Research Statement

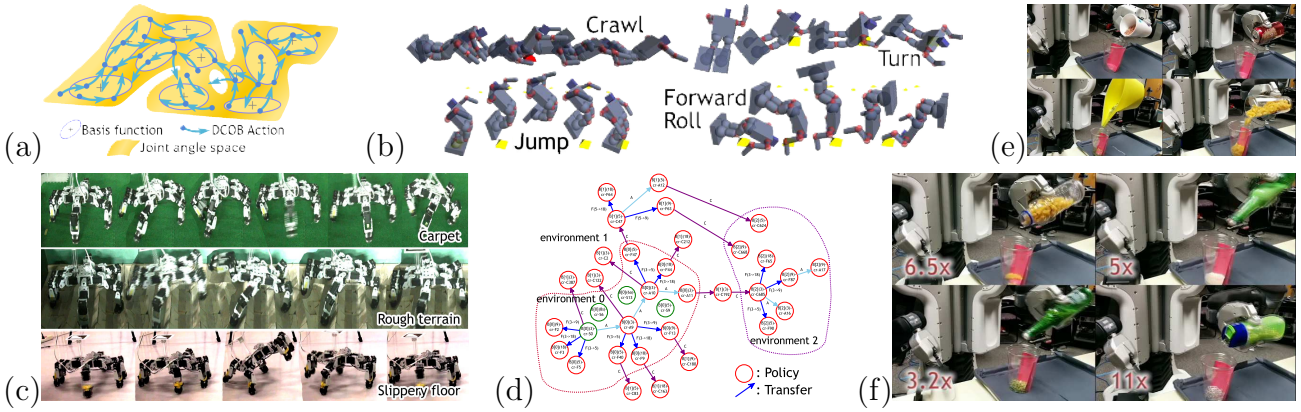
Akihiko Yamaguchi (`info@akihikoy.net`)

Introduction

My main research interest is autonomous behavior generation of robots. Current robotics technologies have succeeded in motion planning for collision avoidance, and grasping and manipulating rigid objects. However robots still have a limited use; one reason is their less capability in manipulating non-rigid objects. Our everyday activities involve many such manipulations, that are necessary to home-care robots. For example in a cooking video of humans, we will see many complicated manipulation skills such as cutting vegetables/meats, and pouring salt/ketchup/cheese. Programming such behaviors on robots is very expensive; even if a behavior is programmed, it will not generalize widely, i.e. will fail at slightly different situations. We need a behavior generation system that can **generalize** behaviors to unseen situations, and even when the generalization fails, it should **adapt** behaviors through learning. Such a system should be **scalable** to complicated tasks such as cooking. The central problem is planning under unknown (unmodeled) dynamics, known as **reinforcement learning** (RL) problem. So far I studied model-free RL approach (mostly in doctoral thesis) and model-based RL approach (postdoctoral work at CMU).

Previous and Current Work

In my PhD thesis I explored model-free reinforcement learning (RL) of robot behaviors with less prior knowledge about the tasks. A popular approach of RL in robotics is a direct policy search where policies are directly trained from experience typically using gradients (e.g. [9, 4, 13]). However these methods often converge to poor local maxima due to their exploration noise model (typically Gaussian noise around a policy). In practical domains (e.g. ball-in-cups), human demonstrations are used as initial policies. We developed a method DCOB to generate a set of primitive actions that are small transitions in state space (Fig.(a)), and applied the Peng's Q(λ)-learning algorithm [8] with a Gaussian-network function approximator and a softmax exploration [25]. The exploration noise is similar to a multimodal Gaussian, which provides wider exploration than the (unimodal) Gaussian noise around a policy. We applied the proposed method to obtaining whole-body motions such as crawling, turning, and jumping in simulation (Fig.(b)). The robot obtained these motions only from reward functions. We also applied it to a crawling task of a 6-legged spider-like robot where the robot could obtain a motion in 20 min from scratch (no simulation). The developed methods were also published as an open source software SkyAI (`http://skyai.org/`)[22].

