2009 IEEE International Conference on Robotics and Automation, Kobe, Japan, May, 2009.

Robust Hitting with Dynamics Shaping

Tasuku Yamawaki, Yusuke Tsuzuki and Masahito Yashima

***Abstract*— The present paper proposes the motion planning based on "the dynamics shaping" for a robotic arm to hit the target robustly toward the desired direction, of which the concept is to shape the robot dynamics appropriately in order to accomplish the desired motion. According to the linear system theory, the positional error of the end-point converges onto near the singular vector corresponding to its maximum singular value of the output controllability matrix of the robotic arm. Therefore, if we can control the direction of the singular vector by applying the dynamics shaping, we will be able to control the direction of the positional error of the end-effector caused by the disturbance. We propose a novel motion planning based on the dynamics shaping and verify numerically and experimentally that the robotic arm can robustly hit the target toward the desired direction with a simple open-loop control system even though the disturbance is applied.**

I. INTRODUCTION

In general, we cannot obtain exact models of real robotic systems, because the real robotic systems include various uncertainties. One of the uncertainties is noise and disturbance which are excluded from the nominal models. For example, sensor noise and quantization error of the angular velocity are applied to the robot as disturbance. To cope with the disturbance, many researchers have focused on the controller design such as the robust control [2]. However, the controller design is very complicated.

In contrast, the present paper proposes the novel approach called "*the dynamics shaping*", which is to shape the dynamics of a robot appropriately in order to accomplish the desired motion. The motion planning based on the dynamics shaping is effectively making use of the robot dynamics so as to accomplish the task robustly even though a controller is very simple. Therefore we focus on the motion planning rather than the controller design. Our approach is to plan the motion which reduces the influence of the disturbance on the task by shaping the robot dynamics appropriately. To our knowledge, while many motion planning approaches for robotic arms have been proposed [1], [10], [11], none of these schemes based on the dynamics shaping have been addressed in order to constrain the disturbance [13], [14].

In several tasks, the generation direction of the positional error is important. One of such tasks is robust hitting of a stationary target by the robotic arm as shown in Fig. 1. If the end-effector is deviated vertically from the desired path by disturbance, the robot fails to hit the target as described in Fig. 1(a). If the direction of the positional error can be controlled to be tangential to the desired path as shown in Fig. 1(b), the robotic arm can hit the target even though the disturbance is applied. We have revealed that the hitting motion based on the dynamics shaping could achieve high robustness by gathering the positional error onto near the desired path [14]. However, its motion planning does not take into consideration hitting the target in the desired direction, which is the important performance for the hitting motion. Thus, the present paper aims to hit the target robustly toward an arbitrary desired direction by utilizing the dynamics shaping.

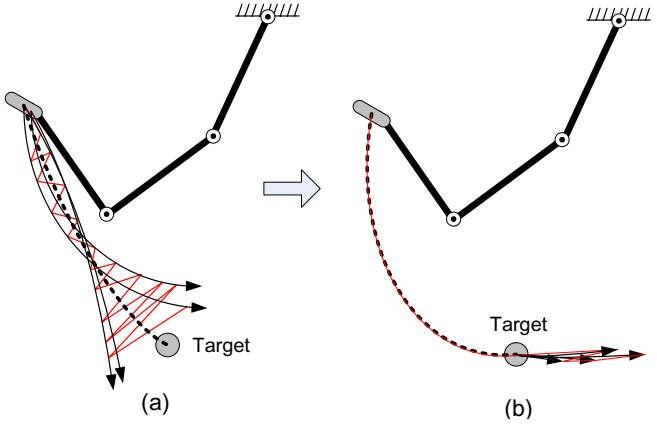

Fig. 1. Robust hitting of a stationary target by the robotic arm with "Dynamics Shaping."

To verify the validity of our motion planning, we conduct two simulations and one experiment. According to the linear systems theory, section II shows that the end-effector position error converges onto near the singular vector of the output controllability matrix of the robotic arm system. Section III proposes the motion planning based on the dynamics shaping which enables the robotic arm to hit the target in the desired direction. Section IV applies the proposed motion planning to the three-revolute-joint robotic arm and verifies numerically and experimentally that the robotic arm can robustly hit the target toward the desired direction with a simple open-loop control system even though the disturbance is applied.

