Path-tracking Control of Bevel-Tip Needles Using Model Predictive Control

Benyan Huo
Shenyang Institute of Automation,
Chinese Academy of Sciences
University of Chinese Academy of Sciences
Shenyang, China
huobenyan@sia.cn

Xingang Zhao, Jianda Han
Shenyang Institute of Automation,
Chinese Academy of Sciences
Shenyang, China
zhaoxingang@sia.cn, jdhan@sia.cn

Weiliang Xu
Department of Mechanical Engineering,
University of Auckland
Auckland,
New Zealand
p.xu@auckland.ac.nz

***Abstract*—The bevel-tip flexible needle is an improvement of the clinical puncture needle and has the potential to reduce surgery traumas and improve puncture accuracy. Due to the nonholonomic kinematics model, it is difficult to drive flexible needles to track a pre-plan path. In this paper, we propose an effective path tracking control method based on Model Predictive Control (MPC). The kinematics model of flexible needles is described firstly, according to which the control method is designed. Then, the path tracking control method is presented in detail. The finite prediction horizon of MPC is extended to infinite in order to obtain a global optimal solution. Due to the nonlinear model, the objective function of receding horizon optimization is also nonlinear, and Particle Swarm Optimization (PSO) is employed to search optimal predictive control series. At last, simulations are conducted and results illustrate that the proposed controller performs well to precisely drive the flexible needle to track a desired path.**

Keywords—bevel-tip needle; path-tracking; MPC; PSO

I. INTRODUCTION

Percutaneous punctures are the least invasive surgeries to reach a nidus requiring diagnose or cure and are widely performed in biopsy, injection, radiofrequency ablation, drug delivery and so on. The clinical efficacy of these surgeries depend heavily on the precision that the puncture needle reaches the desired nidus position. However, there are some factors that drive the needle tip to deviate from the desired position, such as tissue inhomogeneity, tissue deformation and surgeon's hand-eye incoordination and fatigue. What's worse, being rigid, the clinical puncture needle is lack of controllability. When the deviation occurs, the only way to correct it is adjusting the needle tail, which will increase the surgery trauma. In order to improve the controllability of puncture needles, a bevel-tip flexible needle which has a prominent bevel tip and is made up of elastic alloy material has been proposed [1]. The bevel-tip needle is more flexible and controllable. Thus, the puncture accuracy is improved via precise planning and controlling, and nidus blocked by bone, vessel or other sensitive structures can be reached by the flexible needle.

To drive the flexible needle precisely, dynamics models and kinematics models of the flexible needle have been developed. Dynamics models focus on the needle tip cutting force analysis and the tip deflection modeling. The relationship between the cutting force and the deflection are set up. Because the needle tip is too tiny to mount a force sensor, the cutting force has to be obtained indirectly via two methods. One method is modeling the cutting force directly [2, 3]. Another method is minus the needle body friction force which can be modeled from the whole needle suffered force measured by a force sensor mounted on the needle tail [4, 5]. The tip deflection models are mainly based on the cantilever beam model [6, 7], segmented cantilever beam model[8] and spring-beam-damping model [9]. However, the dynamics models are very complex and remain in theoretical analysis. The kinematics model, bicycle model [1], is established by analyzing the needle tip motion and the model parameters are recognized via experiment datum. And Wooram et al. [10] went further to simplify the model by decreasing the model parameters to one. Contrasting with the complex dynamics model, kinematics models are widely employed in puncture path planning methods and driving methods.

In order to driving the flexible needle to an expected target, the problem that how to drive needles to a desired plane was researched. A sub-model extracted from the kinematics model was linearized via the back-stepping method, a linear full-state feedback controller [11] and a nonlinear adaptive output-feedback controller [12] were designed. But the problem that driving needles to a target was not solved completely until that the sliding mode control [13] and reachable decision control [14] were employed. When performing percutaneous punctures, flexible needles should avoid obstacles such as vessels, nerves etc. However, the obstacle avoidance is not taken into account in above methods. In order to deal with obstacle avoidance, the path-planning problem has been researched. Alterovitz et al. [15] turned the path planning in 2D plane into nonlinear optimization problem via random markov process and got the optimized puncture path by minimizing the object function. The path planning problem in 3D was solved directly by employing rapidly exploring random tree [16, 17] and multi-objective particle swarm optimization [18]. Once a puncture path is determined, path-tracking control is needed to execute the desired path safely and effectively. This article focuses on the path tracking

This work was supported in part by the National Natural Science Foundation of China(61273356).

