A study of the AADL mode change protocol

D. Bertrand, A.-M. Déplanche, S. Faucou and O. Roux

Université de Nantes / IRCCyN

Modes in AADL

An AADL system can operate in different modes =

Modes in AADL

An AADL system can operate in different modes =

- each mode is associated with a configuration

Modes in AADL

An AADL system can operate in different modes =

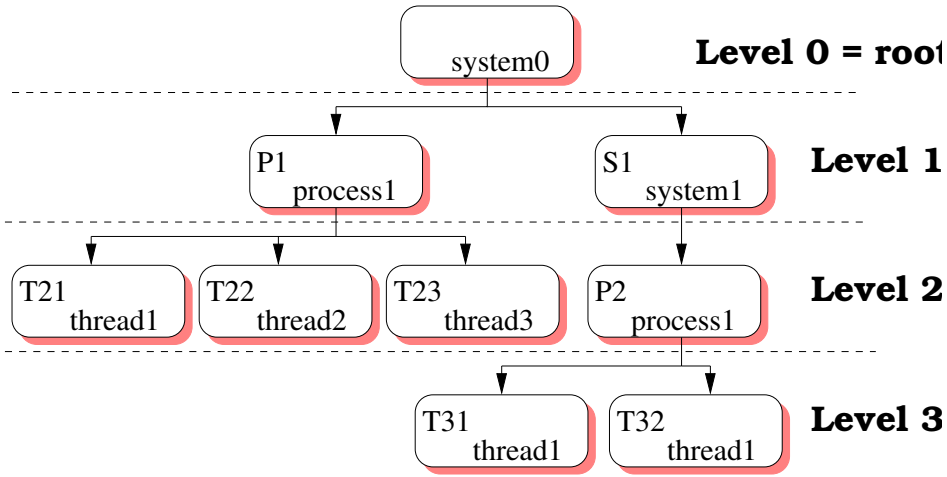
- each mode is associated with a configuration
- events can trigger a change from one system operational mode (SOM) to another

Modes in AADL

An AADL system can operate in different modes =

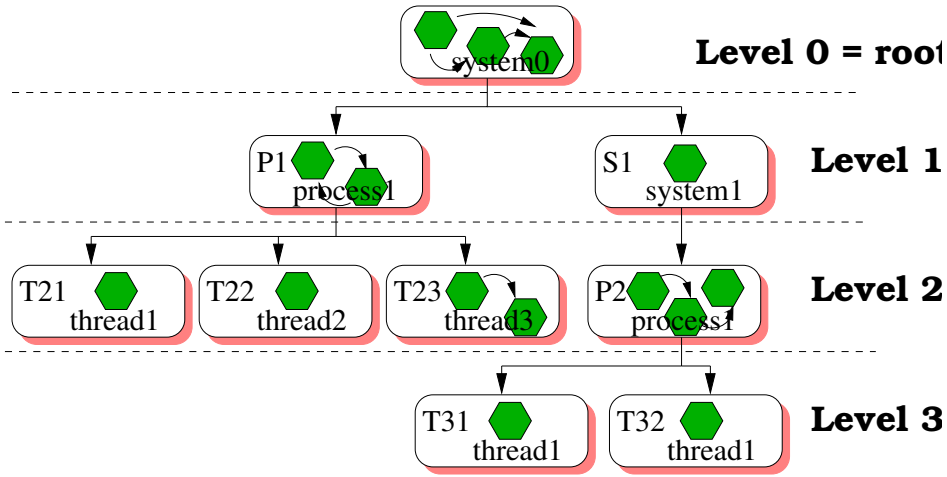
- each mode is associated with a configuration
- events can trigger a change from one system operational mode (SOM) to another
- a SOM is a vector of modes
  - current mode of each active components
  - we note $\perp$ the mode of a component inactive in the current SOM

Example



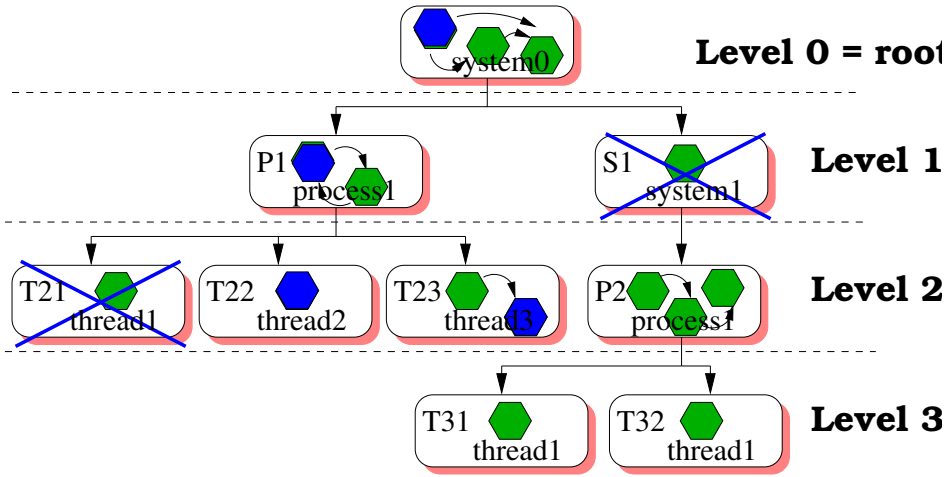
- AADL model = tree of components;