II. GENERATING MECHANISM OF END-EFFECTOR POSITION ERROR

A. Formulation based on Output Controllability

A n degree-of-freedom robotic arm is a nonlinear system, which is expressed by the following equation of motion:

$$\boldsymbol{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \boldsymbol{g}(\boldsymbol{\theta}) = \boldsymbol{\tau} \tag{1}$$

where $\boldsymbol{\theta} \in \Re^n$ is the joint position, $\boldsymbol{\tau} \in \Re^n$ is the joint torque, $\boldsymbol{M}(\boldsymbol{\theta}) \in \Re^{n\times n}$ is the inertia matrix, $\boldsymbol{h}(\boldsymbol{\theta},\ \dot{\boldsymbol{\theta}}) \in \Re^n$

The authors are with the Department of Mechanical Systems Engineering, National Defense Academy of Japan, 1-10-20, Hashirimizu, Yokosuka, JAPAN {yamawaki, yashima}@nda.ac.jp

is the centrifugal force and the Coriolis force, and $\boldsymbol{g}(\boldsymbol{\theta}) \in \Re^n$ is the gravitational force acting on the robotic arm.

To apply the linear systems theory [3], [5], we deal with the linearized model of the robotic arm, of which the input and the output are the joint torque $\boldsymbol{\tau}$ and the Cartesian coordinate position of the end-effector described as $\boldsymbol{p} \in \Re^m$ $(m \leq 3)$, respectively. A system is said to be output controllable if it is possible to construct control inputs $\widehat{\boldsymbol{\tau}}$ that will transfer any given initial output $\boldsymbol{p}(0)$ to any final output $\boldsymbol{p}(t_f)$ until a finite time interval $0 \leq t \leq t_f$ [8], [9]. The output controllability matrix $\boldsymbol{N}$ of the robotic arm can be described as

$$\boldsymbol{N} = \boldsymbol{J}\Big[\ \boldsymbol{0} \quad \boldsymbol{M}^{-1} \quad \boldsymbol{0} \quad (-\boldsymbol{M}^{-1}\boldsymbol{G})\boldsymbol{M}^{-1}\cdots \\ \cdots(-\boldsymbol{M}^{-1}\boldsymbol{G})^{2n-1}\boldsymbol{M}^{-1}\Big] \qquad (2)$$

where $\boldsymbol{J} = \partial \boldsymbol{p}(\boldsymbol{\theta})/\partial \boldsymbol{\theta} \in \Re^{m\times n}$ is the Jacobian matrix of the robotic arm, $\boldsymbol{G} = \partial \boldsymbol{g}(\boldsymbol{\theta})/\partial \boldsymbol{\theta} \in \Re^{n\times n}$, $\widehat{\boldsymbol{\tau}} = [\ \widehat{\boldsymbol{\tau}}_1^T,\ \widehat{\boldsymbol{\tau}}_2^T, \ldots,\ \widehat{\boldsymbol{\tau}}_n^T]^T$ and $\widehat{\boldsymbol{\tau}}_i = \int_0^{t_f} q_i(-t)\boldsymbol{\tau}(t)dt$.

The maximum amplifying ratio of the input $\widehat{\boldsymbol{\tau}}$ to the output $\boldsymbol{p}$ is given by the maximum singular value σ_{max} of the matrix $\boldsymbol{N}$, which can be written as [12]

$$\max_{\widehat{\boldsymbol{\tau}}\neq 0} \frac{\|\boldsymbol{N}\widehat{\boldsymbol{\tau}}\|}{\|\widehat{\boldsymbol{\tau}}\|} = \sigma_{max} \qquad (3)$$

The singular vector $\boldsymbol{u}_{max}$ corresponding to the maximum singular value σ_{max} indicates its direction. The singular vector and singular value can be easily obtained from the singular value decomposition of the matrix $\boldsymbol{N}$.

Therefore, the singular vector $\boldsymbol{u}_{max}$ indicates the direction in which the effect of the joint torque (input) on the end-effector position (output) is maximized. If there is noise in the input torque, the singular vector $\boldsymbol{u}_{max}$ indicates the direction in which the effect of the noise has the maximum intensity with the end-effector position. Thus if we can control the direction of the singular vector, we will be able to control the direction of the end-effector position error caused by the disturbance.

B. Relationship between Singular Vector and Direction of Position Error Caused by Disturbance

According to the above-mentioned linear systems theory, the end-effector position error caused by the disturbance converges onto near the singular vector corresponding to the maximum singular value. In order to verify the validity, we conduct example simulations using the three-revolute-joint robotic arm which moves within a horizontal plane as shown in Fig. 2. We define the joint torque and the end-effector position $(x,\ y)$ as the input and output variables, respectively. To simplify, we neglect the gravitational effect.

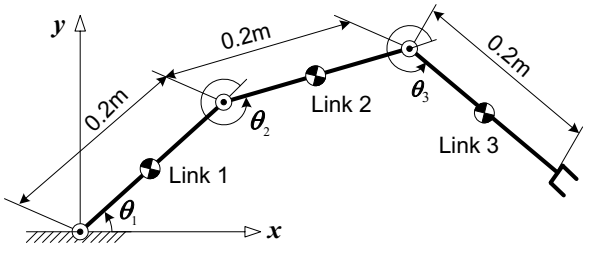

Fig. 2. Three-revolute-joint robotic arm which moves within a horizontal plane. The maximum values of the diagonal element of the inertia matrix $\boldsymbol{M}$ are $(1.880,\ 0.200,\ 0.024)$ kgm^2.

