Performance Analysis, Scheduling and Synthesis of Embedded Systems

Kim G. Larsen

CISS – Aalborg University

DENMARK

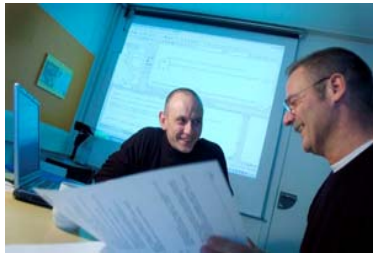
CISS in Numbers

- National ICT Comptetence Center 2002:

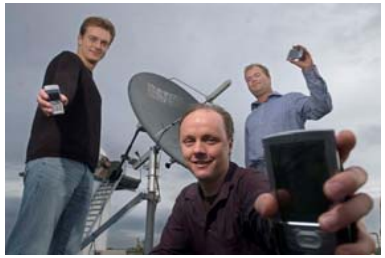
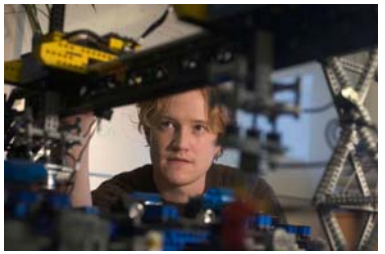
  31,5 MDKK Ministry
  8,5 MDKK North Jutland
  7,5 MDKK Aalborg City
  16,00 MDKK Companies
  16,00 MDKK AAU

- **45** projects
- **20** CISS employees
- **25** CISS associated researcher at 3 different research groups at AAU!
- >20 industrial PhDs







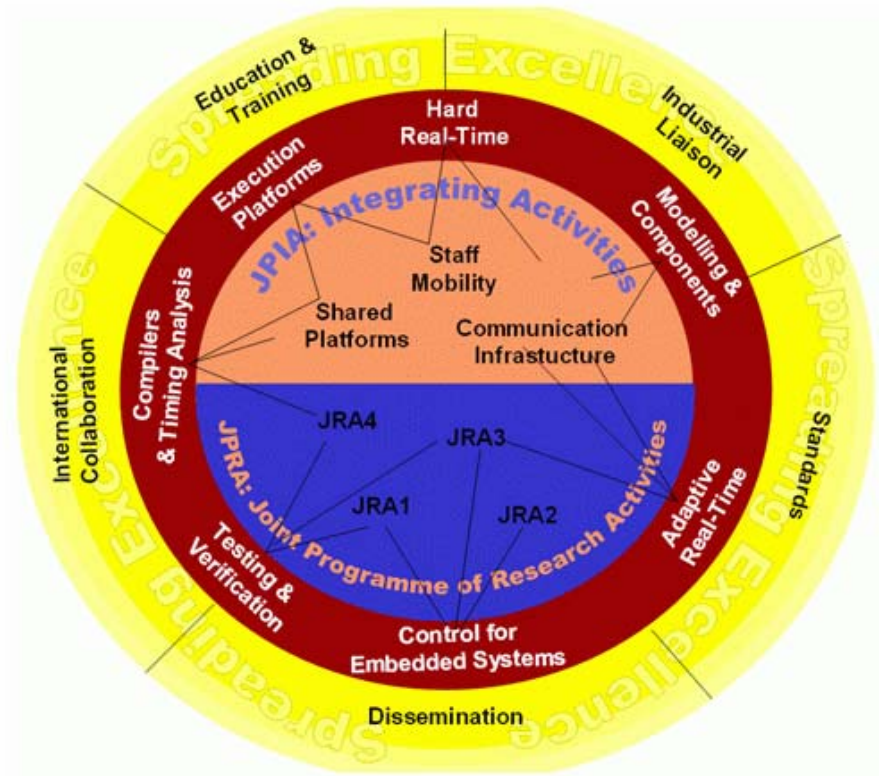



European Projects

- **ARTIST2** (NoE FP6)
  - coordinator for Testing & Verification Cluster

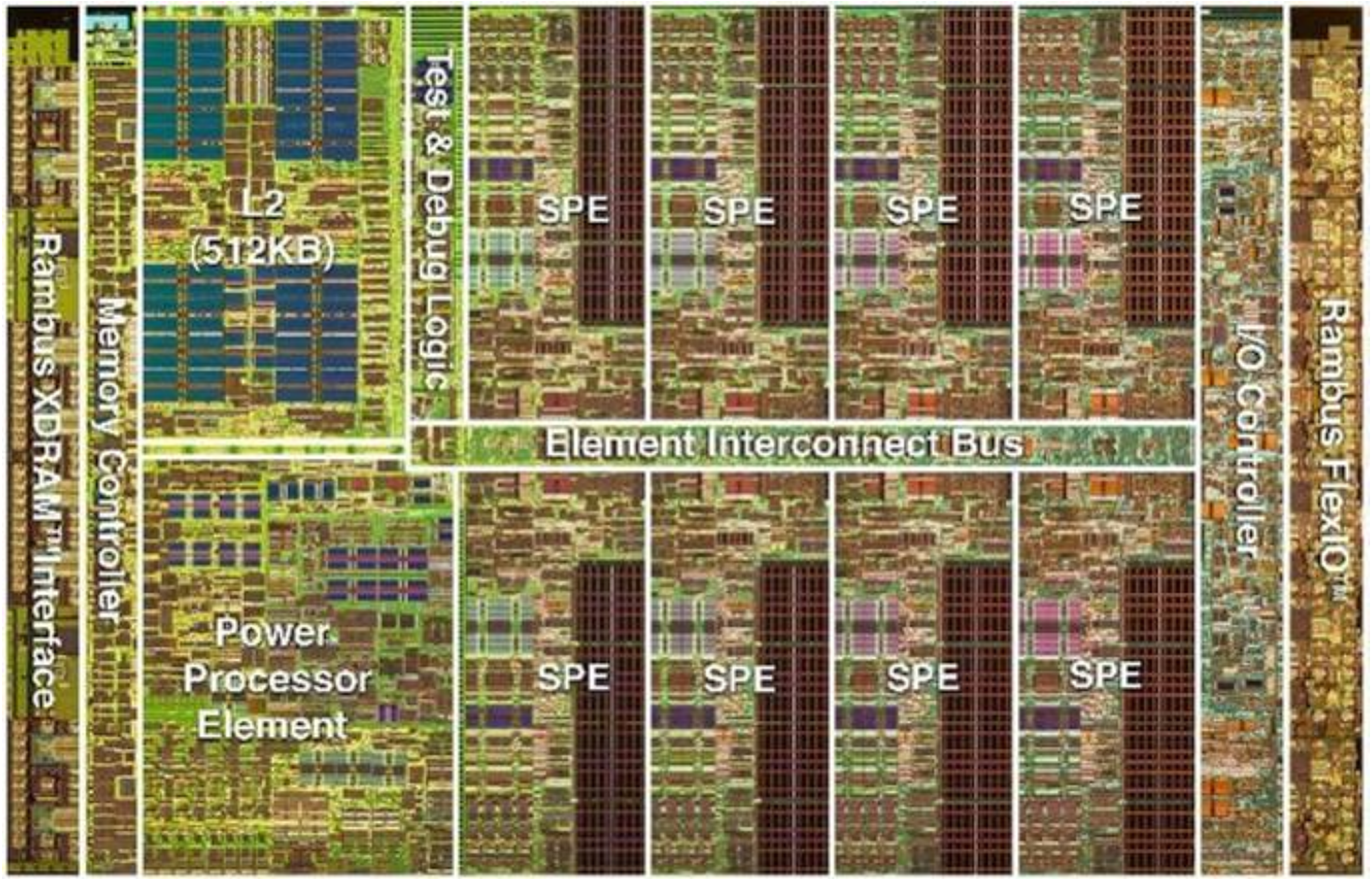
- **ARTIST Design** (NoE FP7)
  - kick-off meeting end of January
  - co-coordinator of Modeling and Validation w Tom Henzinger)

- Other new STREPs (FP7) Quasimodo, Multiform

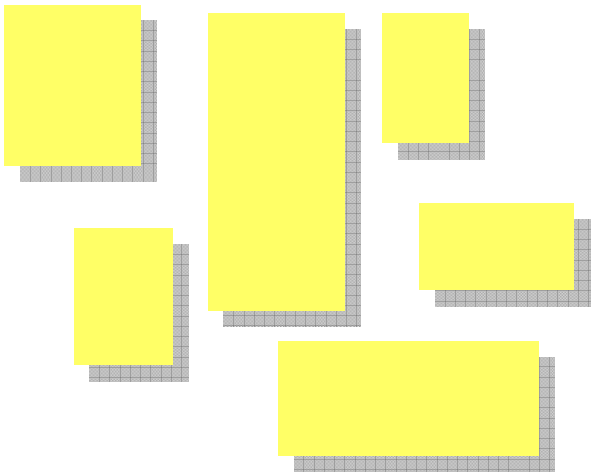


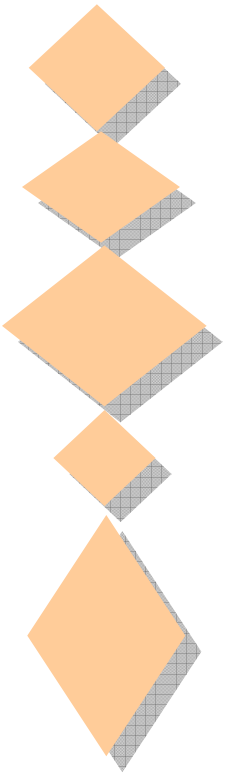
Motivation – MPSoC

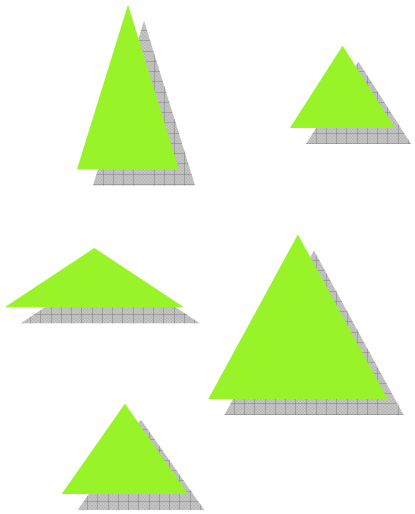
CELL processor



Scheduling… in ES





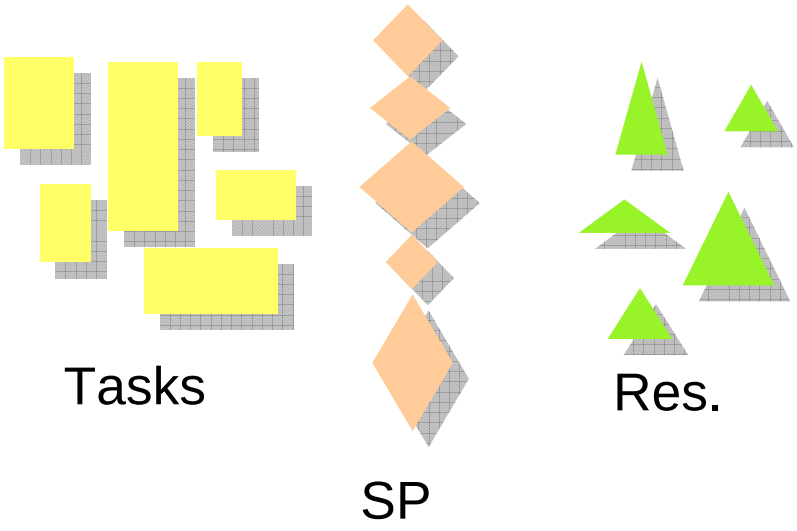


Tasks:
Computation times
Deadlines
Dependencies
Arrival patterns
uncertainties

Scheduling Principles (OS)
EDF, FPS, RMS, DVS, ..

Resources
Execution platform
PE, Memory
Networks
Drivers
uncertainties

Issues



- Performance Evaluation
  - Estimate resources (e.g. energy) required by given SP.
- Scheduling & Synthesis
  - Synthesize (optimal) SP ensuring given objective.
  - *Scheduling*: SP controls everything (including ex.time).
  - *Synthesis*: scheduling under uncertainties (e.g. execution time, availability of resources).

- Schedulability Analysis
  - Verify that given SP ensures deadlines.

Approach – TA

- Performance Evaluation
  - Estimate resources (e.g. energy) required by given

CLASSIC

- Synthesis
  - optimal) SP given objective.
  - *Scheduling*: SP controls

CORA

TIGA

- Sch
  - Verify that given SP ensures deadlines.

TALK:
What can we do?
What can we do efficiently?
What can not be done?
What would we like to do?

