Computational Science and Engineering Program: Graduate Student Handbook

September 2009

1. Program Description and Objectives

Computation has become accepted as the third mode of discovery, along with theory and experimentation, in the advancement of scientific knowledge and engineering practice. Computational models are now used routinely in virtually all fields of science and engineering to better understand systems and phenomena as large as the universe, or as small as the tiniest microelectronic circuits and nanomaterials. Computation is essential in critical applications with great societal impact such as the sustainable growth of cities, the design of power-efficient buildings and aircraft, and the creation of novel biomedical devices, effective drugs, and efficient health care delivery systems, to mention a few. Computational science and engineering is a discipline devoted to the systematic study of computer-based models of natural phenomena and engineered systems. The CSE discipline is defined around the body of knowledge and professional practices that are integrated by combining elements from computer science, applied mathematics, engineering and science. It is a distinct discipline in its own right. For example, modeling and simulation is one sub-field of the CSE discipline and have become indispensable in defense applications such as war-gaming, logistic support, test and evaluation of new equipment, and training. It is concerned with creating useful, practical abstractions of systems that are grounded in mathematical and theoretical foundations, are represented using computing hardware and software, and are customized to meet requirements and constraints arising in science and engineering. Like the CSE discipline, the modeling and simulation sub-field is domain independent. However, exploration of this area requires a deep appreciation and knowledge of a specific domain in order to provide context and concreteness to guide the definition of compelling research problems and appropriate solutions. A second subfield of the CSE discipline involves algorithms for solving and manipulating continuous and discrete computational models. A third concerns the analysis, knowledge discovery, and visualization of models derived from empirical or simulation data. A forth is concerned with computational techniques for high performance computing, focusing on the integration of algorithms (e.g., parallel algorithms), data structures, software, machine architecture, and domain knowledge to achieve efficient realization of computational models that cannot be realized on conventional computing platforms because of their large computational and/or storage requirements.

The objectives of the CSE graduate programs include:

- Create students that are able to advance the state of knowledge and/or practice in the computational science and engineering discipline through their innovative contributions.
- Create students who are able to integrate and apply principles from mathematics, science, engineering, and computing to innovate, and create computational models and apply them to solve important real-world problems.

- Create students who are able to innovate and contribute in multidisciplinary teams that include individuals whose primary background is in computing, mathematics, science, and/or engineering.
- Prepares individuals for careers in industry, government (e.g., national laboratories), and academia, both in terms of knowledge and computational (e.g., software development) skills.

The program will meet many of its educational objectives through its curriculum. The curriculum is designed with the following principal educational goals:

- Students will develop a solid understanding of fundamental principles across a range of core areas in the computational science and engineering discipline.
- Students will develop a deep understanding and set of skills and expertise in a specific computational specialization of the computational science and engineering discipline.
- The students will be able to apply and integrate the knowledge and skills they have developed and demonstrate their expertise and proficiency in an application area of practical importance.
- Students will be able to engage in multidisciplinary activities by being able to communicate complex ideas in their area of expertise to individuals in other fields, be able to understand complex ideas and concepts from other disciplines, and be able to incorporate these concepts into their own work.

2. Desired Qualification of Students

Students admitted to the program must be able to demonstrate the following competencies:

- An undergraduate level understanding of concepts from computer science, applied mathematics, a physical science (e.g., physics, chemistry, or biology) or engineering. This will typically be demonstrated by a bachelor's degree in one of these subject areas. Students with a different background may also apply, and will be evaluated on a case-by-case basis. All aspects of the applicant's background will be considered, including work or other academic experience in deriving admission decisions.
- Computing skills in algorithms, data structures, and programming in a high level language such as C or FORTRAN is required. This requirement is minimally satisfied by an introductory computer science course, however, at least two semester courses are strongly recommended.
- Undergraduate mathematics in calculus is required. Undergraduate course work in areas such as mathematical analysis, numerical differential equations, linear algebra, discrete mathematics, and probability and statistics are also recommended, and may be required for certain programs of study selected by the student.

Students failing to meet one or more of these competencies may apply, however, they will be expected to satisfy deficiencies in their background, e.g., by complete preparatory coursework.

3. Home Units and Home Unit Requirements

Each student in the CSE graduate program is affiliated with a *home unit*. A home unit is an academic unit (Department, Division, or School) at Georgia Tech that has agreed to formally participate in the CSE program. The home unit is determined by mutual consent of the student and the home unit. An initial home unit is determined during either the admissions process or in the process of transferring to the CSE program from another academic program at Georgia Tech. Once admitted, the student may change to a new home unit with the concurrence of that unit.

Allocation of space and financial assistance (e.g., teaching and research assistantships) are governed by the rules of the home unit in which the student resides. For doctoral students, the PhD advisor should have an appointment in the student's home unit, in addition to being a member of the CSE program faculty. Students are welcome to explore research opportunities with faculty in other units beyond the student's home unit; if a faculty member in another home unit becomes the student's dissertation advisor, the student would normally change their home unit to that in which the advisor holds an appointment.

All students, regardless of their home unit, must fulfill the degree requirements specified in this document to complete the program. Some home units have additional degree requirements beyond those specified for all students. The home units and specific degree requirements, if any, are specified below:

- *School of Aerospace Engineering*.
    - MS students *must* complete the thesis option of the master's degree program.
- *School of Biology*.
    - No additional degree requirements are specified.
- *Coulter Department of Biomedical Engineering*.
    - Students must take application specialization courses from the BME department, as approved by the BME home unit coordinator.
    - At least two individuals of the PhD dissertation committee must be faculty member from the Coulter BME Department.
    - The student's principal research advisor must be a member of the Coulter BME Department faculty.
    - PhD students must participate in the BME teaching practicum program, and serve as a teaching assistant in BME courses for two semesters and in the BME seminar course for their first two academic years in residence.
- *School of Chemistry and Biochemistry*.
    - Students must take application specialization courses from the School of Chemistry and Biochemistry, as approved by the School's home unit coordinator. Students should enroll in at least one course offered by the school during each semester in which the student serves as a teaching assistant for the school.

- At least three individuals of the PhD dissertation committee must be faculty members from the School of Chemistry and Biochemistry.
- Beyond the second semester, the student's principal research advisor must be a member of the School of Chemistry and Biochemistry faculty to be eligible to serve as a teaching assistant in that school.

- *School of Civil and Environmental Engineering*.
  - No additional degree requirements are specified.
- *Computational Science and Engineering Division (College of Computing)*.
  - No additional degree requirements are specified.
- *School of Industrial and Systems Engineering*.
  - No additional degree requirements are specified.
- *School of Mathematics*.
  - No additional degree requirements are specified.

4. Program Administration and Points of Contact

The programs are administered by the CSE graduate program office, in conjunction with administrative personnel from each of the participating units. The CSE Program office is a centralized location with overall responsibility for administration of the program. It coordinates the various program activities and provides a single interface to the program both from outside Georgia Tech as well as other parts within Tech (e.g., the registrar's office). The *CSE program director* is a faculty member who has overall responsibility for the management and administration of the program.

In addition, each participating unit designates a *home unit coordinator*. The home unit coordinator is a faculty member with overall responsibilities for CSE program activities as they pertain to that home unit, and represents that home unit in administrative activities that pertain to the program as a whole.

Students enrolled in the CSE programs should first consult with their home unit coordinator for advice and recommendations concerning the program, and consult with the CSE program director as needed.

4.1 CSE Program Director

Dr. Richard Fujimoto
Regents' Professor and Chair
Computational Science and Engineering Division, College of Computing
Office: 1320A Klaus Advanced Computing Building
Phone: (404) 894-5615
Email: fujimoto@cc.gatech.edu

4.2 CSE Home Unit Coordinators

Aerospace Engineering
Dr. P. K. Yeung
Professor
Office: 361 Guggenheim
Phone: (404) 894-9341
Email: yeung@mail.ae.gatech.edu

Biology
Dr. Jung Choi
Associate Professor
Office: 213/210 Cherry Emerson
Phone: (404) 894-0519
Email: jung.choi@biology.edu

Biomedical Engineering
Dr. Stephen P. DeWeerth
Professor and Associate Chair for Graduate Studies
Office: 3107 U.A. Whitaker Biomedical Engineering Building
Phone: (404) 894-4738
Email: steve.deweerth@bme.gatech.edu

Chemistry and Biochemistry
Dr. C. David Sherrill
Associate Professor
Office: 2100N Molecular Science and Engineering
Phone: (404) 894-4037
Email: sherrill@chemistry.gatech.edu

Civil and Environmental Engineering
Dr. Leroy Emkin
Professor
Office: 427 Mason
Phone: (404) 894-2260
Email: leroy.emkin@ce.gatech.edu

Computational Science and Engineering (College of Computing)
Dr. Hongyuan Zha
Professor and Director of Graduate Studies
Office: 1314 Klaus Advanced Computing Building
Phone: (404) 385-1491
Email: zha@cc.gatech.edu

Industrial and Systems Engineering
Dr. David Goldsman
Professor
Office: 433 Groseclose
Phone: (404) 894-2365
Email: sman@gatech.edu

Mathematics
Dr. Luca Dieci
Professor and Graduate Coordinator
Office: 215 Skiles
Phone: (404) 894-9209
Email: dieci@math.gatech.edu