978-1-4799-8464-0/16/$31.00 ©2016 IEEE

control and proposes a method based on model predictive control (MPC).

MPC was proposed in 1987 firstly and is a popular control technique used in industry. However, it is appropriate for path tracking control due to two reasons. Firstly, the MPC method is suitable to the path tracking control where the future reference is provided from the desired path. Secondly, the MPC method can easily deal with the nonholonomic constraints due to its model-based design. Due to these advantages, the MPC method for path tracking control has been studied. In this article, an infinite prediction horizon MPC method is presented to drive flexible needles to track a desired path.

The rest of this paper is organized as follows: Section II presents the flexible needle kinematics model. Then, the path tracking method is established in section III. In order to get the global optimal control series, the finite-horizon optimization is extended to infinite-horizon. Section IV demonstrates simulations to verify the performance of methods proposed. The simulation results are presented and analyzed. Finally, in Section V, we conclude our works.

II. Kinematics Model of Flexible Needles

In order to design a path-tracking controller based on the MPC method, a predictive model is essential. The kinematics model of flexible needles is presented in the following.

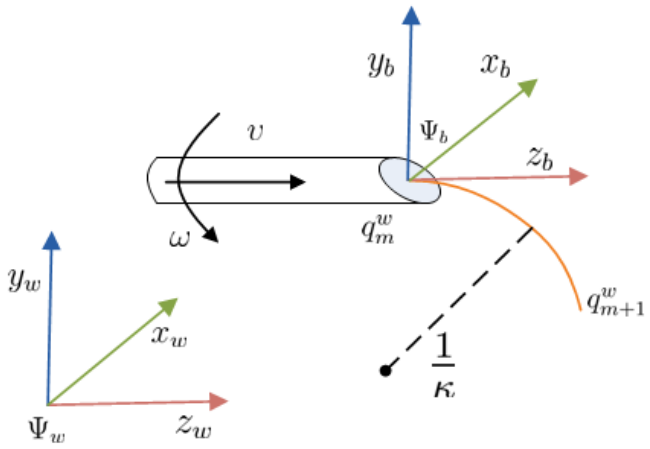


Fig. 1. The kinematics model of flexible needles

Under the following assumptions, the flexible needle will follow an arc when driven in tissue. The assumptions are:

1). Tissue is homogeneous and undeformed.

2). The flexible needle is stiffness in axial direction. Thus, the tip orientation follows the needle tail exactly.

3). The needle body follows the tip exactly, and the needle position will not change in radial direction after inserted into tissue.

As shown in Fig. 1, the curvature of the arc followed by flexible needles is κ, and the position and attitude of the needle tip determine the arc orientation. At the needle tail, the insertion speed is υ and the rotational speed is ω. The tip position and attitude in the world coordinate system Ψ_w is $\boldsymbol{q^w} = \begin{bmatrix} x & y & z & \alpha & \beta & \gamma \end{bmatrix}$, α, β, γ are the attitude angles. Thus, the needle tip velocity in Ψ_w is

$$\boldsymbol{V^w} = \dot{\boldsymbol{q}}^{\boldsymbol{w}}.$$

Due to the needle following an arc, the needle tip velocity in its body coordinate system Ψ_b is $\boldsymbol{V^b} = \upsilon \boldsymbol{V_1} + \omega \boldsymbol{V_2}$. Here, $\boldsymbol{V_1} = \begin{bmatrix} 0 & 0 & 1 & \kappa & 0 & 0 \end{bmatrix}^T$ and $\boldsymbol{V_2} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T$.

The translation of the needle tip in the world coordinate Ψ_w system and the body coordinate system Ψ_b is

$$\boldsymbol{V^b} = \boldsymbol{J}\boldsymbol{V^w}.$$

Here, the translation matrix is $\boldsymbol{J} = \begin{bmatrix} \boldsymbol{R}_{wb}^T & 0_{3\times3} \\ 0_{3\times3} & \boldsymbol{R}_\Omega \end{bmatrix}$, $\boldsymbol{R_{wb}^T}$ is the rotation matrix between Ψ_b and Ψ_w, $\boldsymbol{R}_\Omega = \begin{bmatrix} \cos\beta\cos\gamma & \sin\gamma & 0 \\ -\cos\beta\sin\gamma & \cos\gamma & 0 \\ \sin\beta & 0 & 1 \end{bmatrix}$ is the coefficient matrix of Euler kinematics equations of rigid body motion.

