Teaching Statement of Albert Jiang

I understand the challenges facing education in the computing and mathematical sciences, where the subjects are often seen as too dense and difficult by students. I believe that learning computer science can and should be fun, and my teaching goal is to develop students' **interest** in the topic, especially the joy of problem solving. I believe that a key technique for effective teaching is to incorporate **interactive** elements, such as games, lab exercises, and group discussions. I believe the most important lesson one can learn is **how to learn**. Computer science is a young and fast-evolving field, without lifelong learning the students' knowledge and skills will grow outdated. Students should be encouraged to learn beyond the scope of the course material, via (e.g.) extra-credit exercises and course projects. I believe that it is important to recognize that students are **diverse** with different backgrounds, goals and learning styles, and to adapt my teaching to allow them to succeed.

I have extensive experience as instructor, research supervisor, tutorial lecturer, teaching assistant and private tutor. Computer science topics that I am able to teach include introductory computer programming and architecture, advanced undergraduate-level artificial intelligence and algorithms, as well as advanced research topics in my research field of computational game theory and multiagent systems. I am also passionate about mentoring and supervising students in doing cutting-edge research. Below I will briefly highlight my teaching experience, and then outline my future teaching interests.

Teaching Experience

Together with Prof. Milind Tambe, I co-taught *CSCI 543: Software Multiagent Systems* in Spring 2012 and *CSCI 599: Security and Game Theory* in Spring 2013, both graduate-level courses at the Computer Science Department at University of Southern California. The former is an introduction to multiagent systems and game theory, with emphasis to applications to real-world challenges. The latter focuses on applications of computational game theory to security. My responsibilities included giving lectures, designing and marking assignments and exams, and meeting with students to advise on their course projects. Furthermore, I helped revise and expand the course materials. For instance, I developed and gave a lecture on linear programming duality and applications to game theory, and another lecture on correlated equilibrium and its relation to Stackelberg equilibrium, based on recent progress in this research area.

I have participated in the **writing of teaching materials** aimed towards advanced undergraduate and graduate students, which were included in Yoav Shoham and Kevin Leyton-Brown's textbook Multiagent Systems: Algorithmic, Game-theoretic and Logical Foundations, published by Cambridge University Press in 2009. In particular, I wrote Section 3.3.4, which provides a detailed and elementary derivation of Nash's famous theorem on the existence of Nash equilibria in finite games. I also contributed to portions of Sections 6.5.2 and 6.5.3 of the textbook, on compact representations of games. From this experience of writing teaching materials on advanced technical topics, I learned to emphasize the high-level ideas, while at the same time maintaining mathematical rigor.

I have given tutorials on the research topic of game theory for security, at the ACM Conference on Electronic Commerce in June 2012, at the AAAI conference in July 2013, and at the Center for Risk and Economic Analysis of Terrorism Events (CREATE) at the University of Southern California in August 2012 and again in September 2013. These half-day to full-day tutorials were aimed to introduce the key concepts of the research topic to a non-expert audience. I gained valuable experience developing and giving lectures.

Teaching Statement of Albert Jiang

From 2003 to 2010, I worked as a teaching assistant for various computer science courses at University of British Columbia, including first- and second-year computer programming and computer architecture, advanced undergraduate level algorithms and artificial intelligence, and graduate level multiagent systems. My duties included leading tutorials and labs, marking assignments and exams, and assisting students in online discussion boards. I was especially experienced in interacting with students in labs, where I tried to guide the students' thinking processes as they try to write programs to solve exercise problems.

I have worked as a private tutor on computer science topics. In the tutoring setting there is more freedom in tailoring the course to the specific interests and needs of the students, and more opportunities to interact with the students and sense their progress. I will highlight two instances. In 2007, I tutored a student who enrolled in an undergraduate artificial intelligence course but had to miss a majority of lectures due to work. I taught the missed lectures using the slides from the course website. In 2010, I taught computer programming to a middle school student with no prior programming experience. After discussion with the student we decided on the Python programming language. Once the student was familiar with the language, I started to teach him basic algorithms.

