

# Compliant Manipulation for Peg-in-Hole: is Passive Compliance a Key to Learn Contact Motion?

Seung-kook Yun

Computer Science and Artificial Intelligence Laboratory  
MIT, Cambridge, MA, USA

yunsk@mit.edu

**Abstract**— We examine the usefulness of passive compliance in a manipulator that learns contact motion. Based on the notice that humans outperforms robots with the contact motion, we follow two aspects of human manipulation: passive compliance and learning. As imitation of human's arm and learning, we use a robot arm with passive compliant joints and it learns a policy for peg in hole by the proposed gradient descending method. We present that the passive compliance provides with quick and stable learning as well as a slow control sampling time.

## I. INTRODUCTION

The *Peg in Hole* problem has been extensively researched because it represents a contact task of manipulation that requires both position and force control.

Most of the approaches assume a manipulator of serially connected rigid links, therefore, they highly depend on excellent resolution of a force/torque sensor that is normally very expensive and quite noisy, a precise model of a dynamic system, and a very fast control sampling time that is generally smaller than 1msec. Even if all these conditions are fulfilled, only a little unexpected uncertainty would be able to make the system very unstable and cause a failure. Peg-in-hole seems to require extremely careful consideration to perform well even with a good manipulator hardware. However, consider that human babies play with LEGO blocks. A two-year-old baby can easily assemble the blocks even with his tiny arm and limited capacity for manipulation. For older kids, assembling LEGOs is trivial. What makes the difference? We examine of the two aspects of human manipulation: passive compliance and learning. While a robot manipulator pursues high speed and precision relying on high stiffness and good sensory measurements, a human depends on one's experience and passive nature of muscles. We let a robot to imitate these abilities in this paper.

The goal of this research is to incorporate passive compliance and reinforcement learning for the peg-in-hole problem as the representative of the contact motion, and to show the importance of the passive compliance in learning of the contact motion. We show it enables a faster learning rate, robustness to noise and a slow control sampling time.

This paper is organized as following. Section II talks about the various approaches to the peg-in-hole. Section III examines the differences between a robot manipulator and a human arm, and let a robot imitate a human. Section IV constructs the control scheme by reinforcement learning, and

shows the results. We describes usefulness of the passive compliance in Section V. Section VI conclude research and address some future works.

## II. RELATED WORK

There have been numerous research on the *peg-in-hole* problem. Our approach is new in that we imitate both hardware and software of human manipulation with a passive compliant arm and reinforcement learning.

The milestone paper on the control strategy was given by Lozano Perez, Mason and Taylor [1]. They described how to synthesize compliant motion strategies from geometric constraints including uncertainty. Many researchers solved the problem by hybrid of position and force control. By various methods, they reduce uncertainty in position and force [2]–[5].

Several works have been done on learning of peg-in-hole. Hovland, Sikka and McCarragher [6] proposed skill learning by human-demonstration. They implemented a hidden markov model to find out the relations among the motions. Ogawara and etc. [7] also suggested to learn a skill from visually learning a human demonstration. Lee and Kim [8] used reinforcement learning on 2D peg-in-hole simulation to develop an expert system. Gullapalli, Barto and Grupen [9] set up an associative reinforcement learning system based on the neural network. They let a 6-DOF manipulator learn to insert a peg by relationship between position and force sensing values, and output velocities.

Research have also been done on exploiting passive compliance in peg-in-hole, and they focus on special devices to help a robot with insertion. Southern and Lyons [10] analyzed the usefulness of a passive accommodation device in robotic insertion processes. Haskiya W., Maycock K. and Knight J. [11] developed a hardware frame attached to a peg, in order to ensure good insertion.

Past research on a passively compliant robot also considers safety. Zinn and etc [12] developed DM<sup>2</sup> with a passive spring in a joint to ensure safety and good force control behavior. They used two actuators per a joint for a manipulator to have a flat force-control bandwidth over the entire range. Morita and Sugano [13] proposed MIA which has 7 passive compliant joints with variable stiffness springs and dampers. They used this arm to develop safety strategies. Yun and etc. [14] proposed a safe robot arm based on a torsional spring, a variable viscosity damper and soft skin.