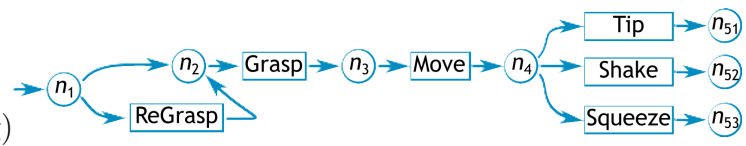


With DCOB, I achieved an adaptation of robots with degrees of freedom (DoF) up to 7 (some joints were coupled in the experiments). For a larger DoF and wider generalization, I explored a curriculum learning [3], and a modular system. The system has a library of policies. The policies are automatically created by combining different strategies, including learning-from-scratch with DCOB, transfer-learning-with-freeing-DoF (increasing the DoF by decoupling joint constraints), and inter-environment-transfer. The combination of learning-from-scratch and transfer-learning-with-freeing-DoF enables a curriculum learning like learning from easy missions [1]. For example in learning crawling, robots learn gradually from a simpler configuration (lower DoF, lower speed) to complicated configurations (higher DoF, higher speed). The robots could learn behaviors with full DoF (such as 18) from less prior knowledge [24]. The inter-environment-transfer enables learning in different types of environments (e.g. terrain types)[26]. We introduced an environment class estimation which may be an unseen environment. In case of an unseen environment, the system uses inter-environment-transfer to learn a new policy based on existing ones. The experiments of a crawling task of a spider-like robot in different terrain types demonstrated the adaptability and the generalization over the environments (Fig.(c))[23]. We found that a second-best policy was better for the inter-environment transfer (Fig.(d)), which might be due to overfitting of the best policy.

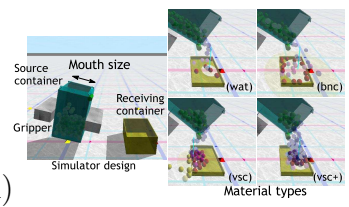
Through the above work, I noticed that it was still far from practical domains. For example applying to cutting or pouring task is difficult, since in these tasks the state-action space is huge and the solutions exist only on narrow manifolds. Discrete dynamics also makes learning more difficult; e.g. robots can move an object during grasping it. In Carnegie Mellon University, I started to work on robot pouring as a case study of complicated manipulation in order to establish an effective reinforcement learning framework. We considered a general pouring task: moving material including liquids and powders from a container to a receiver. This work followed learning from demonstration framework. Humans use many strategies in pouring, for example pouring water by tipping a bottle, shaking a salt bottle, and tapping a coffee powder bag. Our hypothesis is that using a range of strategies (skills) enables a behavior to generalize over situations. On a PR2 robot, we implemented some skills, motion planning for skill parameters (e.g. grasping poses and collision free trajectories), and learning methods for selecting a skill and adjusting skill parameters (e.g. shaking axis and speed). Our behavior model could pour various materials from a range of containers (Fig.(e)(f); video: `https://youtu.be/GjwfbOur3CQ`)[21]. We achieved adaptation to variations including shapes of containers, materials, initial poses of containers and the robot, and target amount. However it was difficult to generalize the behavior to unseen container shapes and materials. Since pouring involves dynamics that are hard to model such as liquid flow, and we need to plan a sequence of primitive actions, it is formulated as a reinforcement learning (RL) problem. We considered many RL approaches, and decided to focus on model-based approach since we can expect a good generalization and reusability of learned components [15, 18]. In a model-free RL, we train policies directly, while in a model-based RL, we train dynamical models and optimize policies by solving dynamic programming. We can use existing engineered models such as collision models together with learned models.