4.3 CSE Program Advisor

Ms. Sheila Williams
CSE Graduate Program Advisor
Office: 1321 Klaus Advanced Computing Building
Phone: (404) 385-8529
Email: sheilaw@cc.gatech.edu

4.4 CSE Program Faculty

CSE/CoC
Alberto Apostolico, David Bader, George Biros, Richard Fujimoto, Alex Gray, Guy Lebanon, Haesun Park, Rich Vuduc, Hongyuan Zha

Aerospace Engineering
PK Yeung, Suresh Menon, Sathya Hanagud, Marilyn J. Smith, Olivier A. Bauchau, Lakshmi Sankar

Biomedical Engineering
Mark Borodovsky, Stephen DeWeerth, Melissa Kemp, Robert Lee, Brani Vidakovic, Eberhard Voit, May Wang, Ajit Yoganathan

Civil and Environmental Engineering
Armistead G. Russell, James David Frost, Nelson C. Baker, Leroy Z. Emkin, Rami Haj-Ali, Rafi Muhanna, Don White, Kenneth M. Will, Michael Hunter, John D. Leonard, Jorge Laval, Jochen Teizer, Aris Georgakakos, Jian Luo, Phil Roberts, Thorsten Stoesser

Industrial and System Engineering
Christos Alexopoulos, Dave Goldsman, Seong-Hee Kim, Sigrun Andradottir, Martin Savelsbergh, William Cook, Ellis Johnson, Craig Tovey, George Nemhauser, Arkadi Nemirovski

Biology
Stephen Harvey, King Jordan, Joshua Weitz, Eric Gaucher, Soojin Yi, Nicholas Bergman, Jeff Skolnick

Chemistry and Biochemistry
Ken Brown, David Sherrill, Rigoberto Hernandez, Jean-Luc Brédas

Mathematics
Luca Dieci, Guillermo Goldsztein, Evans Harrell, Vladimir Koltchinskii, Lew Lefton, Yingjie Liu, John McCuan, Haomin Zhou, Silas Alben ,Christine Heitsch, Sung Ha Kang, Jeff Geronimo Tom Morley

5. MS Program Degree Requirements

The program of study is designed to provide a base of knowledge and skills in core CSE areas, in-depth knowledge of advanced computational methods, and experience in applying computational methods to relevant and important problems within the context of at least one specific application domain. The program was designed to provide much flexibility in order to allow students to tailor the program to their individual career objectives.

The Masters in CSE requires a minimum of 30 semester hours. As shown in table 1, this includes a set of required courses defining a core body of knowledge in CSE (12 hours). A set of technical elective courses focusing on developing a more in-depth knowledge of computational techniques as well as the application of computational methods in an application domain (12 hours) is also required. This set of courses will typically form a body of material in close alignment with the student's selected home unit, and form a minor course of study aligned with the home unit. Finally, either a thesis or additional technical electives (6 hours) are required.

Each student is expected to maintain the GPA of at least 3.0 for all courses listed on his/her degree program. All courses listed in the degree program must be taken on the A-F grading basis if the A-F grading basis is offered.

A MS degree must be completed within six years from the date of the first coursework on the degree program, including any transfer credits.

Table 1. Curriculum Overview (30 hours)

Curriculum Component	Semester Hours
CSE core courses	12
Computation and application specialization – home unit minor	12
Additional computation and application electives (non-thesis option)	6
CSE Thesis (thesis option)	6
Total	**30 Hrs**

5.1 CSE Core (12 Semester Hours)

Each student must complete four courses among the five listed in Table 2. Students who have completed a core course, or equivalent course at another institution, prior to entering the program will be allowed to substitute a core course with an additional specialty course, consistent with the student's intended specialization.

Five courses make up the CSE core. These courses are designed to have several objectives:

- Provide the student with knowledge of a variety of areas within the CSE discipline.
- Ensure the students have strong software development skills so that they are able to develop substantial computational artifacts.
- Train the students to integrate and synthesize concepts from mathematics, computing, science and engineering to solve computational problems.
- Develop the student's ability and skills to perform multidisciplinary research involving complex concepts from computing, mathematics, science, and engineering.

Table 2. CSE Core (12 hours; pick any four courses)

CSE/Math 6643 Numerical Linear Algebra	3
CSE 6140 Computational Science and Engineering Algorithms	3
CSE 6730 Modeling and Simulation: Fundamentals & Implementation	3
CSE/ISYE 6740 Computational Data Analysis	3
CSE 6220 High Performance Computing	3

5.2 Computation and Application Specialization – Home Unit Minor (12 Semester Hours)

Twelve hours of technical electives on computation and application areas are required that form a focused area of specialization in CSE in conjunction with the student's declared home unit. The objectives of this specialization include: (1) increase the student's depth of knowledge and skills in CSE computational techniques, (2) equip the student with knowledge of a particular application domain to enable one to attack problems within that domain by applying advanced computational techniques, (3) provide sufficient flexibility for a student to tailor course selections to his or her individual needs and long-term career objectives, and (4) ensure a well-structured, coherent program of study is created. The CSE program is designed to ensure each student develops multidisciplinary skills in at least two areas among computation, science, and engineering.

To fulfill this requirement, the student must take an additional 12 hours of courses that meet the following criteria:

- The set of courses must clearly support graduate work in the computational science and engineering discipline.
- The courses must include coverage of at least one application domain, in addition to providing advanced knowledge of computational techniques; at least one course applying CSE techniques to a specific domain is necessary to fulfill this requirement.
- The set must include at least 6 hours of coursework offered outside computing (i.e., not carrying a CS or CSE course designation).

- The technical specialization to be applied to the CSE degree must be proposed by the student and approved by the home unit coordinator and the CSE program director during the first semester of study in the program to ensure the above requirements will be met upon completion of the proposed program of study.

There are numerous courses offered at Georgia Tech that are appropriate for "graduate work in the computational science and engineering discipline." A partial list of courses is included in an Appendix to provide guidance. Students should consult with their home unit coordinator and/or the CSE graduate program coordinator for guidance in constructing an appropriate course of study.

Consideration will be given to students participating in an approved PhD program. Courses within an approved course of study leading to a Ph.D. dissertation may be considered sufficient to meet the technical electives requirement.

Students admitted to certain home units have additional requirements or constraints. Students in the School of Aerospace Engineering must take the MS Thesis option to complete program requirements.

5.3 Program of Study Approval

Students must acquire the approval of their proposed program of study in their first semester of enrollment in the CSE program from both the student's home unit coordinator and the CSE program director. A suitable selection of technical electives that meets the requirements described above, and is consistent with the student's career objectives will be determined at this time.

5.4 CSE Master's Thesis

The master's thesis option is intended for students wishing to perform gradate research in a topic in the CSE discipline. The student must demonstrate the ability to perform independent research in collaboration with a faculty advisor that can be defended to a committee of faculty. To fulfill the thesis requirement, the student must:

- Define a suitable research problem and approach in consultation with a thesis advisor. The thesis advisor should be a faculty member from a unit participating in the CSE program.
- Complete 6 hours of the course CSE 7000 (Master's Thesis).
- Document the results of this body of work in a thesis document. Typically this document will describe the research problem being addressed, relevant related work, the research approach used to attack the problem, and results, conclusions, and areas meriting further investigation.
- Defend the research and results of the work to a thesis committee that includes at least three individuals.

The thesis committee must include at least one faculty member with an appointment in the College of Computing and one with a faculty appointment in the College of Sciences or the College of Engineering.

The research problem statement and thesis committee must be approved by both the home unit coordinator and the CSE program director.

5.5 Obtaining a CSE Master Degree while Pursuing a Ph.D. Degree

A student pursuing a Ph.D. degree has the option to obtain a CSE master's degree if the CSE master's degree program requirements are fulfilled. The student does not need to officially apply to the CSE master's degree program to be awarded this degree.

5.6 Transfer of Credits

Any transfer of credit must be requested during the first semester in residence at Georgia Tech. A student may receive up to six semester hours of transfer credit toward the M.S. degree for graduate-level courses taken at an institution accredited by a Canadian or U.S. regional accrediting board, or at a foreign school or university that has a signed partner agreement with Georgia Tech-Lorraine, and not used for credit toward another degree.

The student must confer with the CSE program director to ascertain whether the courses to be transferred are a logical part of the graduate program at Georgia Tech. If the courses are appropriate, the student will need to give the CSE graduate program advisor a copy of his/her current transcript displaying the course(s) and some descriptive course materials, including catalog descriptions, syllabi, and textbooks, which will be used to evaluate the course. The CSE program director will consult with faculty of the appropriate Georgia Tech department to determine the equivalent Georgia Tech course and the number of credit hours to be accepted. Once the CSE program director approves the course(s) for transfer credit, the Non-Resident Credit Report is prepared and sent directly to the Georgia Tech Registrar with the supporting documentation.

For more information on transfer of graduate credits, see

http://www.catalog.gatech.edu/students/grad/geninfo/transfercredit.php

5.7 Sample Program

The following is a sample program that a student might follow to complete the MS program under the course-only option:

Semester 1 (Fall)	**Semester 2 (Spring)**	**Semester 3 (Fall)**
CSE-core (3)	CSE-core-3 (3)	Specialization (3)
CSE-core (3)	CSE-core-4 (3)	Specialization (3)
Specialization (3)	Specialization (3)	Specialization (3)
Specialization (3)		

The following is a sample program that a student might follow to complete the MS program under the MS thesis option. This table represents a three semester program; however, we anticipate that many students will require four semesters to compile adequate background knowledge to successfully complete thesis research.