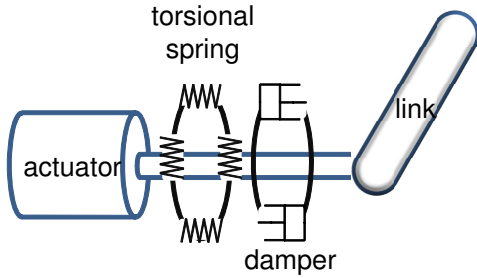


Fig. 1. a rotational passive compliant robot joint with a torsional spring and a damper

### III. HOW TO IMITATE HUMAN'S CONTACT MOTION?

In this paper, a robot manipulator has passive compliance by adding torsional springs and dampers to joints, and learns like a human by reinforcement learning.

While robot manipulators have a precise and quick movement in free space and clearly outperform a human, contact motion of a human outperforms that of a robot even though a robot manipulator generally has a much faster sampling time and preciser sensors. Humans do not decide and re-evaluate motion thousands of times per second. As for peg-in-hole, tons of research proves that a human is much better at this kind of contact motions, because no human has trouble in this simple job. We believe this difference comes from two aspects of human manipulation: passive compliance and learning.

#### A. Passive compliance

The nature of muscles gives a human passive compliance in one's arm. Instead of reacting quickly at contact by accurate force sensing and high powered actuators, the muscles make an arm adapt to the external shape(hole in case of peg-in-hole) or force so that it may have high stability and a slow sampling time. Research on a series elastic actuator [15] have shown that passive compliance can greatly enhance stability of force control.

A torsional spring serially connected to an actuator provides our manipulator with passive compliance as shown in Fig 1. A damper is also added in order to reduce high frequency vibration and to dissipate some energy. Values of the spring constants and the damping ratios will be shown in Section IV-A, which are selected for a manipulator to have less than 10mm deflection by gravity when it is fully stretched horizontally. Note that springs can be deflected not only by gravity but also by dynamics of a manipulator. Therefore, this amount of the deflection appears huge in a view of traditional approaches in which extreme precision and high speed are the virtues of manipulation. However, we will show that a flexible arm can be better at the contact motion even with this deflection.

#### B. Learning

Learning is another big challenging issue for a robot. Whereas most robots make a decision only based on the

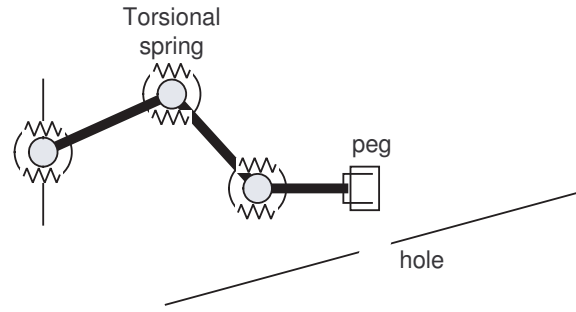


Fig. 2. A system is composed of a 3-DOF manipulator with passive compliant joints and the environment including a hole

current states, accumulated experience shapes manipulation skills of a human. We use reinforcement learning, because it resembles a way a human learns. A simple structure of reinforcement learning is developed to prove our hypothesis. Details are shown in Section IV-B.

### IV. PEG-IN-HOLE BY REINFORCEMENT LEARNING

We solve a 2D peg-in-hole problem by the passive compliant joints and reinforcement learning. A square peg and a hole are implemented in the environment where contact model is a virtual spring, and a 3DOF manipulator is modeled with passive compliance. A robot learns its control policy by a policy gradient method [16].

#### A. System description

A system diagram including a manipulator and the environment is shown in Fig 2. Dynamic equations of a 3DOF manipulator with passive joints are obtained and implemented in MATLAB. More specifically, we assume:

- Each link has its point mass at the end tip.
- A peg and a hole are square.
- Contact between a peg and a hole is elastically modeled.
- No friction at contact
- No gravity

Considered parameters in the dynamic equations are following:

- state  $x = [\theta_1, \theta_2, \theta_3, \varphi_1, \varphi_2, \varphi_3, ]$ , where  $\theta$  is a joint angle and  $\varphi$  is a spring displacement
- $J, M$ : inertia of actuators and mass of links
- $L$ : length of links
- $K, b$ : spring constant and damping of the passive joints

Lengths and Inertia are given by considering the size of an adult human. The parameter values used in the simulation are shown in Table I.