As a model-based method, we developed a stochastic version of neural networks to learn models [19], and a differential dynamic programming for optimization over graph-structured

(g)



(h)

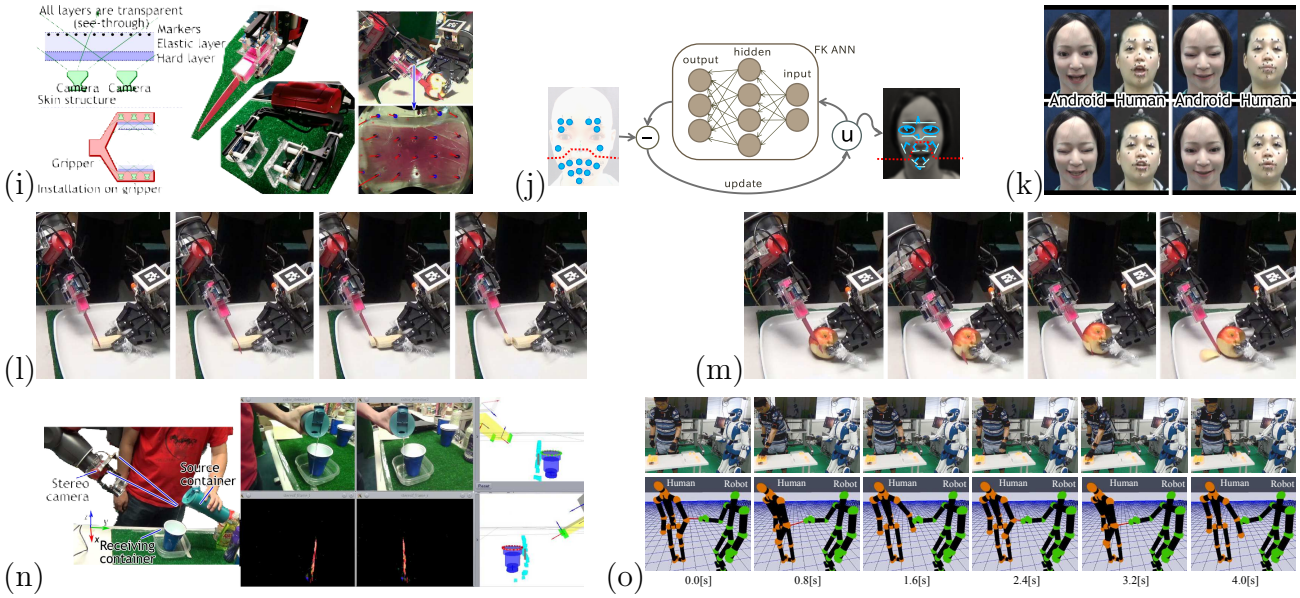


dynamical systems [17]. The proposed method is a version of deep reinforcement learning [18]. Pouring behavior involves selections of skills (discrete-action optimization) and adjustments of skill parameters (continuous-action optimization). Such a dynamical system can be described as a graph structure (Fig.(g)). Differential dynamic programming (DDP; [6]) is for a sequence of continuous actions, and is applicable to linear structured dynamical systems only. We used graph theory to analyze graph structures (e.g. unrolling a graph), and derived DDP equations for structures with bifurcations. The proposed method is referred to as Graph DDP [17]. In order to handle uncertainty of models, Graph DDP is based on a stochastic version of DDP [7].

On the other side, we explored modeling methods of component dynamical systems. A non-parametric method (locally weighted regression; LWR [2]) and a parametric method (neural networks) are investigated [14, 19]. Since we are using stochastic DDP, these regression models need to be capable of modeling uncertainty and propagating probability distributions. In [19], we extended neural networks for such capabilities. Compared to LWR, the extended neural networks performed better in the model-based RL scenario of pouring. With Graph DDP and the extended neural networks, we achieved generalization of pouring behavior over material types and container shapes (Fig.(h))[17].

I emphasize that a reason of this success is supported by **task-level** modeling of dynamics. Task-level dynamics model the input and output relation of a skill. Task-level models can avoid cumulative error of time-integral unlike a differential equation model of dynamics. Recently, in order to make behavior generation more flexible, I started to work on semantic (symbolic) representation and reasoning. This work is done as a collaboration with Professor Michel Beetz in University of Bremen where we are extending an ontology for robot reasoning KnowRob [12] to be capable of behavior reasoning. With a flexible reasoning, we can generate graph structures of behaviors, which is useful for, for example, failure recovery.