Semester 1 (Fall)	**Semester 2 (Spring)**	**Semester 3 (Fall)**
CSE-core (3)	CSE-core-3 (3)	Specialization (3)
CSE-core (3)	CSE-core-4 (3)	Specialization (3)
Specialization (3)	CSE 7000 (MS Thesis) (3)	CSE 7000 (MS Thesis) (3)
Specialization (3)		

6. PhD Program Degree Requirements

The program was designed to provide much flexibility in order to allow students to tailor the program to their individual career objectives. The program of study shall be defined by the student with the approval of the student's dissertation advisor, home unit coordinator and CSE program director, but must satisfy the following minimum course requirement. The Ph.D. degree in CSE requires a minimum of 31 semester hours of coursework (see Table 3). The program of study is designed to give the student breadth of knowledge in computational science and engineering, depth in specific computational methods and techniques, and knowledge to apply these techniques to problems within the context of a specific application domain. Required coursework includes:

- *CSE core*. Twelve semester hours of CSE core courses are required to give breadth of knowledge in major areas of computational science and engineering. These courses are designed to provide a base of knowledge and skill spanning several core areas of computational models. In addition, a one-hour introductory course must be taken by students, preferably in their first year in the program, that focuses on an introduction to the CSE discipline and multidisciplinary communications.
- *Computation specialization*. Nine semester hours are required to give the student focused in-depth knowledge of advanced computational methods and techniques.
- *Application specialization*. Nine semester hours are required to give the student sufficient knowledge and experience in an application domain in order to apply computational methods to relevant and important problems within the context of that domain.

These requirements constitute the minimum amount of coursework to fulfill degree requirements. The dissertation advisor or the student's home unit may impose additional course requirements in accordance with the student's specific research activities and long-term professional objectives.

Each student is expected to maintain the GPA of at least 3.3 for all courses listed on his/her degree program. All courses listed in the degree program must be taken on the A-F grading basis if the A-F grading basis is offered.

A PhD degree must be completed within ten years from the date of the first coursework on the degree program, including any transfer credits.

Table 3. Curriculum Overview (31 hours)

Curriculum Component	Semester Hours
CSE core courses	13
Computation specialization	9
Application specialization	9
Total	**31 rs**

6.1 CSE Core (13 Semester Hours)

Each student must complete four courses among the five listed in Table 2. Students who have completed a core course, or equivalent course at another institution, prior to entering the program will be allowed to substitute a core course with an additional specialty course, consistent with the student's intended specialization.

In addition, students must take CSE 6001 (Introduction to Computational Science and Engineering), preferably during the first year of enrollment in the CSE doctoral program. It is recommended that students participate in this seminar course every semester of enrollment in the CSE program to keep abreast with recent advances in the field, however, students need only take this course one time for credit.

Finally, students are also required to take CSE 890x (or equivalent in the student's home unit), a special problems course with a member in the CSE program faculty. The purpose of this requirement is to engage the student in preliminary research activities with CSE faculty members early in the PhD program. It is recommended that students take this course in their first semester of enrollment in the CSE PhD program.

6.2 Computation Specialization (9 Semester Hours)

Students must complete nine hours of technical electives focusing on advanced computational methods. This specialization increases the student's depth of knowledge and skills in computation. The set of courses must clearly support graduate work in the computational science and engineering discipline by providing advanced knowledge of computational techniques and methods.

Georgia Tech currently offers a wide range and variety of courses that include computational methods. A recommended list of suitable courses and concentrations is included as in Appendix 1. Selection of courses to fulfill this requirement must be done in consultation with an academic advisor. In general, simply utilizing computer software in the course is not by itself sufficient to qualify a course for the computation specialization; the course must include intellectual content in computational methods or techniques, often in the context of some domain or class of applications. Courses listed under "CSE core" (Table 2) that are not used to satisfy the core course requirement may be used to partially fulfill the computation specialization requirement.

6.3 Application Specialization (9 Semester Hours)

Students must complete an additional nine hours of technical electives focusing on an application domain where advanced computational techniques may be applied. Courses fulfilling this requirement need not necessarily have a computation focus. For instance, coursework may provide knowledge of an application area (e.g., a science or an engineering field) to enable the student to apply computational techniques within the context of that domain.

The computation and application specializations taken together must provide a coherent program of study that will enable the student to develop and apply advanced computational techniques and methods to relevant problems in a specific field of study.

Students who declare the School of Chemistry and Biochemistry as their home unit are required to complete Chemistry courses to fulfill the Application Specialization requirement.

6.4 Minor Requirement

Students must complete a focused program of study including at least 9 semester hours of courses outside the computational science and engineering field. These courses will normally consist of courses that carry neither the CSE nor the CS course designation.

Because the curriculum is inherently multidisciplinary and requires study in at least two fields (computational science and engineering and a domain of application), *courses that do not carry the CS/CSE designation that are used to fulfill the computation and application specialization requirements may also be used to fulfill the minor requirement*.

6.5 PhD Review

All PhD. students are reviewed on an annual basis to determine progress and performance in the PhD program. Details of this review process are specified by the home unit. For students homed in the College of Computing, the review begins with the student completing a self evaluation of his/her performance. This is followed by a review by the faculty advisor. A PhD Review Committee then assigns a status. There are four possible status designations: satisfactory, concern, warning, and probation. Students assigned to warning or probation status will be reevaluated by the committee in the Spring semester. Students that receive a probation status are in jeopardy of losing financial support. Students placed on academic probation by the institute, e.g., due to a low GPA, shall automatically be placed on probation status. Students will be notified of the result of the review in writing prior to the end of the Fall semester.

6.6 CSE Qualifying Examination

6.6.1. CSE PhD Qualifying Exam Format

The Ph.D. qualifying examination consists of two components. This examination is designed to ensure the student has achieved sufficient knowledge in core areas of computational science and engineering as well as in their chosen specialization area as preparation for advanced research.

- Written qualifying exam: This written exam encompasses core areas of Computational Science and Engineering. Students select two areas among numerical methods, discrete algorithms, modeling and simulation, computational data analysis, and high performance computing. Material covered by the examination will normally include topics covered in core courses in the core area augmented with a reading list provided to the student as preparation for the examination. The format is a day-long written examination.

- Specialization exam and artifact defense: This portion of the exam has two purposes: to ensure that the student has acquired sufficient knowledge in his/her specialization area in a computing, engineering or science discipline for advanced research, and to ensure that the student can demonstrate an ability to integrate knowledge in mathematical foundations/computational methods and knowledge in a specific engineering or science discipline to synthesize a concrete computational artifact, e.g., a significant computer program.

Specifics concerning the specialization exam and artifact defense depend on the home unit, as discussed below.

School of Biology. The second portion of the qualifying exam will cover both the computational artifact and the student's specialization area of Biology. This exam will consist of a formal written grant proposal following National Institutes of Health (NIH) or National Science Foundation (NSF) guidelines that will normally outline the student's thesis research proposal. The grant proposal is expected to describe, as part of the preliminary results, the student's prior research and development of a computational artifact that is related to the student's proposed thesis research. It will also include an oral presentation to the student's thesis committee of the student's prior research accomplishments working under the direction of his or her principal research advisor, with the biological research aspects of the work highlighted. The student will then defend the artifact and the thesis proposal, answering questions orally from the committee.

Frequently, the computational artifact will have been developed or will be under development as part of the student's research project. In such cases, the student must be sure to explain the biological relevance of this work and how it has or will be applied to biological problems. Students will be expected to demonstrate an understanding of basic biological concepts as they relate to their research project. The grant proposal should be submitted to the committee at least two weeks prior to the oral exam.

Finally, completion of the second portion of the qualifying exam fulfills the CSE program requirement for a dissertation proposal defense. Therefore, students homed in the School of Biology are not required to complete a separate thesis proposal defense in addition to the qualifying examination.

School of Chemistry and Biochemistry. The specialization and artifact defense is an exam that will cover both the computational artifact and the student's specialization area of Chemistry in a single oral examination session. The computational artifact defense is an oral defense of the artifact (typically a software program developed by the student). The specialization part of the exam will consist of an oral presentation of the student's prior research accomplishments working under the direction of his or her principal research advisor, with the chemical aspects of the work highlighted. The student should also explain the relevance of this research and discuss their current and future research plans. Frequently, the computational artifact will have been developed as part of the student's research project. In such cases, the student must be sure to explain the chemical relevance of this work and how it has or will be applied to chemical problems. Students will be expected to demonstrate an understanding of basic chemical concepts as they relate to their research project. A written description of both the computational artifact and a summary of prior and current research (no more than 10 pages) should be submitted to the committee at least two weeks prior to the oral exam.

Computational Science and Engineering Division (College of Computing). An oral exam is used that consist of two parts: prior and current research accomplishments of the student carried out under the direction of his/her principal research advisor; and a computational artifact created by the student based on the above-mentioned research accomplishments. The student should explain the relevance of this research and discuss his/her current and future research plan. The student will have created and documented the computational artifact prior to the examination, and must answer questions regarding the artifact itself. For example, the student may be required to describe the purpose of the artifact and access its strengths, weaknesses and aspects of its design such as the choice of computational algorithms or data structures. A written description of both the computational artifact and a summary of prior and current research (no more than 30 pages) should be submitted to the committee at least two weeks prior to the oral exam.

