Priority Inheritance Protocols: An Approach to Real-Time Synchronization

L. Sha, R. Rajkumar, and J. P. Lehoczky

Priority Inversion Problem

➢ In traditional synchronization techniques
  ▪ Higher priority tasks can be blocked by lower priority tasks

➢ Motivating example
  ▪ Three tasks: P1, P2, P3
    • P1 has the highest priority
  ▪ P3 locks the semaphore S
  ▪ P1 preempts P3 and then attempts to lock S, but is blocked
    • This necessary for mutual exclusion
  ▪ P2 preempts P3, delaying P1
Priority Inversion Example

- The blocking of P1 is affected by independent P2
  - In the interval (t5, t6)
- The blocking duration is unpredictable
Priority Inheritance Protocol

➢ A simple, but not good solution
  ▪ A task in its critical section is not allowed to be preempted
  ▪ But, higher priority tasks may suffer unnecessary blocking

➢ Priority Inheritance Protocol
  ▪ When a task blocks one or more higher priority tasks
  ▪ It ignores its original priority
  ▪ It executes its critical section at the highest priority level of all the tasks it is blocking

➢ Priority inheritance is transitive
  ▪ If P3 blocks P2, and P2 blocks P1, P3 inherits P1’s priority
➢ **P3 inherits the P1’s priority from t4**
➢ **P2 cannot preempt P1 at t5**
Limitations of PIP

- PIP have two problems
  - Do not prevent deadlocks
  - Chain of blocking can be formed

P2 and P3 are in nested critical sections resulting in deadlocks!
**Chained Blocking Example**

- **P1 is blocked two times**
  - When it attempts to lock S1
  - When it attempts to lock S2
Priority Ceiling Protocol

➢ Based on Priority Inheritance Protocol

➢ A priority ceiling is assigned to each semaphore
  ▪ The highest priority of tasks that may use the semaphore

➢ Task P is allowed to start a new critical section
  ▪ Only if P's priority is higher than all priority ceilings of all the semaphores locked by jobs other than P

➢ PCP can
  ▪ Prevent deadlocks
  ▪ Avoid chained blocking
PCP Example

- P1: \{S1\}, P2: \{S2, S3\}, P3: \{S3, S2\}
- At t3, P2’s priority is higher than P3
- At t4, P2’s priority is not higher than S3’s ceiling
- At 6, P1’s priority is higher than P2’s priority and S3’s ceiling
Schedulability Analysis

- Schedulability condition

\[
\forall i, 1 \leq i \leq n, \quad \frac{C_1}{T_1} + \frac{C_2}{T_2} + \cdots + \frac{C_i}{T_i} + \frac{B_i}{T_i} \leq i(2^{1/i} - 1).
\]

- \(B_i\) is the worst case blocking time of task \(i\)