The UPPAAL Team

@UPPsala

- Wang Yi
- Paul Pettersson
- John Håkansson
- Anders Hessel
- Pavel Krcal
- Leonid Mokrushin
- Shi Xiaochun

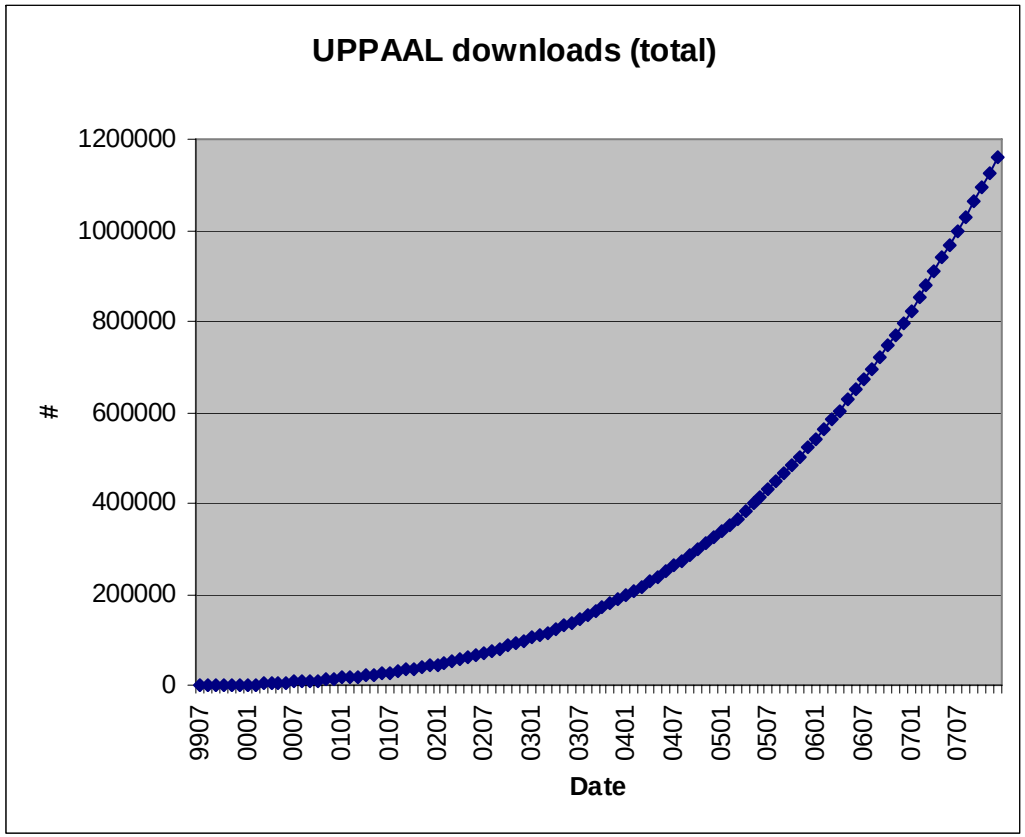
@AALborg

- Kim G Larsen
- Gerd Behrman
- Arne Skou
- Brian Nielsen
- Alexandre David
- Jacob I. Rasmussen
- Marius Mikucionis
- Thomas Chatain

@Elsewhere

- Emmanuel Fleury, Didier Lime, Johan Bengtsson, Fredrik Larsson, Kåre J Kristoffersen, Tobias Amnell, Thomas Hune, Oliver Möller, Elena Fersman, Carsten Weise, David Griffioen, Ansgar Fehnker, Frits Vandraager, Theo Ruys, Pedro D'Argenio, J-P Katoen, Jan Tretmans, Judi Romijn, Ed Brinksma, Martijn Hendriks, Klaus Havelund, Franck Cassez, Magnus Lindahl, Francois Laroussinie, Patricia Bouyer, Augusto Burgueno, H. Bowmann, D. Latella, M. Massink, G. Faconti, Kristina Lundqvist, Lars Asplund, Justin Pearson...

“Impact”



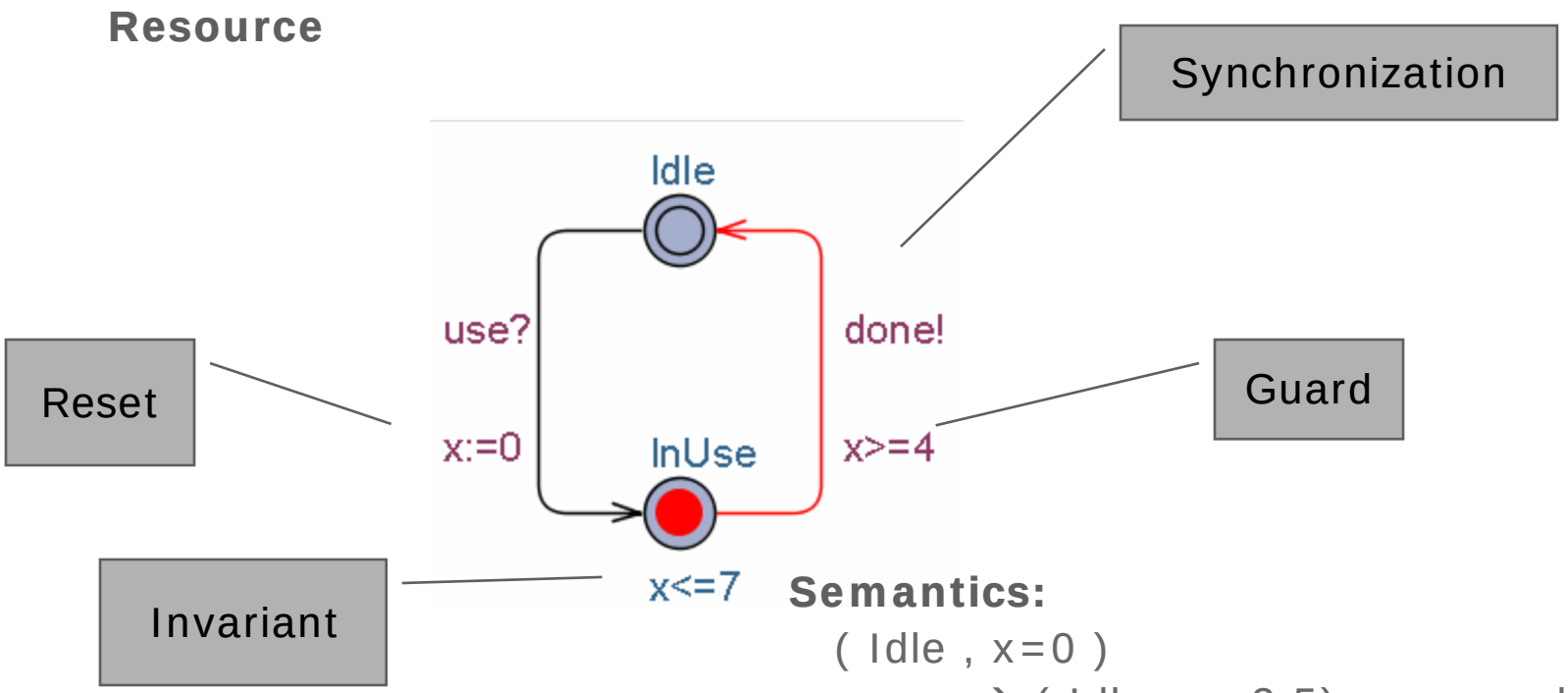
Google:

UPPAAL:	134.000
SPIN Verifier:	242.000
nuSMV:	57.700

> 2.900
Google Scholar Citations
(Rhapsody/Esterel < 5.000)

Timed Automata

[Alur & Dill'89]

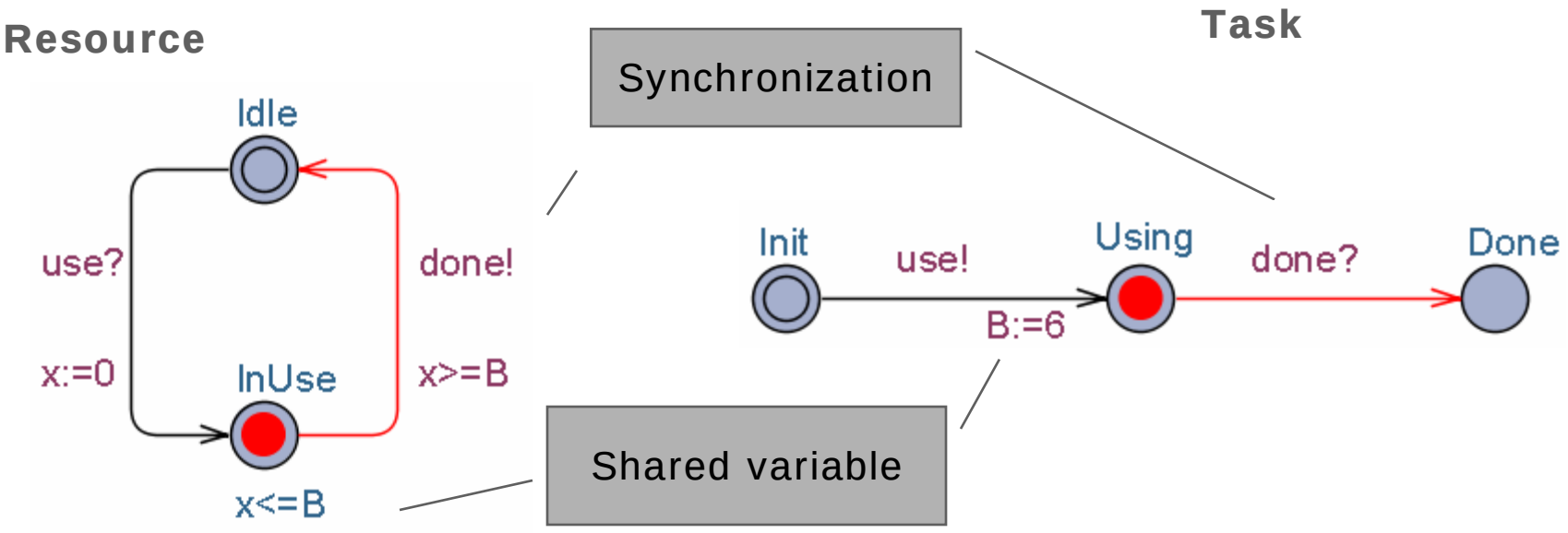


Semantics:

(Idle , x=0)

→ (Idle , x=2.5) d(2.5)
→ (InUse , x=0) use?
→ (InUse , x=5) d(5)
→ (Idle , x=5) done!
→ (Idle , x=8) d(3)
→ (InUse , x=0) use?

Composition



Semantics:

(Idle , Init , B=0, x=0)

→ (Idle , Init , B=0 , x=3.1415) d(3.1415)

→ (InUse , Using , B=6, x=0) use

→ (InUse , Using , B=6, x=6) d(6)

→ (Idle , Done , B=6 , x=6) done

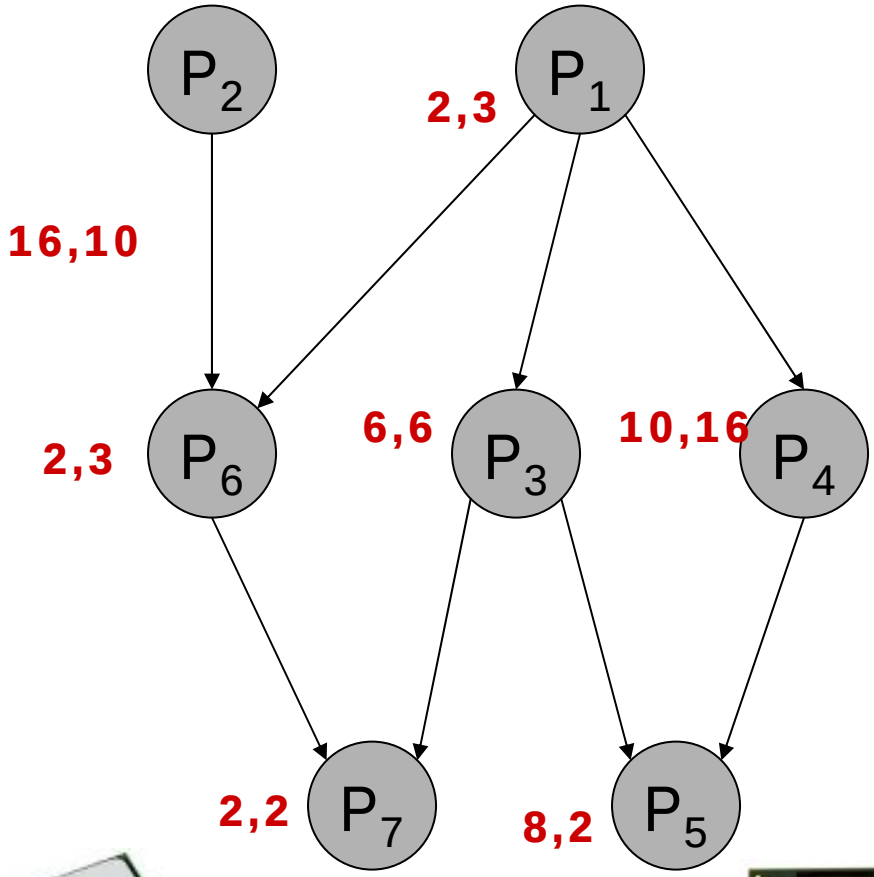
Task Graph Scheduling

Optimal Static Task Scheduling

- Task **P**={P_1,.., P_m}
- Machines **M**={M_1,..,M_n}
- Duration Δ : **(P**×**M)** → $\mathbf{N}_\infty$
- < : p.o. on **P** (pred.)