Side Projects: In addition to the behavior learning and reasoning research, I also work on relative projects. Major ones include multimodal optical skin sensor for manipulation of deformable objects (Fig.(i))[16] with which we implemented a cutting motion (Fig.(l)(m)), stereo vision of liquid and particle flow for robot pouring (Fig.(n))[20], human-safe robot control and motion planning (Fig.(o))[10, 11], and learning inverse kinematics of an android robot face with neural networks (Fig.(j)(k))[5].



Future Directions

From the previous work I have learned that a modular or library based approach where we combine many alternative strategies is effective in intelligent and robust behavior generations. Unification of many different types of reasoning, learning, and representations would advance the intelligence and robustness. My future research direction is such a unification and its verification in practical domains. A behavior generation system I will create consists of different types of libraries, reasoning and generating modules, learning modules, an ontology describing the relations among elements in libraries, and an execution system. Completing and advancing such a system is the main direction of the next few years. Through this work, following intellectual benefits are expected: 1) Developing a unified architecture of reasoning, learning, actions, perceptions, and ontology in different levels including numeric, symbolic, and task-level representations. 2) Adaptation and generalization of robotic behaviors in practical domains such as everyday activities of humans. 3) Robust behavior generation by autonomously combining many different alternative strategies. 4) Learning effective dynamical models and reasoning behaviors by combining numeric, symbolic, and task-level representations. 5) Improving efficiency of learning behaviors across many tasks as the library-based approach increases the reusability. 6) Failure detection with forward estimation models, and failure recovery by symbolic and numeric reasoning with different strategies and learning. 7) Finding analogies based on abstraction of ontology, and reasoning behaviors with them. 8) Increasing transferability of knowledge (elements in libraries) among robots and humans. It will enable humans to teach robots in many different ways, such as kinesthetic teaching, and abstract-level programming like "making coffee by pouring coffee powder and hot water." 9) Learning functionality of tools, and reasoning about when and how to use tools. 10) Manipulation of non-rigid objects including liquids, powders, food, thermal and chemical processes, and concepts like "tasty". In short, I will contribute in the fields of robotics, machine learning, and artificial intelligence. I will collaborate internally and externally. My current collaborators include Carnegie Mellon University, Nara Institute of Science and Technology (Japan), and University of Bremen (Germany). I will also continue side projects including optical skin sensor [16].

References

[1] Minoru Asada, Shoichi Noda, Sukoya Tawaratsumida, and Koh Hosoda. Purposive behavior acquisition for a real robot by vision-based reinforcement learning. *Machine Learning*, 23(2-3):279–303, 1996.

[2] Christopher G. Atkeson, Andrew W. Moore, and Stefan Schaal. Locally weighted learning. *Artificial Intelligence Review*, 11:11–73, 1997.

[3] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML'09)*, pages 41–48, 2009.

[4] Jens Kober and Jan Peters. Learning motor primitives for robotics. In *the IEEE International Conference on Robotics and Automation (ICRA'09)*, pages 2509–2515, 2009.

[5] Emarc Magtanong, Akihiko Yamaguchi, Kentaro Takemura, Jun Takamatsu, and Tsukasa Ogasawara. Inverse kinematics solver for android faces with elastic skin. In *Latest Advances in Robot Kinematics*, pages 181–188, Innsbruck, Austria, 2012.

[6] David Mayne. A second-order gradient method for determining optimal trajectories of non-linear discrete-time systems. *International Journal of Control*, 3(1):85–95, 1966.

[7] Yunpeng Pan and Evangelos Theodorou. Probabilistic differential dynamic programming. In *Advances in Neural Information Processing Systems 27*, pages 1907–1915. Curran Associates, Inc., 2014.

[8] Jing Peng and Ronald J. Williams. Incremental multi-step Q-learning. In *International Conference on Machine Learning*, pages 226–232, 1994.

[9] Jan Peters, Sethu Vijayakumar, and Stefan Schaal. Reinforcement learning for humanoid robotics. In *Humanoids2003, IEEE-RAS International Conference on Humanoid Robots*, 2003.

[10] Gustavo Alfonso Garcia Ricardez, Akihiko Yamaguchi, Jun Takamatsu, and Tsukasa Ogasawara. Asymmetric velocity moderation for human-safe robot control. *Advanced Robotics*, 29(17):1111–1125, 2015.

