

Mitchell Keith Bloch

PhD, Computer Science & Artificial Intelligence

 University of Michigan

+1 (734) 926-5042 

bazald@gmail.com 

bazald.com 

github.com/bazald 

[linkedin.com/in/bazald](https://www.linkedin.com/in/bazald) 

Education

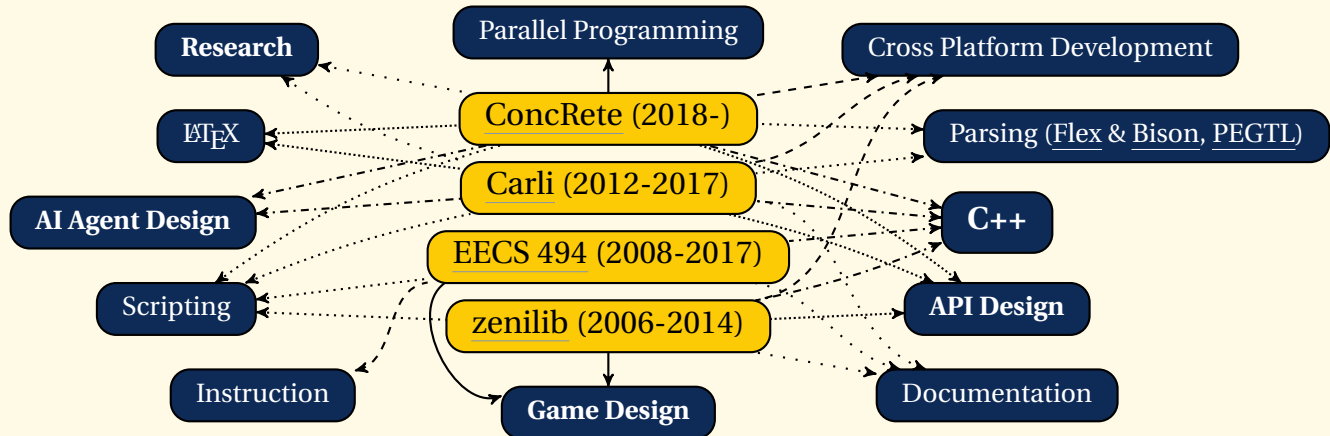
Doctor of Philosophy, Computer Science and Engineering (University of Michigan, August 2018)

Master of Science in Engineering, Computer Science and Engineering (University of Michigan, 2010)

Bachelor of Science in Engineering, Computer Science, *Summa Cum Laude* (University of Michigan, 2008)

Projects

Skills



Since graduating, I have explored lock-free algorithms and created **ConcRete**, a C++ implementation of a parallel Rete. It supports lock-free rule insertion and excision as well as Working Memory Element (WME) insertion and removal. Hash Array Mapped Trie (HAMT) and Ctrie variants are used extensively. I additionally implemented a lightweight rule based system on top of ConcRete which uses PEGTL for parsing.

I implemented the **Carli** Relational Reinforcement Learning (RRL) architecture for my thesis, *Computationally Efficient RRL*. Using Rete as the basis for implementing Adaptive Tile Codings (ATCs) that can incorporate relational features, my agents demonstrated more advanced capabilities than the previous state of the art and achieved a several orders of magnitude reduction in computation time.

I taught **EECS 494 – Game Design and Implementation** for two semesters after having assisted with the course for many years. We taught the course using **zenilib** for six years before switching to Unity. I was responsible for engine development and support up until the time of that switch. I was additionally responsible for redesigning the class syllabus and schedule, changing the final projects to consist of multiple sprints in the context of agile development, the decision to incorporate a formal presentation into the final showcase for the class, and managing two teaching assistants.

I started developing a cross-platform game engine, **zenilib**, as a personal project to learn more about game development. I ultimately used it to teach for Wolverine Soft, Camp CAEN, and EECS 494. I helped high school students to make their first games in C++ in a mere two weeks. I enabled students in EECS 494 to move on from making only 2D games to making 3D games. I additionally used **zenilib** for personal projects, including Coriolis Station which my teammates and I designed for Sid Meier's Game Design Boot Camp.


Posters and Presentations

Deciding to Specialize and Respecialize a Value Function for RRL (RLDM 2017)


Automatic Value Function Refinement and Unrefinement for RRL (The 36th Soar Workshop)

The Carli Architecture—Efficient Value Function Specialization for RRL (RLDM 2015)

Hobbies

 Argentine Tango

 Karaoke

 Burning Man