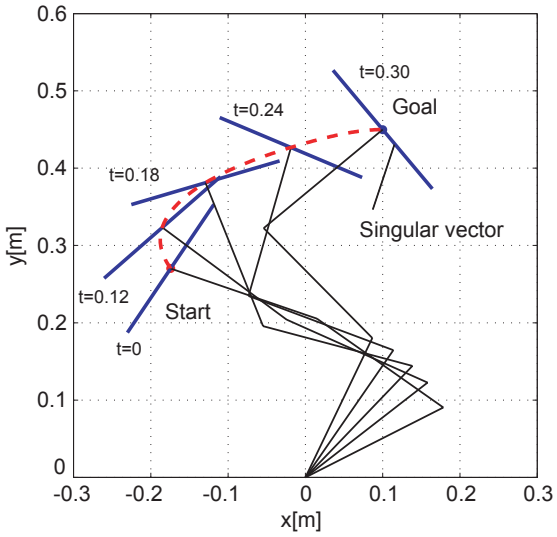

Fig. 3. Singular vectors at each end-effector position

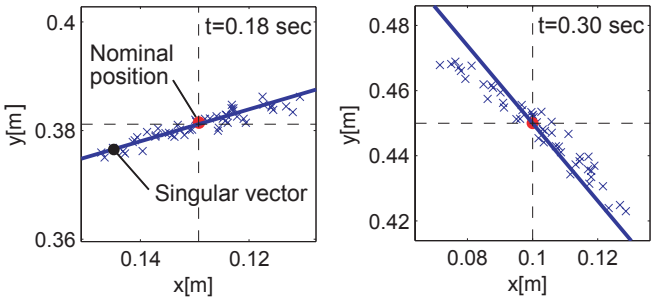

Fig. 4. Position error distribution and singular vector at each nominal end-effector position.

The joint torque to trace the nominal end-effector's trajectory is obtained by using the equation of motion (1) in advance. The input torque is set as the sum of its nominal joint torque and the Gaussian white noise. Applying the input torque to the robotic arm, we obtain the locus of the end-effector. This simulation is repeated 50 times and obtains the variance of the end-effector position error caused by the disturbance. A quintic polynomial is used for the joint trajectory which connects an initial joint angle position with a goal position.

We let the joint vectors at the start and goal be $\boldsymbol{\theta}_0 = (26.85,\ 118.0,\ 16.25)^T$ deg and $\boldsymbol{\theta}_f = (64.34,\ 70.35,\ -95.07)^T$ deg, the total motion time t_f be 0.3 sec, and the variance of the Gaussian white noise be $(4,\ 4,\ 4)$ N^{2}m^2. Fig. 3 shows the nominal arm movement and the singular vectors drawn at each position of the end-effector. Fig. 4 shows the position error distributions denoted by $\times$ at $t = 0.18$ sec and the goal. The center position of

each figure indicates the nominal position of the end-effector at each time. The straight line indicates the direction of the singular vector at each nominal end-effector position.

According to the simulation results, the end-effector constantly converges near onto the singular vector, even though the direction of the singular vector changes with the arm movement. If we can fit the direction of the singular vector to the tangential direction of the nominal path by using the dynamics shaping, we will be able to obtain the robust trajectory such that the end-effector does not deviate greatly from the nominal path even though noise is applied to the input torque.

III. HITTING TRAJECTORY GENERATION BASED ON DYNAMICS SHAPING

This section shows that we can generate the trajectory for the robotic arm to hit robustly the target toward the desired direction by applying the motion planning based on the dynamics shaping which fits the direction of the singular vector to the tangential direction of the hitting path as much as possible.

The hitting trajectory generation is composed of three phases as shown in Fig. 5. Phase 1 finds the hitting configuration whose singular vector points to the desired hitting direction. Next, Phase 2 backwardly searches for the path from the hitting configuration so that the direction of the singular vector can be fitted to the tangential direction of the path. Finally, Phase 3 generates the hitting trajectories of the end-effector position and the joint torque by tracing forwardly the reference path obtained in Phase 2 as much as possible from the initial point to the hitting point.

Phase 1, 2, and 3 can correspond to three phases of a golf swing, which are addressing, takeaway and downswing, respectively. The trajectory generation based on the dynamics shaping is similar to that of a human being. The detail of each phase is described below.

