

Control of Microrobots with Data and Computationally Efficient Reinforcement Learning

Nathan Lambert, Kristofer Pister, Roberto Calandra

Progress

Substantial progress has been made in recent time on dynamics modeling for control. Two manuscripts are nearing publication:

1. Learning Accurate Long-term Dynamics for Model-based RL in the Conference on Decision and Control
2. Compounding Error in One-step Deep Dynamics Models is under preparation.

Planning

It is an exciting time where finally we have some funding, students, and therefore broader impetus to work on the intersection of machine learning and Microrobots. In this memo I address three things:

1. The motivation of why this is an interesting problem at multiple levels of the stack,

2. My contributions, both published and unpublished,
3. The many paths we have going forward from here.

I. Motivation: We have almost no data, it is hard to understand what we have

Microrobotics, and any Microsystems research, has a very different publication pathway than most of computer science research. The problem is primarily getting something to work once. This can be exemplified in two ways: fabrication / yield problems and testing / experimental problems. The ionocraft has a simple fabrication process but was hard to test (and assemble), but now the MEMs devices for the motors have a more complicated process (due to many, many more moving parts and constraints), but assembly and testing is a bit lower risk.

Therefore, given that one of these data constraints is likely to always exist **we want a method to be extremely sample efficient**. When Googling this, I don't even think sample efficient is the right word, because we are not doing theoretical work where we *want* it to be more sample efficient, we *need* it to be so, or we will not solve the task. I call this in my work **minimum-data reinforcement learning**.

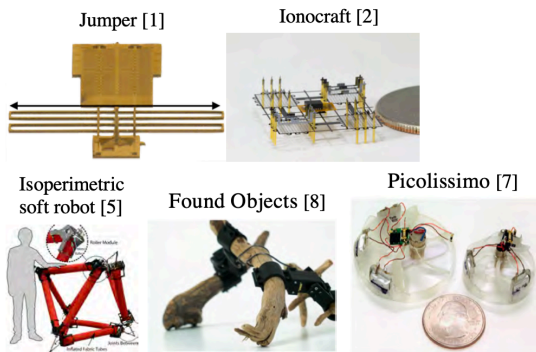
Second is an unavoidable problem of cutting edge research. With microrobotics, we will likely never have a perfect model of the dynamics. Our simulations will never be perfect. We do not operate like TSMC & ASML. Therefore, we want methods (or students), but hopefully methods, that and learn from data and **reason with uncertainty** to update the

design and or control. Progress in machine learning and deep learning particularly has shown a ton of promise in utilizing function approximation for complex tasks and datasets.

Historically, model-based methods have filled the void of sample-efficient and handling uncertainty for people. This is where I started. If you want to watch a talk with a intro to why microrobots, you can see one of my last **practice quals** or the beginning of a **seminar** I gave at Cornell, **slides**.

Novel robotic platforms

Microrobots



Demonstrating control for the first time:

1. No strong prior on robot dynamics
2. High cost-per-test

The method for control needs to:

1. Manage uncertainty
2. Be sample efficient

[1] C. B. Schindler, J. T. Greenspan, H. C. Gomez and K. S. J. Pister, "A Jumping Silicon Microrobot with Electrostatic Inchworm Motors and Energy Storing Substrate Springs," 2019 20th International Conference on Solid-State Sensors, Actuators and Microsystems & Eurosensors XXXIII (TRANSDUCERS & EUROSENSORS XXXIII), Berlin, Germany, 2019, pp. 88-91.
[2] Drew, Daniel S., et al. "Toward controlled flight of the ionocraft: a flying microrobot using electrohydrodynamic thrust with onboard sensing and no moving parts." *IEEE Robotics and Automation Letters* 3.4 (2018): 2807-2813.
[3] Contreras, Daniel S., Daniel S. Drew, and Kristofer SJ Pister. "First steps of a millimeter-scale walking silicon robot." 2017 19th International Conference on Solid-State Sensors, Actuators and Microsystems (TRANSDUCERS). IEEE, 2017.
[4] Rauf, Abdul M., et al. "Towards Aerodynamic Control of Miniature Rockets with MEMS Control Surfaces." 2020 IEEE 53rd International Conference on Micro Electro Mechanical Systems (MEMS). IEEE, 2020.
[5] Usevitch, Nathan S., et al. "An untethered isoperimetric soft robot." *Science Robotics* 5.40 (2020).
[6] Yim, Justin K., and Ronald S. Fearing. "Precision jumping limits from flight-phase control in salto-1p." 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2018.
[7] Piccoli, Matthew, and Mark Yin. "Picolissimo: The smallest micro aerial vehicle." 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2017.
[8] Makiwaka, Azumi, et al. "Improvised Robotic Design with Found Objects."

Slide from my quals summarizing this problem.