Every possible case of the contact and corresponding force vectors are shown in Fig 3, and it shows contact happens at only four points: The left and right lower-corners of a peg, and the left and right upper-corners of a hole, if a peg is rectangle and so is a hole.

Simulation runs by a 4th order of Runge-Kunta with a 1kHz sampling time.

parameters	1st joint	2nd joint	3rd joint
Point mass(kg)	0.5	0.5	0.1
link length(m)	0.3	0.3	0.05
inertia of actuators(kgm <sup>2</sup> )	0.1	0.1	0.01
stiffness(Nm/rad)	50	20	5
damping constant(Nm/rad <sup>2</sup> )	1.0	1.0	0.2

TABLE I  
PARAMETERS OF A MANIPULATOR

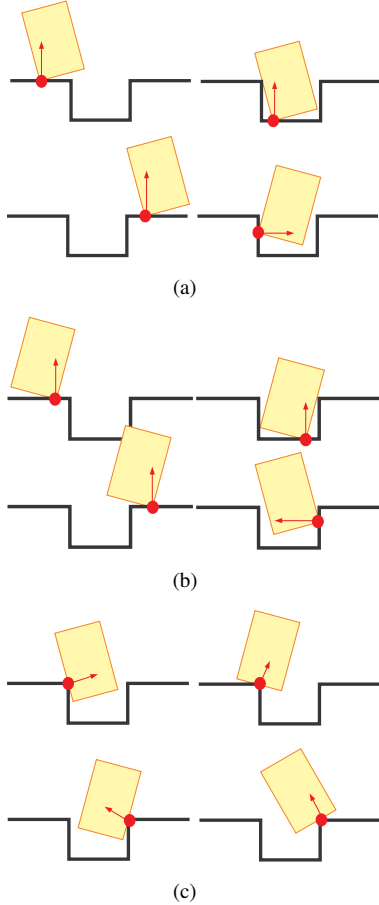


Fig. 3. Possible contacts and corresponding force vectors at (a) lower left corner of a peg (b) lower right corner of a peg (c) upper corners of a hole

### B. Algorithm for updating a policy by reinforcement learning

In order to find the optimal control policy, we implement a gradient descending method [16] with a slow control sampling time.

To exploit reinforcement learning techniques for discrete space, we use the tiling method [16], [17] in which the state space is divided into grids that have parameters of the policy. In general we make the grids for every state. However, our system has 6 states of joint angles and velocities, and covering the whole workspace requires a huge size of dimensions. To reduce them, we propose two methods. Firstly, we assume that a manipulator always starts from the estimated starting position which is nearby the hole so that we focus only around that position. We narrow down the space for the tiles

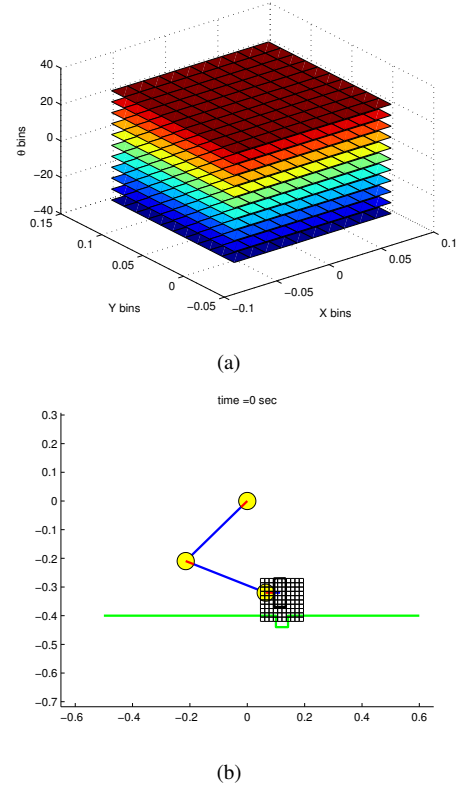


Fig. 4. Proposed tiling: (a) tiles for  $x$ ,  $y$  and  $\theta$  around the estimated starting position (b) overlaid  $x$  and  $y$  grids on the workspace of the manipulator

to  $150\text{mm} \times 150\text{mm} \times 60^\circ$  in Cartesian space. Each state has 11 bins which cover  $-75 \sim +75\text{mm}$ ,  $-30 \sim +120\text{mm}$  and  $-30^\circ \sim +30^\circ$  for  $x$ ,  $y$  and  $\theta$  from the hole. The bins in 3-dimensional space and  $x, y$  grids are shown in Fig 4.