Then, we get the kinematics model of flexible needles:

$$\dot{\boldsymbol{q}} = \boldsymbol{A}(\boldsymbol{q}) \begin{bmatrix} \upsilon \\ \omega \end{bmatrix} = \begin{bmatrix} \sin\beta & 0 \\ -\cos\beta\sin\alpha & 0 \\ \cos\alpha\cos\beta & 0 \\ \kappa\cos\gamma\sec\beta & 0 \\ \kappa\sin\gamma & 0 \\ -\kappa\cos\gamma\tan\beta & 1 \end{bmatrix} \begin{bmatrix} \upsilon \\ \omega \end{bmatrix}. \quad (1)$$

Here, the attitude angle β is not equal $n\pi + \frac{\pi}{2}, n \in \mathbb{N}$ to keep the kinematics model nonsingular.

As shown in (1), the kinematics model is strongly nonlinear and the model states are coupling with the inputs. The model is also nonholonomic because the needle tip position cannot be integrated from the velocity directly. It is an impossible mission to linearize the model or to transform it to standard state space equations. Therefore, we discretize (1) as following:

$$\boldsymbol{q}_{m+1} = \boldsymbol{q}_m + \boldsymbol{A}(\boldsymbol{q}_m) \begin{bmatrix} \upsilon_m \\ \omega_m \end{bmatrix} * \Delta T. \quad (2)$$

Here, $m = 0, 1, 2, \cdots$ is the time step. To reduce the discretization error, the sampling time ΔT should be short enough.

As the assumption 3 says, the needle position will not change in radial direction after inserted into tissue. That means states of the flexible needle remain unchanged when the insertion process is stopped. It allows us to pause the insertion process to correct the path tracking error or to ensure the insertion is safe enough without worrying about the real-time problem.

III. Design of Path Tracking controller

The MPC is an important method to solve the path-tracking problem. The control system includes four mainly parts: a desired path, a predictive model, an optimization function and a rolling optimization method. At each control period, the control system generates an optimal control series by employing the optimization method to solve an optimization problem that minimize the error between the predictive path and the desired path. The first control quantity of this series is applied to the controlled plant. This optimization process is executed again at the next control period using the updated states and measurements.

In order to drive flexible needles to follow a desired path, the four mainly parts of the MPC are also needed. Fig. 2 shows the schematic of the MPC control loop. In the following, how the optimization function established is described firstly. The desired path and the predictive model are presented at the same time. Then, the optimization method is proposed to solve the optimization problem.

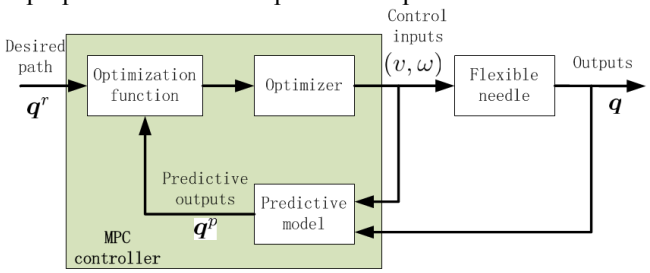


Fig. 2. The schematic of the MPC control loop

A. Optimization function

The predictive model shown in (2) is redefined by a nonlinear state space form as

$$\begin{aligned} &\boldsymbol{q}_{m+1} = f(\boldsymbol{q}_m, \boldsymbol{u}_m) \\ &here\ \boldsymbol{u}_m = \begin{bmatrix} \Delta T v_m & \Delta T \omega_m \end{bmatrix}^T. \\ &subject\ to:\ v_m \geq 0 \end{aligned} \tag{3}$$

Here, v_m, ω_m is the control input at the time $m * \Delta T$. As mentioned above, pausing the insertion process does not effect the insertion trajectory. There may be several pauses in one insertion process. Hence, it is not convenient to describe a insertion trajectory in time domain. In the insertion process, it is not allowed to pull the flexible needle back. It means that the insertion speed v_m should not be negative and the relation between the insertion length and the insertion trajectory is one-to-one correspondence. Thus, the insertion trajectory could be described in the domain of insertion length. The control input is redefined as $\boldsymbol{u}_m = \begin{bmatrix} l_m & \delta_m \end{bmatrix}^T$, m represents the whole insertion length, l_m is the insertion length in the current control period, δ_m is the rotational angle. We should notice that the predictive model is also redefined as $\boldsymbol{q}_{m+l_m} = f(\boldsymbol{q}_m, \boldsymbol{u}_m)$, which is different from the conventional form. If the current insertion length l_m constantly equals to one, the predictive model is the same as (3). The optimization function is based on the predictive model described in the domain of insertion length.

