IJARST

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science & Technology (IJARST)

Volume 5, Issue 6, May 2020

Motion Planning Of Robotics Manipulator with Deep Reinforcement Learning

Mr. Navlesh Gavhale[1], Mr. S. S Patil[2], Miss. Prajakta Koratkar[3]

[1]Student, Defense Institute of Advanced Technology (Deemed University) Girinagar, Pune.

[2,3]Faculty, Defense Institute of Advanced Technology (Deemed University) Girinagar, Pune.

navleshgavhale777@gmail.com

Abstract: Motion planning is a primary task for a robotic or mechanical system with closed kinematic chains that let the robot find the optimal path in order to reach its goal position or destination. Sampling-based methods like RRT, RRT* provide a good solution for motion planning in basic robot systems. However, complex robotic systems such as hand manipulators have a large number of DoF. This makes motion planning extremely complicated and difficult. Further, these systems also have restrictions such as time and to do real-world tasks in an unstructured environment. Thus, incorporating these restrictions in existing motion planners is complex and computationally expensive. So, real-time calculation i.e. very fast computation of inverse kinematics (IK) in complex robotics systems with dynamically stable configuration is of high priority .as they are very vulnerable to do tasks in an unstructured environment. In this thesis, a methodology to motion planning for complex robotics manipulators using Deep Reinforcement Learning (RL) is proposed. Where the robotics manipulator autonomously learns the optimal behavior through a series of trial-and-error interactions with the environment. It is based on the Markov Decision Process (MDP), Bellman's Equation, Q learning, Deep Deterministic Policy Gradient (DDPG), and Hindsight Experience Replay (HER). The proposed strategy was developed for various motion planning tasks in Robotics. Our goal is to develop ways to train different kinds of complex Robotics Manipulator (gripers).To learn to interact with different objects, especially to grasp, perform some maneuvers to these objects. Achieving this would open up the possibility of the robot agent learning to interact with the environment without prior knowledge.

Keywords: Hybrid power systems, micro grid, power management strategies, smart grid , MATLAB simulation

1. INTRODUCTION

Recent technological Researches are aggressively pushing the state of the art in robotics. We know natural manipulators like Hand, leg, etc. are very complex in design. Because they have a lot of soft tissue, complicated networks of the sensory system, so it is hard to replicate. To design such complex systems come with a cost, nowadays a lot of manipulators are available but they are simple in design.

Figure 1.0.1: Simple Manipulators

DOI: XX.ABCDE/IJARST

They are specially designed for a particular job they do that one job and do it accurately. They don't do other jobs. This does not mean that we don't have a flexible manipulator, flexible manipulator exits but we don't know who to control this manipulator properly. They like if we increase the flexibility of manipulator than the performance of the manipulator goes down and if we increase performance then we cannot get flexibility.

There has been an increasing demand for reinforcement learning (RL)[1]within the fields of Robotics and intelligent systems. Reinforcement learning deals with learning actions in a given environment in order to maximize the growing reward. Classic reinforcement learning (RL) technique makes use of tabular methods or linear approximation to learn this correlation between actions and tasks.[2]RL over other methods is RL doesn't require prior information on the dynamics of the system. It enables the robot to learn optimal behaviour through a trial and error cooperation with the environment. Instead of detailing the solution of the problem.

With the research in deep neural networks within recent times, learning feature extraction and non-linear approximations has become much simpler. It was believed that non-linear approximations like neural networks are very hard to train in the reinforcement learning scenario. However recent research in RL has successfully combined the deep neural networks ANN with RL and improve the learning process. Deep Q Networks [3](DQN) used fully connected ANN layers and convolutional neural networks (CNN to make the RL agents learn to play the ATARI games. From the success of DQN, several variants of this architecture like Double DQN, Prioritized Replay, Duelling Network are proposed which propelled the use of Deep RL in multi-agents. Deep Deterministic Policy Gradient (DDPG) algorithm was proposed by [4] for learning continuous control and high action space tasks. which we use to motion planning of robotics hand manipulator. Robotic systems like open and closed kinematic chains offer fresh perspectives to use deep RL algorithms.