From 2007 to 2011, I was a chess instructor at Ho Math & Chess in Richmond, BC, Canada. I taught two one-hour sessions each Saturday; each session had two to five teenaged students. The content of the classes included lectures, puzzles and practice games. Here my goal was to make sure that the students were having fun and at the same time gaining experience on problem-solving. I was able to assess the strength of each student and decide on the right level of handicap when playing chess games, and to capture their interest by having prizes and introducing them to variants of chess.

In August 2010, I attended a three-day instructional skills workshop held by the Centre for Teaching and Academic Growth (TAG). The workshop was designed for graduate students interested in teaching, and consisted of sessions on relevant topics as well as teaching practice. During the workshop each participant taught three short lessons and received feedback from peers. For one of the short 10-minute lessons I chose a challenging topic: a proof that the square root of two is irrational, and tried to explain the proof in a way that my audience (graduate students from various disciplines) could understand. From the workshop I gained valuable knowledge and experience on designing and delivering interactive lessons. More information about the workshop is available at http://ctlt.ubc.ca/programs/all-our-programs/instructional-skills-workshops/.

Mentoring Experience

I have extensive experience mentoring and co-supervising students. As a senior PhD student in Prof. Leyton-Brown's game theory and multiagent system research group at University of British Columbia, I worked with undergraduate student Damien Bargiacchi from 2007 to 2009 to develop a graphical user interface for a research project (http://agg.cs.ubc.ca). In 2008, I mentored graduate student Kevin Swersky on his course project for Prof. Leyton-Brown's multiagent systems course. As a postdoctoral researcher in Prof. Tambe's research group at University of Southern California, I unofficially co-supervised seven students (Fei Fang, Leandro Marcolino, Thanh Nguyen, Eric Shieh, Chao Zhang, Yundi Qian, Kim Chu), ranging from undergraduate to junior PhD level. My responsibilities typically included meeting with the students regularly, advising on research directions and priorities, and attending some of the students' weekly meetings with Prof. Tambe. I also

co-authored papers with all of these students, resulting in publications at top conferences and journals including AAMAS, IJCAI, GameSec, and JAIR. From these experiences I learned how to listen to students, to ask relevant questions, to explain technical concepts, to suggest research topics based on students' different interests and backgrounds, and to share my research philosophy and experiences.

Teaching Interests

Given my teaching experience and knowledge of computer science, I believe I am qualified to teach any core computer science course in the at the introductory undergraduate level, as well as advanced undergraduate and graduate courses in the areas of artificial intelligence, multiagent systems/game theory, optimization, and algorithms. I outline below the courses and topics that I feel best qualified for and most excited to teach.

- **Artificial Intelligence.** This can be an advanced undergraduate- or graduate-level course. Topics may include: advanced search and problem-solving techniques, knowledge representation, reasoning under uncertainty, Bayesian networks and other graphical models, decision theory and game theory, planning, Markov Decision Processes, machine learning, reinforcement learning.

- **Algorithms,** an advanced undergraduate course. Potential topics: advanced data structures, dynamic programming, greedy algorithms, graph-based problems, Monte Carlo methods, computational complexity.

- **Optimization**, an advanced undergraduate- or graduate-level course. Topics may include: convexity, linear programming, duality, the simplex algorithm, the ellipsoid algorithm, integer programming and applications to combinatorial optimization, convex optimization, gradient methods.

- **Introductory Multiagent Systems/Game Theory,** an advanced undergraduate- or graduate-level course that introduces the key concepts in this young and rapidly-growing research area. Topics may include: utility theory, normal form games, extensive form games, Bayesian games, Nash equilibrium and other solution concepts, basic equilibrium computation, social choice, mechanism design, auction theory, price of anarchy, repeated games, stochastic games, multiagent reinforcement learning, bounded rationality and behavioral game theory, security games, cooperative game theory.

- **Advanced Computational Game Theory and Applications.** This is an advanced PhD-level course that gives an in-depth look at the cutting edge of a hot research topic: game-theoretic computation and its applications to security and other real-world challenges. I will develop the course materials from existing textbooks as well as recent research papers. Topics may include: complexity of Nash equilibrium computation, linear complementarity program, game representations, Action-Graph Games with application to auctions, algorithms for computing solution concepts, sequence form with application to poker playing, no-regret learning, Stackelberg equilibrium computation with applications to security, scaling up to large security games, robustness and uncertainty, optimizing against human adversaries.