- A task can be executed only if all predecessors have completed
- Each machine can process at most one task at a time
- Task cannot be preempted.

- Compute schedule with minimum completion-time!

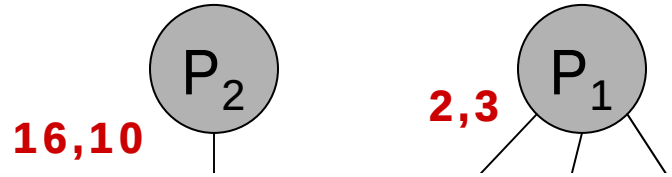


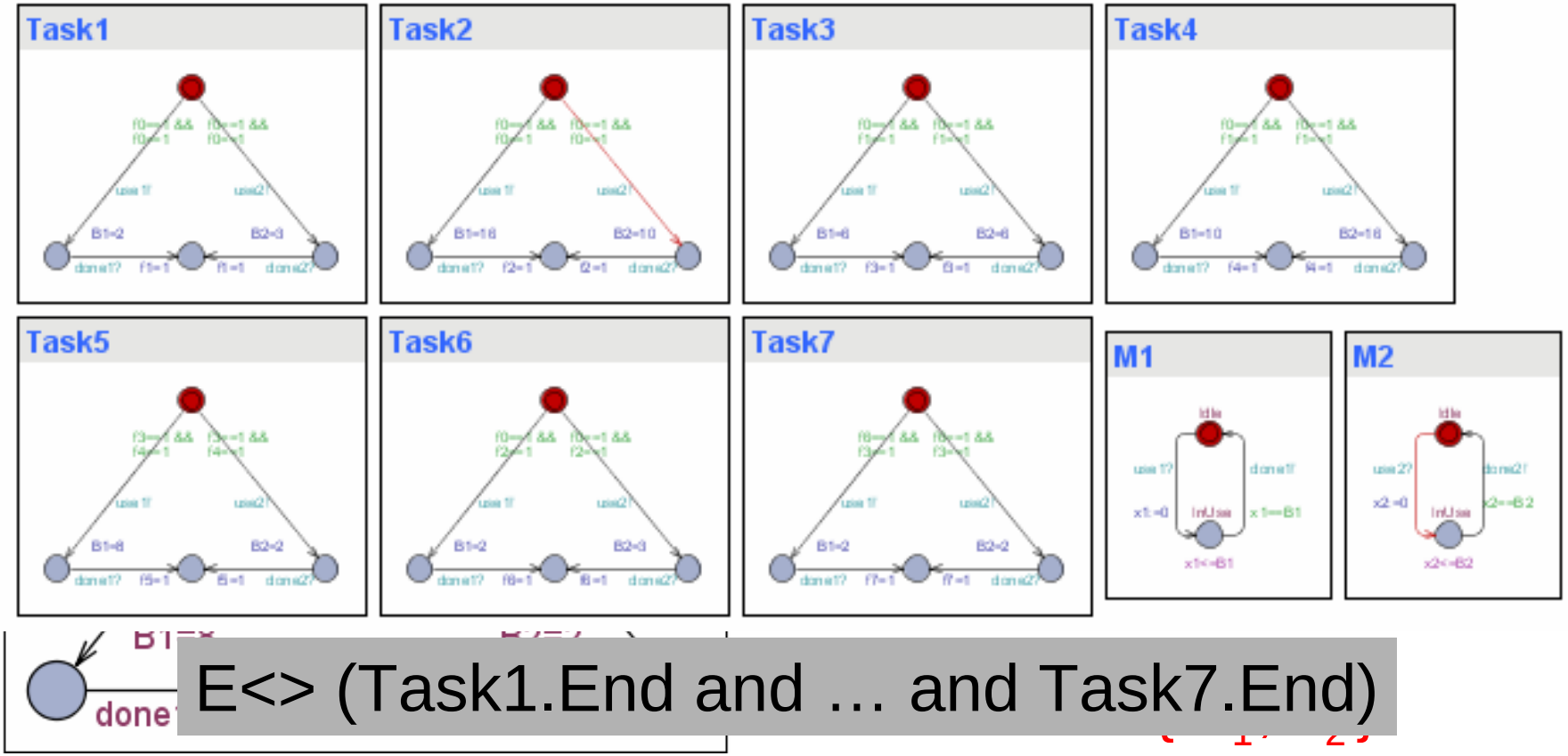
M = {M_1,M_2}

Task Graph Scheduling

Optimal Static Task Scheduling

- Task **P**={P_1,.., P_m}
- Machines **M**={M_1,..,M_n}





E<> (Task1.End and … and Task7.End)

Experimental Results

name	#tasks	#chains	# machines	optimal	TA
001	437	125	4	1178	1182
000	452	43	20	537	537
018	730	175	10	700	704
074	1007	66	12	891	894
021	1145	88	20	605	612
228	1187	293	8	1570	1574
071	1193	124	20	629	634
271	1348	127	12	1163	1164
237	1566	152	12	1340	1342
231	1664	101	16	t.o.	1137
235	1782	218	16	t.o.	1150
233	1980	207	19	1118	1121
294	2014	141	17	1257	1261
295	2168	965	18	1318	1322
292	2333	318	3	8009	8009
298	2399	303	10	2471	2473

AMETIST
advanced methods for timed systems

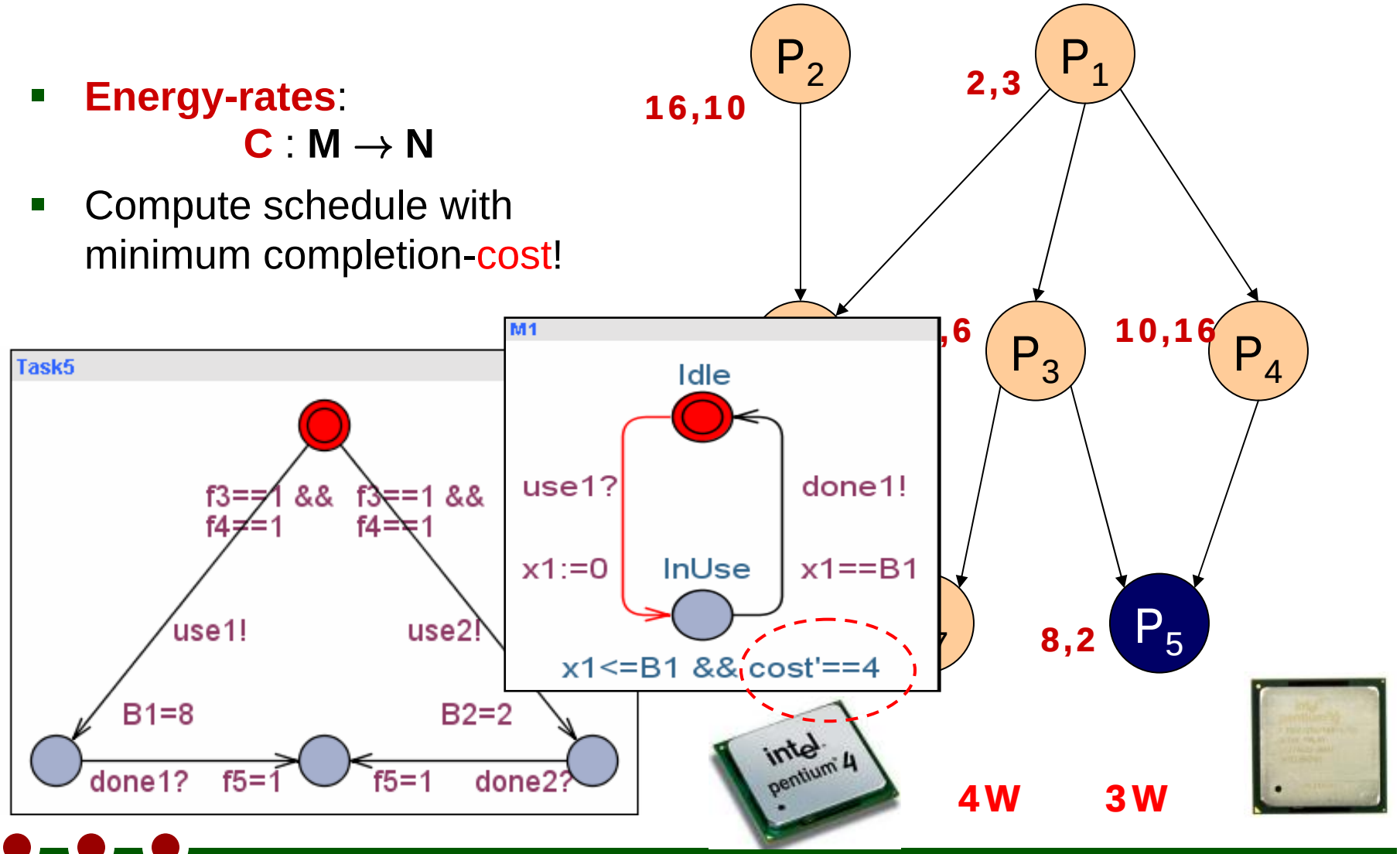
Symbolic A*
Brand-&-Bound
60 sec

Abdeddaïm, Kerbaa, Maler

Optimal Task Graph Scheduling

Power-Optimality

- **Energy-rates**:
  $C : M \rightarrow N$
- Compute schedule with minimum completion-cost!

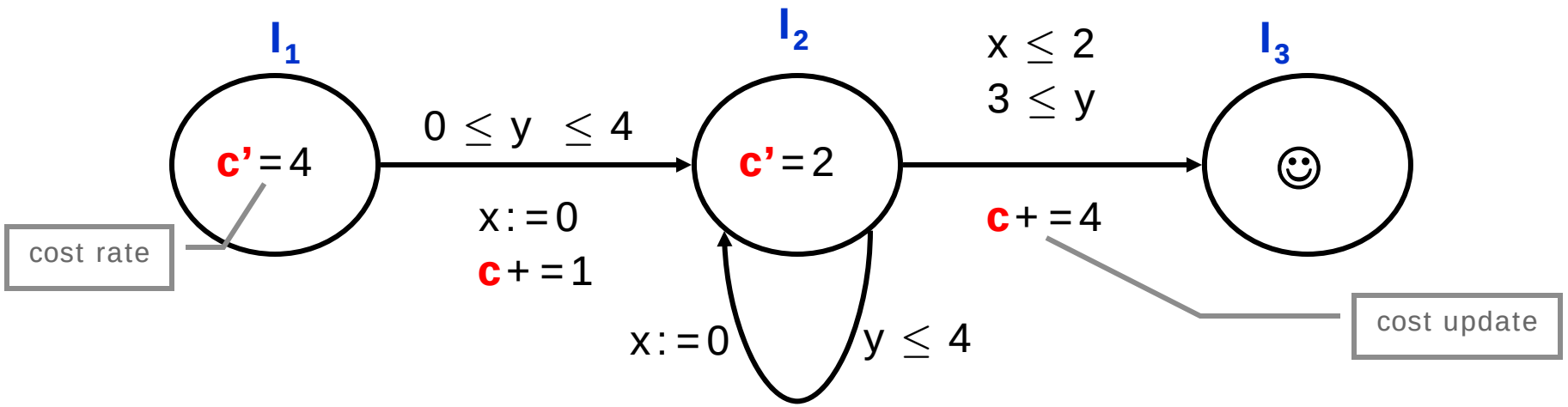


Priced Timed Automata

Behrmann, Fehnker, et all (HSCC'01)

Alur, Torre, Pappas (HSCC'01)