Hand manipulator robotic systems have constantly fascinated the research society for the past few decades. They have relatively complex architecture, high degrees of freedom (DoF), and balancing requirements which make control tasks of those robots difficult. Motion planning in Complex Robotics (humanoid) systems requires collision-free whole-body motions along with active balancing. Even though control in robotics systems is a relatively old research topic, still, the technology development has not been reached and many aspects of motion planning in humanoids need to be improved. Sampling-based planning algorithms, like RRT and PRM are capable of generating globally collision-free solutions.

Within the past few decades, a variety of approaches based on Sampling based planning (SBP) algorithms, such as RRT connect, RRT* connect, RRT*/PRM*. Some recent[5,6,7] works demonstrated motion planning of humanoid robots using these SBP algorithms in different environmental settings proposed an optimal motion planner by posing it as an optimal control problem and combining it with RRT. These SBM based planners were designed for mobile robots and low DoF robotic systems, using them on high DoF humanoid systems which require active balancing is still a difficult task. Also, these are computationally expensive and thus cannot search for the optimal path in real-time. Besides, all the works mentioned in path planning considered only static obstacles. Proposed a stereo vision-based approach[8] for dynamic obstacles avoidance in humanoid robots with predefined motions for maneuver. In the recent DARPA Robotics Challenge (DRC), humanoids were required to perform a variety of maneuvers and coordinated tasks that show the need for real-time controllers and motion planning in complex 3D environments. Robotics systems constitute of kinematics open and closed chains. Even though they are not multi-agent systems, they can be posed as multi-tasking systems where there are shared and non-separable actions (common kinematic chains). In the humanoid model, the spine is the common chain which grants the reachability tasks of both the hands. The problem of motion planning in branched robotic systems can be seen as an optimal control problem of multiple kinematic chains where these may have common sub-chains. There are different mathematical ways that try to address the problem of multi-tasking in branched manipulators. However, classical methods based on Jacobian[9,10] have very limited capability in branched manipulators. Methods like Augmented Jacobian have constrained solution spaces, while methods based on optimization do not provide real-time control.

2. LITERATURE REVIEW

This section discusses the various methods used to control complex manipulators. There are a few more fields related to this project, containing algorithms and simulation environments. In this section, we discuss these related techniques.

2.1 Sampling-based planning (SBP)

Within the past few decades, a variety of approaches based on Sampling based planning (SBP) algorithms, such as RRT connect, RRT* connect, RRT*/PRM*. Some recent[5,6,7] works demonstrated motion planning of humanoid robots using these SBP algorithms in different environmental settings proposed an optimal motion planner by posing it as an optimal control problem and combining it with RRT. These SBM based planners were designed for mobile robots and low DoF robotic systems, using them on high DoF humanoid systems which require active balancing is still a challenging task. Also, these are computationally expensive and thus cannot search for the optimal path in real-time. Besides, all the works mentioned in path planning considered only static obstacles. Proposed a stereo vision-based approach[8] for dynamic obstacles avoidance in humanoid robots with predefined motions for maneuver.

2.2 Grasping and Manipulation[26,27]

From this paper [26,27] we get to know Grasp quality measures and GraspIt simulator platform. Grasping get more attention from the robotics community than dexterous hand manipulations. Researchers have proceeded to grasp primarily from two different directions. One approach target on improving the design of the manipulator so as to improve its capabilities in terms of making stable grasps. The other approach utilizes optimization-based techniques to evaluate the stability of the grasp using grasp metrics [26]. Various grasp metrics have been proposed but no clear advantage of using one over other. These methods are less suited to generalize for dexterous manipulation but are based on important ideas.

