

CITS7211 Modelling Complex Systems

Lecture 6: Modelling Concurrent Systems

Overview

In this lecture we examine concurrent software as a complex system. Concurrent software exhibits many of the properties we have seen in complex physical system, including a high degree of unpredictability, emergent behaviour and local simplicity. As with many of the systems we have seen, concurrent software systems may be represent using interacting automata, except that rather than using spatial dimensions, the topology of the system represents communication channels between the automata. We examine an automata based technique for modelling concurrent systems, and show how these models may be used to guarantee the correct behaviour of the system.

Contact: Tim French, CSSE Rm 2.14 (Tuesday 11-1), tim@csse.uwa.edu.au.

Topics:

1. Concurrent Systems:
 - a. Finite state machines (revision)
 - b. Propositions, states, transitions, invariants.
 - c. Concurrency
2. Specifications:
 - a. Propositional Logic
 - b. Temporal Logic
 - c. Timed automata
3. Modelling and Verification
 - a. Non-determinism
 - b. Model checking
 - c. Theorem proving
4. Modelling with Uppaal
 - a. Specifying a system.
 - b. Simulating a system
 - c. Formalizing and verifying properties
5. Examples:
 - a. Mutual Exclusion
 - b. Vikings on a bridge
 - c. Dining Philosophers

Reading:

1. *Uppaal in a Nutshell*, Larsen, Pettersson and Yi, 1997. Download and install Uppaal and try out the demos. See if you can define some queries for the Dining Philosophers.
2. Dining Philosophers in Uppaal, (see webpage).

Notices:

- a. Could everyone please arrange a time in the next two weeks to meet me to discuss their progress in the project.
- b. Next week is a guest lecture by Prof George Milne and Joel Kelso. There is some preparatory reading on the unit website.