School of Mathematics. This will be an oral exam covering both a computational artifact and the student's specialization area of "Applied and Computational Mathematics”. The goal of the exam is for the students to present the chosen topic of their eventual dissertation to a core group which will likely become part of the Dissertation Committee.

The computational artifact defense will follow the same format as for all CSE PhD students. It is expected that the computational artifact will have been developed as part of the student’s coursework and directed
study under the direction of their advisors. The students will need to explain the relevance of this work in the context of applied and computational mathematics. The students will also need to explain how the computational artifact will be used as a platform for future computational methodology, theory and code developments.

The specialization part of the exam will consist in a report from the student on the research papers read, and research accomplishments to date, highlighting the components related to applied and computational mathematics. It is expected that students will demonstrate an understanding of basic concepts in applied and computational mathematics as they relate to their research project. The students will also be asked to explain the relevance of their specialization in the broad context of the CSE focus.

A short description of the computational artifact and a list of selected readings, coursework, and relevant references (not to exceed 5 pages in total), will need to be submitted before the oral exam.

For information concerning the qualifying examination for other home units, please contact the CSE Program Director.

6.6.2 Ph.D. Qualifying Exam Committee

Each student's Ph.D. Qualifying Exam Committee consists of the student's advisor and three additional faculty members. At least one of them should be a member of the CSE written exam committee that includes those individuals responsible for developing and grading the written portion of the qualifying examination that semester. The committee must be approved by the CSE program director. The qualifying exam committee shall be present for the artifact defense, which will take place after the written core examination has been completed and passed. The qualifying exam committee is responsible for making an overall recommendation concerning the outcome of the qualifying examination, covering both the written and the home unit / artifact components.

6.6.3 CSE Qualifying Exam Administration

The written qualifying exam is to be offered once in the Fall semester and once in the Spring semester each year, usually around the 10th week of the term. The second part of the CSE exam (i.e., the artifact oral exam) must be attempted within four weeks of completing the written qualifying exam, excluding semester breaks. Each student is to schedule the artifact oral exam only after passing the written qualifying exam. The qualifying exam must be attempted by the end of the second year of enrollment in the CSE doctoral program. If the exam is failed, the student is allowed to retake it one more time in the next semester when the exam is offered. However, for those students with the CSE Division as their home unit, the written qualifying exam must be passed by the end of the second year of enrollment in the CSE doctoral program. Students who are not able to pass the exam after two attempts will be encouraged to seek a Master's degree, and will not be able to continue in the CSE doctoral program.

6.6.4 Declare Intent

At the beginning of each semester, each PhD. student wishing to take the Qualifying exam that semester must complete the CSE Qualifying Exam Form and return it to the CSE Graduate Program Advisor. Students who intend to take the exam must specify the following information:

- Requested Ph.D. Qualifying Exam Committee members.
- Declaration of the two core areas among numerical methods, discrete algorithms, modeling and simulation, computational data analysis, and high performance computing for the first portion of the qualifying examination.

6.6.5 Composition of the Written Qualifying Exam

The written qualifying exam is subdivided into the five core areas: numerical methods, discrete algorithms, modeling and simulation, computational data analysis, and high performance computing. Each of these core areas provides a reading list composed of books and articles, and its scope covers the general topics taught in the corresponding courses plus more advanced materials and application-oriented special topics (listed in the Appendix of this handbook).

The written exam contains **four** questions from each of the above core areas. The written exam for a student will contain a total of **eight** questions (**four** questions from each of these two selected core areas), and the student is expected to answer **six** of these questions (**three** questions from each core area) during the written exam.

6.6.6 Grading and Results

Each question in the core written exam is graded using the scale 0-10. Each student is expected to answer exactly six questions. If a student answers more than six questions, then only the lowest scored six answers will be counted toward the written exam result. There will be three possible outcomes for a written exam: "pass", "conditional pass", and "fail". Students with either "pass" or "conditional pass" will be allowed to go on with the specialization exam and artifact defense. Students with "fail" will need to retake the written exam. The result of the written exam is determined by the CSE Written Exam Committee that is responsible for developing and grading the exam. The final overall outcome is determined by the student's Ph.D. Qualifying Exam Committee based on the results of all the components of the qualify exams. To pass the qualifying exam, a majority of the committee members, including at least three individuals, must vote "pass". The students will be notified of the result within 2 weeks of taking the exam.

6.7 Program of Study Approval

The student must file an approved program of study indicating which courses will be used to fulfill the degree requirements after successfully passing the Qualifying Exam or by the end of the student's second year in the program. The student's dissertation advisor, the home unit coordinator, and CSE program director must approve the proposed program of study.

6.8 Applying to PhD Candidacy

After successfully presenting the research proposal, students must petition for admission to doctoral candidacy. This is done via the Georgia Tech Request for Admission to PhD Candidacy form. To qualify for PhD Candidacy, a student must complete all course work requirements; achieve a satisfactory scholastic record (3.3 GPA); pass the CSE Qualifying Examination; submit an approved program of study and an approved thesis committee member form to the CSE graduate programs advisor.

6.9 CSE Doctoral Dissertation

The doctoral dissertation (thesis) forms a central component of the CSE Ph.D. program. The student must demonstrate the ability to perform independent research in collaboration with a faculty advisor that can be defended to a committee of faculty. To complete the doctoral thesis, the student must complete three principal milestones:

- *Ph.D. proposal defense*. The proposal defense should be completed after some preliminary research has been conducted. The student must submit a written proposal documenting the research problem being addressed, discussion of related work, discussion of the research approach used to attack the problem, preliminary research results, and plans to complete the doctoral dissertation research. The proposal must be defended to the doctoral dissertation committee in an oral defense. For students homed in the School of Biology, this requirement is combined with the second part of the qualifying examination.

- *Ph.D. dissertation*. The body of work and results in the Ph.D. program must be documented in the dissertation that must be approved by the research advisor and the doctoral dissertation committee.
- *Ph. D. dissertation defense*. The student must present an oral defense of the body of work included in the doctoral dissertation to the doctoral dissertation committee.

The doctoral dissertation committee includes at least five individuals and must include a balance of faculty spanning multiple disciplines. This latter requirement will typically be satisfied with at least two members of the committee with an appointment in the College of Computing, and at least two with an appointment in the College of Engineering or the College of Science. The main Ph.D. advisor should be a member of the home unit and also be a member of the CSE program faculty. Students who declare the School of Chemistry and Biochemistry as their home unit are required to have at least three committee members that are faculty members of the School of Chemistry and Biochemistry.

6.10 Obtaining a CSE Master Degree while Pursuing a Ph.D. Degree

A student pursuing a Ph.D. degree has the option to obtain a CSE master degree if the CSE master degree program requirements are fulfilled. The student does not need to officially apply to the CSE master degree program to be awarded this degree.

7. Recommended Computation Specialization Courses

The following is the list of the recommended computational specialization courses. The courses are grouped based on the five CSE core areas. (the courses marked with '+' after the course number indicates the CSE core courses). Courses marked '*' are offered through the distance learning program at the time of this writing.

7.1 Numerical Computing and Geometric Computing

CSE/MATH 6643*+ Numerical Linear Algebra

CSE/MATH 6644* Iterative Methods for Systems of Equations

MATH 6640* Introduction to Numerical Methods for Partial Differential Equations

MATH 6641 Advanced Numerical Methods for Partial Differential Equations

MATH 6645 Numerical Approximation Theory

MATH 6646 Numerical Methods for Ordinary Differential Equations

MATH 6647* Numerical Methods for Dynamical Systems

ISYE 6669* Deterministic Optimization

ISYE 6679* Computational Methods

CEE 6507 Nonlinear Finite Element Analysis

ME 6104* Computer Aided Design

ME 6758* Numerical Methods in ME

ME/MSE/PTFE 6795 Mathematical, Statistical, and Computational Techniques in Material Science