While grasping has always the center of hand research, one common approach towards reactive manipulation is Tele-operation using data sensor gloves. In teleoperation, the hard part of the planning and higher cognitive process (decision making) is left to the intelligent user while the controllers blindly follow the joint trajectories. Contact invariant trajectory optimization is capable of synthesizing hand manipulation but it's slow and trades physics for expressive manipulation behavior. Such an approach is hard to generalizable and is less likely to scale in real applications. Also, see the review paper [28] [29].

2.3 Dimensionality Reduction and Synergy Based Control

Although rich and diverse, animal forms exhibit characteristic movements that will be attributed to its morphology, neural system, and habitat. Researchers have credited these to bio-muscular and neural factors. Low dimensional embeddings have been found at the level of kinematics [32], instantaneous muscle activity, Spatiotemporal muscle activity, and feedback control law[33]. These low dimensional embeddings have long been utilized in the graphics community to reduce the dimensionality of search spaces to synthesize full-body movements. It has also been demonstrated that such embedding exists in hand movements. Approx. 95% of the (positional) postural variance associated with hand grasping can be explained using four principal components. Such low dimensional embedding and synergy spaces have been utilized by the robotics community to accelerate the pace of grasping research [27]. However, it has constrained the capabilities of present robotic devices to simple grasps. Similar to biological systems, now day robots have many Dofs. While low dimensional spaces and synergy have helped us control and emulate some of the functionalities; they have restricted the behaviors to simple movements that cover up the expressiveness and dexterity of these robots.

2.4 Capturing Hand Manipulation

In this papers [38,39,40] they work on Gesture-based programming for robotic hands. Also, vision-based techniques to observe and record real-life hand manipulation. They also use 3D visualization devices to achieve desired visual effects, and for motion capture, these techniques have been used for hand movement synthesis. Human hand function has been recorded and studied extensively within the past, mostly in the context of static grasping as opposed to more dexterous manipulation. Investigations have to lead to different shape based grasp taxonomies and grasp evaluation metrics [26]. These metrics have been widely leveraged by data-driven and optimization-based techniques[27] for static grasp synthesis. However, they have very limited utility for more dexterous manipulation involving dynamic contact phenomena. To the best of our knowledge, there has not been prior work done that successfully captured physically consistent dynamic interactions for hand manipulation. This is not surprising; given that the full suite of sensor technologies needed to do this is not readily available. Researchers [37] have however extensively looked at using human motion data for robot programming and robot teaching. Human demonstrations were recorded using a combination of vision, hand tracking, and motion tracking systems. Recorded demonstrations were then first segmented and classified into predetermined steps (reaching, pre grasp, grasp types etc), and then appropriately sequenced to generate a robot program equivalent to the human demonstration. These approaches [37] propelled the advancement in grasp planning towards manipulation, but have been limited to basic reaching and pick-place operations with simple grippers.

Scaling attempts towards manipulation using dexterous manipulators were challenged by the low fidelity Scaling attempts towards manipulation using dexterous manipulators were challenged by the low fidelity recordings of human demonstration. Obstruction due to compact workspaces-inhabited by the object being manipulated significantly impacts the ability of motion tracking and vision-based techniques to observe and record real-life hand manipulation demonstration. a lot of manual work [38] was required to clean up the recorded data sets with specific attention towards individual hand object interaction. Furthermore, phenomena such as sliding, rolling, deformations, compliance which heavily dominate manipulation, and geometric inconsistencies are very difficult to fix in data sets recorded using such techniques. Technological limitations and physically inconsistent datasets considerably impact the pursuit of understanding manipulation from empirical data.

