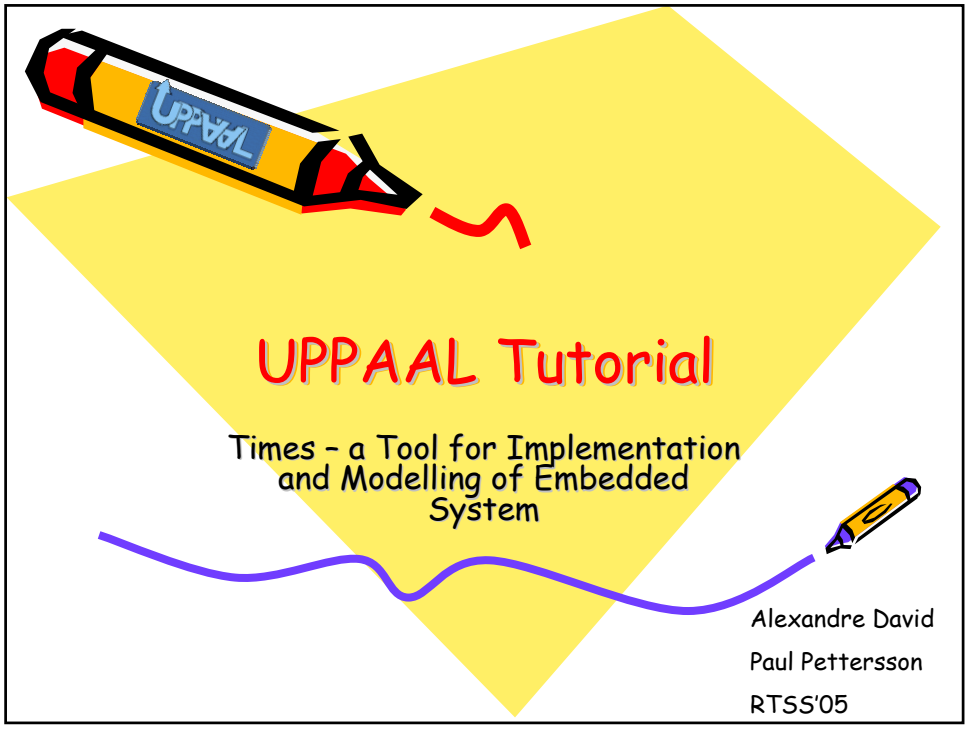
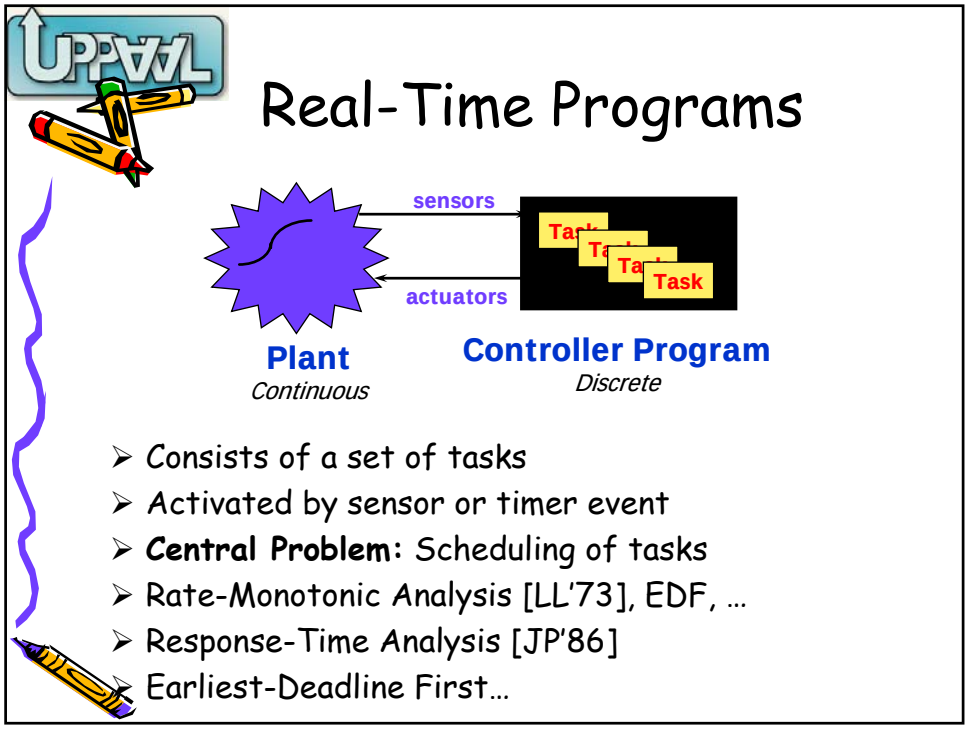
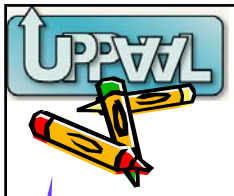