ME 6124 Finite-Element Method: Theory and Practice

CEE 6507 Nonlinear Finite Element Analysis

CS 6764 Geometric Modeling

7.2 Computational Data Analysis and Visualization

CSE/ISyE 6740*+ Computational Data Analysis

CSE 6240 Web Search and Text Mining

CSE 6241 Pattern Matching

CS 6480 Computer Visualization Techniques

CS 6485 Visualization Methods for Science and Engineering

ISYE 6402* Time Series Analysis

ISYE 6404 Nonparametric Data Analysis

ISYE 6414* Statistical Modeling and Regression Analysis

ISYE 6416 Computational Statistics

ISYE 6783* Financial Data Analysis

ISYE 7406 Data Mining and Statistical Learning

7.3 Modeling and Simulation

CSE 6730*+ Modeling and Simulation: Fundamentals and Implementation

CSE/INTA 6742 Modeling, Simulation, and Military Gaming

CSE/CS 6236 Parallel and Distributed Simulation

ISYE 6644* Simulation

ISYE 6650* Probabilistic Models

ISYE 6645 Monte Carlo Methods

MATH 4255* Monte Carlo Methods

ISYE 7210 Real-Time Interactive Simulation

ME 6105 Modeling and Simulation in Design

AE/ISYE 6779 Dynamic System Simulation and Modeling

INTA 6004 Modeling, Forecasting and Decision Making

7.4 CSE Algorithms

CSE 6140*+ CSE Algorithms

CSE 6301* Algorithms for Bioinformatics and Computational Biology

CS 6505* Computability, Algorithms, and Complexity

CS 6550 Design and Analysis of Algorithms

CS 7510 Graph Algorithms

7.5 High Performance Computing

CSE 6220*+ High Performance Computing

CSE 6221* Multicore Computing: Concurrency and Parallelism on the Desktop

CSE/CS 6230* High Performance Parallel Computing: Tools and Applications

CSE/CS 6236* Parallel and Distributed Simulation

CS 6290 High Performance Computer Architecture

CS 7110 Parallel Computer Architecture

CS 7210 Distributed Computing

ECE 6101 Parallel and Distributed Computer Architecture

7.6 Optimization

ISYE 6644* Simulation

ISYE 6661 Linear Optimization

ISYE 6662 Discrete Optimization

ISYE 6663 Nonlinear Optimization

ISYE 6669* Deterministic Optimization

ISYE 6679* Computational Methods in Operations Research

MATH 4580* Linear Programming

CSE/MATH 6643*+ Numerical Linear Algebra

CSE/MATH 6644* Iterative Methods for Systems of Equations

MATH 6640* Introduction to Numerical Methods for Partial Differential Equations

MATH 6641 Advanced Numerical Methods for Partial Differential Equations

MATH 6645 Numerical Approximation Theory

MATH 6646 Numerical Methods for Ordinary Differential Equations

MATH 6647* Numerical Methods for Dynamical Systems

ISYE 6669* Deterministic Optimization

ISYE 6679* Computational Methods

CEE 6507 Nonlinear Finite Element Analysis

ME/MSE/PTFE 6795 Mathematical, Statistical, and Computational Techniques in Material Sci.

ME 6124 Finite-Element Method: Theory and Practice

CEE 6507 Nonlinear Finite Element Analysis

CS 6764 Geometric Modeling

7.7 Computational Data Analysis and Visualization

CSE/ISyE 6740*+ Computational Data Analysis

CSE 6240 Web Search and Text Mining

CSE 6241 Pattern Matching

CS 6480 Computer Visualization Techniques

CS 6485 Visualization Methods for Science and Engineering

ISYE 6404 Nonparametric Data Analysis

ISYE 6414* Statistical Modeling and Regression Analysis

ISYE 6416 Computational Statistics

ISYE 7406 Data Mining and Statistical Learning

7.8 Modeling and Simulation

CSE 6730*+ Modeling and Simulation: Fundamentals and Implementation

CSE/INTA 6742 Modeling, Simulation, and Military Gaming

CSE/CS 6236 Parallel and Distributed Simulation

ISYE 6644* Simulation

ISYE 6650* Probabilistic Models

ISYE 6645 Monte Carlo Methods

ISYE 7210 Real-Time Interactive Simulation

ME 6105 Modeling and Simulation in Design

AE/ISYE 6779 Dynamic System Simulation and Modeling

INTA 6004 Modeling, Forecasting and Decision Making

7.9 CSE Algorithms

CSE 6140*+ CSE Algorithms

CSE 6301* Algorithms for Bioinformatics and Computational Biology

CS 6505* Computability, Algorithms, and Complexity

CS 6550 Design and Analysis of Algorithms

CS 7510 Graph Algorithms

7.10 High Performance Computing

CSE 6220*+ High Performance Computing

CSE 6221* Multicore Computing: Concurrency and Parallelism on the Desktop

CSE/CS 6230* High Performance Parallel Computing: Tools and Applications

CSE/CS 6236* Parallel and Distributed Simulation

CS 6290 High Performance Computer Architecture

CS 7110 Parallel Computer Architecture

CS 7210 Distributed Computing

ECE 6101 Parallel and Distributed Computer Architecture

7.11 Optimization

ISYE 6661 Linear Optimization

ISYE 6662 Discrete Optimization

ISYE 6663 Nonlinear Optimization

ISYE 6679 Computational Methods in Operations Research

8. Sample Application Specialization Courses

The number and type of CSE related courses in application areas at Georgia Tech is large and varied. Students should work with their academic advisors to formulate sequences of coherent application specialization elective courses that best meet the students' PhD research topics and objectives. The following is a sample list of application specialization courses. The list is by no means exhaustive but is provided to give students some guidance.

8.1 Fluid Dynamics and Turbulence

AE 6009 Viscous Fluid Flow

AE 6012* Turbulent Flows

AE 6042* Computational Fluid Dynamics

AE 6412* Turbulent Combustion

8.2 Structural Analysis

CEE 6501 Matrix Structural Analysis

CEE 6504 Finite Element Method of Structural Analysis

CEE 6507 Nonlinear Finite Element Analysis

CEE 6510 Structural Dynamics

CEE 6513 Computational Methods in Mechanics

CEE 6551 Advanced Strength of Materials

8.3 Computational Mechanics

CEE 6513 Computational Methods in Mechanics

8.4 Computational Chemistry

CHEM 6472 Quantum Chemistry and Molecular Spectroscopy

CHEM 6491 Quantum Mechanics

CHEM 8843 (temporary number) Computational Chemistry

CHEM 8873 (temporary number) Computational Chemistry Applied to Electronic and Optical Organic Materials

CHBE/CHEM/MSE/PTFE 6751 Physical Chemistry of Polymer Solutions

CHBE/CHEM/MSE/PTFE 6755 Theoretical Chemistry of Polymers

CHEM 6481 – Statistical Mechanics

8.5 Theoretical Ecology and Evolutionary Modeling

BIOL/MATH 4755 Mathematical Biology

BIOL 6422 Theoretical Ecology

BIOL 6600 Evolution

BIOL 7101 Advanced Sensory Ecology

8.6 Bioinformatics

Biol 8803 (temporary number) Genomics and Applied Bioinformatics

Biol 7023 Bioinformatics

Biol 8803 (temporary number) Molecular Evolution

Biol 8804 (temporary number) Macromolecular Modeling

CSE 6301 Algorithms for Bioinformatics and Computational Biology

8.7 Transportation Systems

CEE 4600 Transportation Planning, Operations, and Design

CEE 6601 Linear Statistical Models in Transportation

CEE 6602 Urban Transportation Planning

CEE 6603 Traffic Engineering

CEE 6621 GIS in Transportation

CEE 6622 Travel Demand Analysis

CEE 6631 Signalized Intersections and Networks

CEE 6632 Simulation Models in Transportation

CEE 6636 Traffic Flow Theory

CP 6514 Introduction to Geographic Information Systems

8.8 Gaming and Defense Modeling and Simulation

CS 7497 Virtual Environments

INTA 6004 Modeling, Forecasting and Decision Making

AE/ISYE 6779 Dynamic System Simulation and Modeling

8.9 Computational Electromagnetics

ECE 6350 Applied Electromagnetics

ECE 6380* Introduction to Computational Electromagnetics

ECE 7380 Topics in Computational Electromagnetics

8.10 Manufacturing and Logistics

ISYE 6201* Manufacturing Systems

ISYE 6202* Warehousing Systems

ISYE 6203* Transportation and Supply Chain Systems

8.11 Computational Finance

MGT 6078 Finance and Investments

MGT 6081 Derivative Securities

MATH 6635 Numerical Methods in Finance

ISYE/MATH 6759 Stochastic Processes in Finance I

ISYE/MATH 6767 Design and Implementation of Systems to Support Computational Finance

ISYE/MATH/MGT 6769 Fixed Income Securities

ISYE/MATH 6783* Statistical Techniques of Financial Data

Appendix A

Distance Learning Courses

These are Distance Learning courses and are not directly related to the CSE degree program. Students wishing to take any of the courses in Appendix A to fulfill the CSE degree requirements **must get the course(s) pre-approved** by the CSE program director. Additionally, students planning their CSE graduate studies using distance learning courses are advised to have flexibility in their course schedules as not all distance learning courses are offered every year.

Aerospace Engineering

AE 6450 Rocket Propulsion
AE 7772 Fund-Fracture Mechanics
AE 6440 Turbine Engine Aerothermodynamics
AE 6042 Computational Fluid Dynamics
AE 6320 Astronautics
AE 6412 Turbulent Combustion
AE 8804 QM Advanced Design Methods I
AE 8803 SCH Aerospace Systems Engineering
AE 6333 Rotorcraft Design I
AE 8803 MAV Fixed-Wing Design I
AE 6070 Rotary Wing Aerodynamics
AE 6020 High-Speed Flow
AE 6211 Advanced Dynamics II
AE 6334 Rotorcraft Design II
AE 6374 Advanced Design Methods II
AE 8803 Special Topics - Planetary Entry
AE 6343 Aircraft Design I
AE 6445 Combustor Fundamentals
AE 6354 Advanced Orbital Mechanics
AE 6050 Gas Dynamics ELECTIVE
AE 8803 QME Special Topics - Combustor Design Modeling
AE 6030 Unsteady Aerodynamics
AE 6050 Gas Dynamics
AE 6060 Aeroacoustics
AE 6362 Safety by Design
AE 6210 Advanced Dynamics I
AE 8803 Orbital Mechanics
AE 6779 Dynamic System Modeling NEW
AE 8803 QPR Human Contribution to Safety
AE 6322 Spacecraft Launch and Vehicle Design