Timed Automata + **COST** variable

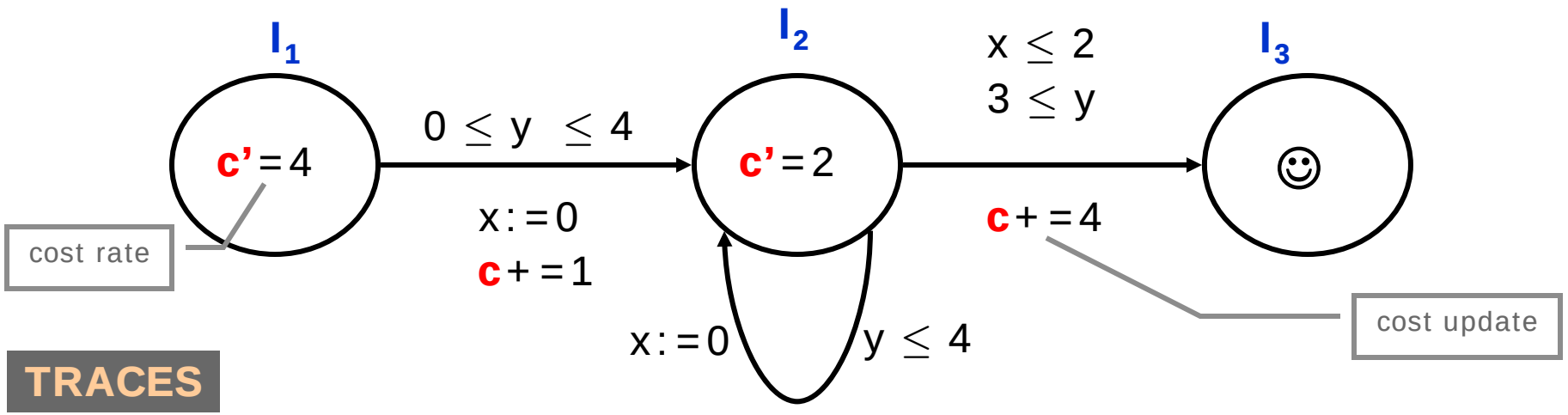


Priced Timed Automata

Behrmann, Fehnker, et all (HSCC'01)

Alur, Torre, Pappas (HSCC'01)

Timed Automata + **COST** variable



TRACES

$(l_1, x=y=0) \xrightarrow[12]{\varepsilon(3)} (l_1, x=y=3) \xrightarrow{1} (l_2, x=0, y=3) \xrightarrow{4} (l_3, \_, \_)$

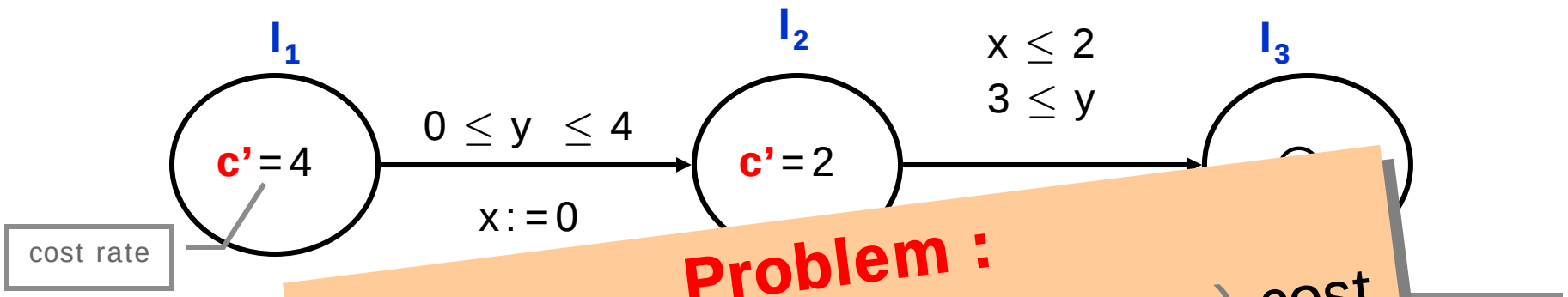
$\sum c = 17$

Priced Timed Automata

Timed Automata + **COST** variable

Behrmann, Fehnker, et all (HSCC'01)

Alur, Torre, Pappas (HSCC'01)



Problem :

Find the minimum (maximum) cost of reaching location l_3

Efficient Implementation:
CAV'01 and TACAS'04

TRACES

$(l_1, x=y=0)$

$(l_1, x=y=0) \xrightarrow{\varepsilon}$

$(l_1, x=y=0) \xrightarrow{1} (l_2, x=0, y=0$

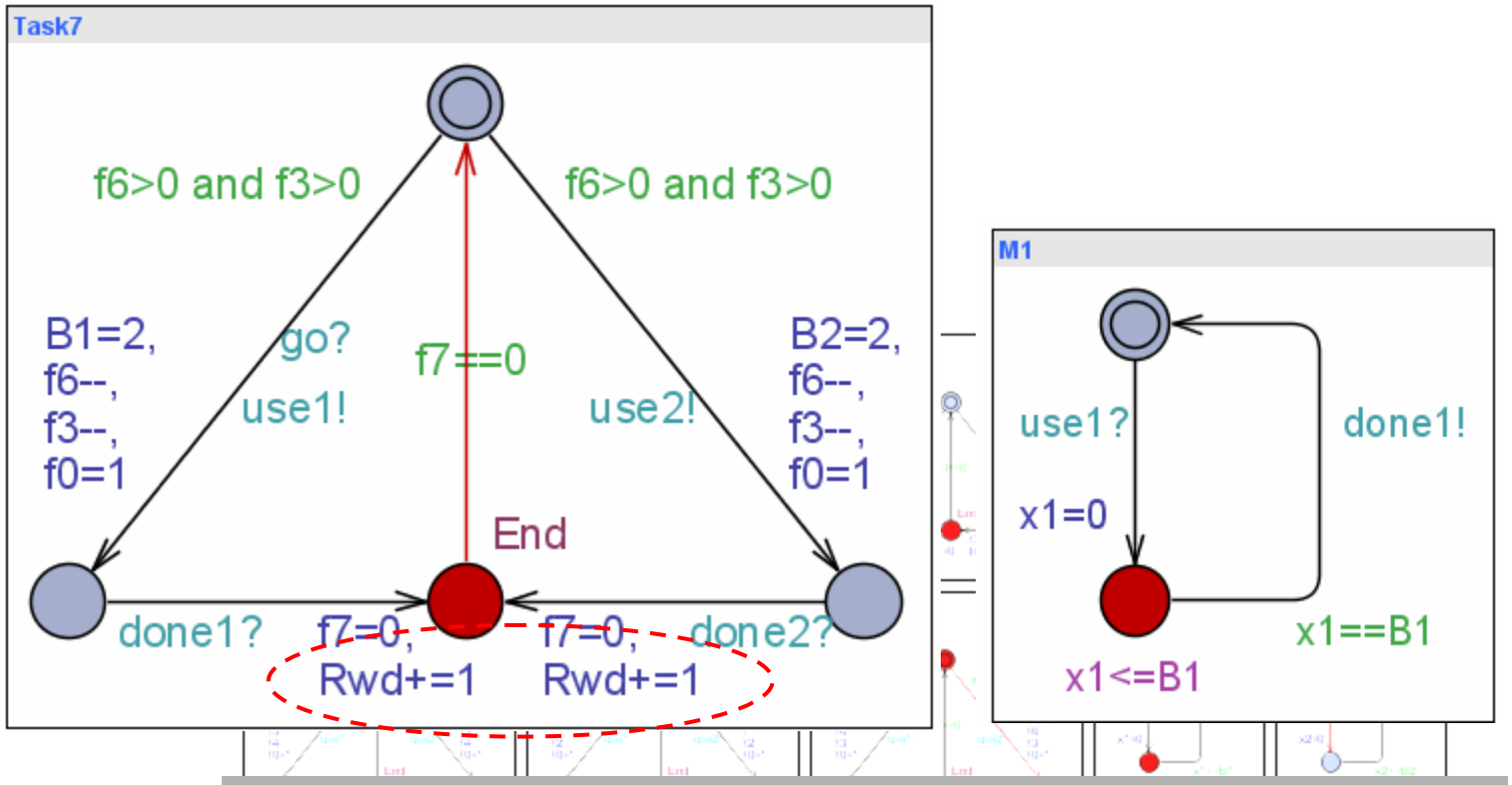
Σ c=17

$(l_3, \_, \_)$
Σ c=16

$\xrightarrow{4}$ $(l_3, \_, \_)$
Σ c=11

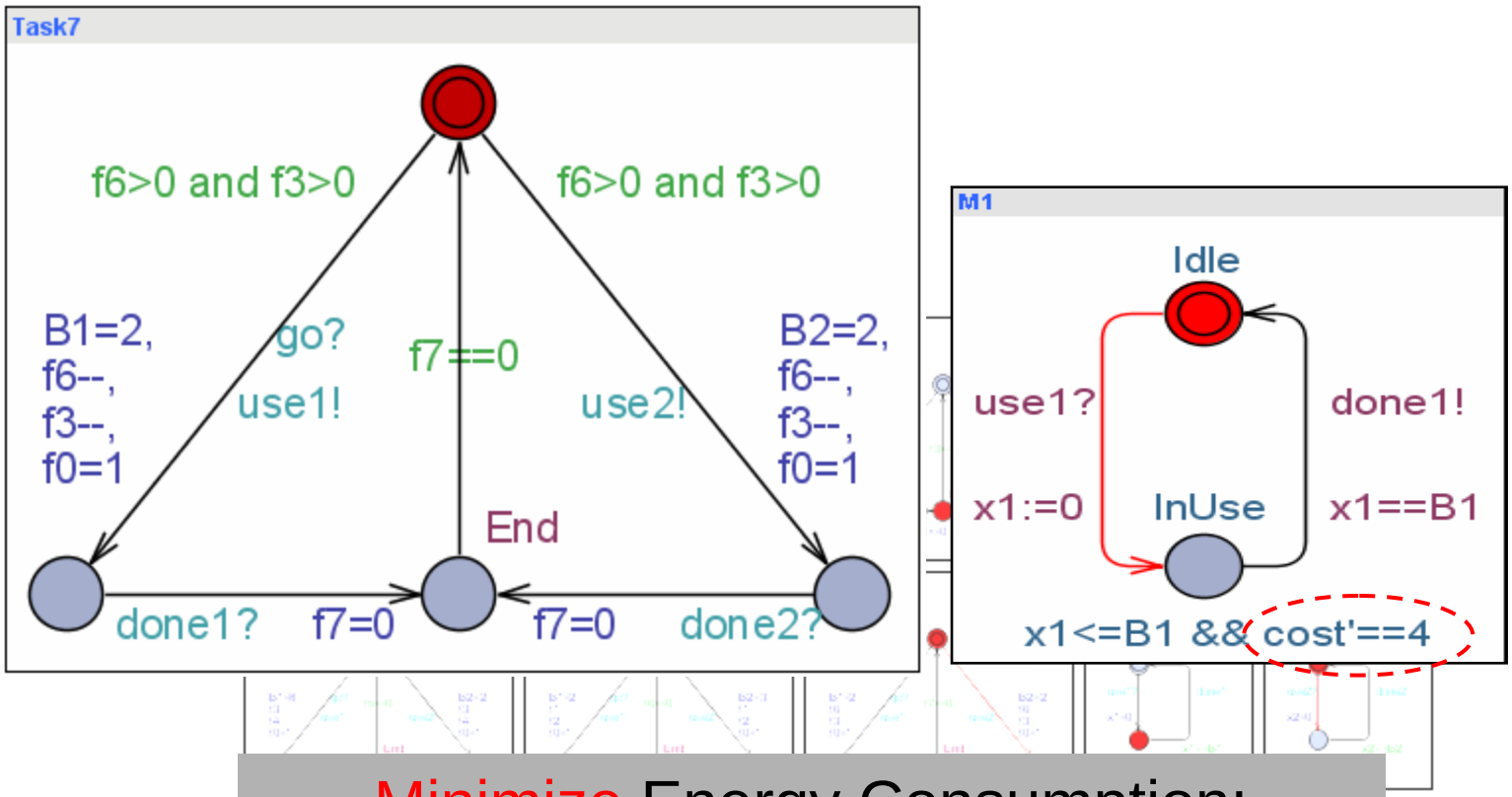
Competitive with MILP and commercial tool (Axxon)

