}$$

Where, $\boldsymbol{B}(\boldsymbol{q}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -\dfrac{m(l+d)^2}{I+m(l+d)^2} & 0 \end{bmatrix}$, $\boldsymbol{u} = \begin{bmatrix} \dot{\psi} \\ \dot{l} \end{bmatrix}$

Formula (3) is a non-integrable form, which shows that the system is subject to non-integrable kinematic constraint, that is, the leg of the robot system can generate disturbance to the attitude of the robot body in the course of expansion and rotation, which is a typical nonholonomic constraint (Janiak and Tchoń, 2011).

3. Optimal control of attitude motion based on curve fitting method

In Formula (3), $\boldsymbol{q}$ is a function of $\boldsymbol{u}$ and t, which can be regarded as two control input variables controlled by the angular speed and the linear velocity of the drivable leg when the attitude motion of robot system is represented by three configuration variables. The system motion planning with nonholonomic constraint is usually to determine the control input $\boldsymbol{u}$ so that the system moves from a given initial configuration $\boldsymbol{q}_0 = (\psi_0, l_0, \theta_0)^{\mathrm{T}} \in \boldsymbol{R}^3$ to a given end configuration $\boldsymbol{q}_f = (\psi_f, l_f, \theta_f)^{\mathrm{T}} \in \boldsymbol{R}^3$ over a set period of time while optimizing the objective function to minimize the energy consumed by the system. A set of optimal control input law $\boldsymbol{u}(t) \in \boldsymbol{R}^2, t \in [0,T]$ can be obtained by optimizing the objective function. Within the set time period T, the control system moves from the initial configuration $\boldsymbol{q_0}$ to the end configuration $\boldsymbol{q_f}$ (Wu *et al.*, 2011).

The robot system consumes energy to realize the attitude motion. When the attitude of the system changes smoothly, the consumed energy is less (Su *et al.*, 2012). Therefore, according to the minimum energy principle, the energy consumption of the robot system is selected as the optimization objective function, which can be expressed as:

$$\boldsymbol{J}(\boldsymbol{u}) = \int_0^T < \boldsymbol{u}, \boldsymbol{u} > \mathrm{d}t \tag{4}$$

Assuming that the system is controllable, there exists optimization solution $\boldsymbol{u}(t)$ $\in L_2([0,T])$, where L_2 is Hilbert space consisting of measurable vector function $\boldsymbol{u}(t)=[u_1(t), \ldots, u_n(t)]^{\mathrm{T}}$. The general method to solve $\boldsymbol{u}(t)$ is to use Fourier series expansion method to discretize the optimization problem of infinite dimension, transforming the problem into an optimization problem to find a solution to finite dimensional constraint. One of the main shortcomings of this method is that the initial value and the final value of the control input are usually not zero, so it is difficult to be directly applied in engineering. In order to solve this problem, the method of curve fitting is used for numerical approximation.

A set of data points (x_i, y_i)(i=1,2,3, …, n) is known, an approximate curve analytic function $y \approx \varphi(x)$ is obtained, and $\varphi(x)$ is made to approximate the value of y_i at each data point x_i, which is the curve fitting method. A linear least square method (Bingül and Karahan, 2011) is used, and the analytic function can be expressed as:

$$f(x) = \sum_{i=0}^{m} a_i k_i(x), \quad m \le n \tag{5}$$

Where, $f(x)$ is a linear combination of function systems $k_i(x)$, a_i(i=1,2,3, …, m) is a combination coefficient, $k_i(x)$ is a set of function selected in advance. Look for a series of date a_0, a_1, …, a_m to minimize $I = \sum_{i=0}^{m} (f(x_i) - y_i)^2$.