AE 6520 Advanced Flight Dynamics
AE 6361 Air Breathing Propulsion System Design I
AE 6503 Helicopter Stability and Control
AE 6766 Combustion-I
AE 8803 PRI Special Topics - Humans and Autonomy NEW
AE 8803QWA Special Topics - Electric Propulsion
AE 8803 QMN Combustion Modeling
AE 8803 QOL Advanced Design Methods II
AE 8804 QMA Fixed-Wing Aircraft Design II
AE 8804 QSC Rotorcraft Design II
AE 8803 QSC Intro to Prod Life Cycle Mgt & Des Tools
AE 4803 Restricted Course-ELDP Program (AE)
AE 8803 Restricted Course-ELDP Program 2 (AE)
AE 6220 Rotorcraft Dynamics
AE 6372 Aerospace Systems Engineering
AE 6373 Adv Design Methods I
AE/ME 6760 Acoustics I
AE/ME 6762 Applied Acoustics
AE 6012 Turbulent Flows
AE 6344 Aircraft Design II NEW
AE 8803 Adv Design Methods III
AE 6414 Multiphase Combustion
AE 6445 Combustor Fundamentals
AE 8803 Cognitive Engineering
AE 6410 Combustion Dynamics

Computational Science and Engineering

CSE/MATH 6643 Numerical Linear Algebra*
CSE/MATH 6644 Iterative Methods for Systems of Equations
CSE/ISyE 6740 Computational Data Analysis*
CSE 6730 Modeling and Simulation: Fundamentals and Implementation*
CSE/CS 6236 Parallel and Distributed Simulation
CSE 6140 CSE Algorithms*
CSE 6220 High Performance Computing*
CSE 6221 Multicore Computing: Concurrency and Parallelism on the Desktop
CSE/CS 6230 High Performance Parallel Computing: Tools and Application
CSE/CS 6236 Parallel and Distributed Simulation

"*" indicates CSE core courses

Electrical and Computer Engineering

ECE 4270 Fund-Digital Signal Proc
ECE 6100 Adv Comput Architecture
ECE 6250 Adv Digital Signal Proc

ECE 6277 DSP Software Sys Design
ECE 6320 Power Sys Ctrl & Operation
ECE 6521 Optical Fibers
ECE 6550 Linear Sys and Controls
ECE 6557 Manufacturing Sys Design
ECE 6601 Random Processes
ECE 6602 Digital Communications
ECE 6605 Information Theory
ECE 6606 Coding Theory & Applications
ECE 6607 Computer Comm Networks
ECE 8843 Computer Network Security
ECE 6272 Fundamentals of Radar Signal Processing
ECE 4321 Power System Engineering
ECE 6101 Parallel& Dist Comp Arch
ECE 6254 Stat Digit Sig Proc & Mod
ECE 6255 Digit Proc-Speech Signal
ECE 6340 Electric Power Quality
ECE 6522 Nonlinear Optics
ECE 6552 Nonlinear Systems and Control
ECE 6553 Optimal Control and Optimization
ECE 6322 Power System Planning and Reliability
ECE 6500 Fourier Techniques and Signal Analysis
ECE 8803 Special Topics - Radar Imaging
ECE 6551 Digital Control
ECE 6530 Modulation, Diffractive and Crystal Optics
ECE 6140 Digital Systems Test
ECE 8843 Computer Network Security
ECE 6271 Adaptive Filtering
ECE 6331 Power Electronic Cicuits
ECE 4320 Power Systems Analysis & Control
ECE 6258 Digital Image Processing
ECE 6279 Spatial Array Processing
ECE 6555 Optimal Estimation
ECE 6612 Computer Network Security
ECE 6556 Intelligent Control
ECE 6604 Personal & Mobile Commun
ECE 6609 ATM Networks
ECE 6610 Wireless Networks
ECE 7142 Fault Tolerant Computing
ECE 7102 RISC Architectures
ECE 6323 Power System Protection
ECE 6335 Electric Machinery Analysis
ECE 6273 Pattern Recognit-Speech
ECE 6520 Integrated Optics NEW
ECE 6380 Intro Computational EM
ECE 4753 Topics in Engr Practice

ECE 6560 Partial Differential Equations in Image Processing and Computer Vision
ECE 6744 Topics in Engineering Practice
ECE 8863 Special Topics-Sensor Networks
ECE 8873 Special Topics..Radar Imaging
ECE 6780 Medical Image Processing
ECE 6276 DSP Hardware Sys Desgin
ECE 6321 Power System Stability Techniques
BMED/ECE 6780 Medical Image Processing
ECE 8893 Special Topics - Embedded Video Surveillance Systems NEW
ECE 6603 Adv Digital Communications NEW
ECE 6430 Digital MOS ICs
ECE 4420 Digital Integ Circuits
ECE 6510 Electro-Optics
ECE 6453 Theory Electronic Device
ECE 6554 Adaptive Control

Industrial and System Engineering

ISYE 6201 Manufacturing Systems
ISYE 6202 Warehousing Systems
ISYE 6669 Deterministic Optimiz
ISYE 6225 Engineering Economy
ISYE 6644 Simulation
ISYE 6650 Probabilistic Models
ISYE 6203 Transp & Supply Chain Sys
ISYE 6307 Scheduling Theory
ISYE 6402 Time Series Analysis
ISYE 8813 Industrial Ecology and Natural Systems NEW
ISYE 6783 Financial Data Analysis NEW
ISYE 6413 Design and Analysis of Experiments
ISYE 6781 Reliability Theory
ISYE 6205 Cognititive Engineering
ISYE 6230 Economic Decision Analy
ISYE 6401 Stat Models & Dsgn Expts
ISYE/Math 6781 Reliability Theory
ISYE 8803 Special Topics - Energy Technology & Policy NEW
ISYE 6679 Computational Methods
ISyE 6414 Regression Analysis
ISYE 6669 Deterministic Optimiz
ISYE 6650 Probabilistic Models
MATH 4261 Math Statistics I
ISYE 6644 Simulation
ISYE 6781 Reliability Theory
ISYE 6679 Computational Methods
ISYE 6225 Engineering Economy
ISYE 6307 Scheduling Theory

ISYE 6230 Economic Decision Analy
ISYE 6401 Stat Models & Dsgn Expts
ISYE 6203 Transp & Supply Chain Sys
ISYE 6402 Time Series Analysis

Mathematics

MATH 4150 Introduction to Number Theory
MATH 4255 Monte Carlo Methods
MATH 4261 Math Statistics I
MATH 4262 Mathematical Statistics II
MATH 4280 Intro to Information Theory
MATH 4305 Linear Algebra
MATH 4317 Analysis I
MATH 4320 Complex Analysis
MATH 4580 Linear Programming
MATH 4581 Classical Mathematical Methods in Engineering
MATH 4581 Math Methods in Engr
MATH 4640 Numerical Analysis I
MATH 4641 Numerical Analysis II
MATH 4803 Graph Theory **(Undergraduate)**
MATH 6014 Graph Theory
MATH 6021 Topology of Euclidean Spaces
MATH 6221 Classical Probability **NEW**
MATH 6241 Probability I
MATH 6263 Testing Statistical Hypotheses
MATH 6266 Linear Statistical Model
MATH 6321 Complex-Analysis
MATH 6327 Real Analysis
MATH 6337 Real Analysis I **NEW**
MATH 6421 Algebraic Geometry I
MATH 6455 Differential Geometry I
MATH 6514 Industrial Mathematics I
MATH 6580 Introduction to Hilbert Spaces
MATH 6583 Integral Equations and Transforms
MATH 6640 Num Meth-Part Diff Eqns
MATH 6644 Iterative Methods for Systems of Equations
MATH 6647 Numeric Meth:Dynamic Sys
MATH 6701 Mathematical Methods of Applied Sciences I
MATH 6702 Math Meth-Appl Sci II
MATH 6761 Stochastic Processes I **NEW**
MATH 7334 Operator Theory
MATH 7581 Calculus Variations **NEW**

Mechanical Engineering

ME/HP 6758 Numerical Methods in ME
ME 6242 Mechanics of Contact NEW
ME 6201 Principle-Continuum Mech
ME 6201 Principle-Continuum Mech
ME 6222 Mfg Processes & Systems
ME 6244 Rotordynamics
ME 6301 Conduction Heat Transfer
ME 6304 Prin of Thermodynamics
ME 6406 Machine Vision
ME 6441 Dynamics of Mechanical Sys
ME 6601 Intro to Fluid Mechanics
ME 6770 Energy Meth - Elast & Plast
ME 7442 Vibration - Continuous Sys
ME 7301 Trans Ph Multiphase Flow
ISYE 6401 Stat Models & Dsgn Expts
ME 6792 Manufacturing Seminar
AE/ME 6766 Combustion
AE 6765 Kinetics & Thermo Gases
ME 6101 Engineering Design
ME 4193 Tribological Design
ISYE 6739 Statistical Methods
ME 6103 Optimization Engr Design
ME 6602 Viscous Flow
ME 6102 Designing Open Engr Sys
ME 6104 Computer-Aided Design
ME 6203 Inelastic Deform Solids
ME 6223 Automated Manufacturing Process Planning
ME 6224 Machine Tool Analysis and Control
ME 6243 Fluid Film Lubrication
ME 6302 Convection Heat Transfer
ME 6402 Nonlinear Control System
ME 6403 Digital Control Systems
ME 6442 Vibration-Mechanical Sys
ME 6622 Experimental Methods
ME 6758 Numerical Methods in ME
ME 7772 Fund-Fracture Mechanics
ME 4172 Design Sustainable Eng Sys
ME 6201 Principle-Continuum Mech
ME 6404 Adv Ctrl Dsgn Implementation
ME 8933 Special Problems-Therm Sciences
ME 4753 Topics in Engr Practice.ME
ME 4189 Structural Vibrations
ME 6744 Topics in Engineering Practice
ME 6401 Linear Control Systems
ME 7227 Rapid Prototype-Engr

