Real-time Scheduling of Aperiodic and Sporadic Tasks

Advanced Operating Systems
Tutorial 2
Tutorial Outline

• Review of lectured material
• Worked examples
Review of Lectured Material

• Aperiodic and sporadic tasks; acceptance tests

• Scheduling aperiodic jobs
  • Background execution
  • Periodic servers: polling, deferrable, and sporadic
  • Critical instant analysis for fixed-priority deferrable server; maximum utilisation test for deferrable server in EDF systems
  • Sporadic server budget consumption/replenishment; proofs of correctness

• Scheduling sporadic jobs
  • Acceptance test in EDF systems: density of intervals
  • Acceptance test in rate monotonic systems: maximum usage over periods

• Implementation choices
Key Learning Outcomes

- Understand scheduling of aperiodic jobs using periodic server
- Understand design trade-off between types of periodic server
- Know scheduling guarantees for various types of periodic server
- Know how to accept and schedule sporadic jobs; scheduling tests
Worked Examples

• Deferrable server
• Sporadic tasks – acceptance test
• Sporadic tasks – in hard real-time systems
Example 1(a): Deferrable Server

• Consider a system with three periodic tasks:
  • $T_1 = (6, 1)$
  • $T_2 = (10, 1)$
  • $T_3 = (14, 3)$

• Questions:
  • What does an EDF schedule for this system look like?
Example 1(b): Deferrable Server

- Consider a system with three periodic tasks:
  - $T_1 = (6, 1)$
  - $T_2 = (10, 1)$
  - $T_3 = (14, 3)$

- Questions:
  - What does an EDF schedule for this system look like?
  - A deferrable server with period 4 and budget 1 is added. Can the system be scheduled?
Example 1(c): Deferrable Server

- Consider a system with three periodic tasks:
  - $T_1 = (6, 1)$
  - $T_2 = (10, 1)$
  - $T_3 = (14, 3)$

- Questions:
  - An aperiodic task arrives with $r_A = 6$ and $e_A = 2$. What does the schedule look like? How does the budget of the server vary?
Example 2: Sporadic tasks – Acceptance Tests

• When sporadic tasks are introduced into a priority-scheduled system of periodic tasks, it becomes necessary to incorporate an acceptance test into that system.

• Describe the purpose of an acceptance test, and why is it important for error handling.
Example 3: Sporadic tasks – Hard Real-Time

• Are sporadic tasks incompatible with hard real-time systems?
Discussion

• Should understand how to evaluate the schedules for various types of server
• Should know how to demonstrate correctness of a system with aperiodic or sporadic tasks scheduled using a periodic server

• Exercise 1 is due now
• Exercise 2 is available – due in tutorial 3