Optimal Infinite Scheduling



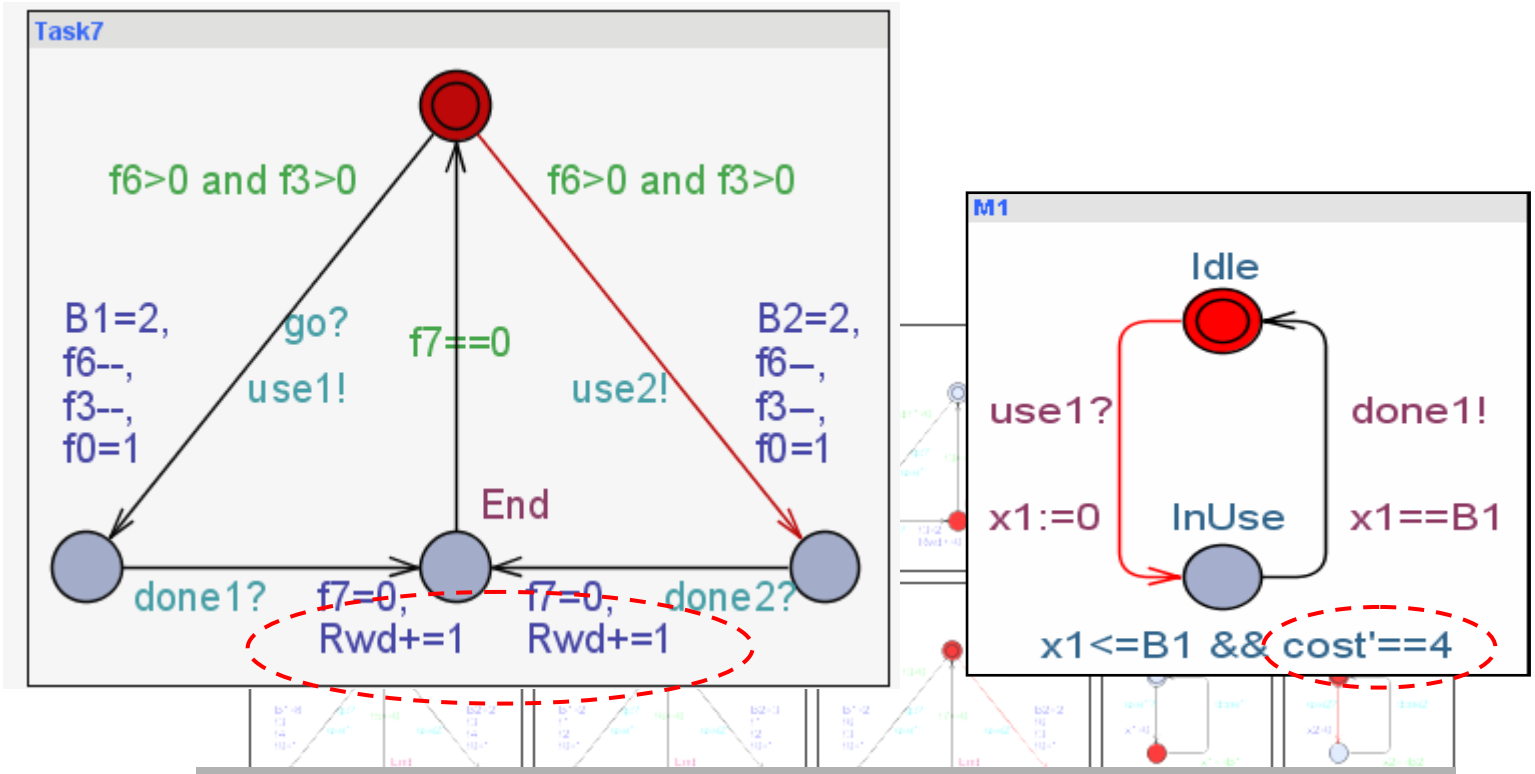
Maximize throughput:
i.e. maximize Reward / Time in the long run!

Optimal Infinite Scheduling



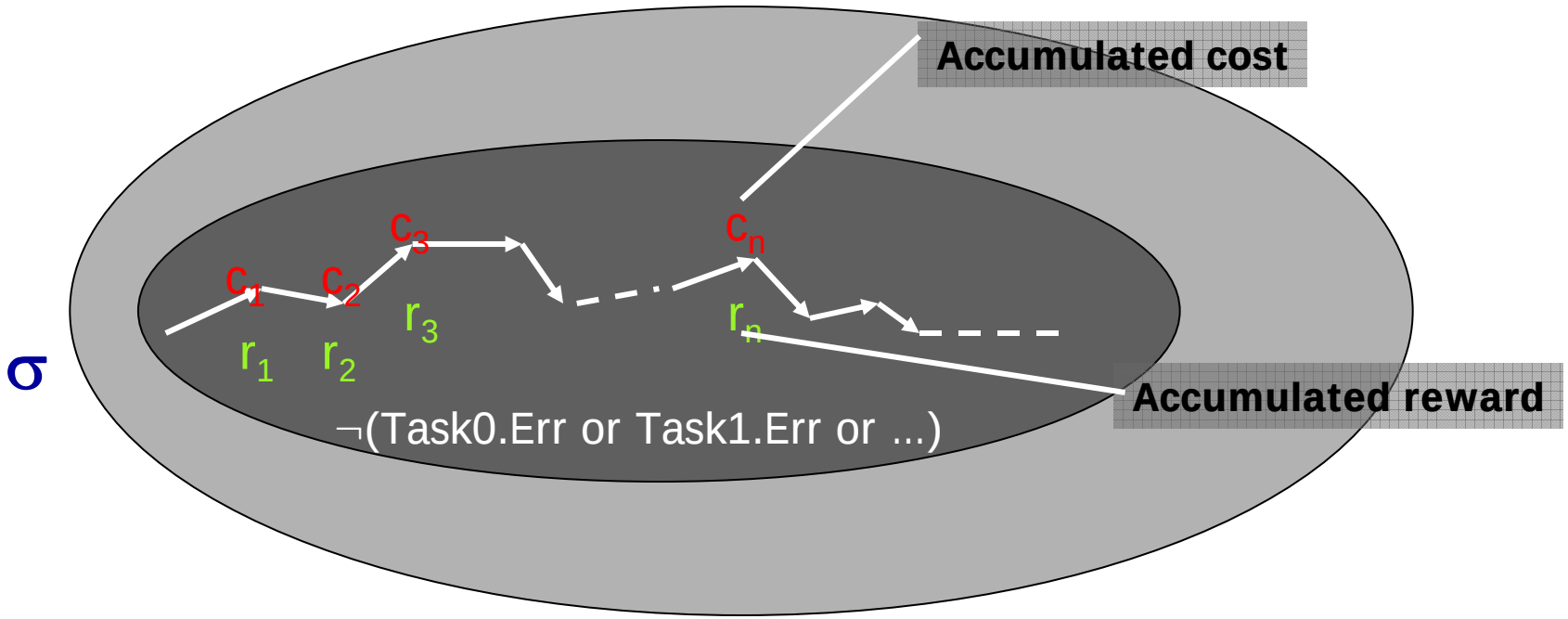
Minimize Energy Consumption:
i.e. minimize Cost / Time in the long run

Optimal Infinite Scheduling



Maximize throughput:
i.e. maximize Reward / Cost in the long run

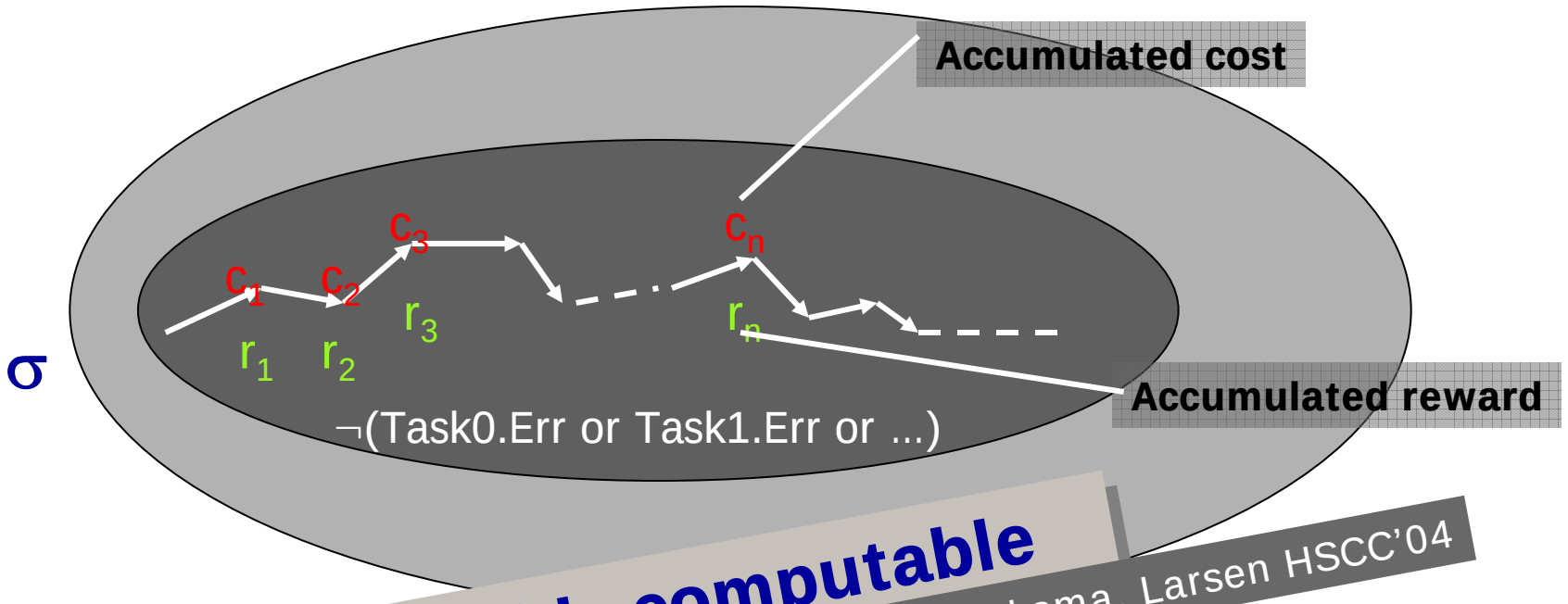
Cost Optimal Scheduling =
Optimal Infinite Path