ME 6303 Radiation Heat Transfer
ME 6305 Apps of Thermodynamics
ME 8833 Spec Top-Thermal Science
ME 6766 Combustion I
ME 6203 Inelastic Deform Solids
ME 4791 Mech Behavior-Composites
ME 7774 Fatigue-Materials & Struct
ME 6105 Modeling&Simulation-Dsgn

Appendix B

Numerical Methods

Faculty

George Biros
Haesun Park
Richard Vuduc
Hongyuan Zha

Scope

direct and iterative methods for linear systems
eigenvalue decomposition
numerical optimization
interpolation and approximation
numerical solutions of ordinary differential equations
parallel numerical algorithms

The student is expected to have a general knowledge of the topics listed above. Standard questions that might be asked include definitions, existence, uniqueness, characterization, derivation, proof, applicability, sensitivity, stability, accuracy, convergence, computational complexity, etc., as may be relevant.

Suggested readings

Matrix Computations, by Gene H. Golub and Charles F. van Loan, John Hopkins, 1996.
Linear and Nonlinear Programming 2/e by D. Luenberger, Springer, 2003
Scientific Computing: An Introductory Survey, Second Edition, by Michael T. Heath, McGraw Hill, 2002.

Other references

Applied Numerical Linear Algebra by J.W. Demmel, SIAM, 1997
Numerical Linear Algebra by Lloyd N. Trefethen and David Bau, SIAM, 1997
A First Course in the Numerical Analysis of Differential Equations, by A. Iserles, Cambridge, 1996.

Related courses

CS8803 NMC : Numerical Methods in Computational Science and Engineering
Iterative Methods for Solving Systems of equations (CSE/MATH 6644)
Numerical Methods for Mechanical Engineers (ME 6758)
Parallel Numerical Algorithms (CSE 8803 PNA) [Spring 2008]

Discrete Algorithms

Faculty

Alberto Apostolico
David Bader

Scope

algorithm design, complexity analysis, experimentation, and optimization

Suggested readings

Book

Cormen, Leiserson, Rivest, and Stein, Introduction to Algorithms, Second edition, MIT Press, 2001.

Articles

All downloadable from the webpage http://www-static.cc.gatech.edu/~bader/COURSES/GATECH/CS8803-Fall2006/

Guy Blelloch, Algorithms in the Real World, Lecture Notes
Alok Aggarwal and Jeffrey Scott Vitter, The Input/Output Complexity of Sorting and Related Problems, Communications of the ACM, 31:1116-1127, 1988.

Sandeep Sen, Siddhartha Chatterjee, Neeraj Dumir, Towards a theory of cache-efficient algorithms, Journal of the ACM, 49(6):828-858, 2002.
A. LaMarca and R.E. Ladner, The Influence of Caches on the Performance of Heaps, Journal of Experimental Algorithmics, Vol 1, 1996.
A. LaMarca and R.E. Ladner, The Influence of Caches on the Performance of Sorting, Journal of Algorithms, 31:66-104, 1999.
R.E. Ladner, J.D. Fix, and A. LaMarca, Cache Performance Analysis of Traversals and Random Accesses, Tenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 1999.
John W. Romein, Jaap Heringa, Henri E. Bal, A Million-Fold Speed Improvement in Genomic Repeats Detection, Supercomputing 2003.
Joon-Sang Park, Michael Penner, Viktor K Prasanna, Optimizing Graph Algorithms for Improved Cache Performance, International Parallel and Distributed Processing Symposium, Fort Lauderdale, FL, April 2002.
Markus Kowarschik, Ulrich Rüde, Christian Weiss, and Wolfgang Karl, Cache-Aware Multigrid Methods for Solving Poisson's Equation in Two Dimensions, Computing, 64:381-399, 2000.
M. Frigo, C. E. Leiserson, H. Prokop, and S. Ramachandran, Cache-Oblivious Algorithms, IEEE Symposium on Foundations of Computer Science, 1999.
Stephen Alstrup, Michael A. Bender, Erik D. Demaine, Martin Farach-Colton, J. Ian Munro, Theis Rauhe, Mikkel Thorup, Efficient Tree Layout in a Multilevel Memory Hierarchy, Extended version of ESA 2002 paper, November 2002.
Lars Arge, Michael A. Bender, Erik D. Demaine, Bryan Holland-Minkley, J. Ian Munro, Cache-Oblivious Priority Queue and Graph Algorithm Applications, 34th ACM Symposium on Theory of Computing (STOC), 2002.
William E. Lorensen, Harvey E. Cline. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. Computer Graphics, Volume 21, Number 4, July 1987.
M. Erez, J. H. Ahn, A. Garg, W.J. Dally, and E. Darve, Analysis and Performance Results of a Molecular Modeling Application on Merrimac, SC'04, Pittsburgh, PA, November 2004.
D.S. Johnson, A Theoretician's Guide to the Experimental Analysis of Algorithms, in Proceedings of the 5th and 6th DIMACS Implementation Challenges, M. Goldwasser, D. S. Johnson, and C. C. McGeoch, Editors, American Mathematical Society, Providence, 2002.
The Buffer Tree: A Technique for Designing Batched External Data Structures, Lars Arge, Algorithmica, 37(1):1-24, 2003.

Suggested courses

CS 8803-DA: Computational Science & Engineering (CSE) Algorithms

Modeling and Simulation

Faculty

Richard Fujimoto

Scope

- Discrete event simulation
  - Conceptual models (e.g., queueing networks, petri nets, cellular automata), formalisms
  - DES world views and paradigms (e.g., event-oriented, process-oriented, agent-based simulation)
  - Implementation issues (e.g., event list data structures, threads)
  - Random number and random variate generation
  - Input and output analysis
  - Verification and validation
- Parallel discrete event simulation
  - conservative synchronization: Chandy/Misra/Bryant, deadlock detection and recovery, synchronous execution, lookahead
  - optimistic synchronization: Time Warp, GVT algorithms, memory management, limiting optimism
  - hybrid approaches
- Time parallel simulation
- Distributed virtual environments
  - Clock synchronization
  - Data distribution
  - Dead reckoning
- Simulation interoperability, High Level Architecture
- Continuous simulation
  - Boundary value problems for ODEs
  - Elliptic and parabolic PDEs
  - Finite difference methods
  - Finite element methods
  - Consistency, convergence, stability

Suggested readings

Books

L. G. Birta and G. Arbez, Modeling and Simulation: Exploring Dynamic System Behavior, Springer, 2007

R. M. Fujimoto, Parallel and Distributed Simulation Systems, Wiley, 2000.

A. Law and W. D. Kelton, Simulation Modeling and Analysis, 3rd Edition, McGraw-Hill, 2000.

L. M. Leemis and S. K. Park, Discrete-Event Simulation: A First Course, Prentice Hall, 2006

B. Zeigler, H. Praehofer, T. G. Kim, Theory of Modeling and Simulation, 2nd Edition, Academic Press, 2000.

Articles

D. Jones, "An Empirical Comparison of Priority-Queue and Event-Set Implementations," *Communications of the ACM*, Vol, 29, No. 4 April 1986, pp300-311.

R. Brown, "Calendar Queues: A Fast O(1) Priority Queue Implementation for the Simulation Event Set Problem," *Communications of the ACM*, Vol. 31, No. 10, Oct. 1988 pp. 1220-1227.

W.-T. Tang, R. Goh, I. Thng, "Ladder Queue: An O(1) Priority Queue Structure for Large-Scale Discrete Event Simulation," *ACM Transactions on Modeling and Computer Simulation*, Vol. 15, No. 3, July 2005, pp 175-204.

T. Schriber and D. Brunner, "Inside Discrete-Event Simulation Software: How it Works and Why it Matters," *Proceedings of the Winter Simulation Conference*, December 2006.

K. Perumalla, and R. M. Fujimoto, "Large-scale Process-Oriented Optimistic Parallel Simulation," *Proceedings of the Winter Simulation Conference*, pp. 459-466, December 1998.

K. Chandy and J. Misra, "Distributed Simulation: A Case Study in Design and Verification of Distributed Programs," *IEEE Transactions on Software Engineering*, vol. SE-5, no. 5, 1978, pp. 440-452.

D. Jefferson, "Virtual Time," *ACM Transactions on Programming Languages and Systems*, vol. 7, no. 3, 1985, pp. 404-425.

F. Mattern, Efficient Algorithms for Distributed Snapshots and Global Virtual Time Approximation. *Journal of Parallel and Distributed Computing*, 1993. 18(4): p. 423-434.

M. Hybinette and R. M. Fujimoto "Cloning Parallel Simulations," *ACM Transactions on Modeling and Computer Simulation*, Vol. 11, No. 3, pp. 378-407, October 2001.