2. History: Model-based Reinforcement Learning (MBRL) as a case study

This is the level of the stack where I took a total leap of faith in the spring of 2018. We wanted to control the ionocraft because a) it is super hard to do with any method and b) would be super cool to use a learning method rather than struggling through hand PID tuning. This turned into controlling a quadrotor, which was the closest system we could come up with.

Starting with class projects in **SP18** (Hybrid Systems with Tomlin and Machine Learning with Sahai) I proceeded to continue working on this over the summer, while helping Dan and Kris with the Ionocraft. In the **FA18** Deep RL course, the project was starting to take form. After a **rejected version** at ICRA that year, the final version was **accepted into IROS** the following spring. So, it took a solid year+ of work to get this out the door (looking at the versions is interesting, but mostly to show the process this can take if you want to take a leap of faith and try to work on this).

Now, there is *way* more support of projects in this vein. There is more work on it directly with Kris and the **old guard** and some **M.S. students**. There is work I have had generous opportunities to work with Roberto on external to Berkeley (**example, example**). Working on this stuff has opened a lot of doors for me, it's really exciting. There is generally a lack of people a) willing to work on it seriously and b) with a certain niche of skills that makes them useful — for all of you that is easily microrobotics and related EE hardware design.

3. Future Work: optimization across the stack of novel robotics

I have ordered these in what I view as most possible in a learning side to least possible.

The X-factor of applying these to hardware makes this trickier. For example, doing any control synthesis on hardware is a huge win, and I would **love** to work on this for something like the walking robot. It seems like the timeline for this does not overlap with my critical path, so this may turn into a collaboration at whatever job I get next (hopefully industry research that allows collaborations).

Here is a short summary of problem spaces I discuss:

- **Black box optimization:** data-driven optimization of design where human tuning parameters is hard (high dimensions, nonlinear);
- **Morphology learning:** specific design optimization applied to how a robot will move and or how it operates;
- **Co-adaptation:** jointly optimizing the robot design and the downstream controller for single- or multi-task control;
- **Multi-agent control:** applying our background in swarms to high-dimensional control tasks (kind of a long shot);
- **Controller synthesis:** continuing some of my algorithmic work on model-based RL for controlled generation (kind of a long shot as well);

Black box design optimization

There is also a white paper I started that someone could take over, and potentially turn into a high-impact paper with **any** real world results (**pdf**, **overleaf**). It is the idea of using machine learning to optimize design. We have a metric function we want to optimize, this takes the high-dimensional optimization off your hands.

In the past I even made a **GitHub repo** to try and make it easier to onboard people here.

Linking to some of the work I have done in model-based RL, there is some **recent works** on model-based optimization of design. Microrobots / real hardware are way more interesting than what has been done, but would likely be worth starting in a simulator. Kris can chime in with some talks he made 20+ years ago on Sugar and accurate MEMs simulation.

Example deliverable: Optimize yield of MBRL robots with human-computer joint design optimization.

Morphology learning & co-adaptation

I broke morphology off into its own category because it is much more focused on locomotion and structural design of how it operates, rather than just optimizing an arbitrary MEMS function (it is a subset of above). Kris has already had multiple generations of undergrads working on things related to this (for example, here is **work** from Brian Yang and Grant Wang on gaits— they presented it at one of the first group

meetings I attended in graduate school). Co-adaptation is when you include the controller design in the problem formulation (a bit more advanced). Roberto has a lot more work here (some with Kris, some ongoing with me and Mark):[example 1](#) [example 2](#).

Example deliverable(s): Apply a black-box optimization task to an already existing robot structure (leg length, number of legs, motor force, etc.) and show performance tuning over iterations; Apply an existing MBRL algorithm to joint optimization of design and control on a simulated Microrobot control task.

Bridging multi-agent learning and hierarchical control

Something that the framing of working on microrobots gives you is a **fundamental expectance of multi-agency**. What I mean here is: anyone working on microrobots bakes into the motivation of their work that eventually we will make a lot of these. Therefore, any method we have to control them must be able to hand high-dimensional input spaces (many agents). There is a fundamental problem in model-based RL methods that they have not solved higher dimension tasks. Therefore, there is potential high impact work with a potentially different approach: break down the complicated control problem for one agent as if different sections are sub-agents in a multi-agent control problem.

This came up when discussion BotNet ([code](#), [paper](#)) and our continued working group on multi-agent control.

Example deliverable: use hierarchical / partially centralized MBRL to control the Humanoid environment.

Control synthesis

This work would be addressing many of the open questions in model-based RL research. There is plenty to do, but it may be easier to learn and enter from starting in another area with lower hanging fruit (application work).

Example deliverable(s): apply MBRL to a walking real-world hexapod; developing further sample-efficient MBRL algorithms by considering objective mismatch.

Conclusions & calls to action

There is a lot of work here. It is hard to make time to dive into something new, but there is also a support network to work on it, so who's up for it?