2.5 Model-based Reinforcement Learning

In this paper [45, 48] author give a basic understanding of model-based reinforcement learning. Depending on one's preference of terminology, the methods we will detail in Chapter3 and Chapter6 can be classified as model-free Reinforcement Learning (RL). While RL aims to solve the same general problem as optimal control, its uniqueness comes from model-free learning in stochastic domains [45]. The idea of learning policies without having models still dominates RL, and forms the basis of the most remarkable success stories, both old [46] and new [47]. However, RL with learned models has also considered. Adaptive control on the other hand mostly focuses on learning the parameters of a model with predefined structure, essentially interleaving system identification with control [48]. Our approach here lies somewhere in between (to fix terminology, we call it RL in subsequent sections). We rely on a model, but that model does not have any informative predefined structure. Instead, it is a time-varying linear model learned from data, using a generic prior for regularization. Related ideas have been pursued previously [49]. Nevertheless, as with most approaches to automatic control and computational intelligence in general, the challenge is not only in creating, formulating ideas but also in getting them to scale to hard problems. which is our main contribution here. In particular, we demonstrate scaling from a14-dimensional state space in [49] to a 100-dimensional state-space here. This is important in light of the curse of dimensionality. Indeed RL has been successfully applied to a range of robotic tasks [50], however, dimensionality and sample complexity have presented major challenges [51]

From paper [1,3,46,47].The main disadvantage is that a ground-truth model of the environment is usually not available to the agent. If agents want to use a model in this case, it has to learn the model purely from experience, which creates several challenges. Like bias in the model can be utilized by the agent, resulting in an agent which performs well with respect to the learned model, but behaves super terribly in the real environment.

3. Methodology

3.1 Robotic Hand Manipulator

Three important decisions have been made at the early stage of this project: the simulation tools we needed design environment as well as to carry out our training experiment; a language not only to implement our reinforcement learning algorithms but also with support from a machine learning library to construct and train the deep neural networks; the structure to combine these two-part together.

3.2 DDPG (Deep Deterministic Policy Gradients.) [4]

In paer[4] we get to know that DQN solves problems with high-dimensional observation spaces and discrete low dimensional action space pretty well.

Many real tasks i.e. physical control tasks have continuous and high dimensional action space. Like in our task action space is very large it is 24 DOF and 20 control joint.so action space is 20. To find an action that maximizes action valve function which requires optimization of at every step. In discrete and low dimensional action space we can simply use the max function but we cannot use it in this case. So we can't use Q-learning straightforwardly to continuous action spaces.

To solve this problem [4] paper proposes the RL algorithm. DDPG is RL based algorithm for continuous action space. This combines DQN and Deterministic policy gradient (actor-critic) which use deep neural networks as function approximate for actor and critic. Which we optimize by minimizing the loss. In DQN optimization too slow to be practical with large, unconstraint function approximators and action spaces instead we used an actor-critic approach based on the deterministic policy gradients (DPG) which we see in 3.9.

In our case actor and critic networks are approximated using a fully connected neural network with MLP 3 layers with 256 unit each (ReLU).the actor neural network has state vector has input and action space that contain angular velocities of all the joint needed to achieve goals as output. The critic network takes the state-action vector as input and gives corresponding action-value as outputs. Batch normalization used to avoid over-fitting. The critic Q(s; a) network is learned using the Bellman equation shown in3.12. actor updates the policy in the direction that improves Q value, i.e., critic provides the loss function for the actor.

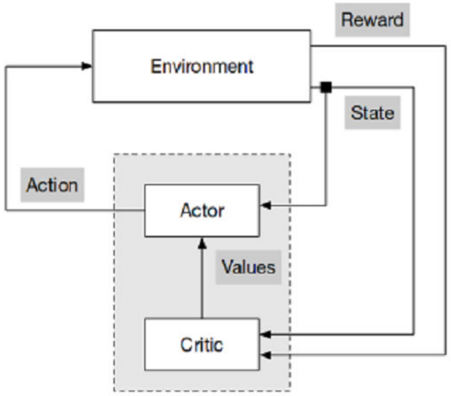

Figure 3.2: Actor and critic Model

The Deterministic Policy Gradient algorithm maintains a parameterized actor function μ(s/Θμ) which defines the present policy by deterministically mapping(give) states to a particular action. The actor is updated by applying rules to the expected return from the start distribution J with respect to the actor parameters:[4]