The desired path $\boldsymbol{q}^r$ is a preplan path from an initial point to an expected target, which is generated by path planning methods. In the MPC methods, the error between the desired path $\boldsymbol{q}^r$ and the prediction path $\boldsymbol{q}^p$ will be calculated. There is a question that how to get the error. In general, $\boldsymbol{q}^r$ and $\boldsymbol{q}^p$ are described in time domain and the error is the difference value between $\boldsymbol{q}^r$ and $\boldsymbol{q}^p$ at a given time. However, the predictive model in our article is described in the domain of insertion length. The error is chosen to be the difference value at a given insertion length. Thus, a discrete point in the domain of insertion length is represented as $\boldsymbol{q}^r_{i*\Delta L}$ and simplified as $\boldsymbol{q}^r_i$, here ΔL is the sampling length.

In order to formulate a path-tracking controller, the quadratic optimization function is established, which is used to minimize the path-tracking error and control inputs.

$$\begin{aligned} J(m) =& \boldsymbol{e}^T_{m+N} \boldsymbol{R} \boldsymbol{e}_{m+N} + \sum_{i=1}^{N} \boldsymbol{e}^T_{m+i} \boldsymbol{P} \boldsymbol{e}_{m+i} \\ &+ \sum_{i=0}^{N-1} \boldsymbol{u}^T_{m+i} \boldsymbol{Q} \boldsymbol{u}_{m+i} \end{aligned} \tag{4}$$

Here, N is the number of prediction horizon. $\boldsymbol{e}_{m+N} = \boldsymbol{q}^r_{m+N} - \boldsymbol{q}^p_{m+N}$, $\boldsymbol{e}_{m+i} = \boldsymbol{q}^r_{m+i} - \boldsymbol{q}^p_{m+i}$, $\boldsymbol{q}^r_{m+i}$ is a discrete point on the desired path, $\boldsymbol{q}^p_{m+i}$ is a discrete point on a predictive path. $\boldsymbol{R}, \boldsymbol{P}$ and $\boldsymbol{Q}$ are the diagonal positive definite weighting matrixes which are used for penalizing the predictive error and control input separately. In current work, the control energy is unlimited. Therefore, the weighting matrix $\boldsymbol{Q}$ is set to zero.

To get the global optimal control inputs and guarantee the path-tracking controller stable, we extend the prediction and control horizons from finite to infinite. In practical, the length of prediction and control horizons strongly influences the computation time to optimize the control inputs at each sample. Fortunately, the clinical puncture process does not have a strong requirement for real-time performance. Since the length of the desired path is fixed, the length of prediction and control horizons is gradually shortened with the puncture being executed; meanwhile, the computation time is also shortened.

To reduce the model discretization error, the sampling length ΔL should be short enough. Thus, the predictive control series has a short control period. Generally, the first control input of the predictive control series is applied. It is unreasonable to the path-tracking controller of flexible needles due to the difficulty of getting the position feedback of the needle tip. This is because the needle tip is too tiny to mount a position sensor. In clinical, the needle tip position is measured by computed tomography, which is not a real-time device and emits electromagnetic radiation. In order to reduce the scanning times, front N_c control

inputs are applied. We define N_c as executive horizon. Obviously, a large N_c will increase the path-tracking error. A compromise is needed between the tracking accuracy and the scanning times.

B. Optimization Method

The predictive model is strongly nonlinear and is also impossible to be linearized properly. Hence, the optimization function shown in (4) only can be solved by a nonlinear optimization method. In this paper, the particle swarm optimization (PSO) method is employed to solve the path-tracking optimization function, though it is a stochastic optimization method and may obtain a suboptimal solution.