A. Phase 1: Setting of Hitting Configuration

If a robotic arm has redundant degrees of freedom, the direction of the singular vector can be changed by varying the arm configuration without change of the end-effector's position at the hitting point $\boldsymbol{p}_h$ as shown in Fig. 5(a). A randomized algorithm, which has high performance for a global searching, is applied to obtain the arm configuration whose singular vector points to the desired hitting direction $\boldsymbol{u}_t = \dot{\boldsymbol{p}}_h / ||\dot{\boldsymbol{p}}_h||$. The algorithm is given as follows.

1) We select the joint angle θ_1 at random in a specified range and obtain the initial configuration $\boldsymbol{\theta}^{(1)} = (\theta_1, \theta_2, \theta_3)^T$, whose end-effector is located at the hitting point $\boldsymbol{p}_h$. The deviation angle q between its singular vector $\boldsymbol{u}_{max}$ and the desired hitting direction $\boldsymbol{u}_t$ is derived from $q = \cos^{-1}(\boldsymbol{u}_{max}^T \boldsymbol{u}_t)$.
2) We set the infinitesimal relative angle $\Delta\theta_1 \geq 0$ and set the joint angle θ_1 as $\theta_1 = \theta_1 \pm \Delta\theta_1$. For each of θ_1, we obtain the arm configuration so that the position of the end-effector is located at the hitting point $\boldsymbol{p}_h$.
3) We obtain the deviation angle q for each of the arm configuration given in step 2. If the smaller q satisfies $|q| < |q^{(i-1)}|$, where $q^{(i-1)}$ is the deviation angle of the previous search step, we update the arm configuration as $\boldsymbol{\theta}^{(i)}$. If not, we go back to step 1 and update the initial configuration for search.
4) If the deviation angle satisfies $|q^{(i)}| < \epsilon$, we stop searching and set the arm configuration as the hitting configuration.
5) If the iteration number i exceeds a specified upper limit $i > i_{max}$, we terminate searching.
6) We go back to step 2.

B. Phase 2: Generation of Reference Path

Phase 2 generates the reference path of the end-effector position and the joint angle on which the singular vectors point to the tangential direction of the end-effector path, which is used to find a hitting trajectory in Phase 3. The backward search is carried out from the hitting configuration at the hitting point $\boldsymbol{p}_h$ as shown in Fig. 5(b).

We apply similar techniques to the potential-guided path planning [4], [7], [14] to find the reference path. The end-effector is steered toward the direction of the singular vector $\boldsymbol{u}_{max}$ by applying a virtual force $\boldsymbol{f}$ at the end-effector $\boldsymbol{p}$. Such virtual force can be described as

$$\boldsymbol{f} = -k_s \mathrm{sgn}\left(\boldsymbol{u}_{max}^T (\boldsymbol{p}_h - \boldsymbol{p})\right) \boldsymbol{u}_{max} \tag{4}$$

where k_s is gain and $\mathrm{sgn}()$ gives the direction of the virtual force $\boldsymbol{f}$ such that the end-effector can move forward. To accomplish the motion of the end-effector generated by the virtual force $\boldsymbol{f}$, the robotic arm should be actuated by the joint torque $\boldsymbol{\tau}$, which is given by

$$\boldsymbol{\tau} = -\boldsymbol{C}_p \dot{\boldsymbol{\theta}} + \boldsymbol{J}^T \boldsymbol{f} \tag{5}$$

where $\boldsymbol{C}_p$ is the dumping matrix which modulates the fluctuation of the joint angle due to the arm's redundancy. The reference paths of the end-effector and the joint angle, which are generated by the joint torque $\boldsymbol{\tau}$ of (5), are obtained by integrating (1) twice. When the joint angle reaches the limit angle, we terminate the path generation and set its end-effector position as the initial position $\boldsymbol{p}_0$ of the hitting motion.

C. Phase 3: Generation of Hitting Trajectory

Phase 3 generates the hitting trajectory of the joint angle and the joint torque. The forward search is carried out from the initial configuration at the point $\boldsymbol{p}_0$ as shown in Fig. 5(c). The hitting trajectory is divided into two trajectories before and after hitting, which are shown below.

1) Trajectory from Initial Configuration to Hitting Configuration: We generate the hitting trajectory from the initial configuration to the hitting configuration at the point $\boldsymbol{p}_h$, which passes through via points on the reference path of the joint angle obtained in Phase 2 and is represented by a cubic spline function between via points [6]. Before the trajectory generation, we set the three terms shown below.