$$\begin{aligned}\nabla_{\theta^\mu} J &\approx \mathbb{E}_{s_t \sim \rho^\beta}\left[\nabla_{\theta^\mu} Q(s,a|\theta^Q)|_{s=s_t,a=\mu(s_t|\theta^\mu)}\right]\\ &= \mathbb{E}_{s_t \sim \rho^\beta}\left[\nabla_a Q(s,a|\theta^Q)|_{s=s_t,a=\mu(s_t)} \nabla_{\theta_\mu} \mu(s|\theta^\mu)|_{s=s_t}\right]\end{aligned}$$

Parameter noise lets us train agents tasks much more rapidly. Parameter noise adds adaptive noise to the parameters of the ANN neural network policy, rather than to its action space. Earlier RL uses action space noise [54] to change the likelihoods associated with each action the agent might take from one moment to the next. Parameter space noise adds randomness into the parameters of the agent, modifying the types of decisions it makes such that they depend on what the agent currently senses.

Deep reinforcement learning approaches like ,DQN, and DDPG [54].where you don't touch the parameters, but add noise to the action space of the policy.

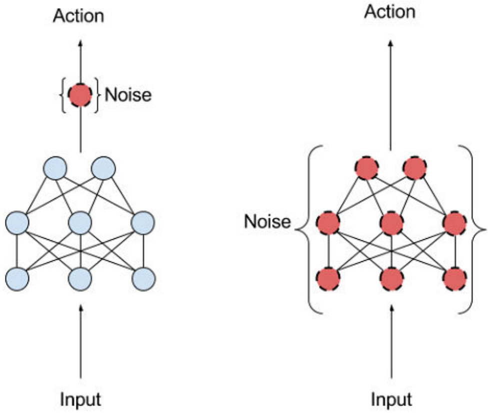

Figure 3.3: Parameter Noise

Parameter noise helps algorithms explore their environments more preciously and effectively, leading to higher rewards and more elegant behaviours. Deliberately adding noise to the parameters of the policy makes an agent's exploration consistent across different time steps, whereas adding noise to the action space leads to more unpredictable exploration. which isn't like anything unique to the agent's parameters.

3.3 Hindsight Experience Replay (HER)

From this paper[55] we can increase the training rate significantly. Hindsight Experience Replay (HER)[55] which allows the algorithm to perform exactly this kind of reasoning and can be combined with any off-policy Algorithm. Not only does HER improve the sample efficiency, but more importantly, it makes learning possible even if the reward signal is sparse and binary. Our approach is based on training generalize policies that take as input not only the current state but also a goal state. The main idea behind HER is to replay each episode with a different goal than the one the agent was trying to achieve, e.g. one of the goals which were achieved in the episode. This reinforcement learning algorithm that can learn from failure.

Let take an example to stabilize cartpole. Our first attempt mostly will not be a successful one. Unless we get lucky, the next few attempts will also likely not succeed. Typical RL algorithms like DDPG,DQN would not learn anything from this experience since they just obtain a constant reward in failure this is -1 that does not contain any learning signal.

The key insight that HER formalizes is what humans do: Even though we have not succeeded at a specific goal, we have at least achieved a different goal. So why not just pretend that we wanted to achieve this goal only to begin with, instead of the one that we set out to achieve originally. By doing this substitution, the RL algorithm can obtain a learning signal since it has achieved some goal; even if it wasn't the one that we meant to achieve originally. If we repeat this process, we will eventually learn how to achieve random goals, including the goals that we really want to achieve. it often used in off-policy RL algorithms like DQN and DDPG[1,2,4] with goals. which are chosen in goal(hindsight) after the episode has finished. HER can, therefore, be combined with any off-policy RL algorithm (for example, HER can be combined with DDPG, which we write as "DDPG + HER").[55]