Each bin has three parameters:  $w_x$ ,  $w_y$  and  $w_\theta$  denoting two forces and one torque. Control inputs for the joints are given by projecting them on Jacobian of the manipulator.

$$\tau = J^T \begin{bmatrix} w_x \\ w_y \\ w_\theta \end{bmatrix}$$

Since the system has a continuous domain of the states, parameters of a policy at a arbitrary point is interpolated by 8 neighborhood grid points. This is trilinear interpolation that interpolates a point within a 3D box [18], and convergence of the reinforcement learning parameters is proven with this method [19].

Given the space of the grids, the policy update algorithm works as in Algorithm 1. Firstly, gaussian noises are added to control inputs from the policy.

$$u = w + Z$$

$$Z \sim \mathcal{N}(\vec{0}, \vec{\sigma}^2)$$

where  $w$  is a policy and  $u$  is a control input. This perturbation occurs every control sampling time - 0.01 sec in our simulation. Note that the sampling is very slow considering the

contact motion, since it is known that manipulation for the contact motion requires over 1kHz sampling rate to ensure stability.

---

**Algorithm 1** Policy gradient learning algorithm for *peg in hole*

---

- 1: **for** Each trial **do**
  - 2: Starts randomly deviated from the estimated starting position
  - 3: Obtain  $E(w)$  by simulation without noise
  - 4: **for** Each control sampling time **do**
  - 5: Add a gaussian noise  $Z$  to control input
  - 6: **end for**
  - 7: Obtain  $E(w + Z)$
  - 8: Update the policy by the difference between the value functions
  - 9: **end for**
- 

The policy is updated by the gradient descending method:

$$\Delta w = -\eta(E(w + Z) - E(w))e_N$$

where  $\eta$  is a matrix of learning rates,  $E(\cdot)$  is a value function defined as the sum of the cost function:

$$E(w) = \sum_{n=1}^N g(x, u)$$

and  $e_N$  is an eligibility vector, updated by:

$$e_{k+1} = e_k + \frac{Z}{\sigma^2}$$

The unperturbed value function is obtained by simulation without noise. We select the cost function as the sum of the position and the force error.

$$g(x, u) = C_1 \|x - x_D\|^2$$

when

$$|k_1\phi_1| + |k_2\phi_2| + |k_3\phi_3| < \Gamma$$

otherwise,

$$g(x, u) = C_1 \|x - x_D\|^2 + C_2 (|k_1\phi_1| + |k_2\phi_2| + |k_3\phi_3|)$$

where  $x_D$  is hole position,  $k_1 \sim k_3$  are spring constants,  $\Gamma$  is a threshold for the force error, and  $C_1$  and  $C_2$  are constants. Note that we do not use explicit force but measured torques by the spring displacements.

### C. Learned motion and performance

We assume a completely deterministic system with no noise. We consider noise in Section V-B. A policy is updated 300 times with 2sec duration. We use a 40mm width peg and a 50mm hole for training. The manipulator starts from around the randomly estimated starting position.

The learning curve is shown in Fig 5. After 300 trials, the robot can successfully insert the peg with 100% success. We see a trial as a success when the bottom of the peg approaches within 5mm of the hole bottom. Learned parameters of the policy at  $\theta = 0$  are visualized in Fig 6. These patterns can be

