In my undergraduate career, I have had the opportunity to work on several diverse research projects. While working on both mostly individual projects and as a member of a larger research team, I have both discovered my passion for Computer Science research as well as my interest in novel computing architectures. I have learned firsthand that academic research provides both the flexibility and opportunity to make a broader contribution to other fields and industry.

Working with the Harvard Robobees project, I have learned the complexities of designing systems software for novel computing architectures, specifically swarms of micro aerial vehicles. As a member of Professor Matt Welsh's group, I have contributed to the design of a new distributed operating system for our future MAV swarms, Karma. Throughout several months of design-by-consensus meetings, we designed interfaces for coordination and control of the swarm, including resource management, communication, and programmability that would be robust in the face of regular node failure, communication unreliability, and a lack of global knowledge. While group members built Karma in simulation, I concurrently worked on building a real-world MAV testbed using micro-helicopters in order to rigorously evaluate our design with realistic constraints. I developed an API and computer control, low-level feedback loops, and a TinyOS AVR32 port and drivers that we are integrating with a Vicon motion capture system for greater positional accuracy. With Bryan Kate, a graduate student, I also explored the feasibility of constructing a radio-based communication and navigation backbone using landed bees both in simulation and in limited human-directed trials using sensor motes that we plan to integrate as a system service. This work is currently ongoing.

Working on Karma exposed me to systems building from the ground up and has taught me not only how difficult real distributed systems design is, but also how collective design can lead to more robust problem solutions. Our RoboBee swarms will exhibit many of the properties that make distributed systems both fun and challenging, and reasoning about the right programming abstraction and set of services that we should present the programmer while still allowing robust operation has been exciting. RoboBees is intended for real-world deployment and will eventually contribute to UAV solutions for commercial pollination (counteracting colony collapse disorder), hazardous search and rescue, and other tasks. Therefore, the design decisions we make have real implications for eventual end-users, requiring additional consideration. Working with four other members on this project has been a great experience, as we each are able to contribute our strengths.

I have also worked with Professors Margo Seltzer and David Brooks on software-assisted microarchitecture in a data center context. Because power and cooling costs represent a significant portion of data center operating costs, we developed an operating system solution that allows applications to absorb portions of normal cooling overheads the instead of relying solely on physical infrastructure like chillers and air conditioners. Dimetrodon targets average-case reductions in data center operating temperatures by flexibly degrading application performance, allowing decreased energy costs which are critical for both environment-friendly data centers and for energy efficiency.. Unlike reactive techniques such as Dynamic Thermal Management, we proactive target reductions in cooling costs. I implemented Dimetrodon in FreeBSD, patching the scheduler to allow each processor core to cool by periodically scheduling idle quanta. With the help of Dr. Vijay Reddi, I evaluated Dimetrodon's effect on system heat, performance, and efficiency across several workload types on an actual server. From these results, we generated predictive models for the tradeoff between performance and average heat dissipation (typically at least 1:1, but up to 1:15). We applied them to actual data center cost and operation models, and our results are currently under submission.

This work taught me the difficulty of working with real hardware and operating systems, but also the benefit of doing so and the need for ground truth. While we could have explored our ideas in simulation, hacking a commodity operating system and gathering real data gives our results more authenticity and weight. Taking an idea, developing and evaluating it, and driving a project from inception to completion was a valuable experience. Unlike the Karma project, which was more group-oriented, I was the lead on this project, and I learned that with the benefits of creative direction come the requirements of moving the work forward.

With Professor Radhika Nagpal and Dr. Justin Werfel, I spent several months exploring the role of private and social information in bee foraging models. This work began as a class project in which, with a classmate, I built a bee foraging model and simulation, drawing from the biological literature. Shortly afterwards, Justin and I learned about a recent controversy in biology regarding the usefulness of sharing positional information about food sources; several studies found that recruited bee foragers chose to ignore positional information, seemingly contradicting years of evidence supporting a cooperative-behavior hypothesis. Under Radhika and Justin's guidance, I expanded our initial work to shed further light on this controversy by refining our model, rewriting the simulator, and performing extensive multi-parameter studies to determine when sharing positional information is beneficial to bee colonies. We subsequently found that in several environmental types, relying solely on private information can actually be more beneficial than sharing positional information, largely due to overconcentration of foraging efforts. I wrote all code in our final simulator, performed all data collection and most of the analysis, and again, with advisory support, wrote our manuscript. This work, while not expressly systems research, taught me how to build a model based on relevant literature, how to perform large state-space exploration of a model, and how to interpret the results, which we subsequently shared with several biologists actively studying bee foraging. I presented our findings in Brussels at ANTS 2010, a swarm intelligence conference, where our manuscript received the Best Student Paper award.

Conference Papers

- "Positional Communication and Private Information in Honeybee Foraging Models." Peter Bailis, Radhika Nagpal, and Justin Werfel. In *Proceedings of the Seventh International Conference on Swarm Intelligence (ANTS 2010)*, September 2010, Brussels, Belgium. Received **Best Student Paper Award**. Also gave talk and poster at conference.
- "Dimetrodon: Mitigating Cooling Overheads in Data Center Thermal Management via Idle Cycle Injection." Peter Bailis, Vijay Janapa Reddi, Sanjay Gandhi, David Brooks, and Margo Seltzer. Under submission.
- "The Perfect Swarm: Challenges and Opportunities in Coordinating Large-scale, Real-world MAV Deployments," Peter Bailis, Karthik Dantu, Bryan Kate, Jason Waterman, and Matt Welsh. Submitted to *Twelfth Workshop on Mobile Computing Systems and Applications (HotMobile 2011)*.
- "All the World's a Data Source, and All Its Databases Merely Caches." Peter Bailis and Margo Seltzer. Submitted to *Fifth Biennial Conference on Innovative Data Systems Research (CIDR 2011)*, Outrageous Ideas and Vision Track.

Posters

- "Kárma: A Distributed Operating System for Micro-UAV Swarms." Peter Bailis, Karthik Dantu, Bryan Kate, Jason Waterman, and Matt Welsh. Poster and Extended Abstract at *Ninth USENIX Symposium on Operating Systems Design and Implementation (OSDI 2010)*, October 2010, Vancouver, Canada.