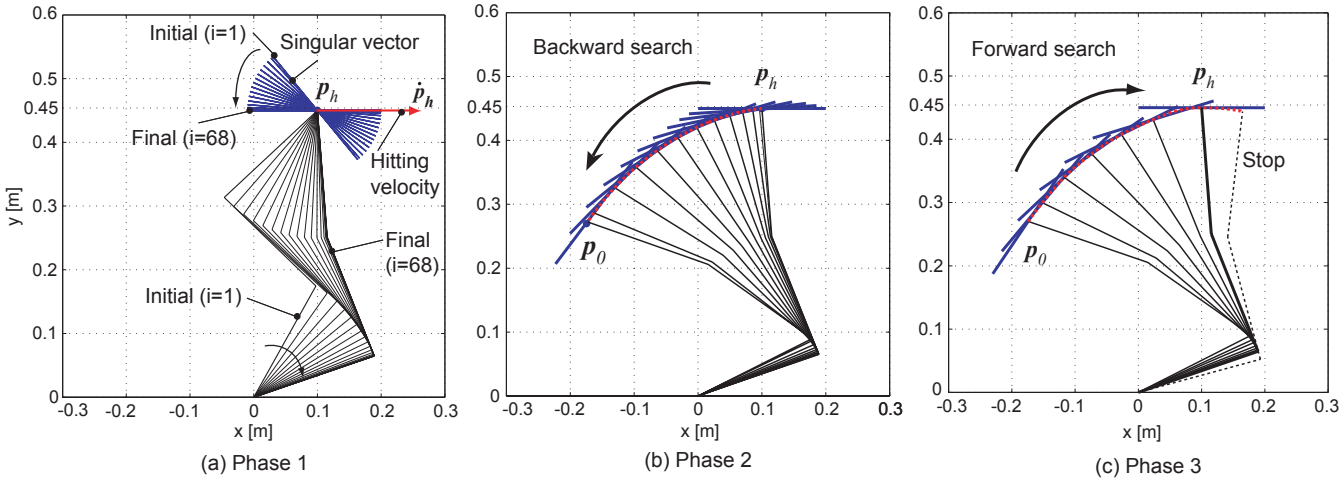

Fig. 5. (a) Phase 1 obtains the hitting configuration. (b) Phase 2 generates the reference path and the initial end-effector position $\boldsymbol{p}_0$. (c) Phase 3 generates the hitting trajectory tracing the via points on the reference path.

a) **Via configurations including the initial point and the final point (hitting point)** : We take the initial configuration at the point $\boldsymbol{p}_0$ and the final configuration at the point (hitting point) $\boldsymbol{p}_h$, respectively. Other via configurations are given to equalize the distance between each via configuration in the task space.

b) **Corresponding traveling time** : We set the corresponding traveling time $t_1 = 0, t_2, \cdots, t_{m-1}, t_m$ by considering that the joint torque does not exceed the maximum torque limit.

c) **Velociy and acceleration of the initial configuration and the hitting configuration** : The initial joint angular velocity is set as zero. The hitting joint angular velocity $\dot{\boldsymbol{\theta}}_h$, which generates the desired hitting velocity $\dot{\boldsymbol{p}}_h$ in the task space, is given by

$$\dot{\boldsymbol{\theta}}_h = \boldsymbol{J}^\sharp \dot{\boldsymbol{p}}_h + \left(\boldsymbol{I} - \boldsymbol{J}^\sharp \boldsymbol{J}\right) \boldsymbol{k} \tag{6}$$

where $\boldsymbol{J}^\sharp$ is the pseudo-inverse of $\boldsymbol{J}$, and $\boldsymbol{k}$ is an arbitrary constant vector, which is chosen so as to exploit the arm's redundancy. We adjust $\boldsymbol{k}$ to reduce the change of the angular velocity between the final point and the via point before it. The joint angular accelerations at the initial point and the final point are set by considering that the joint torque does not exceed the maximum torque limit.

2) Braking Trajectory after Hitting: To stop the arm's motion after hitting, we input the braking force given as

$$\boldsymbol{\tau} = -\boldsymbol{C}_b \dot{\boldsymbol{\theta}} \tag{7}$$

where $\boldsymbol{C}_b$ is the damping matrix. If the end-effector velocity satisfies $\|\dot{\boldsymbol{p}}\| < \epsilon_p$, we stop the trajectory generation.

IV. APPLICATION TO HITTING MOTION

This section shows that the robotic arm shown in Fig. 2 can achieve the robust hitting motion against the disturbance by applying the trajectory generation based on the dynamics shaping described in section III. Letting the hitting point be $\boldsymbol{p}_h = (0.1,\ 0.45)^T$m and the desired hitting velocity be $\dot{\boldsymbol{p}}_h = (2.0,\ 0)^T$m/s.