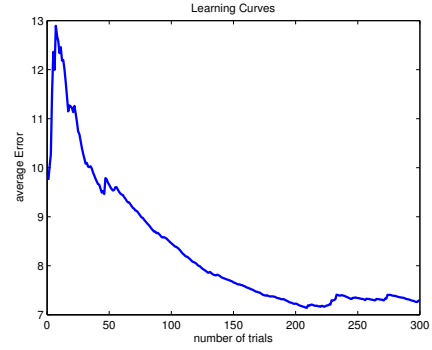


Fig. 5. Learning curve of reinforcement *Peg-in-hole* learning

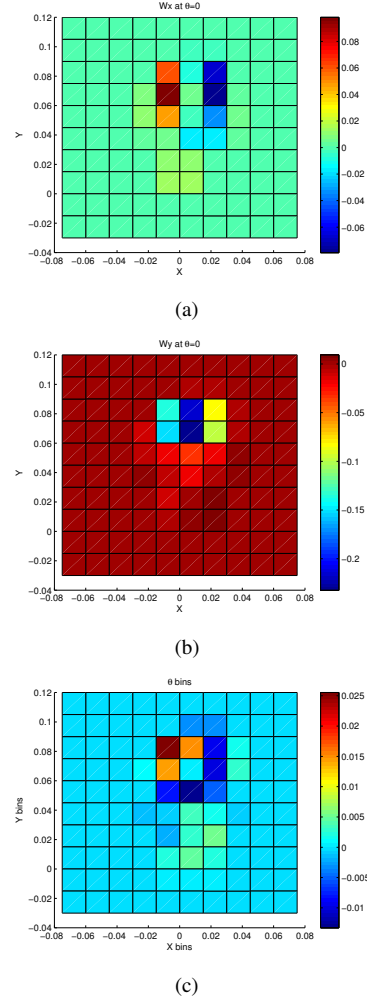


Fig. 6. Learned parameters at  $\theta = 0$ : (a)  $w_x$  (b)  $w_y$  (c)  $w_\theta$

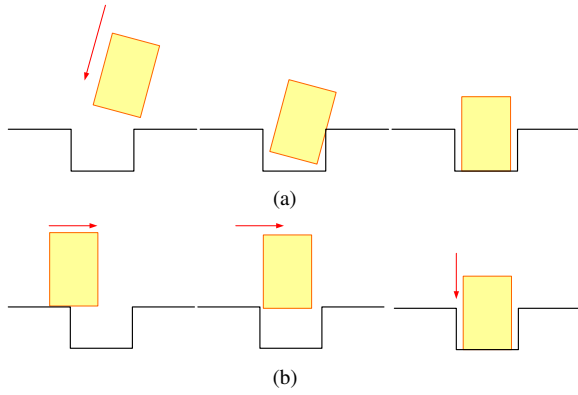


Fig. 7. Learned patterns (a) peg comes from the right side (b) from the left side

understood intuitively:  $w_x$  pushes a manipulator to the goal position,  $w_y$  always pulls it down to the goal and  $w_\theta$  rotates it to the vertically straight.

The robot has learned two patterns according to which side it approaches from: it rotates a peg clockwise a little when it comes from the right side of the hole, while it let a peg just slide on the surface when approaching from the left side. The patterns are shown in Fig 7

## V. DOES PASSIVE COMPLIANCE REALLY HELP?

The proposed algorithm has made a robot learn peg in hole in the environment with no noise. However, it is not clear that how much of this success came from the passive compliance. Now we want to see whether passiveness really helps a robot with the contact motion. We compare simulation results from various stiffness ranges of torsional springs, and also address that the passive compliance gives high robustness to noise.

### A. Performance degradation with stiffer springs

Simulations are implemented in a way that they have the same condition except for the spring constants. Manipulators with  $2\times$ ,  $3\times$ ,  $4\times$ ,  $8\times$ , and  $100\times$  stiffness learn peg-in-hole with 300 trials, and the learning curves are shown in Fig 8. With the learned policies, they try peg-in-hole 50 times, and success ratios are noted in Table II.

The result describes that we have a slower learning rate as the stiffness increases. With almost rigid links ( $100\times$  stiffness), the learning curve does not even converge. This does not directly mean a robot with rigid links cannot learn peg-in-hole by the proposed learning structure. However, at least we can say that we need to be more careful and a learning rate can be slower when we use a stiffer manipulator. With the proper stiffness of the passive compliant joint, a robot appears like being able to learn the contact motion more aggressively and efficiently.

In addition, the same policy works for 0.5mm gap between a peg and a hole even though we trained a manipulator with 10mm gap. Not surprisingly, this comes from a good adaptability of the passivity.