4. CONCLUSION

The objective of the thesis is to address the problem of motion planning in complex, higher DOFs robotic systems such as Hand manipulators, Humanoids using RL reinforcement learning. The advantage of using RL over other methods is RL doesn't require prior knowledge of the dynamics of the system. It enables the robot to learn optimal behavior through the trial-and-error interaction with the environment. Instead of detailing the solution of the problem, in RL the evaluation is done using the feedback provided in the form of the scalar objective function which measures the one-step performance of the robot. Also, unlike sampling-based planners, Inverse kinematic Base planners, we bypass the need for accurate and robust dynamics modeling of the system.

We evaluate the performance of DDPG with and without Hindsight Experience Replay(HER) DDPG+HER with sparse rewards DDPG with sparse rewards DDPG+HER significantly performs well than DDPG.

REFERENCES

[1] R. S. Sutton and A. G. Barto, Reinforcement learning: An introduction. MIT press Cambridge, 1998, vol.1, no.1

[2] G. Konidaris, S. Osentoski, and P. S. Thomas, "Value function approximation in reinforcement learning using the fourier basis." in AAAI, vol. 6, 2011, p. 7.

[3] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning." In AAAI, 2016, pp. 2094–2100.

[4]T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," arXiv preprint arXiv:1509.02971v6, 2019.

[5] K. Hauser, T. Bretl, and J.-C. Latombe, "Non-gaited humanoid locomotion planning," in Humanoid Robots, 2005 5th IEEE-RAS International Conference on. IEEE, 2005, pp. 7–12.

[6] S. Dalibard, A. Nakhaei, F. Lamiraux, and J.-P. Laumond, "Whole-body task planning for a humanoid robot: a way to integrate collision avoidance," in Humanoid Robots, 2009. Humanoids 2009. 9th IEEE-RAS International Conference on. IEEE, 2009, pp. 355–360.

[7] H. Teja, A. Balachandran, and S. V. Shah, "Optimal task planning of humanoid in cluttered environment," in ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers, 2016, pp. V05AT07A051–V05AT07A051.

[8] W. Budiharto, J. Moniaga, M. Aulia, and A. Aulia, "A framework for obstacles avoidance of humanoid robot using stereo vision," International Journal of Advanced Robotic Systems, vol. 10, no. 4, p. 204, 2013.

[9] S. R. Buss, "Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods," IEEE Journal of Robotics and Automation, vol. 17, no. 1-19, p. 16, 2004.

[10] M. F. Møller, "A scaled conjugate gradient algorithm for fast supervised learning," Neural networks, vol. 6, no. 4, pp. 525–533, 1993.

[11] Y. Saito, T. Higashihara, K. Ohnishi, and A. Umemura, "Research on intelligent motorized prosthetic hand by functional analysis of human hand at near future," in Proceedings of the 18th IEEE International Symposiumon Robot andHuman Interactive, RO-MAN 2009, pp. 786–791, jpn, October 2009.

[12] K. Scott and A. Perez-Gracia, "Design of a prosthetic hand with remote actuation," in Proceedings of the 34th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS 2012, pp. 3056–3060, usa, September 2012.

[13] R. Wells and J. Schueller, "Forward and Feedback Control of a Flexible robotic Arm," in Proceedings of the IEEE International Conference on Control System, 1999.

[14] S. Jacobsen, E. Iversen, D. Knutti, R. Johnson, and K. Biggers. Design of the utah/m.i.t. dextrous hand. In Robotics and Automation. Proceedings. 1986 IEEE In- ternational Conference on, volume 3, pages 1520 {1532, apr 1986.

[15] A.D. Deshpande, Z. Xu, M.J.V. Weghe, LY Chang, BH Brown, DD Wilkinson, SM Bidic, and Y. Matsuoka. Mechanisms of the anatomically correct testbed (act) hand. IEEE/ASME Trasactions on Mechatronics, 2011.