The coefficients a_0, a_1, …, a_n can be determined by the least square method so that the extreme values are as follows when $I = \varphi(a_0, a_1, \ldots, a_n)$:

$$\frac{\partial \varphi}{\partial a_j} = 2\sum_{i=0}^{m} x_i^j (a_0 + a_1 x + \ldots + a_n x_i^n - y_i) = 0 \tag{6}$$

From formula (6), the following can be obtained:

$$\sum_{i=0}^{m} (a_0 x_i^j + a_1 x_i^{1+j} + \ldots + a_n x_i^{n+j}) = \sum_{i=0}^{m} x_i^j y_i \tag{7}$$

The fitting polynomial f(x)= $a_0 + a_1 x +$, …, $+ a_n x_n$ can be obtained by solving the coefficients of Formula (7). When n≥7, the rounding error of the solution of equation is large, and the five-order curve is fitted (Chen and Zheng, 2012).

In the calculation, the ordered nodes $0 = t_0 < t_1 < \ldots < t_N = T$are divided and the control input vectors at the ordered nodes are $\lambda = [\lambda_0, \lambda_1, \ldots, \lambda_N]^T$, and the five-order curve fitting technology can be used to obtain:

$$\boldsymbol{u}(t) = s(\lambda, t), \ t \in [0, T] \tag{8}$$

Formula (8) is substituted into Formula (4), and the control input vector $\boldsymbol{\lambda}$ is used as a new control variable and the penalty function method is introduced, then Formula (4) can be expressed as:

$$\boldsymbol{J}(\lambda, \mu) = \int_0^T [s(\lambda, t)]^2 \mathrm{d}t + \mu \left\| \boldsymbol{f}(\lambda) - \boldsymbol{q}_f \right\|^2 \tag{9}$$

Where, $\mu > 0$ is the penalty factor, $f(\lambda)$ is the solution of Formula (3) at the time of the control input $\boldsymbol{u}$ when $t = T$. Therefore, the problem of finding $\boldsymbol{u}$ to minimize value of Formula (4) can be converted to the problem of finding $\boldsymbol{\lambda}$ that makes minimum value of Formula (9).

4. Particle swarm optimization for attitude motion planning of system

The particle swarm optimization algorithm is used to optimize the parameters of Formula (9). As an optimization iterative algorithm to simulate the foraging behavior of bird clusters, the particle swarm optimization algorithm regards each alternative solution as a bird in solution space, which is called "particle". A flock of birds consisting of several birds (several "particles") collectively search for food in a search space (only one piece of food in the space). The position and speed of each particle are initialized to a set of random values in the search space. Its position and speed are updated according to the individual optimal value P_i currently found by the particle and the global optimal value P_g of the current whole bird flock (Poli *et al.*, 2007):

$$\boldsymbol{V}_{id}(t+1) = w \cdot \boldsymbol{v}_{id}(t) + c_1 \cdot r_1 \cdot [P_{id}(t) - \boldsymbol{x}_{id}(t)] + c_2 \cdot r_2 \cdot [P_{gd}(t) - \boldsymbol{x}_{id}(t)] \tag{10}$$

$$\boldsymbol{X}_{id}(t+1) = \boldsymbol{x}_{id}(t) + \boldsymbol{v}_{id}(t+1) \tag{11}$$

Where, the speed vector is $V_i=(v_{i1}, v_{i2}, \ldots, v_{iD})$ and the position vector is $X_i=(x_{i1}, x_{i2}, \ldots, x_{iD})$, D is te dimension of a single particle, w is inertial value; c_1,c_2 are acceleration factors; r_1,r_2 are random numbers. The variation of the position and speed of the d($1 \le d \le D$) dimension of the particle has a fixed value range, and if it exceeds the range, the boundary value is taken.

The calculation steps of the specific algorithm are as follows:

1) Initialization. The basic parameters w, c_1, c_2, r_1, r_2 of the particle swarm are given, the position X_i and speed V_i of each particle are initialized, P_i of individual particle and P_g of the whole population are recorded, and the objective function value of each particle is calculated according to Formula (9).