A. Trajectory Generation

Phase 1 searches the hitting configuration whose singular vector points toward the desired hitting direction $\dot{\boldsymbol{p}}_h$. We let the parameters be $\Delta\theta_1 = q/25$ deg and $\epsilon = 0.001$ deg. Fig. 5(a) shows that the singular vector can converge on the vector of the desired hitting velocity by repeating the search.

Phase 2 generates reversely the reference path whose singular vectors point to the tangential direction of the path. We let the parameters be $k_s = 0.2$ and $\boldsymbol{C}_p = \mathrm{diag}(0,\ 0,\ 0.2)$. By taking consideration of the movable range of the joint angle, we terminate the path generation when $\|\boldsymbol{p}_h - \boldsymbol{p}\| > 0.42$ m. The reference path and the singular vectors are obtained as Fig. 5(b). This figure shows that the proposed path planning algorithm can generate the reference path which can fit the singular vectors to the tangential direction of the path.

Phase 3 generates the hitting trajectory and hitting path which can trace the reference path obtained in Phase 2 as much as possible. We set nine via configurations including the initial configuration and the hitting configuration. We let the parameters be $k = -7$, $\boldsymbol{C}_b = \mathrm{diag}(500,\ 250,\ 300)$ and $\epsilon_p = 0.001$ m/s. Fig. 6 shows the obtained joint trajectories and the via configurations on the reference path. The smooth trajectory at the via points is generated. Fig. 5(c) shows the hitting path corresponding to the hitting trajectory of Fig. 6 and its singular vectors. As seen from this figure, all of the singular vectors almost point toward the tangential direction of the hitting path. These results show that the proposed algorithm of the trajectory planning is valid for the hitting trajectory generation.

B. Simulation

This section shows the error distribution of the end-effector caused by the disturbance. The simulation results in section II-B make it obvious that the position error of the end-effector converges onto near the singular vector. In this section, we obtain the end-effector locus of the robotic arm applied the joint torque including the disturbance. The nominal joint torque is obtained from the proposed trajectory

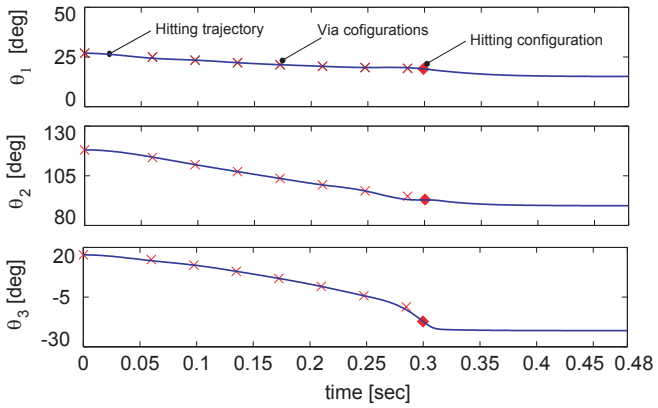

Fig. 6. Hitting trajectory obtained by the cubic spline function and via configurations on the reference path in the configuration space

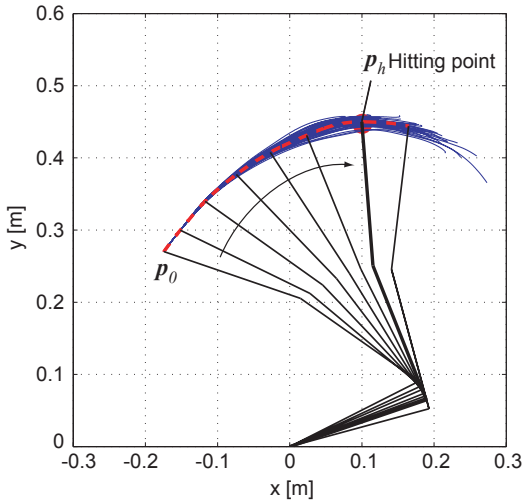

Fig. 7. Simulation results of end-effector distributions for the generated trajectory based on the proposed algorithm.

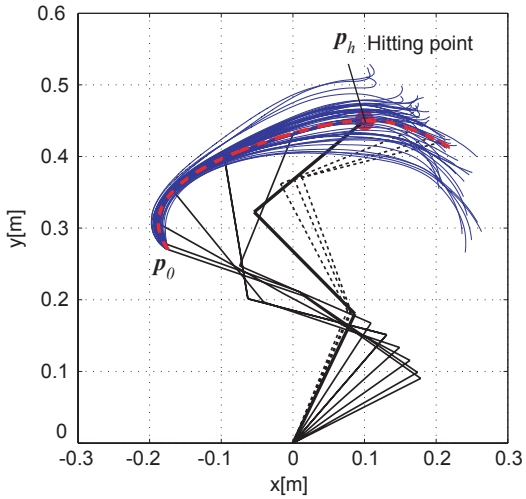

Fig. 8. Simulation results of end-effector distributions for the comparison trajectory.

generation algorithm in advance. The input torque is set as the sum of its nominal joint torque and the Gaussian white noise. Applying the input torque to the robotic arm, we obtain the locus of the end-effector. For comparison, we also conduct similar simulations for the trajectory described in Fig. 3. Since the trajectory is generated without the dynamics shaping, the singular vectors do not point toward the tangential direction of the hitting path of the end-effector. By applying the same initial configuration, the same hitting point and the same hitting velocity to the two trajectories, these simulations are repeated 50 times. The variance of the Gaussian white noise is set as $(16,\ 16)$ $\mathrm{N}^2\mathrm{m}^2$.