[16] Tetsuya Mouri, Haruhisa Kawasaki, Keisuke Yoshikawa, Jun Takai, and Satoshi Ito. Anthropomorphic robot hand: Gifu hand iii. In Proc. Int. Conf. ICCAS, pages 1288{ 1293, 2002.

[17] EY Chao, JD Opgrande, and FE Axmear. Three-dimensional force analysis of finger joints in selected isometric hand functions. Journal of Biomechanics, 9(6):387{396,1976.

[18] A. Crawford, J. Molitor, A. Perez-Gracia, and S. Chiu, "Design of a Robotic Hand and Simple EMG Input Controller with a Biologically-Inspired Parallel Actuation System for Prosthetic Applications," Journal of the Franklin Institute, Elsevier, vol. 344, pp. 36–57, 2007.

[19] Y. Chnag, "A Survey of Robotic Hand- Arm Systems," International Journal of Computer Applications, vol. 109, no. 8, 2015

[20] E.J. Barth, Jianlong Zhang Jianlong Zhang, and M. Goldfarb. Sliding mode approach to PWM-controlled pneumatic systems. Proceedings of the 2002 American Control Conference (IEEE Cat. No.CH37301), 3:2362{2367 vol.3, 2002

[21] G. Carducci, N. I. Giannoccaro, a. Messina, and G. Rollo. Identi_cation of viscous friction coefficients for a pneumatic system model using optimization methods. Math- ematics and Computers in Simulation, 71(4-6):385{394, 2006.

[22] Tassa Yuval, Tingfan Wu, J Movellan, and E Todorov. Modeling and identi_cation of pneumatic actuators. In International Conference on Mechatronics anf Automation (ICMA), 2013.

[23] Edmond Richer and Yildirim Hurmuzlu. A high performance pneumatic force actuator system: Part1 nonlinear mathematical model. Journal of dynamic systems, measure- ment, and control, 122(3), 2000.

[24] Xue Song Wang, Yu H. Cheng, and Guang Zheng Peng. Modeling and self-tuning pressure regulator design for pneumatic-pressure-load systems. Control Engineering Practice, 15(9):1161{1168, 2007.

[25] Yuval Tassa, Tingfan Wu, Javier Movellan, and Emanuel Todorov. Modeling and identi_cation of pneumatic actuators. 2013 IEEE International Conference on Mecha- tronics and Automation, IEEE ICMA 2013, pages 437{443, 2013

[26] Grasp quality measures. http://personalrobotics.ri.cmu.edu/courses/papers/SuarezEtal06.pdf.

[27] Andrew T Miller and Peter K Allen. Graspit! a versatile simulator for robotic grasping. Robotics & Automation Magazine, IEEE, 11(4):110{122, 2004.

[28] Antonio Bicchi and Vijay Kumar. Robotic grasping and contact: A review. In Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on, volume 1, pages 348{353. IEEE, 2000.

[29] Allison M Okamura, Niels Smaby, and Mark R Cutkosky. An overview of dexterous manipulation. In Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on, volume 1, pages 255{262. IEEE, 2000.

[30] Andrew T Miller and Peter K Allen. Examples of 3d grasp quality computations. In Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on, volume 2, pages 1240–1246. IEEE, 1999. pages 20

[31] Andrew T Miller, Steffen Knoop, Henrik I Christensen, and Peter K Allen. Automatic grasp planning using shape primitives. In Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on, volume 2, pages 1824–1829. IEEE, 2003. pages 21, 51

[32] Marco Santello, Martha Flanders, and John F Soechting. Postural hand synergies for tool use. The Journal of Neuroscience, 1998.

[33] Daniel B Lockhart and Lena H Ting. Optimal sensorimotor transformations for balance. Nature neuroscience, 2007.

[34] Andrew T Miller and Peter K Allen. Graspit! a versatile simulator for robotic grasping. Robotics & Automation Magazine, IEEE, 11(4):110{122, 2004.