2) Iteration. The speed and position of each particle are iterated and updated by formulas (10) and (11).

3) Judgment. The objective function value after iteration of each particle is calculated according to Formula (9). If the current objective function value is better than the value before the update, the objective function value is updated. If obtained P_i and P_g are better than P_i and P_g before the update, the obtained P_i and P_g are updated.

4) Test. When the preset maximum number of iteration is reached or the objective function value to be satisfied is reached, the program is stopped and the optimal solution is output, otherwise we will return to step 2) and the iterative optimization is continued.

5. Example simulation

The mass geometric parametersv of the robot system are I=16.66kg·m^2, m=10kg, l=.5m, d=0.5m.

The basic parameters of the initialization particle swarm optimization algorithm (Venter and Sobieszczanski-Sobieski, 2003) are: number n=15, acceleration factor c1=c2=1.49, inertial value w decrease linearly from the initialization 0.9 to the final value 0.4 with 600 iterations. The robot system motion time is T=5s that is divided equally in five parts. The value of the control input at the initial time and the end time is 0, the number of parameters of each control input is 4, and the dimension of λ corresponding to two control inputs is 8, that is, the dimension of the particle is D=8 when the particle swarm optimization algorithm is applied.

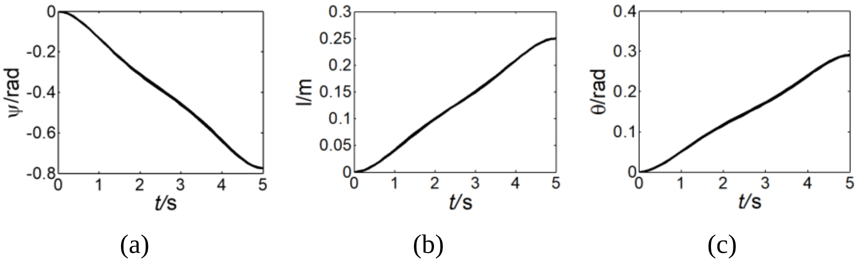

Figure 2. Motion trail of robot configuration parameter

The simulation experiment is carried out by Matlab software. The initial figuration and end figuration of the robot system are assumed to be $[0\ 0\ 0]^T$ and $[-\pi/4\ 0.25\ \pi/12]^T$. The simulation results are obtained, as shown in Figure 2 and Figure 3. It can be seen from the figures that the planned robot attitude motion trail and the rotational angle speed of the drivable leg and the expansion linear speed are smooth and continuous, and the initial value and the final value of the speed are all zero. The five-order polynomial coefficients of the two control inputs obtained by fitting are respectively:

$$\boldsymbol{u}_1 = [0.0002,\quad 0.0089,\quad -0.1030,\quad 0.3584,\quad -0.4557]^T$$

$$\boldsymbol{u}_2 = [-0.0001,\quad -0.0022,\quad 0.0265,\quad -0.0964,\quad 0.1321]^T$$

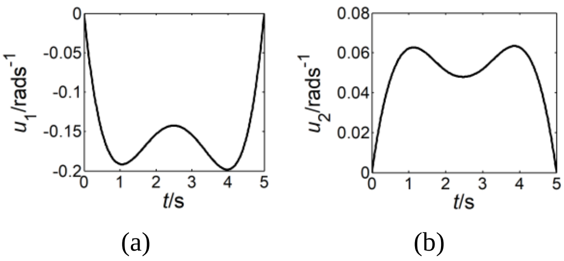

Figure 3. Optimal control input law of robot motion

6. Conclusions

In order to solve the robot motion planning with nonholonomic constraint, an optimal motion planning numerical method based on curve fitting method and particle swarm optimization algorithm is proposed in this study. The example simulation shows that through the method, the attitude of the robot system can be changed only by controlling the leg of robot leg and the motion of the leg can be ensured to set to desired configuration. This method has strong practical application value. In particular, when the motor controlling the motion of the robot body fails under some extreme conditions, such as space exploration and seabed rescue which are not conducive to human direct participation, the method can complete predetermined work by changing attitude motion of robot through controlling the motion of the leg. In addition, the initial value and the final value of the optimal control input of the robot motion planning are all zero by using the curve fitting method. Compared with the case where the initial value and the final value of the control input of the Fourier basis function method are not zero, it is more favorable for engineering application. This method also provides a new way to solve the motion control of other nonholonomic systems.