[11] Gustavo Alfonso Garcia Ricardez, Akihiko Yamaguchi, Jun Takamatsu, and Tsukasa Ogasawara. Human safety index based on impact severity and human behavior estimation. In *the 2nd International Conference on Mechatronics and Robotics Engineering (ICMRE'16)*, 2016.

[12] Moritz Tenorth and Michael Beetz. KnowRob: A knowledge processing infrastructure for cognition-enabled robots. *Int. J. Rob. Res.*, 32(5):566–590, 2013.

[13] E. Theodorou, J. Buchli, and S. Schaal. Reinforcement learning of motor skills in high dimensions: A path integral approach. In *the IEEE International Conference on Robotics and Automation (ICRA'10)*, pages 2397–2403, may 2010.

[14] Akihiko Yamaguchi and Christopher G. Atkeson. Differential dynamic programming with temporally decomposed dynamics. In *the 15th IEEE-RAS International Conference on Humanoid Robots (Humanoids'15)*, 2015.

[15] Akihiko Yamaguchi and Christopher G. Atkeson. A representation for general pouring behavior. In *in the Workshop on SPAR in the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'15)*, 2015.

[16] Akihiko Yamaguchi and Christopher G. Atkeson. Combining finger vision and optical tactile sensing: Reducing and handling errors while cutting vegetables. In *the 16th IEEE-RAS International Conference on Humanoid Robots (Humanoids'16)*, 2016.

[17] Akihiko Yamaguchi and Christopher G. Atkeson. Differential dynamic programming for graph-structured dynamical systems: Generalization of pouring behavior with different skills. In *the 16th IEEE-RAS International Conference on Humanoid Robots (Humanoids'16)*, 2016.

[18] Akihiko Yamaguchi and Christopher G. Atkeson. Model-based reinforcement learning with neural networks on hierarchical dynamic system. In *the Workshop on Deep Reinforcement Learning: Frontiers and Challenges in the 25th International Joint Conference on Artificial Intelligence (IJCAI2016)*, 2016.

[19] Akihiko Yamaguchi and Christopher G. Atkeson. Neural networks and differential dynamic programming for reinforcement learning problems. In *the IEEE International Conference on Robotics and Automation (ICRA'16)*, 2016.

[20] Akihiko Yamaguchi and Christopher G. Atkeson. Stereo vision of liquid and particle flow for robot pouring. In *the 16th IEEE-RAS International Conference on Humanoid Robots (Humanoids'16)*, 2016.

[21] Akihiko Yamaguchi, Christopher G. Atkeson, and Tsukasa Ogasawara. Pouring skills with planning and learning modeled from human demonstrations. *International Journal of Humanoid Robotics*, 12(3):1550030, 2015.

[22] Akihiko Yamaguchi and Tsukasa Ogasawara. Skyai: Highly modularized reinforcement learning library —concepts, requirements, and implementation—. In *the 10th IEEE-RAS International Conference on Humanoid Robots (Humanoids'10)*, pages 118–123, Nashville, TN, US, 2010.

[23] Akihiko Yamaguchi, Masahiro Oshita, Jun Takamatsu, and Tsukasa Ogasawara. Experimental verification of learning strategy fusion for varying environments. In *the 10th ACM/IEEE International Conference on Human-Robot Interaction Extended Abstracts (HRI2015)*, pages 171–172, Portland, 2015.

[24] Akihiko Yamaguchi, Jun Takamatsu, and Tsukasa Ogasawara. Learning strategy fusion to acquire dynamic motion. In *the 11th IEEE-RAS International Conference on Humanoid Robots (Humanoids'11)*, pages 247–254, Bled, Slovenia, 2011.

[25] Akihiko Yamaguchi, Jun Takamatsu, and Tsukasa Ogasawara. DCOB: Action space for reinforcement learning of high dof robots. *Autonomous Robots*, 34(4):327–346, 2013.

[26] Akihiko Yamaguchi, Jun Takamatsu, and Tsukasa Ogasawara. Learning strategy fusion for acquiring crawling behavior in multiple environments. In *the 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO'13)*, pages 605–612, Shenzhen, China, 2013.