Fig. 7 describes the loci of the end-effector for the generated hitting trajectory based on the proposed dynamics shaping techniques. Fig. 8 describes the simulation result for the case of the hitting trajectory described in Fig. 3. The loci of Fig. 7 converges closer to the nominal hitting path than that of Fig. 8. According to these results, by utilizing the dynamics shaping techniques, we can implement the robust motion which can keep the end-effector of the robotic arm close to the nominal path, even if the input torque includes the disturbance.

C. Experiment

This section verifies experimentally that the trajectory obtained by the proposed algorithm is robust to the disturbance. We apply the obtained trajectory to the three-DOF direct-drive arm shown in Fig 9, whose physical parameters are identical to that of the arm described in Fig. 2. The base dynamic parameters of the robotic arm, which are needed to compute the nominal joint torque, are obtained by using parameter identification techniques [6]. We perform experiments on hitting the ball by the planar bat installed at the end-effector of the robotic arm toward the desired hitting direction. The diameter of the ball is $D_t = 10$ mm and the width of the planar bat is $l_b = 24$ mm. The input torque is set as the sum of its nominal joint torque obtained in section IV-A and the Gaussian white noise. Applying the input torque to the robotic arm with a simple open-loop control system, we obtain the locus of the end-effector and the ball. For comparison, we also apply the trajectory described in Fig. 3 to the robotic arm. The variance of the Gaussian white noise is set as $(16,\ 16)$ $\mathrm{N}^2\mathrm{m}^2$. The hitting task is repeated 50 times. The locus of the end-effector is obtained from the joint angles measured by the encoders and the locus of the ball is measured by the high-speed camera whose frame rate is 1 kfps.

Fig. 10(a) describes the loci of the end-point obtained by fitting the direction of the singular vector to be tangential to the path utilizing the dynamics shaping. Fig. 10(b) shows the loci of the end-point for the comparison trajectory described in Fig. 3. The loci of Fig. 10(a) converges closer to the nominal path than that of Fig. 10(b). While the robotic arm without the dynamics shaping only hits the target 10 out of 50 times (its batting average is 20%), the robotic arm with the dynamics shaping hits the target 50 out of 50 times (its batting average is 100%). Both experiments mentioned above are shown in the attached movie.

Fig. 11 shows the loci of the ball hit by the robotic

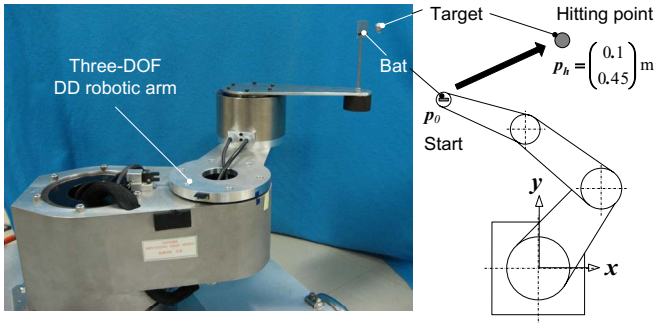

Fig. 9. Experimental setup consisting of the three-DOF Direct Drive robotic arm

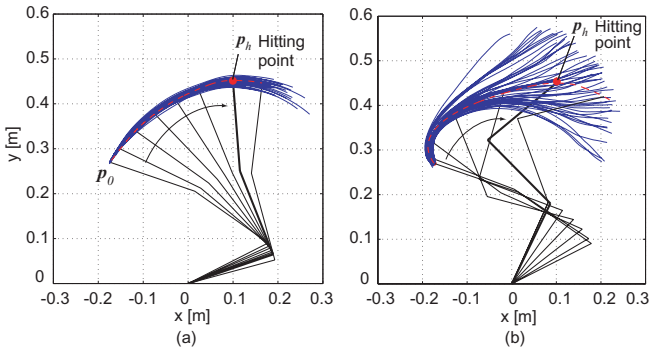

Fig. 10. Experimental results of end-effector distributions. (a) the generated trajectory based on the proposed algorithm. (b) the comparison trajectory.

arm with the dynamics shaping. The deviation angle from the desired hitting direction is below ± 10deg. These results reveal that the robotic arm with the dynamic shaping can hit the target to the desired direction with simple open-loop control systems even though the joint torque includes the disturbance.