Acknowledgement

This work was supported by Zibo science and technology development plan (2016kj010057).

References

Bingül Z., Karahan O. (2011). Dynamic identification of staubli rx-60 robot using pso and ls methods. *Expert Systems with Applications*, Vol. 38, No. 4, pp. 4136-4149. http://dx.doi.org/10.1016/10.1016/j.eswa.2010.09.076

Chen L. B., Zheng Y. Q. (2012). Study on curve fitting based on least square method. *Journal of Wuxi Institute of Technology*, Vol. 11, No. 5, pp. 52-55. http://doi.org/10.3969/j.issn.1671-7880.2012.05.017

Hyon S. H., Emura T., Mita T. (2003). Dynamics-based control of a one-legged hopping robot. *Proceedings of the Institution of Mechanical Engineers Part I Journal of Systems & Control Engineering,* Vol. 217, No. 2, pp. 83-98. http://dx.doi.org/10.1177/095965180321700203

Janiak M., Tchoń K. (2011). Constrained motion planning of nonholonomic systems. *Systems & Control Letters*, Vol. 60, No. 8, pp. 625-631. http://dx.doi.org/10.1016/j.sysconle.2011.04.022

Liu M., Zhong X., Wang X., Chen F., Zha F., Guo W. (2016). Motion control for a single-legged robot. *International Conference on Advanced Robotics and Mechatronics*, pp. 336-341. http://dx.doi.org/10.1109/ICARM.2016.7606942

Naik K. G., Mehrandezh M., Barden J.M. (2006). Control of a One-legged hopping robot using a hybrid neuro-PD controller. *Canadian Conference on Electrical and Computer Engineering*, pp. 1530-1533. http://dx.doi.org/10.1109/CCECE.2006.277578

Poli R., Kennedy J., Blackwell T. (2007). Particle swarm optimization. *Swarm Intelligence*, Vol. 1, No. 1, pp. 33-57. http://dx.doi.org/10.1007/s11721-007-0002-0

Raibert M. H., Tello E. R. (2007). Legged robots that balance. *IEEE Expert*, Vol. 1, No. 4, pp. 89-89. http://dx.doi.org/10.1109/MEX.1986.4307016

Shabestari S. S., Emami M. R. (2016). Gait planning for a hopping robot. *Robotica*, Vol. 34, No. 8, pp. 1822-1840. http://dx.doi.org/10.1017/S0263574714002598

Su P., He G. P., Xu M. (2012). Research on motion simulation of hopping robot based on minimum energy-loss principle. *Machinery Design & Manufacture*, No. 4, pp. 171-173. http://doi.org/10.3969/j.issn.1001-3997.2012.04.064

Venter G., Sobieszczanski-Sobieski J. (2003). Particle swarm optimization. *AIAA Journal*, Vol. 41, No. 8, pp. 1583-1589. http://dx.doi.org/10.2514/2.2111

Wu J. W., Shi S., Liu H., Cai H. (2011). Spacecraft attitude disturbance optimization of space robot in target capturing process. *Robot*, Vol. 33, No. 1, pp. 16-21. http://doi.org/10.3724/SP.J.1218.2011.00016

Yang L. L. (2013). The numerical method to nonholonomic motion planning of a hopping robot. *Manufacturing Automation*, Vol. 35, No. 13, pp. 86-89. http://doi.org/10.3969/j.issn.1009-0134.2013.13.026