The PSO method is proposed in the mid-90s, mainly developed by Dr. Eberhart and Dr. Kenedy[19], which stems from the foraging behavior of birds. It is a stochastic meta-heuristic algorithm that can be applied to optimize a function, which is difficult or impossible to be expressed analytically. A group of randomly particles $\boldsymbol{S} = \{\boldsymbol{s}_1, \boldsymbol{s}_1, \cdots, \boldsymbol{s}_M\}$ are generated in the solution space, M is the number of particles. Each particle $\boldsymbol{s}_i$ represents a feasible solution. The optimization procedure of PSO can be expressed mathematically by the following equations which specify the velocity and position update process of a particle i.

$$\begin{aligned} \boldsymbol{u}_i^{k+1} &= \mathrm{w}\boldsymbol{u}_i^k + c_1 r_1(\boldsymbol{s}_i^b - \boldsymbol{s}_i^k) + c_1 r_1(\boldsymbol{s}_g^b - \boldsymbol{s}_i^k) \\ \boldsymbol{s}_i^{k+1} &= \boldsymbol{s}_i^k + \boldsymbol{u}_i^{k+1} \end{aligned} \tag{5}$$

Here, $\boldsymbol{u}_i^k$ and $\boldsymbol{s}_i^k$ are the velocity and position of the particle i at iteration k. w is an iterative weight coefficient, c_1, c_2 are acceleration coefficients, r_1, r_2 are random numbers between 0 and 1, $\boldsymbol{s}_i^b$ is the best position that the particle i has reached, $\boldsymbol{s}_g^b$ is the best position of the whole particles have reached.

There is an important problem that how to represent a feasible solution, namely a series of control inputs, with a particle. An obvious method is directly establishing particles with the control inputs series.

$$\boldsymbol{s}_i = \{(l_i^m, \delta_i^m), (l_i^{m+1}, \delta_i^{m+1}), \cdots, (l_i^{m+n}, \delta_i^{m+n})\} \tag{6}$$

Here, m is the current insertion step, $m * \Delta L$ represents the whole insertion length till the current step , actually. δ_i^m is the rotational angle in m step. l_i^m is the insertion length in m step, and $l_i^m = l_i^{m+1} = \cdots = l_i^N = \Delta L$. $n * \Delta L$ represents the remaining length from the current step to the end of a desired path.

Due to the small value of the sampling length ΔL, the particle $\boldsymbol{s}_i$ will have a huge number of dimensions, especially at the beginning of a puncture. This will increase the computational complexity and prolong the calculation time to an unacceptable level. From the flexible kinematics model shown in (1), we know that in one control cycle the value change of the needle tip position and attitude, except for the attitude angle γ, is very tiny no matter how much the input of rotational angle is. That means if we allow a small tracking error to exist, there is a lot of rotational angles whose value is zero or close to zero in the predictive control series. Thus, the dimension of particles can be reduced by merging adjacent control inputs whose rotational angle are zero or close to zero. The particle $\boldsymbol{s}_i$ is redefined as

$$\boldsymbol{s}_i = \{(L_i^1, \Theta_i^1), (L_i^2, \Theta_i^2), \cdots, (L_i^{N_s}, \Theta_i^{N_s})\}, \tag{7}$$

where, N_s is the control series length after merged. L_i^k, Θ_i^k are the control inputs after merged. It should be noted that merging control inputs is just a technique to solve the optimization problem conveniently and efficiently. The merged control inputs should be decomposed before being applied. N_s is not a constant value, and is affected by the flexible needle model parameter and the whole length of the desired path.

The procedure of solving the optimization function via PSO is as following:

Step 1. Initializing particles $\boldsymbol{S}$, generating the particles position and velocity randomly, setting the maximum number of iteration I_{max} and the termination condition of PSO.

Step 2. Calculating particles' adaptive values via (4).

Step 3. Updating the personal best position $\boldsymbol{s}_i^b$ and the global best position $\boldsymbol{s}_g^b$.

Step 4. Checking whether the termination condition has been reached. If reached, terminating the algorithm. Otherwise, going to Step 5.

Step 5. Updating every particle's velocity and position via (5). And going to Step 2.

IV. Simulations

The path-tracking controller base on MPC is validated by simulations. Firstly, the tracking ability of the controller is validated via tracking a path whose curvature is not constant. Then, the performance of path-tracking controller is verified and analyzed when the measurement and process noise exists.

