TEACHING STATEMENT

My goal in teaching is to **develop students as critical thinkers and problem solvers**, equipping them with a **solid foundation in mathematics** and the **ability to apply mathematics to real-world challenges**. I aim to help my students develop rigorous thinking and intuition, enabling them to connect abstract models with real-world problems and apply theoretical methods effectively. I have developed my teaching skills and interests through teaching and mentoring students in tutoring calculus and real analysis in the mathematics department at SUSTech, and more recently tutoring GNU $\TeX_{\text{MACS}}$/Mogan and Goldfish Scheme at LIII NETWORK. As a professor, I am qualified to teach **graduate courses on control, reinforcement learning, and game theory**. I am also enthusiastic to teach **undergraduate courses in programming and discrete mathematics.**

TEACHING PHILOSOPHY

When I am preparing a class or a talk, I focus on these 3 questions:

1. What do I want students to remember two years after forgetting the technical details?
2. What real-world problems can the class equip students to solve with diverse approaches?
3. Which skills gained from this course will remain irreplaceable by AI in the next decade?

Keep students motivated and intuitive throughout: These 3 questions guide me in designing a class where students grasp the motivation and intuition behind the technical details. I believe the real goal of teaching is not to make students "memorize" technical details but to help them "understand" intuition and motivation—especially in an era where LLMs remember technical details but lack intuition. Therefore, clearly link each technical detail to its intuition and motivation and organize class materials to follow a logical flow of high-level intuition, supporting better teaching topics and time allocation. By answering these 3 questions, I adopt the backward-design process to align the course content with my teaching philosophy.

I have put the backward design to the linear algebra, calculus, and SICP courses[1] provided by SUSTech and LIII NETWORK. Over 300 students from diverse backgrounds attended these courses, including mathematics, finance, computer science, and other disciplines across various universities. In this role, I have given many lectures to students with diverse mathematical foundations and varying interests. A recurring challenge in teaching students with diverse backgrounds is their struggle to grasp technical details and concepts outside their areas of familiarity. To address this challenge, I designed a high-level motivation with real-world examples throughout the course, simplifying technical details within this framework. For instance, in linear algebra, the overarching motivation represents a linear mapping, illustrated by a 2D rotation mapping. All concepts in linear algebra connect to this example, enabling students to draw and explore 2D examples, fostering a deeper intuitive understanding.

Using interactive toolkits for in-class activities: I strive to include interactive toolkits in my teaching instead of non-interactive ones, such as PPTs. It has been proven to increase student learning and is well-suited to encourage students to connect theories with real-world problems. I am a co-founder of LIII NETWORK that developed GNU $\TeX_{\text{MACS}}$/Mogan, a WYSIWYG $\TeX$-like editor with built-in interactive toolkits such as Python, Julia, and Mathematica REPLs. Mogan is used in the SICP course we offer, a renowned class originally developed by MIT in the 1990s to teach programming and foster a programmer's mindset in students with no prior foundation. In this course, Mogan is a slide player, an IDE for running code, a graphics drawer, and even a virtual blackboard, enabling users to type equations as efficiently as writing on a physical blackboard. Running code is essential for teaching programming, a blackboard is ideal for presenting mathematical details, and drawings and slides effectively illustrate high-level ideas. Mogan integrates all these functionalities with LLMs into a single document, enabling students to connect mathematical concepts (blackboard) with computational details (code) and real-world applications (slides). Additionally, the built-in LLMs allow me to compare traditional and LLM-generated content during class, fostering deeper discussions. Mogan documents are also easily distributable for out-of-class learning. With Mogan, teaching becomes more interactive and engaging in the era of LLMs.

[1] Some lectures are open accessed on the internet. For example, "What is linear, what is algebra, and what is linear algebra?" and "SICP with Mogan and Goldfish Scheme".

TEACHING INTERESTS

I am eager to teach graduate courses on control theory, reinforcement learning, and game theory. I'd also like to teach a class on how these control and learning algorithms can be applied into real-world systems, not just applied in abstract mathematical models. In addition, I'd like to design a seminar on how LLMs can be applied in decision-making to train students to rethink traditional methods in the era of LLMs.

Moreover, I am prepared to teach undergraduate courses in programming, probability, optimization, and discrete mathematics, with the opportunity to relate fundamental mathematical concepts to computer science.

MENTORING

I have mentored many undergraduates in applying for Ph.D. programs in STEM. The case I am most proud of is an undergraduate student majoring in pure mathematics at SUSTech, who faced challenges with a low GPA and struggled with abstract mathematics. I helped him learn several engineering courses closely related to the mathematical concepts he had studied. Through these courses, he improved his mathematical thinking and understanding, which led to a higher GPA. He also developed a strong interest in engineering and successfully secured admission to a Ph.D. program in ECE at the University of Florida.

By mentoring students from the Departments of Mathematics, Statistics, Physics, and Computer Science to pursue Ph.D. programs in engineering, I have learned how to guide students in leveraging their academic backgrounds to effectively transition into other fields. These students are now enrolled in Ph.D. programs at institutions such as the University of Florida, the University of Illinois, Duke University, UW-Madison, and others[2].

LEADERSHIP

I am a co-founder of LIII NETWORK, an open-source company which developed GNU TeXmacs/Mogan, a WYSIWYG TeX-like editor featuring built-in interactive toolkits such as Python, Julia, and Mathematica REPLs. Mogan is an open-source project licensed under the GPL, with its source code available on GitHub: Mogan. Mogan is written in C++ and Scheme, featuring a Scheme interpreter, Goldfish Scheme, which we developed and open-sourced on GitHub: Goldfish.

I led a team of undergraduate and graduate students to develop Goldfish Scheme, a project aimed at enhancing usability by incorporating Python-like features into a Scheme interpreter. Throughout the process, I coordinated the team's efforts, fostering collaboration and guiding them through problem-solving and decision-making. This experience allowed me to demonstrate leadership, inspire creativity, and mentor students in a hands-on development environment, reflecting my commitment to cultivating a supportive and innovative learning atmosphere.

[2] For example, see: Yingyao Zhou, Chongyang Shi, Junze Deng, Du Chen, and Langtian Ma.