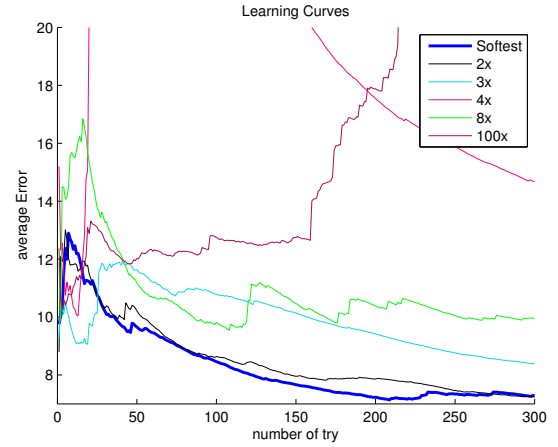


Fig. 8. Learning curves of manipulators with various spring constants

Stiffness	Success ratio
original	100%
$2\times$	90%
$3\times$	50%
$4\times$	0%
$8\times$	0%

TABLE II

SUCCESS RATIOS OF THE MANIPULATORS WITH THE VARIOUS SPRING CONSTANTS

### B. Robustness to noise

We have trained our manipulator in the noise-free environment. However, difficulties of the contact motion emerges from uncertainty. We may not know the exact relative position of a hole from a coordinate of a robot, or a robot may not perfectly follow the control command even if we know the right position. Furthermore, signal from a force sensor is quite noisy.

In order to incorporate the uncertainty and to find out a role of the passive compliance in this case, we add gaussian noises to all the encoders so that a robot has force errors as well as position errors. Note that it senses forces by the angular displacements of the torsional springs. The noises are given:

$$\xi \sim \mathcal{N}(\vec{0}, \vec{\sigma}_n^2)$$

where three kinds of  $\sigma_n$  are chosen as following:

$$\sigma_n = \{0.01, 0.02, 0.04\}$$

The maximum value of the noises is bounded by  $2\sigma_n$ .

By experiments, we figured out that the smallest size of the noise  $\sigma_n = 0.01$  causes maximum  $\pm 17\text{mm}$  and  $\pm 3.5^\circ$  error of a peg position and  $\pm 1.5\text{Nm}$  force error. Considering the width of the peg(40mm) in the simulations, this noise is large. Note that the previous research on reinforcement learning deal with much smaller noise [8], [9]. The larger noises will yield more errors in a proportional way.

Three simulations of reinforcement learning are implemented according to each size of  $\xi$  with  $1\times$ ,  $2\times$ ,  $3\times$  and

Conditions/Stiffness	1×	2×	3×	4×
$\sigma_n = 0.01$	100%	95%	0%	0%
$\sigma_n = 0.02$	100%	70%	-	-
$\sigma_n = 0.04$	82%	14%	-	-

TABLE III  
SUCCESS RATIOS WITH NOISES

Stiffness	Success ratio
original	100%
2x	58%
3x	46%

TABLE IV  
SUCCESS RATIOS WITH 0.1 SEC CONTROL SAMPLING TIME

4× stiffness. Each simulation has 300 trials, and the learned policies are evaluated in 50 tests. The success ratios are shown in Table III.

The result describes that the robots with low stiffness still have high chances of success with noise. The robot with the default stiffness shows almost perfect performance regardless of the size of noise. The second robot also shows robust success ratios though it decrease rapidly as noise increases. The other two robots' performance drops to nothing even with the smallest noise. The result implies passive compliance also provides with more robustness to noise.

### C. Slower control sampling time

Now we see whether we can have a slower control sampling time. Given manipulators with the original, 2x, and 3x stiffness, the same simulations have been done with 10Hz control sampling. Results are noted in Table IV. The original stiffness still produces perfect successes even with ten times slower sampling time, and performances of the others are slightly degraded. Considering that a sampling time is critical for stability and force control performance, the passive compliance gives a large margin for a controller to learn peg in hole.

## VI. CONCLUSION

We show that imitating human's manipulation in hardware and software, with a compliant manipulator and reinforcement learning, brings a convenient way of solving a 2D *peg-in-hole* problem. We design a manipulator with a torsional spring and a damper in every joint, in order to passively adapt a peg to the environment. The gradient descending method is implemented for learning. To reduce dimensions, we assume a small area of peg-in-hole motion and a slow movement. The cost function reflects both of the position and force error. We see 300 trials allows the perfect peg-in-hole motion. By comparison among various stiffness and noises, we show that the passive compliance greatly helps with the contact motion and it yields more stability and a slower sampling time. We think that is what a human does.