A. Tracking Ability Validation

In order to verify the tracking ability of the proposed controller, we design a desired path whose curvature changes along with the change of insertion length. The parameters are set as following. The sampling length ΔL is $1mm$. The whole insertion length is $1000mm$. The insertion length l_m and rotational angle δ_m are constant $1mm$ and $0.05°$ individually. The initial value of curvature κ_m, namely the model parameters, is $1/300mm^{-1}$ and changes in the following manner

$$\kappa_m = \frac{1}{300 - 0.3m}, m \in [1, 1000).$$

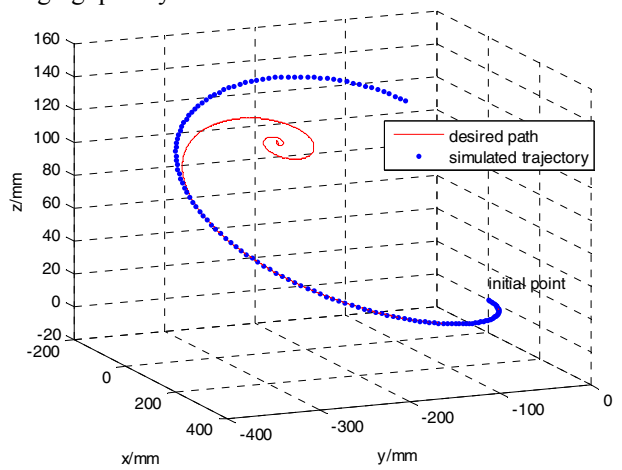
The desired path is generated by the flexible model as shown in (3). The model parameter used in predictive model is

constant and set as $\kappa_p = 1/150mm^{-1}$.

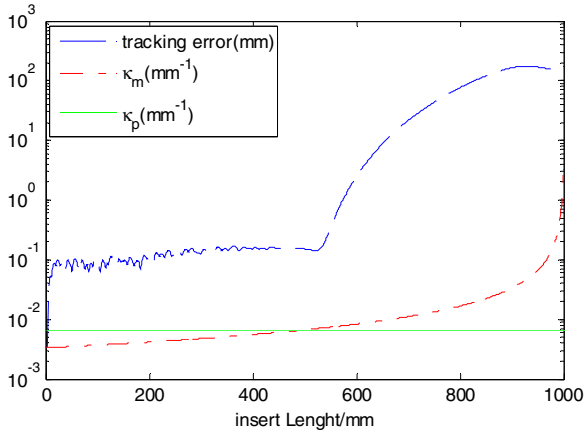
Due to the change of curvature, it is difficult to obtain a global optimal solution. Hence, the predictive horizon is set as $25mm$. The executive horizon is $2mm$.

The parameters of PSO are set as following : the number of particle is $M = 50$, the dimension of particle is $N_s = 10$, the iteration times is $I_{max} = 500$, the iterative weight coefficient is variable, $\mathrm{w} = 1.5 - 0.02i$, i is the iterative time, $c_1 = 2, c_2 = 1$.

The simulation result shown in Fig. 3 indicates that the path-tracking controller drives the flexible needle to track the desired path precisely when the curvature of the desired path is less than the curvature of the flexible needle model. The tracking error is around 0.1mm. Otherwise, the tracking error is diverging quickly.



(a) The desired path and simulated trajectory in 3D



(b) Path-tracking error. The tracking error is around $0.1mm$ when $\kappa_p > \kappa_m$, and diverges when $\kappa_p < \kappa_m$.

Fig. 3. Simulation to verify the tracking ability of the proposed controller. The tracking error is diverging when the curvature of desired path exceeds the curvature of the flexible needle model.

In our simulations, the curvature of the flexible needle model is known exactly, which is impossible in reality due to the inhomogeneity and elasticity of biological tissue. Thus, the value of the predictive model curvature is larger than the curvature recognized from experiment datum to ensure that the desired path is tracked precisely. In the simulation, we do not take into account the measurement noise and process noise, which will heavily affect the tracking accuracy.

B. Path-tracking Control with Noise

In order to verify the performance of the proposed controller when the insertion process is affected by the process and measurement noise, we employ them in our simulations. The parameters are set as above. The desired path curvature is $\kappa_m = 1/300mm^{-1}$, while the predictive model curvature is $\kappa_p = 1/250mm^{-}1$, is larger than the desired path curvature to ensure the tracking accuracy. A path planning method[18] generates the desired path, the initial point of which is $[0, 0, 0]$ and the target point of which is $[5, 10, 100]mm$. From the flexible model shown in (1), we know that the change of needle tip attitude in one control cycle is less than $\kappa_p * \Delta L = 0.004$ radian. Hence, the standard deviation of process noise is set as 0.001 radian, about a quarter of the maximum change value of attitude. The standard deviation of measurement noise is $0.15mm$. The path-tracking trajectory is shown in Fig. 4, in which the desired path and the simulated trajectory is almost the same. Fig. 5 shows the tracking result in XZ plane and YZ plane. Affected by the measurement noise, the trajectory of needle tip drawn in solid line is not smooth. The tracking error is shown in Fig. 6. Affected by noise, the tracking error is enlarged, but is still acceptable. The simulation results indicate that the proposed path-tracking controller performs well in the existence of process and measurement noise.

