

Course Outline

ECE 60872/CS 59000 – Fault-Tolerant Computer System Design

Purdue University Fall 2019

The course provides an introduction to the hardware and software methodologies for specifying, modeling and designing fault-tolerant systems, supported by case studies of real systems. The material presents a broad spectrum of hardware and software error detection and recovery techniques that can be used to build reliable networked systems. The lectures discuss how the hardware and software techniques interplay, what techniques can be provided in COTS hardware, what can be embedded into operating system and network communication layers, and what can be provided via a distributed software layer and in the application itself. It brings in the emerging role of data analytics in building and operating reliable systems.



The course focuses on hands-on learning through the design and development of innovative systems in the course project, which carries 50% of the weightage for the course grade.

This is reinforced by two lectures given by practitioners from the industry who share their experiences and insights in building dependable systems.

*Note: This is **not** an advanced graduate level course. Any student with a strong undergraduate CS or ECE background, i.e., one who is able to program in at least one high level programming language and has a basic knowledge of probability can take the class.*



NEW THIS YEAR

We will devote multiple lectures on the emerging topic of big data for reliability and security. This will cover some of the fundamental algorithms and 2 large use cases with real-world data.

Course Structure:

Class project: There will be different research projects that each team of 2 or 3 students will work on. Each project will focus on one aspect of fault-tolerant system design and will test the ability to design, model or implement, execute experiments and perform evaluation. The target will be to produce work that can be sent for a conference publication, which has happened with many projects in the past. Being graduate students, it is in the best interest of your career to build your publication record.

There will be the following phases in the project, each with their tentative timeline.

List of suggested projects made available	September 3
Project teams formed, discussion of project ideas with instructor	September 3-10
Project proposals submitted	September 12
Interim project presentations (15 minutes each group)	October 10, 12
Preliminary project report	October 15
Final project presentations	Last two days of class

Final project report	Last day of semester
----------------------	----------------------

Lecture Outline

This is the tentative outline of coverage of topics in the class. The new lectures this year are in red.

Introduction: Motivation, System view of high availability design, Terminology	2
Stochastic analysis of reliability <ul style="list-style-type: none"> • Discrete distributions • Continuous distributions 	6
Hardware redundancy: Basic approaches, Static & Dynamic, Voting, Coding for detection and recovery <ul style="list-style-type: none"> • <i>Application: SEC-DED codes</i> 	3
Error detection and correction techniques: Watchdog processors, Heartbeats, Consistency and capability checking, Data audits, Assertions, Control-flow checking <ul style="list-style-type: none"> • <i>Application: Erasure-coded storage</i> 	3
Software fault tolerance: Process pairs, Robust data structures, N version programming, Recovery blocks, Replica consistency & reintegration, Multithreaded programs <ul style="list-style-type: none"> • <i>Application: Quantitative evaluation of NVP and RB</i> 	3
Secure coding practices: Principles and practice <ul style="list-style-type: none"> • <i>Application: Coding examples</i> 	2
Network fault tolerance: Reliable communication protocols, Agreement protocols, Byzantine fault tolerance <ul style="list-style-type: none"> • <i>Application: Bitcoin</i> 	5
Big data for reliability <ul style="list-style-type: none"> • <i>Application: Failure analysis of Purdue compute clusters</i> 	2
Big data for security <ul style="list-style-type: none"> • <i>Application: ML analysis of ransomware</i> 	2
Modeling <ul style="list-style-type: none"> • <i>Application: UltraSAN, Sharpe</i> 	2
Checkpointing & Recovery <ul style="list-style-type: none"> • <i>Application: SCR checkpointing system for DOE supercomputers</i> 	3
Experimental Evaluation: Simulation and Fault-injection based	2
Practical Systems for Fault Tolerance: Putting it all together <ul style="list-style-type: none"> • <i>Application: Amazon Web Service</i> • <i>Application: Hadoop</i> 	2
Industry presentations	2
Discussion of projects	2
Project presentations	2
Tests	1
Total	44