Example



- AADL model = tree of components;
- each component has one or more operating modes;

Example

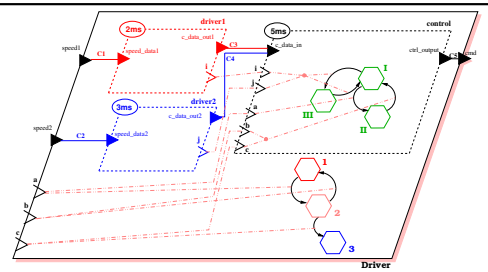


- AADL model = tree of components;
- each component has one or more operating modes;
- each mode is associated with a (synchronised) configuration of subcomponents

Motivations

An AADL model

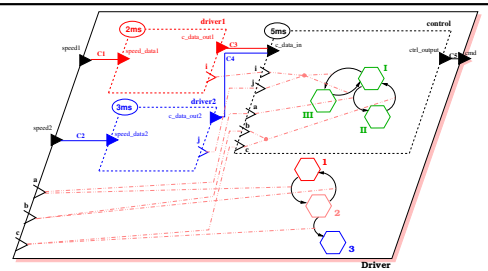
s

$s =$ 

Motivations

An AADL model

s

$s =$ 

A specification

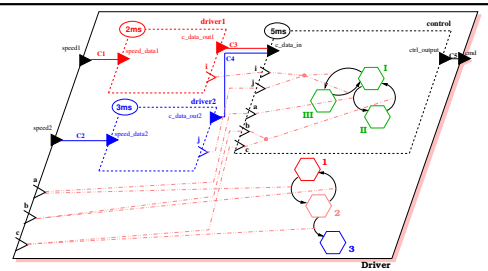
ϕ

Φ = What is the worst-case response time of the system to a SOM change request ?

Motivations

An AADL model

s

$s =$ 

$\models$

A specification

ϕ

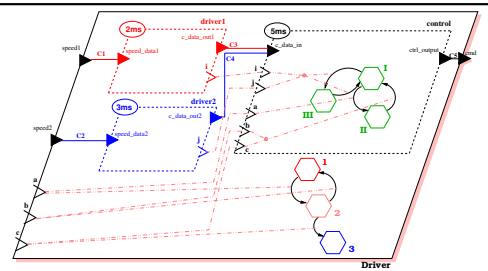
Φ = What is the worst-case response time of the system to a SOM change request ?

?

Motivations

An AADL model

s

$s =$ 

Abstraction

$\models$

A specification

ϕ

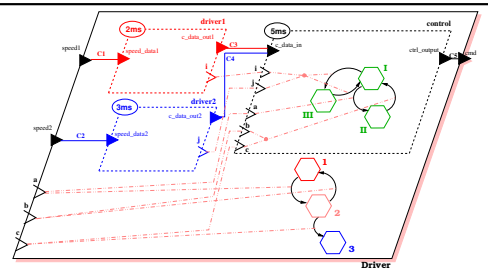
Φ = What is the worst-case response time of the system to a SOM change request ?

?

Motivations

An AADL model

s

$s =$ 

Abstraction

An abstract model

S

Time Petri Nets (TPN)

$\models$

A specification

ϕ

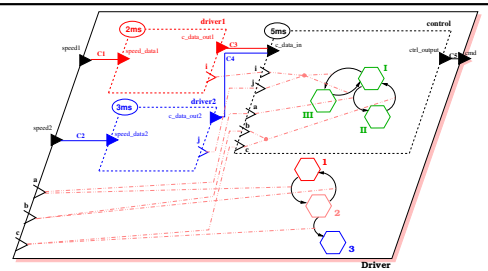
Φ = What is the worst-case response time of the system to a SOM change request ?

?

Motivations

An AADL model

s

$s =$ 

$\models$

A specification

ϕ

Φ = What is the worst-case response time of the system to a SOM change request ?

?

Abstraction

An abstract model

S

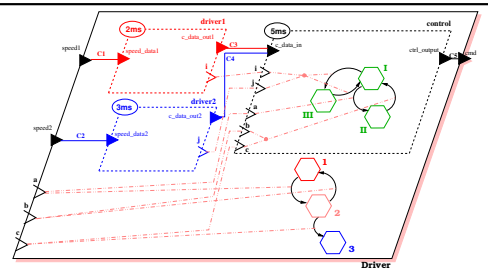
Time Petri Nets (TPN)

Formalisation

Motivations

An AADL model

s

$s =$ 

Abstraction

An abstract model

S

Time Petri Nets (TPN)

$\models$

A specification

ϕ

Φ = What is the worst-case response time of the system to a SOM change request ?

?