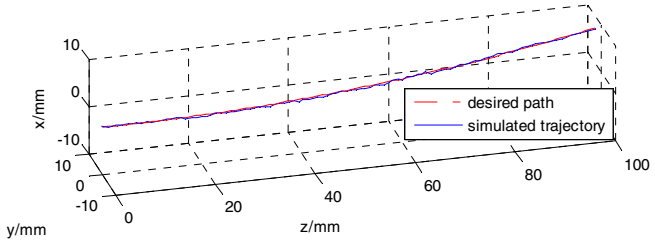


Fig. 4. The result of the path tracking control with process and measurement noise

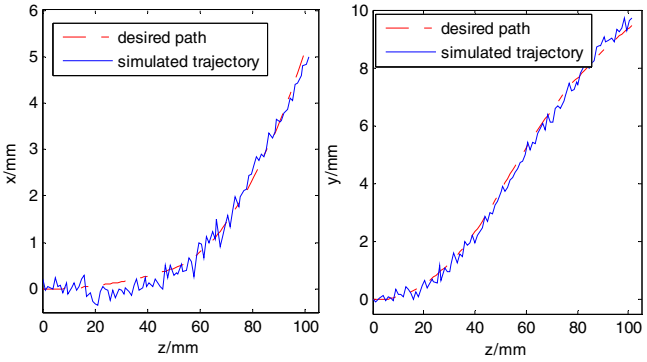


Fig. 5. The desired path and simulated trajectory in XZ plane and YZ plane

V. CONCLUSION

In this article, we focus on the path-tracking problem of the flexible needle. A path-tracking controller based on the MPC method is proposed. In order to calculate the path-tracking error conveniently and effectively, we translate

the flexible model from time domain to insertion length domain. The optimization function is established and the particle swarm optimization is employed to solve it. The predictive horizon of the MPC method is extended to infinite horizon on the intention to obtain a globally optimal solution. In the end, the proposed controller is verified by simulations. We validate the tracking ability by tracking a desired path whose curvature value is variable. In addition, the path tracking simulations with process and measurement noise are performed. The simulation results show that the proposed path-tracking controller can precisely track a desired path whose curvature does not exceed the curvature of the predictive model.

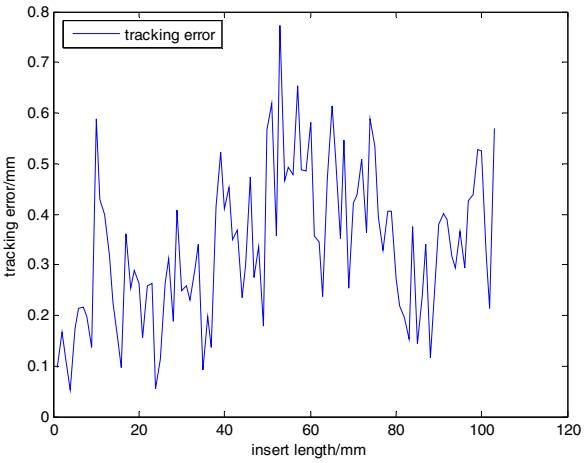


Fig. 6. The path tracking error when the process and measurement noise exists.

REFERENCES

[1] R. Webster, N. Cowan, G. Chirikjian, and A. Okamura, 'Nonholonomic Modeling of Needle Steering', ed. by Marcelo Ang and Oussama KhatibSpringer Berlin / Heidelberg, 2006, pp. 35-44.

[2] S. Misra, KB Reed, BW Schafer, KT Ramesh, and AM Okamura, 'Mechanics of Flexible Needles Robotically Steered through Soft Tissue', *The International Journal of Robotics Research,* 29 (2010), 1640-60.