In future, we want to include a more noble model of the contact. Also more efficient and realistic estimation of the value function is required to use the proposed method in the real world. Dynamic insertion is another challenging issue.

## VII. ACKNOWLEDGMENTS

Seung-kook Yun is supported in part by Samsung We are grateful to Prof. Russ Tedrake and Prof. Daniela Rus for advices. Thanks Paulina Varshavskaya and Soho Park for insightful discussion on reinforcement learning.

## REFERENCES

- [1] T. Lozano-Perez, M. Mason, and R. H. Taylor, "Automatic synthesis of fine-motion strategies for robots," *International Journal of Robotics Research*, vol. 3, no. 1, 1984.
- [2] S. R. Chhatpar and M. S. Branicky, "Search strategies for peg-in-hole assemblies with position uncertainty," in *International Conference on Intelligent Robots and Systems*, 2001, pp. 1465–1470.
- [3] M. A. Peshkin, "Programmed compliance for error corrective assembly," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 4, pp. 473–482, 1990.
- [4] Y. Li, "Hybrid control approach to the peg-in hole problem," *Robotics and Automation Magazine, IEEE*, vol. 4, no. 2, pp. 52–60, 1997.
- [5] J. Morrow and P. Khosla, "Manipulation task primitives for composing robot skills," in *IEEE International Conference on Robotics and Automation (ICRA '97)*, vol. 4, April 1997, pp. 3354–3359.
- [6] G. Hovland, P. Sikka, and B. McCarragher, "Skill acquisition from human demonstration using a hidden markov model," pp. 2706–2711.
- [7] K. Ogawara, J. Takamatsu, S. Iba, T. Tanuki, H. Kimura, and K. Ikeuchi, "Acquiring hand-action models in task and behavior levels by a learning robot through observing human demonstrations," in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, September 2000.
- [8] S. Lee and M. Kim, "Learning expert systems for robot fine motion control," in *Proc. of IEEE Int. Sym. Intelligent Cont.*, 1998, pp. 534–544.
- [9] V. Gullapalli, A. G. Barto, and R. A. Grupen, "Learning admittance mappings for force-guided assembly," in *In Proceedings, IEEE International Conference on Robotics and Automation*, 1994, pp. 2633–2638.
- [10] S. W.R. and L. C.G., "The study of a passive accommodation device in robotic insertion processes," *Journal of Materials Processing Technology*, vol. 124, no. 6, pp. 261–266, 2002.
- [11] H. W., M. K., and K. J., "A passive compliant wrist for chamferless peg-in-hole assembly operation from vertical and horizontal directions," *Proceedings of the I MECH E Part B Journal of Engineering Manufacture*, vol. 212, no. 6, pp. 473–478, 1998.
- [12] M. Zinn, O. Khatib, B. Roth, and J. Salisbury, "Playing it safe [human-friendly robots]," *Robotics and Automation Magazine, IEEE*, vol. 11, pp. 12–21, 2004.
- [13] T. Morita and S. Sugano, "Development and evaluation of seven dof mia arm," in *IEEE International Conference on Robotics and Automation*, Apr 1997, pp. 462–467.
- [14] S. kook Yun, S. Kang, M. Kim, and S.-S. Yoon, "Input preshaping control of the safe arm with mr-based passive compliant joints," in *IEEE International Conference on Robotics and Automation*, Apr 2004, pp. 2788–2793.
- [15] "Series elastic actuators," in *IROS '95: Proceedings of the International Conference on Intelligent Robots and Systems-Volume 1*. Washington, DC, USA: IEEE Computer Society, 1995, p. 399.
- [16] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," 1998.
- [17] R. Tedrake, T. W. Zhang, and H. S. Seung, "Stochastic policy gradient reinforcement learning on a simple 3d biped," in *In Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 2004, pp. 2849–2854.
- [18] [Online]. Available: <http://local.wasp.uwa.edu.au/~pbourke/other/interpolation>
- [19] R. Munos and A. Moore, "Barycentric interpolators for continuous space and time reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 10, 1998.