UPPAAL Tutorial
Times – a Tool for Implementation and Modelling of Embedded System
Alexandre David
Paul Pettersson
RTSS'05


Real-Time Programs
sensors
actuators
Task
Plant
Continuous
Controller Program
Discrete
Consists of a set of tasks
Activated by sensor or timer event
Central Problem: Scheduling of tasks
Rate-Monotonic Analysis [LL'73], EDF, …
Response-Time Analysis [JP'86]
Earliest-Deadline First…



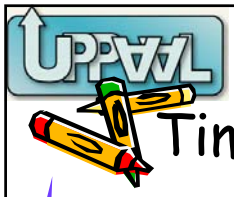
Classical vs. Automata-based Approach

Classic Scheduling Theory

- Periodic Tasks
- Well studies algorithms
- Efficient
- Pessimistic (when tasks are not periodic)
- Extensions for sporadic, jitter, etc.

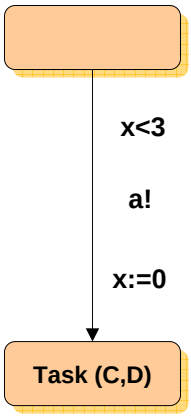
Automata-based Approach

- Automata used to describe task arrival
- General release patterns
- Precise
- Schedulability?
- Decidability?
- Efficiency?

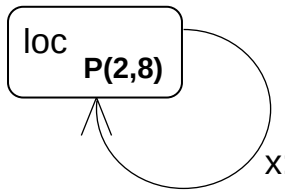


Timed Automata with Tasks

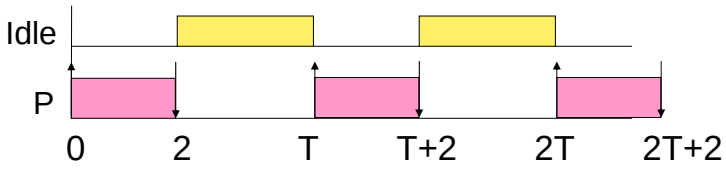
- Events
  - synchronization
  - interrupts
- Timing constraints
  - specifying event arrivals
  - e.g. periodic or sporadic
- Tasks (executable programs)
  - asynchronous execution
  - interrupt processing
  - internal computation



Periodic Task

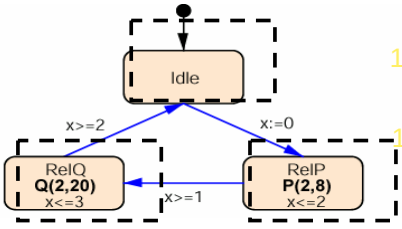


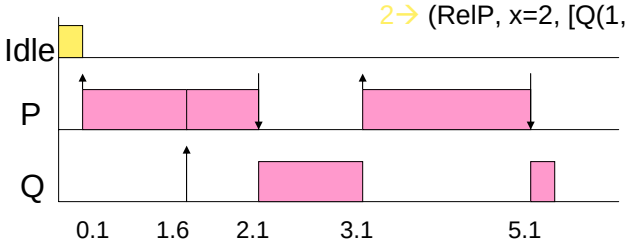
(loc, x=0, [])
(loc, x=0, [P(2,8)])
(loc, x=2, [])
(loc, x=T, [])
(loc, x=0, [P(2,8)])
(loc, x=2, [])
...



Example

(Idle, x=0, [])
0.1→ (Idle, x=0.1, [])
→ (RelP, x=0, [P(2,8)])
1.5→ (RelP, x=1.5, [P(0.5,6.5)])
→ (RelQ, x=1.5, [P(0.5,6.5),Q(2,20)])
1.5→ (RelQ, x=3, [Q(1,18.5)])
→ (Idle, x=3, [Q(1,18.5)])
→ (RelP, x=0, [P(2,8),Q(1,18.5)])
2→ (RelP, x=2, [Q(1,16.5)])





Decidable?

Schedulability Analysis

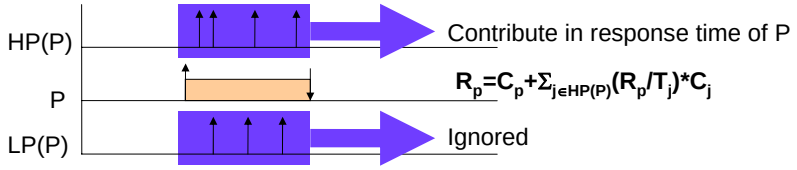
- All tasks meet their deadlines?
- FACT [Wall, Yi, 1998]
  - The schedulability problem is **decidable** for **non-preemptive** tasks.
- Conjecture [since 1998]
  - The schedulability problem is **undecidable** for **preemptive** tasks.

WRONG!