[3] D. Wei, H. Han, and Z.J. Du, 'The Tip Interface Mechanics Modeling of a Bevel-Tip Flexible Needle Insertion', in *International Conference on Mechatronics and Automation (ICMA)* (Chengdu, China, 2012), pp. 581-86.

[4] R. J. Roesthuis, Y. R. J. van Veen, A. Jahya, and S. Misra, 'Mechanics of Needle-Tissue Interaction', in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, 2011, pp. 2557-63.

[5] A. Asadian, R. V. Patel, and M. R. Kermani, 'A Distributed Model for Needle-Tissue Friction in Percutaneous Interventions', in *IEEE International Conference on Robotics and Automation (ICRA)* (Shanghai, 2011), pp. 1896-901.

[6] A. Asadian, M. R. Kermani, and R. V. Patel, 'An Analytical Model for Deflection of Flexible Needles During Needle Insertion', in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (San Francisco, 2011), pp. 2551-56.

[7] M. Abayazid, R. J. Roesthuis, R. Reilink, and S. Misra, 'Integrating Deflection Models and Image Feedback for Real-Time Flexible Needle Steering', *IEEE Transactions on Robotics,* 29 (2013), 542-53.

[8] A. Haddadi, and K. Hashtrudi-Zaad, 'Development of a Dynamic Model for Bevel-Tip Flexible Needle Insertion into Soft Tissues', in *Engineering in Medicine and Biology Society,EMBC, 2011 Annual International Conference of the IEEE*, 2011), pp. 7478-82.

[9] K.G. Yan, T. Podder, Y. Yan, I. Liu Tien, C. W. S. Cheng, and Ng Wansing, 'Flexible Needle-Tissue Interaction Modeling with Depth-Varying Mean Parameter: Preliminary Study', *Biomedical Engineering, IEEE Transactions on,* 56 (2009), 255-62.

[10] P. Wooram, K. Jin Seob, Z. Yu, N. J. Cowan, A. M. Okamura, and G. S. Chirikjian, 'Diffusion-Based Motion Planning for a Nonholonomic Flexible Needle Model', in *IEEE International Conference on Robotics and Automation(ICRA)* (Barcelona, Spain, 2005), pp. 4600-05.

[11] V. Kallem, and N. J. Cowan, 'Image Guidance of Flexible Tip-Steerable Needles', *IEEE Transactions on Robotics,* 25 (2009), 191-96.

[12] M. Motaharifar, H. A. Talebi, F. Abdollahi, and A. Afshar, 'Nonlinear Adaptive Output-Feedback Controller Design for Guidance of Flexible Needles', *Mechatronics, IEEE/ASME Transactions on,* 20 (2015), 1912-19.

[13] D. C. Rucker, J. Das, H. B. Gilbert, P. J. Swaney, M. I. Miga, N. Sarkar, and R. J. Webster, 'Sliding Mode Control of Steerable Needles', *IEEE Transactions on Robotics,* PP (2013), 1-11.

[14] B. Y. Huo, X. G. Zhao, J. D. Han, and W. L. Xu, 'A Method of Flexible Needle Puncture Control Based on Reachable Decision', *Control Theory & Applications,* 31 (2014), 1423-30.

[15] R. Alterovitz, K. Goldberg, and A. Okamura, 'Planning for Steerable Bevel-Tip Needle Insertion through 2d Soft Tissue with Obstacles', in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, 2005, pp. 1640-45.

[16] J. Xu, D. Vincent, A. Ron, and G. Ken, 'Motion Planning for Steerable Needles in 3d Environments with Obstacles Using Rapidly-Exploring Random Trees and Backchaining', in *Automation Science and Engineering, 2008. CASE 2008. IEEE International Conference on*, 2008, pp. 41-46.

[17] S. Patil, J. Burgner, R. J. Webster, and R. Alterovitz, 'Needle Steering in 3-D Via Rapid Replanning', *Robotics, IEEE Transactions on,* 30(2014), 853-64.

[18] B. Y. Huo, X. G. Zhao, J. D. Han, and W. L. Xu, 'Puncture Path Planning for Bevel-Tip Flexible Needle Based on Multi-Objective Particle Swarm Optimization Algorithm', *Jiqiren/Robot,* 37(2015), 385-94.

[19] R. Eberhart, and J. Kennedy, 'A New Optimizer Using Particle Swarm Theory', in *Micro Machine and Human Science, 1995. MHS '95., Proceedings of the Sixth International Symposium on*, 1995, pp. 39-43.