V. CONCLUSION

The present paper revealed that

a) The dynamics shaping, which is to shape the robot dynamics appropriately in order to accomplish the desired motion, is useful for the robotic arm planning to gain the high robustness,

b) The positional error of the end-point converges onto near the singular vector of the output controllability matrix of the robotic arm,

c) The trajectory generation algorithm based on the dynamics shaping enables the singular vector to point toward the tangential direction of the hitting path,

d) The robotic arm with the dynamic shaping is able to hit the target to the desired direction with simple open-loop control systems even though the input torque includes the disturbance.

Our future work is to propose the new trajectory generation algorithm based on the dynamics shaping for the redundant robotic arm to trace an arbitrary desired path with fitting the singular vector to the tangential direction of the desired path.

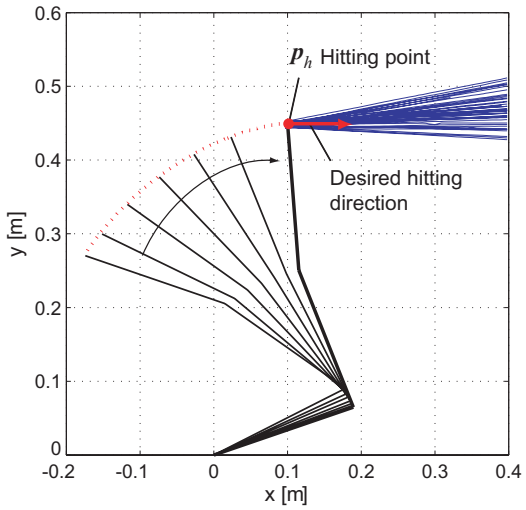

Fig. 11. The loci of the ball hit by the robotic arm with the dynamics shaping.

REFERENCES

[1] J.E. Bobrow, S. Dubowsky, and J.S. Gibson,"Time-optimal Control of Robotic Manipulators Along Specified Paths," Int. J. Robotics Research, vol.4, no. 3, pp. 3-17, 1985.
J. of Neuroscience, vol.5, no.7, pp.1688-1703, 1985.

[2] J.C. Doyle, K. Glover, P.P. Khargonekar, and B.A. Francis, " State-Space Solutions to Standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control problem," IEEE Trans. Automatic Control, vol. 34, no. 8, pp. 831-847, 1989.

[3] B. Friedland, "Controllability Index based on Conditioning Number," Trans. of ASME, J. Dynamics Systems, Measurement, and Control, pp. 444-445, December 1975.

[4] N. Hogan, "Impedance Control: An Approach to Manipulator," Trans. of ASME, J. Dynamics Systems, Measurement, and Control, pp. 1-7, March 1985.

[5] C.D. Johnson, "Optimization of a Certain Quality of Complete Controllability and Observability for Linear Dynamical Systems," Trans. of ASME, J. Basic Engineering, pp.228-238, June 1969.

[6] W. Khalil and E. Dombre, "Modeling, Indentification & Control of Robots,"HPS, 2002.

[7] O. Khatib, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," Int. J. Robotics Research, vol.5, no. 1, pp. 90-98, 1986.

[8] E. Kreindler and P.E. Sarachik, "On the Concepts of Controllability and Observability of Linear Systems," IEEE Trans. on Automatic Control, vol. 9, no. 2, pp. 129-136, 1964.

[9] K. Ogata: Modern Control Engineering 4th Edition. Prentice Hall, 2002.

[10] E. Plaky and L.E. Kavraki, "Distributed Sampling-based Roadmap of Trees for Large-scale Motion Planning," Proc. of the 2005 IEEE International Conference on Robotics and Automation, pp. 3879-3884, 2005.

[11] Z. Shiller and S. Dubowsky, "On Computing the Global Time-optimal Motions of Robotic Manipulators in the Presence of Obstacle," IEEE Trans. on Robotics and Automation, vol. 7, no. 6, pp. 785-797, 1991.

[12] G. Strang: Linear Algebra and Its Applications. Thomson Learning, 1986.

[13] T. Yamawaki and M. Yashima, "Effect of Gravity on Manipulation Performance of a Robotic Arm," Proc. of the 2007 IEEE International Conference on Robotics and Automation, pp. 4407-4413, 2007.

[14] T. Yamawaki and M. Yahisma, "Arm Trajectory Planning by Controlling the Direction of End-point Position Error Caused by Disturbance," Proc. of the 2008 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 1120-1125, 2008.