[35] Grasp quality measures. http://personalrobotics.ri.cmu.edu/courses/papers/SuarezEtal06.pdf.

[36] Andrew T Miller and Peter K Allen. Graspit! a versatile simulator for robotic grasping. Robotics & Automation Magazine, IEEE, 11(4):110{122, 2004.

[37] Richard M Voyles, J Dan Morrow, and Pradeep Khosla. Gesture-based programming for robotics: Human augmented software adaptation. 1999.

[38] Kawasaki Haruhisa, Mouri Tetsuya, and Ueki Satoshi. Virtual Robot Teaching for Humanoid Both-Hands Robots Using Multi-Fingered Haptic Interface, The Thousand Faces of Virtual Reality.

[38] Sing Bing Kang and Katsushi Ikeuchi. Toward automatic robot instruction from perception-mapping human grasps to manipulator grasps. Robotics and Automation,IEEE Transactions on, 13(1):81{95, 1997.

[39] Doug A Bowman and Larry F Hodges. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In Proceedings of the 1997 symposium on Interactive 3D graphics, pages 35ACM, 1997.

[40] Kawasaki Haruhisa, Mouri Tetsuya, and Ueki Satoshi. Virtual Robot Teaching for Humanoid Both-Hands Robots Using Multi-Fingered Haptic Interface, The Thousand Faces of Virtual Reality.

[41] Katsu Yamane, James J Ku_ner, and Jessica K Hodgins. Synthesizing animations of human manipulation tasks. In ACM Transactions on Graphics (TOG), volume 23, pages 532{539. ACM, 2004.

[42] Yuting Ye and C Karen Liu. Synthesis of detailed hand manipulations using contact sampling. ACM Transactions on Graphics (TOG), 31(4):41, 2012.

[43] Microsoft Kinect. www.microsoft.com/en-us/kinectforwindows/.

[44] Tanner Schmidt, Richard Newcombe, and Dieter Fox. Dart: Dense articulated realtime tracking. Proceedings of Robotics: Science and Systems, Berkeley, USA, 2, 2014.

[45] R. Sutton and A. Barto. Reinforcement Learning: An Introduction. MIT Press, 1998.

[46] Gerald Tesauro. Td-gammon, a self-teaching backgammon program, achieves masterlevel play. Neural computation, 6(2):215{219, 1994.

[47] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. Nature,518(7540):529{533, 2015.

[48] Karl J _Astr om and Bj orn Wittenmark. Adaptive control. Courier Corporation, 2013.

[49] S. Levine, N. Wagener, and P. Abbeel. Learning contact-rich manipulation skills with guided policy search. In International Conference on Robotics and Automation (ICRA), 2015.

[50] M. Deisenroth, C. Rasmussen, and D. Fox. Learning to control a low-cost manipulator using data-e_cient reinforcement learning. In Robotics: Science and Systems (RSS),

[51] M. Deisenroth, G. Neumann, and J. Peters. A survey on policy search for robotics. Foundations and Trends in Robotics, 2(1-2):1{142, 2013.

[52] Narasimhan Jegadeesh and Sheridan Titman. “Short-horizon return reversals and the bid-ask spread”. Journal of Financial Intermediation, 4(2):116–132, 1995. pages 20.

[53] Emanuel Todorov, Tom Erez and Yuval Tassa” MuJoCo: A physics engine for model-based control” University of Washington

[54] Matthias Plappertyz, Rein Houthoofty, Prafulla Dhariwaly, Szymon Sidory, Richard Y. Cheny, Xi Chenyy, Tamim Asfourz, Pieter Abbeelyy, and Marcin Andrychowiczy]” PARAMETER SPACE NOISE FOR EXPLORATION” OpenAI z Karlsruhe Institute of Technology (KIT) University of California, Berkeley2018

[55] Marcin Andrychowicz_, FilipWolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel†, Wojciech Zaremba† “Hindsight Experience Replay”OpenAI 2018