- FACT [TACAS 2001]
  - The Schedulability Problem **is decidable for preemptive tasks.**

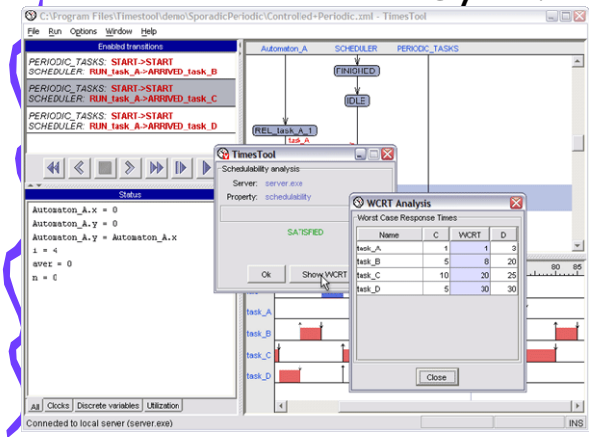
Efficiency [TACAS'03]

- High complexity for the general case, e.g. EDF
- 2 clocks for each task instance n_i of a task type
- Number of clocks for EDF: $2\Sigma_i n_i$
- For fixed priorities the analysis is similar to Response Time Analysis (Joseph, Pandya):



- For fixed priority systems only two clocks are needed for schedulability analysis!

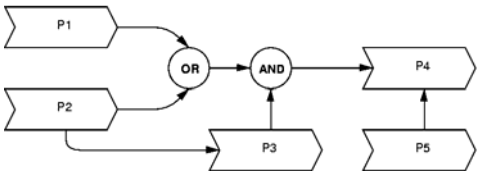
TIMES -Tool for Automatic Schedulability Analysis and Code Synthesis



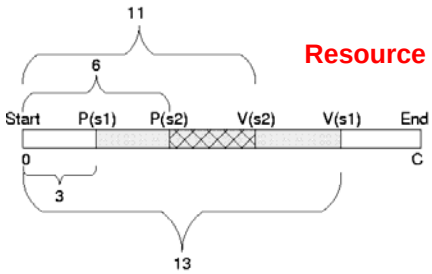
- Timed Automata with Tasks
  - FPS, EDF, …
  - Shared resources (Priority Ceiling Protocol)
- Analysis:
  - Schedulability
  - Response-time
  - Model-checking
- Simulation with MSC and Gantt-charts
- Code synthesis

Extensions [NJC'04]

Precedence Constraints



(P1 or P2) and P3 and P5 → P4

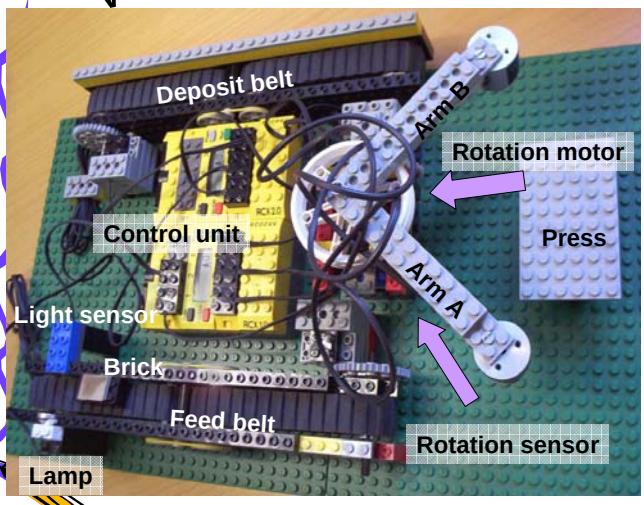


Resource Constraints

(T, OP, A)
3, P(s1)
6, P(s2)
11, V(s2), A2
13, V(s1), A1

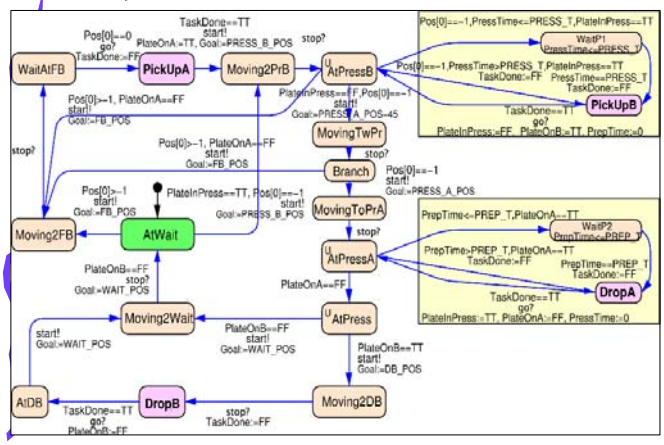
TIMES DEMO

Production Cell [NJC'03]



- Academic example
- LEGO Hitachi H8 processor
- brickOS operating system
- 15 tasks
- Task code written in C

Application: Production Cell



System:

- 15 tasks
- Hitachi H8
- Task code in C

Analysis:

- 9 sec and 13 MB using over-approximation
- Identified dead code

Synthesis:

- 550 lines of C-code excl. task code
- Executed on hardware