Value of path σ: $val(\sigma) = \lim_{n\to\infty} c_n/r_n$

Optimal Schedule σ^*: $val(\sigma^*) = \inf_\sigma val(\sigma)$

Cost Optimal Scheduling =
Optimal Infinite Path



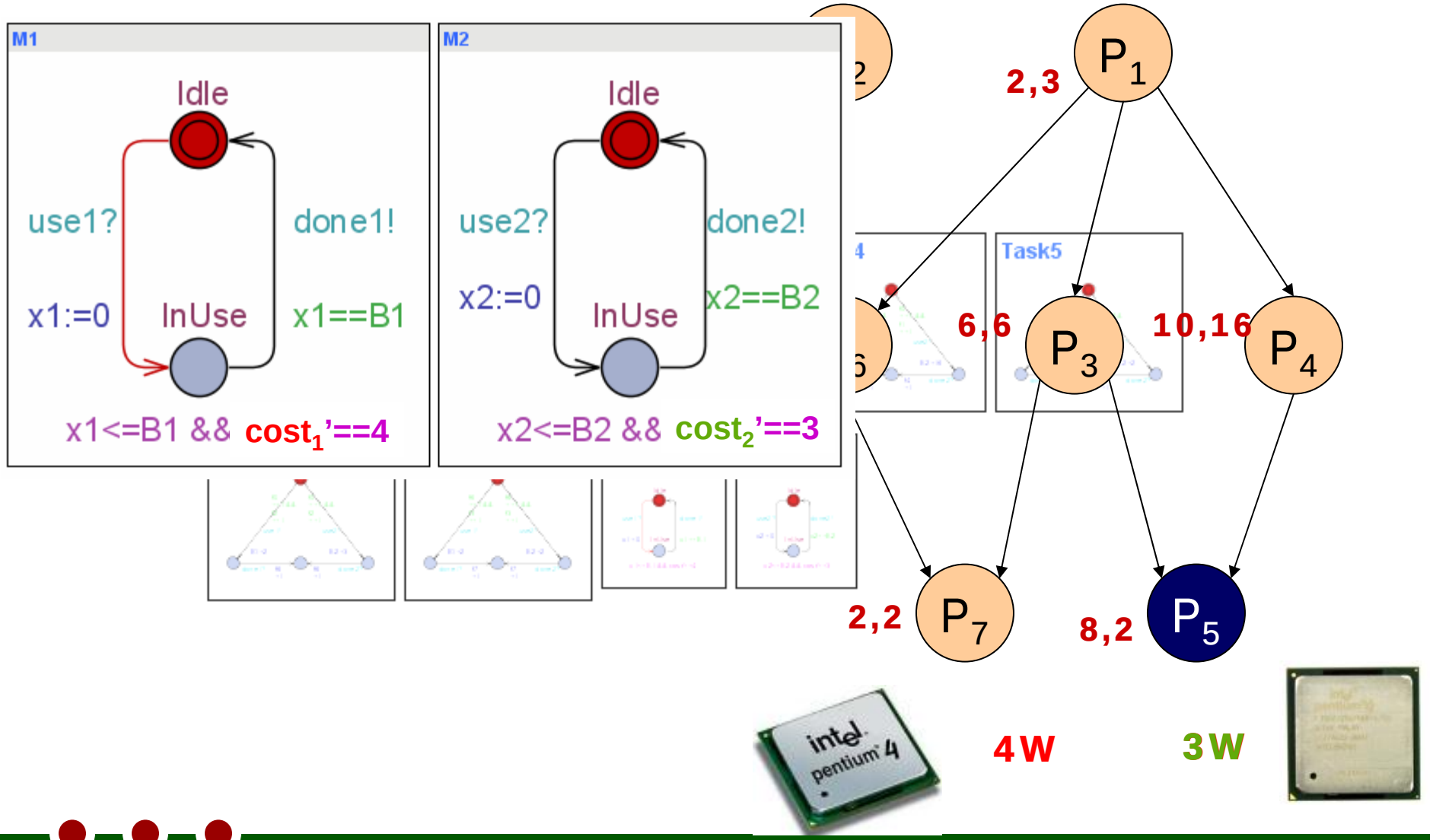
THEOREM: σ^* is computable

Bouyer, Brinksma, Larsen HSCC'04

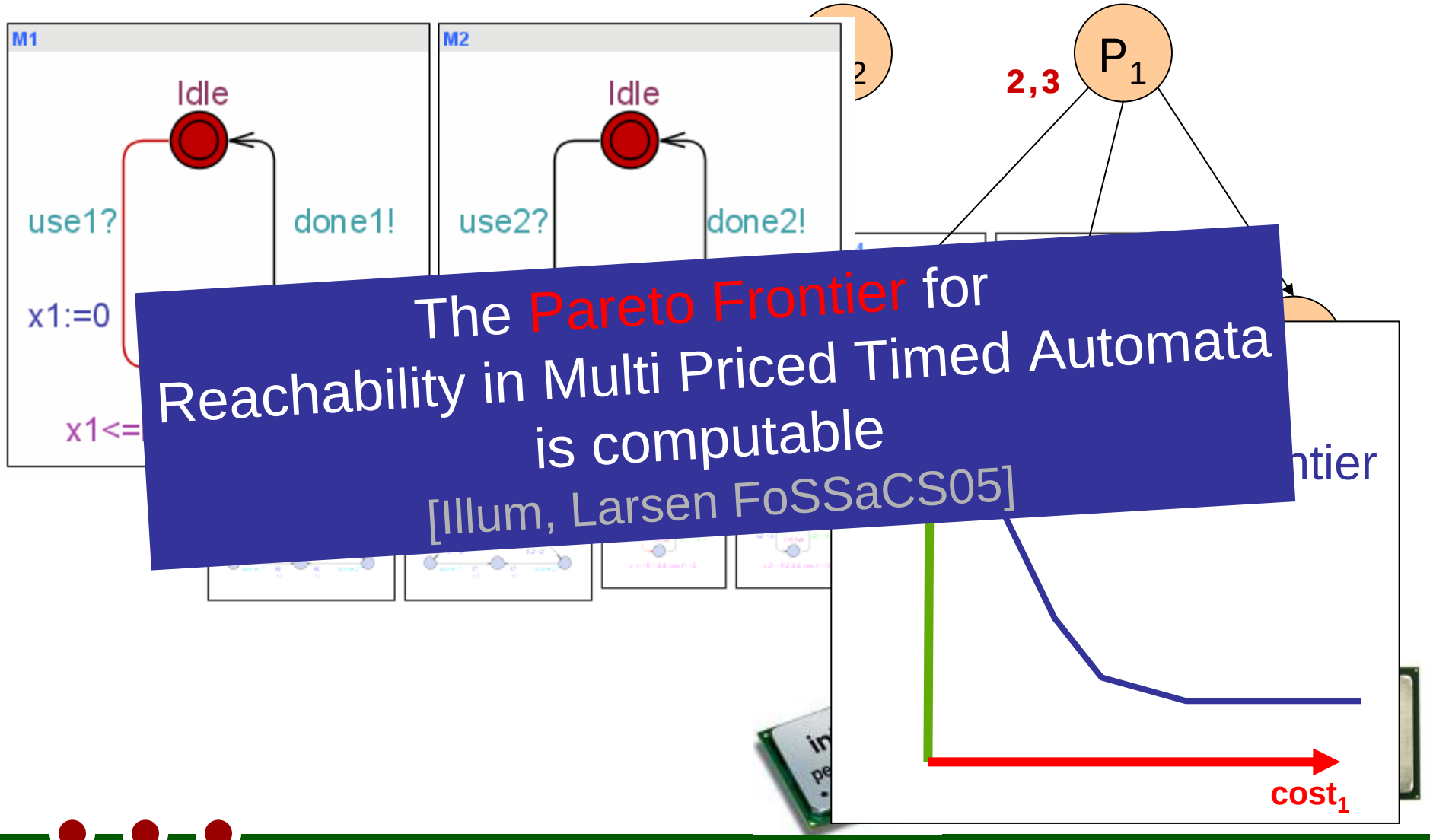
path σ: $val(\sigma) = \lim_{n\to\infty} c_n/r_n$