C. D. Carothers, K. Perumalla, R. M. Fujimoto, "Efficient Optimistic Parallel Simulation Using Reverse Computation, *ACM Transactions on Modeling and Computer Simulation*, Vol. 9, No. 3, pp. 224-253, October 1999.

P. Heidelberger, H. Stone, Parallel Trace-Driven Cache Simulation by Time Partitioning, *Proceedings of the 1990 Winter Simulation Conference*. 1990. p. 734-737.

D. C. Miller and J. A. Thorpe, "SIMNET: The Advent of Simulator Networking," *Proceedings of the IEEE*, vol. 83, no. 8, 1995, pp. 1114-1123.

J. S. Dahmann, R. M. Fujimoto, R. M. Weatherly, "The Department of Defense High Level Architecture," *Proceedings of the 1997 Winter Simulation Conference*, December 1997.

R. M. Fujimoto, "Time Management in the High Level Architecture," *Simulation*, vol. 71, no. 6, 1998, pp. 388-400.

J. L. Peterson, "Petri Nets," *ACM Computing Surveys*, vol. 9, no. 3, 1979, pp. 223-252.

P. Sarkar, "A Brief History of Cellular Automata," *ACM Computing Surveys*, vol. 32, no. 1, 2000, pp. 80-107.

C. Macal and M. North, "Tutorial on Agent-Based Modeling and Simulation," *Proceedings of the 2005 Winter Simulation Conference*, December 2005, pp. 2-15.
C. Macal and M. North, "Tutorial on Agent-Based Modeling and Simulation Part 2: How to Model with Agents" *Proceedings of the 2006 Winter Simulation Conference,* December 2006.

Book chapters on continuous simulations

Chapters 1, 2, 5, 6, *An Introduction to Computer Simulation*, by M. M. Woolfson and G. J. Pert, Oxford, 1999.
Chapters 1-3,5 *Numerical Solution of Partial Differential Equations*, by K. W. Morton and D. F. Mayers, Second edition, Cambridge, 2005.

Related courses

The most relevant course is CSE 6730 (Modeling and Simulation: Fundamentals and Implementation). Other related courses include:

Parallel & Distributed Simulation Systems (CSE 6236/CS4230)
Simulation Systems: Product and Process Life Cycles (AE/CS/ISYE 6778)
Dynamic System Simulation and Modeling (AE/ISYE 6779)
Introduction to Numerical Methods for Partial Differential Equations (MATH 6640)

High-Performance Computing

Faculty

David Bader
George Biros
Jeffrey Vetter
Richard Vuduc

Scope

Parallel algorithms
Architectures (microprocessors, networks; reconfigurable computing)
Programming models (parallel languages and libraries)
Performance metrics and bounds
Memory consistency, synchronization, load balance, scheduling
High-performance compilers
Performance profiling and tuning

Suggested readings

Books

Introduction to Parallel Computing, by Ananth Grama, Anshul Gupta, George Karypis, and Vipin Kumar, Addison Weslesy, 2003.
Algorithms: Sequential, Parallel, and Distributed, by Kenneth Berman and Jerome Paul, Thomson Course Technology, 2005. Chapters 15, 16, 18, 19, and 24.
Computer Architecture: A Quantitative Approach, by John L. Hennessy, David A. Patterson, 3 edition, Morgan Kaufmann, 2002.
Parallel Computer Architecture : A Hardware/Software Approach, by David Culler, J.P. Singh, Anoop Gupta, 1st edition, Morgan Kaufmann, 1998.
Performance Optimization of Numerically Intensive Codes, by Stefan Goedecker and Adolfy Hoisie, SIAM, 2001.
The Sourcebook of Parallel Computing, by Jack Dongarra, Ian Foster, Geoffrey Fox, William Gropp, Ken Kennedy, Linda Torczon, Andy White, editors, Morgan Kaufmann, 2002.
Optimizing Compilers for Modern Architectures, by Randy Allen and Ken Kennedy, Morgan Kaufman, 2001.

Articles

L. G. Valiant, A Bridging Model for Parallel Computation, Communication of the ACM, 33(8):103-111, 1990.
D. E. Culler, R. M. Karp, D. A. Patterson, A. Sahay, K. E. Schauser, E. Santos, R. Subramonian, and T. von Eicken, LogP: Towards a Realistic Model of Parallel Computation, 4th ACM Symp. Principles and Practice of Parallel Programming (PPoPP), pp 1-12, May 1993. [PDF]
V. Ramachandran, A General-Purpose Shared-Memory Model for Parallel Computation, in M. T. Heath and A. Ranade and R. S. Schreiber (eds.), Algorithms for Parallel Processing v 105, pp 1–18, Springer-Verlag, New York, 1999.
M. Snir and D.A. Bader, A Framework for Measuring Supercomputer Productivity, The International Journal of High Performance Computing Applications, 2004.
D.A. Bader, B.M.E. Moret, and P. Sanders, ``High-Performance Algorithm Engineering for Parallel Computation," Lecture Notes of Computer Science, 2002.
Gustafson, "Reevaluating Amdahl's Law," Communications of the ACM, May 1988
Karp and Flatt, "Measuring Parallel Processor Performance," Communications of the ACM, May 1990
Mellor-Crumney and Scott, "Algorithms for Scalable Synchronization on Shared-Memory Multiprocessors," ACM Transactions on Computer Systems (TOCS), Feb 1991.

Sivasubramaniam et al., "An Application-Driven Study of Parallel System Overheads and Network Bandwidth Requirements," IEEE Transactions on Parallel and Distributed Systems (TPDS) 10(3), pp. 193-210, March, 1999.

Seitz, "The CalTech Cosmic Cube," Communications of the ACM, January 1985, pp. 22-33.

Hillis and Steele, "Data Parallel Algorithms," Communications of the ACM, December 1986.

Kung, "Why Systolic Architectures?" IEEE Computer, January 1982, pp. 37-46.

A.H. Veen, "Dataflow Machine Architecture," ACM Computing Surveys, Vol 18, No 4, Dec 1986, PP 365-396.

Kourosh Gharachorloo et al., "Architecture and Design of AlphaServer GS320," Ninth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), 2000.

Luiz Barroso et al., "Piranha: A Scalable Architecture Based on Single-Chip Multiprocessing", Proceedings of the 27th Annual International Symposium on Computer Architecture (ISCA), 2000.

Some aspects about the scalability of scientific applications on parallel computers. By M. Llorente, F. Tirado, and L. Vazquez, *Parallel Computing*, 22, pp 1169–1195, 1996.

Models and languages for parallel computation. by D. Skillicrn and D. Talia. *ACM Computing Surveys*, 30(2), pp. 128-169, 1998.

Concepts and Notations for Concurrent Programming, by Andrews, G., and Schneider, F., *Computing Surveys*, Vol. 15, pp. 3–43, 1983.

Virtue: Immersive Performance Visualization of Parallel and Distributed Applications: Immersive Performance Visualization of Parallel and Distributed Applications, by Eric Shaffer, Shannon Whitmore, Benjamin Schaeffer, and Daniel A. Reed, *IEEE Computer*, December 1999, pp. 44-51.

On the Impact of Communication Complexity on the Design of Parallel Numerical Algorithms, Gannon, D., and Van Rosendale, J., *IEEE Trans. Comput.*, Vol. C-33, pp. 1180–1194, 1984.

Twelve Ways to Fool the Masses When Giving Performance Results on Parallel Computers; by David H. Bailey; *Supercomputing Review*, Aug. 1991, pg. 54–55.

K. Yotov, T. Roeder, K. Pingali, J. Gunnels, F. Gustavson. "An experimental comparison of cache-oblivious and cache-conscious programs." In Proc. SPAA, 2007. [PDF]

M.S. Lam, E.E. Rothberg, M.E. Wolf. "The cache performance and optimizations of blocked algorithms." In Proc. ASPLOS, 1991. [PDF]

J.W. Hong, H.T. Kung. “I/O complexity: The red-blue pebble game.” In Proc. STOC, pp. 326—333, 1981. [PDF]

M.D. Hill, M.R. Marty. “Amdahl’s Law in the multicore era.” IEEE Computer, July 2008. [PDF]

M. Herlihy. “Wait-free synchronization.” ACM TOPLAS, 11(1), pp. 124—149, Jan. 1991. [PDF]

A. Grama, A. Gupta, V. Kumar. "Isoefficiency: Measuring the scalability of parallel algorithms and architectures." IEEE Parallel and Distributed Technology: Systems and Technology, 1(3):12–21, 1993. [WWW]

Related courses

High-Performance Computing (CSE 6220 / CS 6220) [Spring 2008]
High-Performance Computing: Tools and Applications (CSE 6230 / CS 6230; also, High Performance Parallel Computing) [Fall 2008]
High-Performance Computer Architecture (CS 6290)

Data Analysis

Faculty

Alberto Apostolico
Alexander Gray
Guy Lebanon
Haesun Park
Hongyuan Zha

Scope

Machine learning
Parameter optimization
Regression
Classification
Dimension reduction
Manifold learning

Suggested readings

Books

C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
L. Wasserman. *All of Statistics*. Springer, 2006.
Hastie et al. *The Elements of Statistical Learning* . Springer, 2001.
Witten et al. *Managing Gigabytes*. Morgan Kaufmann, 2nd Edition, 1999.

Related courses

Foundation of Machine Learning and Data Mining