Optimal Schedule σ^*: $val(\sigma^*) = \inf_\sigma val(\sigma)$

Multiple Objective Scheduling

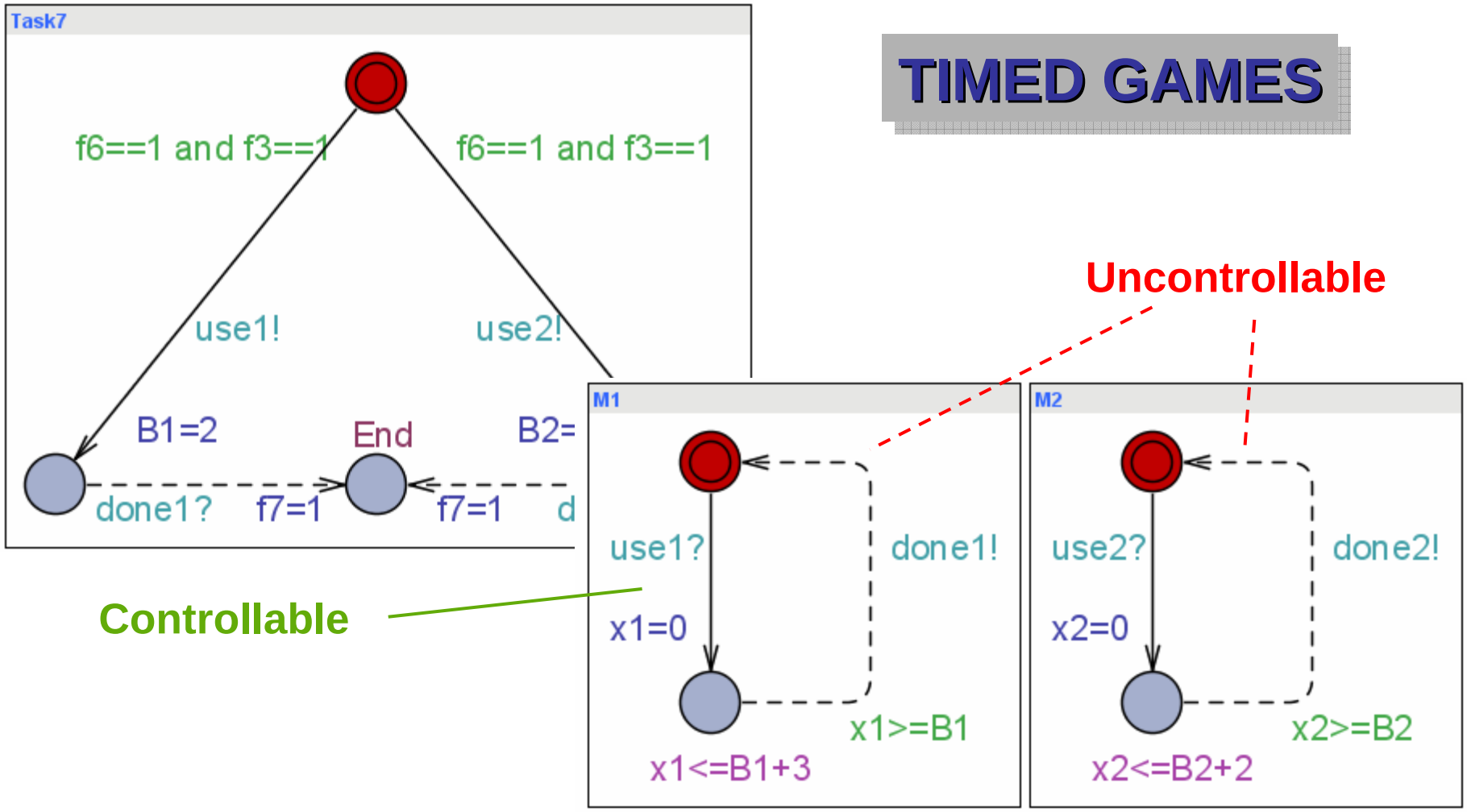


Multiple Objective Scheduling



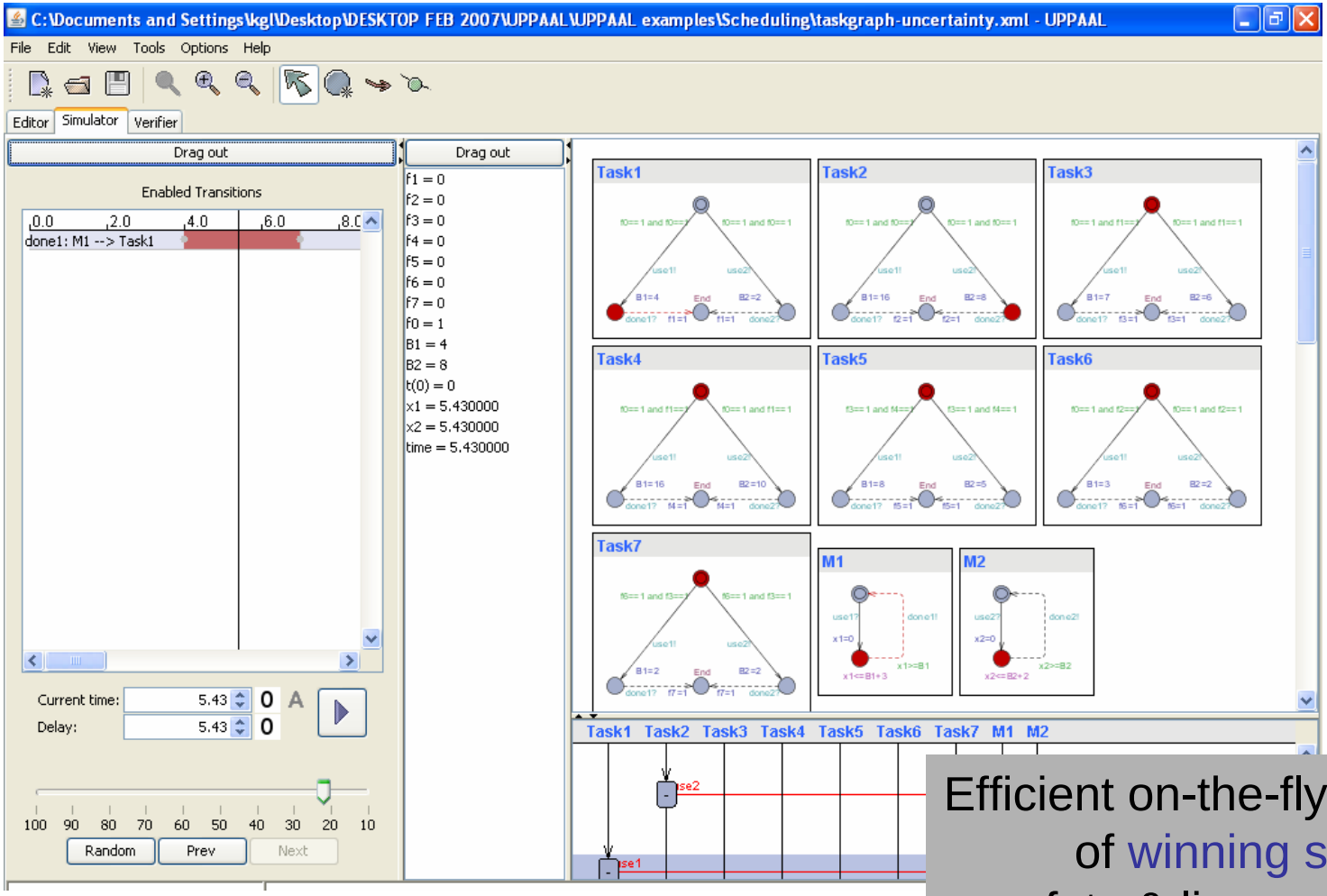
Synthesis =

Scheduling under uncertainty



UPPAAL Tiga

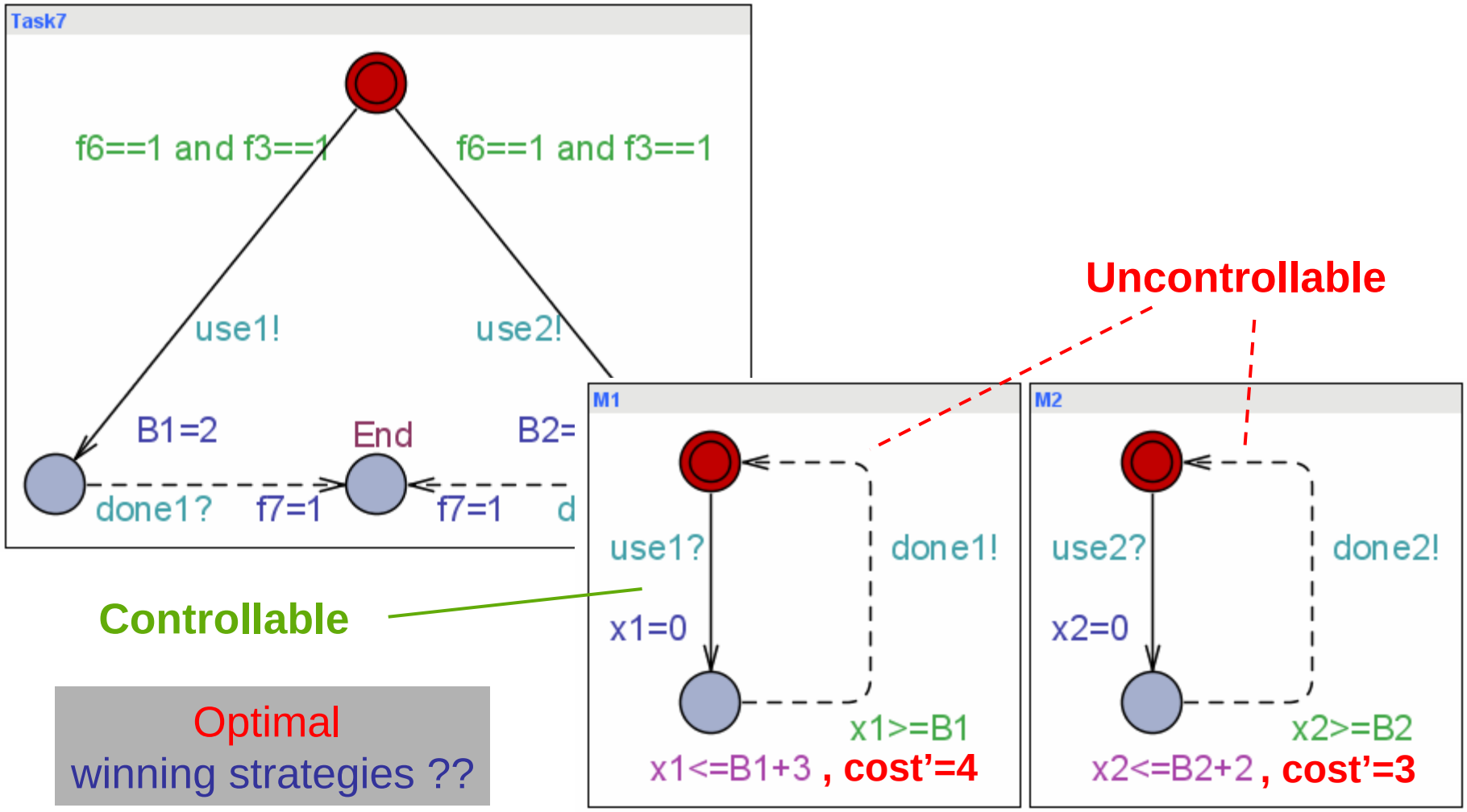
Synthesis of winning strategies for TIMED GAMES



CONCUR05,
CAV07,
FORMATS07

Efficient on-the-fly generation of winning strategies for safety & liveness objectives

Optimal Synthesis =
Priced Timed Games



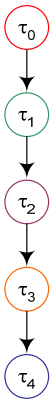
Optimal
winning strategies ??

Priced Timed Games

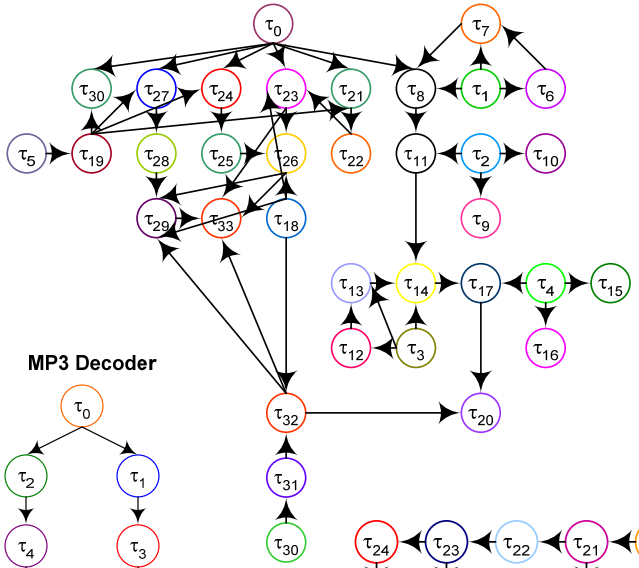
- Price Optimal Control (reachability):
  - Acyclic PTA [LTMM02]
  - Bounded length [ABM04]
  - Strong non-zeno cost-behaviour [BCFL04]
  - Undecidable with 3 clocks or more [BBR05, BBM06]
  - Decidable for PTGs with 1 clock [BLMR06]
- Priced Timed Safety Games
  - Conjectured to be undecidable in general.

Handling realistic applications?

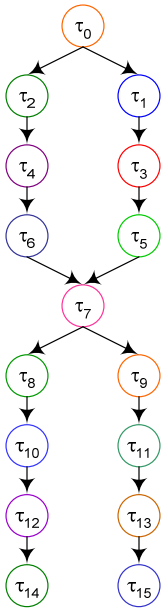
JPEG Encoder



GSM Decoder

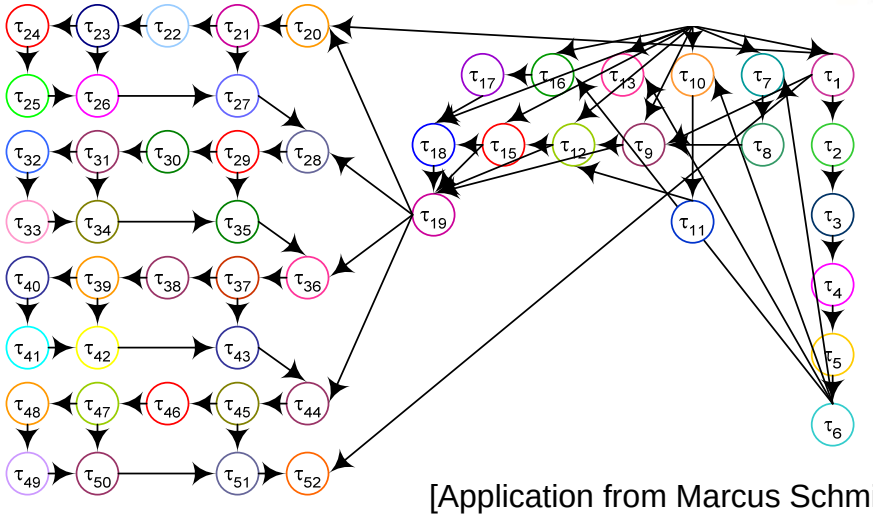


MP3 Decoder



JPEG Decoder

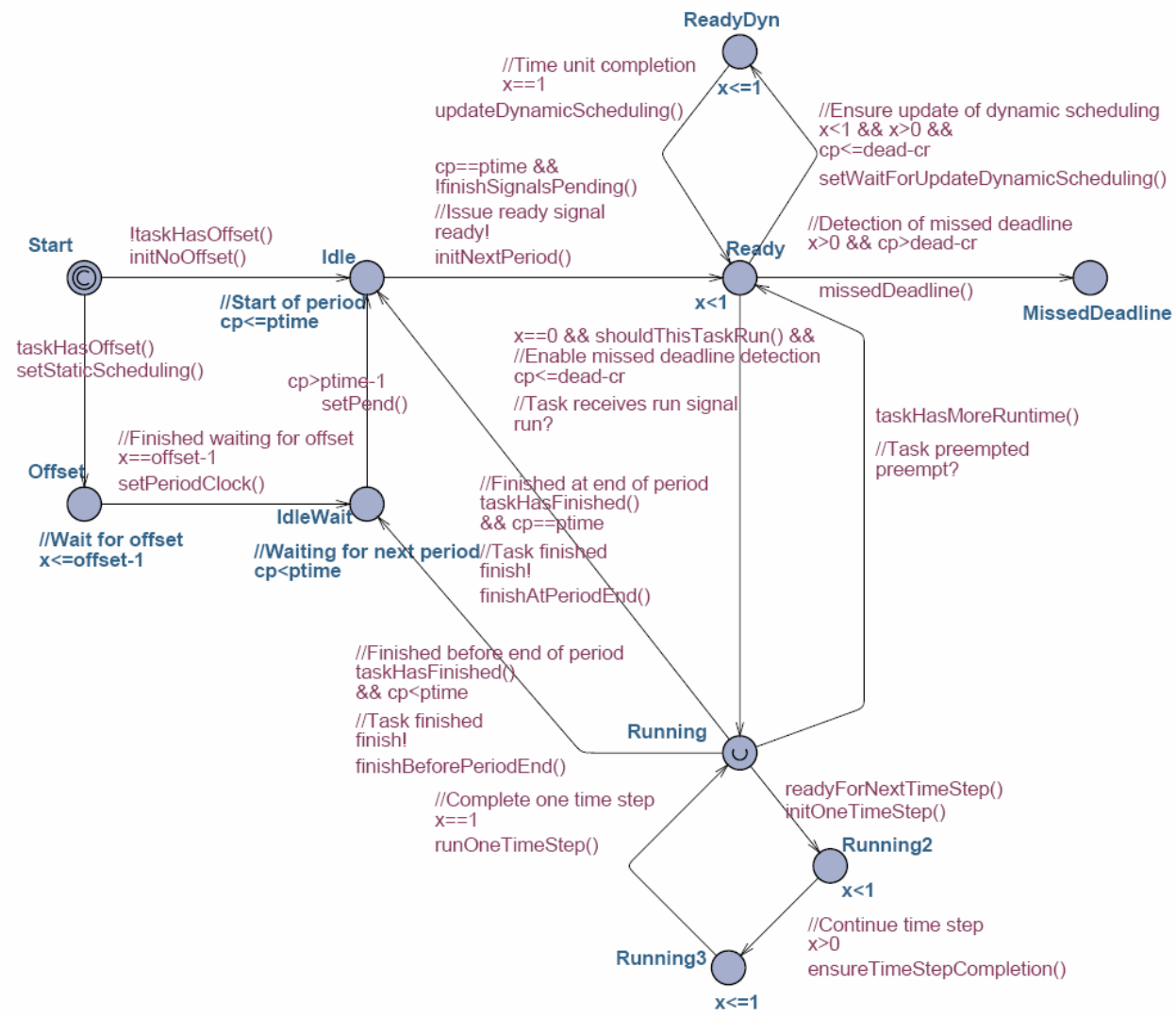
GSM Encoder



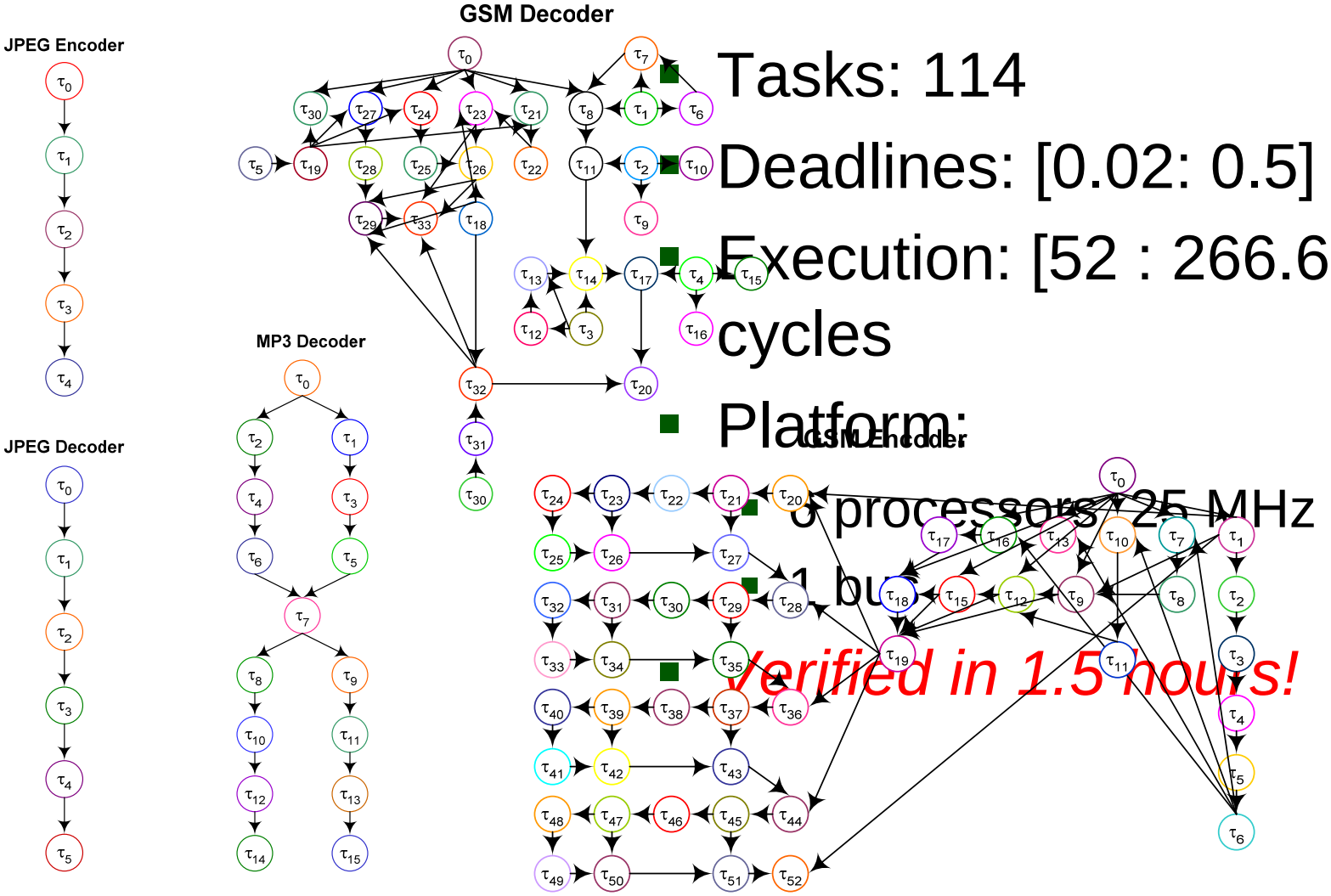
Smart phone:

[Application from Marcus Schmitz, TU Linkoping]

Timed Automata for a task



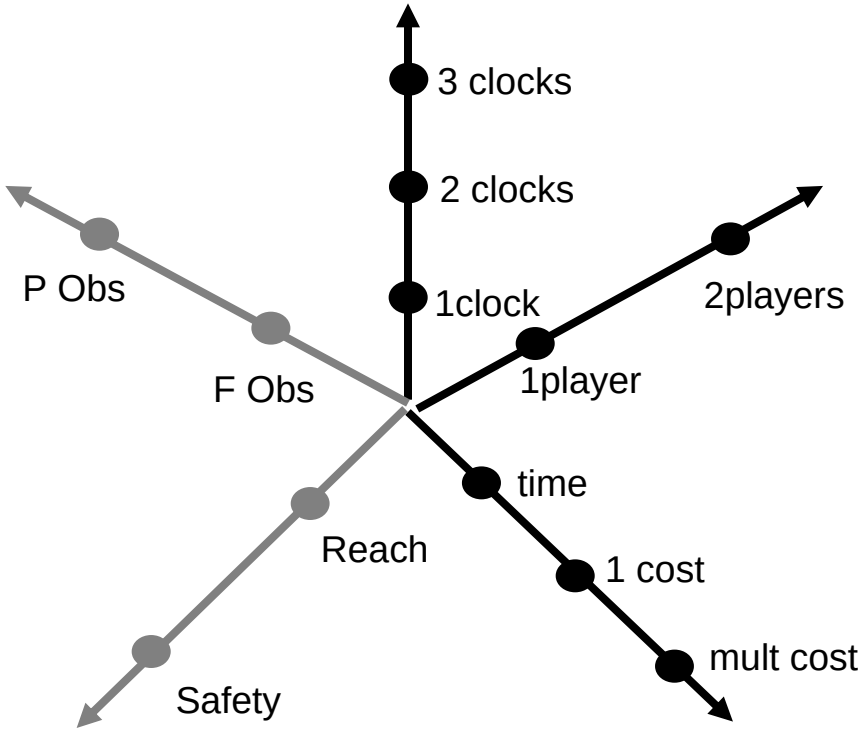
Smart phone



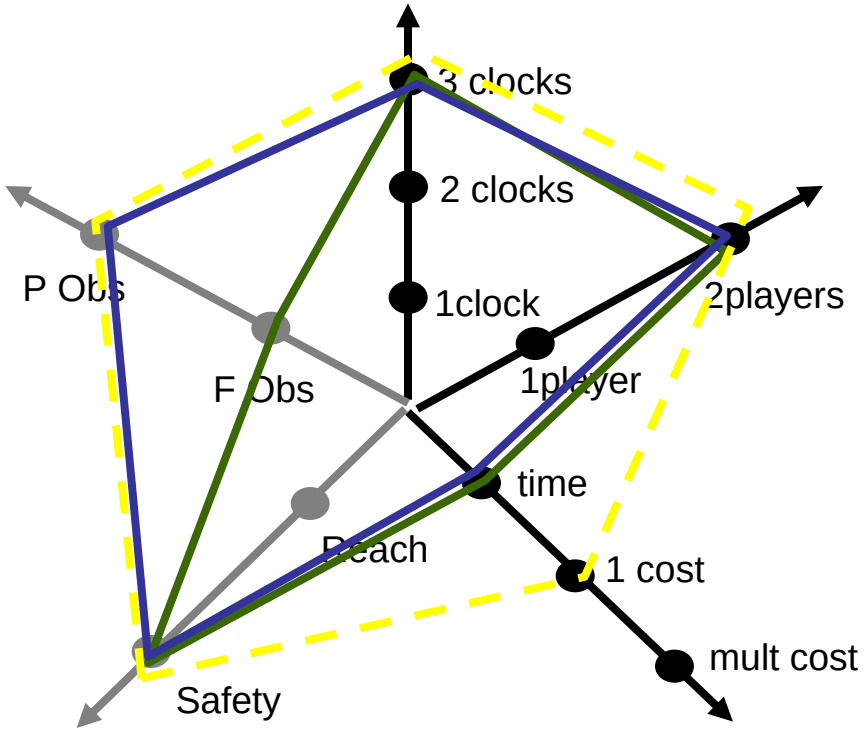
- Tasks: 114
- Deadlines: [0.02: 0.5] sec
- Execution: [52 : 266.687] cycles
- Platform:
  - 6 processors, 25 MHz
  - 1 bu

Verified in 1.5 hours!

Optimal Combinations

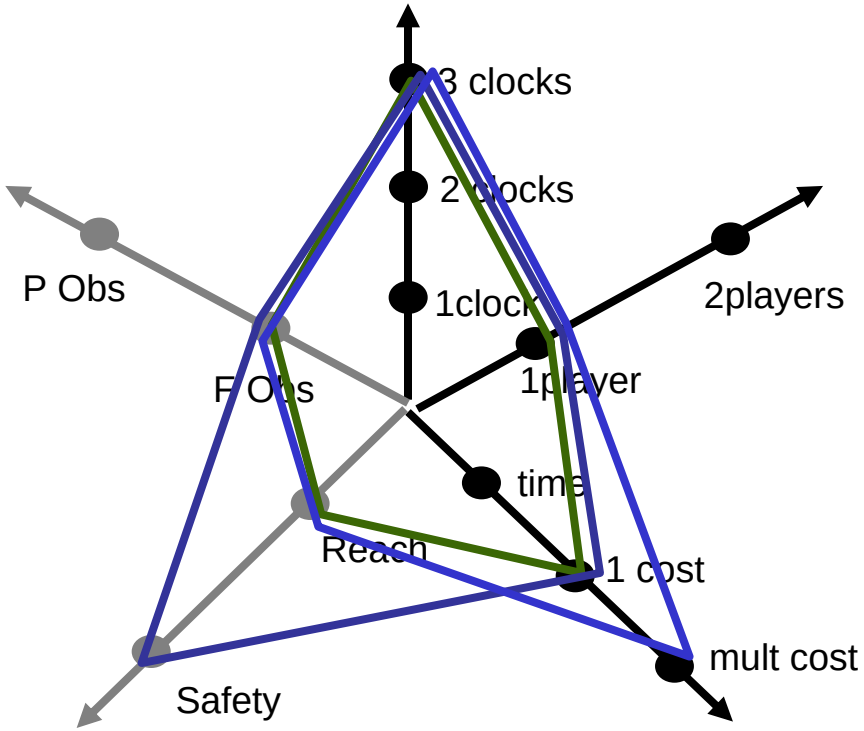


Optimal Combinations



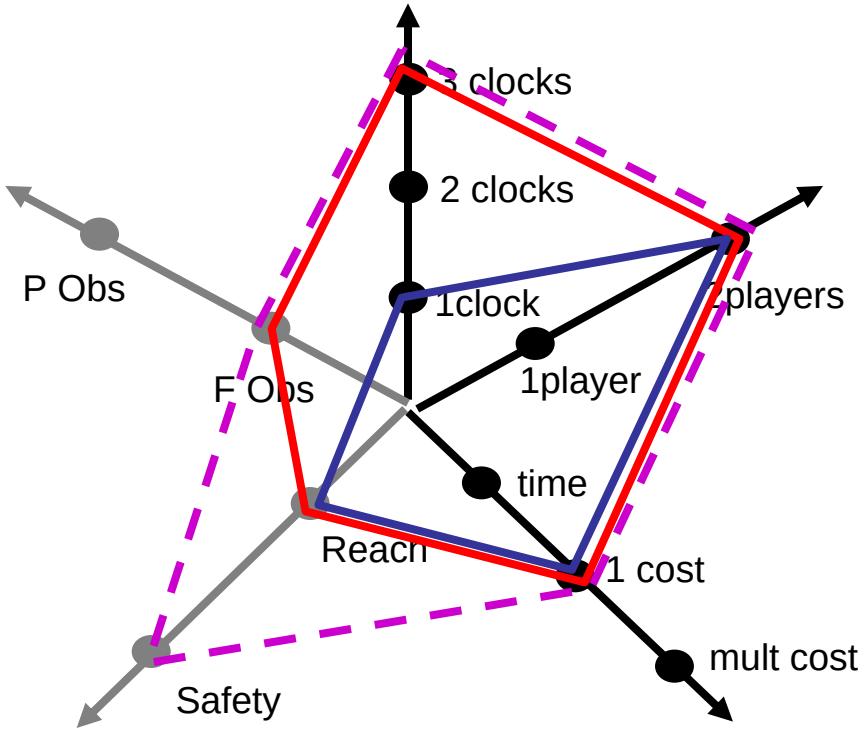
TIGA

Optimal Combinations

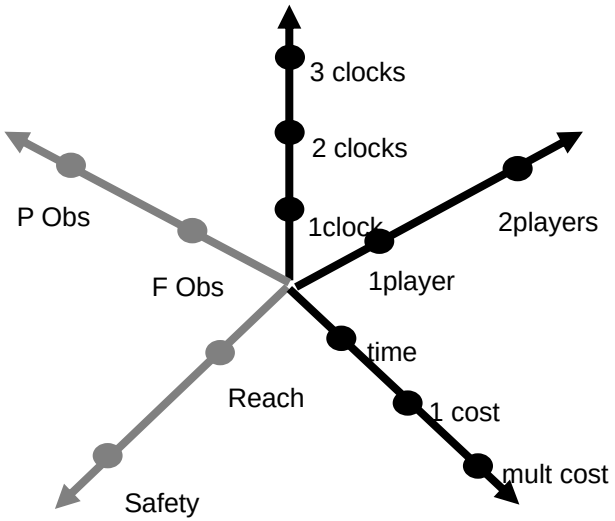


CORA

Optimal Combinations



Conclusion



- Identification of all Pareto optimal combinations!
  - Safety for PTG?
  - Reachability for 2PTG?
  - Safety for MPTA?
  - Safety for 1PTG?
- Efficient realizations
  - TG w PO?
  - Safety for PTA?
  - Reachability for 1PTG?
- Dealing with undecidability.
- $\forall \varepsilon.\ |\text{Th-Prac}| < \varepsilon$

Thanks for your attention!

Please do not hesitate to contact us at:

kgl@cs.aau.dk

or

www.uppaal.com