Formalisation

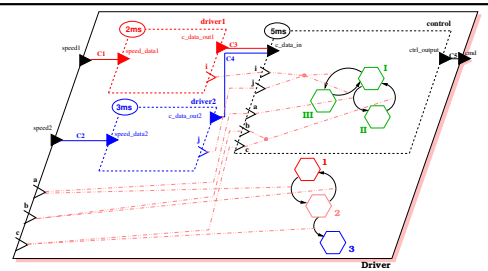
A formal specification Φ

TPN-TCTL logic

Motivations

An AADL model

s

$s =$ 

$\models$

A specification

ϕ

Φ = What is the worst-case response time of the system to a SOM change request ?

?

Abstraction

Formalisation

An abstract model

S

Time Petri Nets (TPN)

$\models$

A formal specification Φ

TPN-TCTL logic

?

Contribution

- Formalisation of the modal behaviour of AADL specifications
  - by putting together the pieces spread over the standard

Contribution

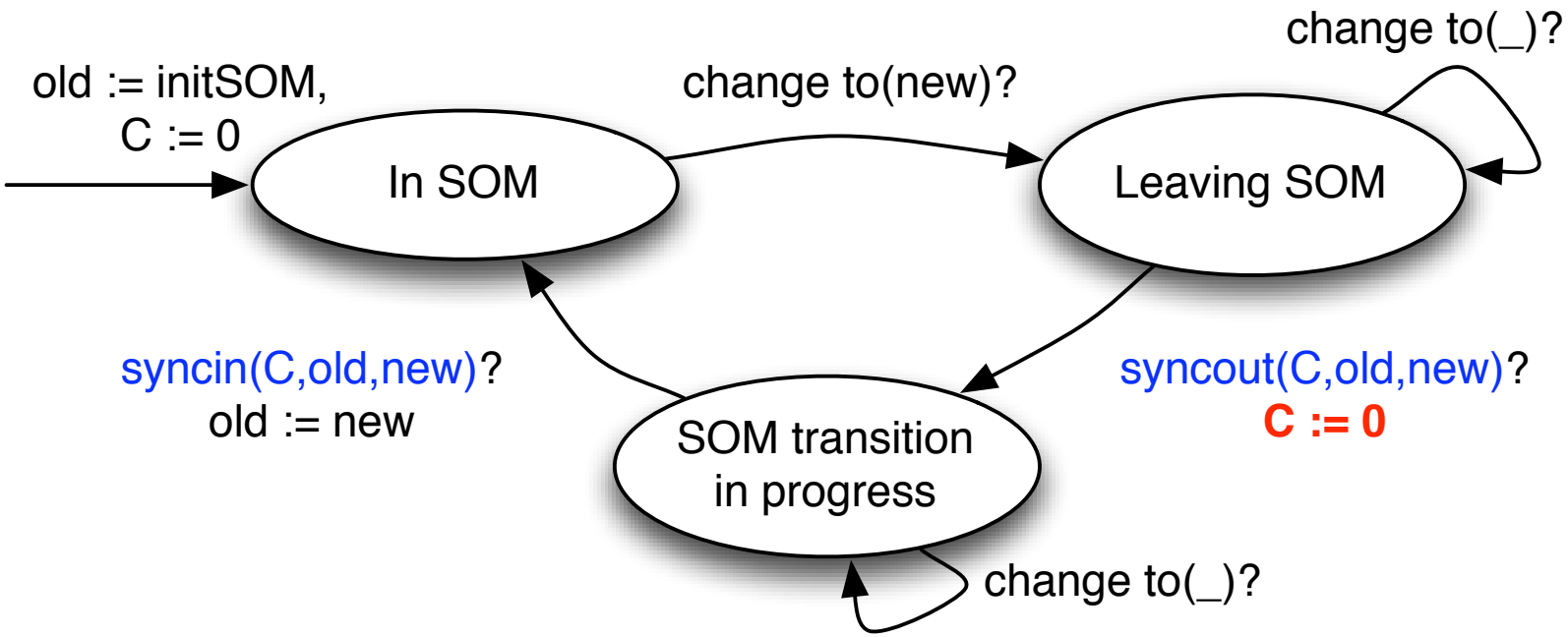
- Formalisation of the modal behaviour of AADL specifications
  - by putting together the pieces spread over the standard
- Translation from a subset of AADL to TPN (timed behaviour of the AADL model abstracted to SOM change)

The mode change protocol

Three steps:

1. The system is in a given SOM
   - a change request arrives
2. The system waits a (absolute) synchronisation point before to take into account this request (out mode synchronisation)
   - the synchronisation point is reached
   - the configuration is changed
3. The system waits a (relative) synchronisation point before to truly enter the new SOM (in mode synchronisation)

Formalisation of the protocol



Focus on the synchronisation points

Synchronisation points are used to ensure the determinism of communication between synchronised threads

- $syncout(C, old, new) \triangleq (C \cong 0 \pmod{h_{out}(old, new)})$
  - $h_{out}(old, new)$: hyperperiod *(least common multiple)* of the threads deactivated and the threads attached to modified or added connections
- $syncin(C, old, new) \triangleq (C = h_{in}(old, new))$
  - $h_{in}(old, new)$: hyperperiod of the threads newly activated and the threads attached to modified or added connections

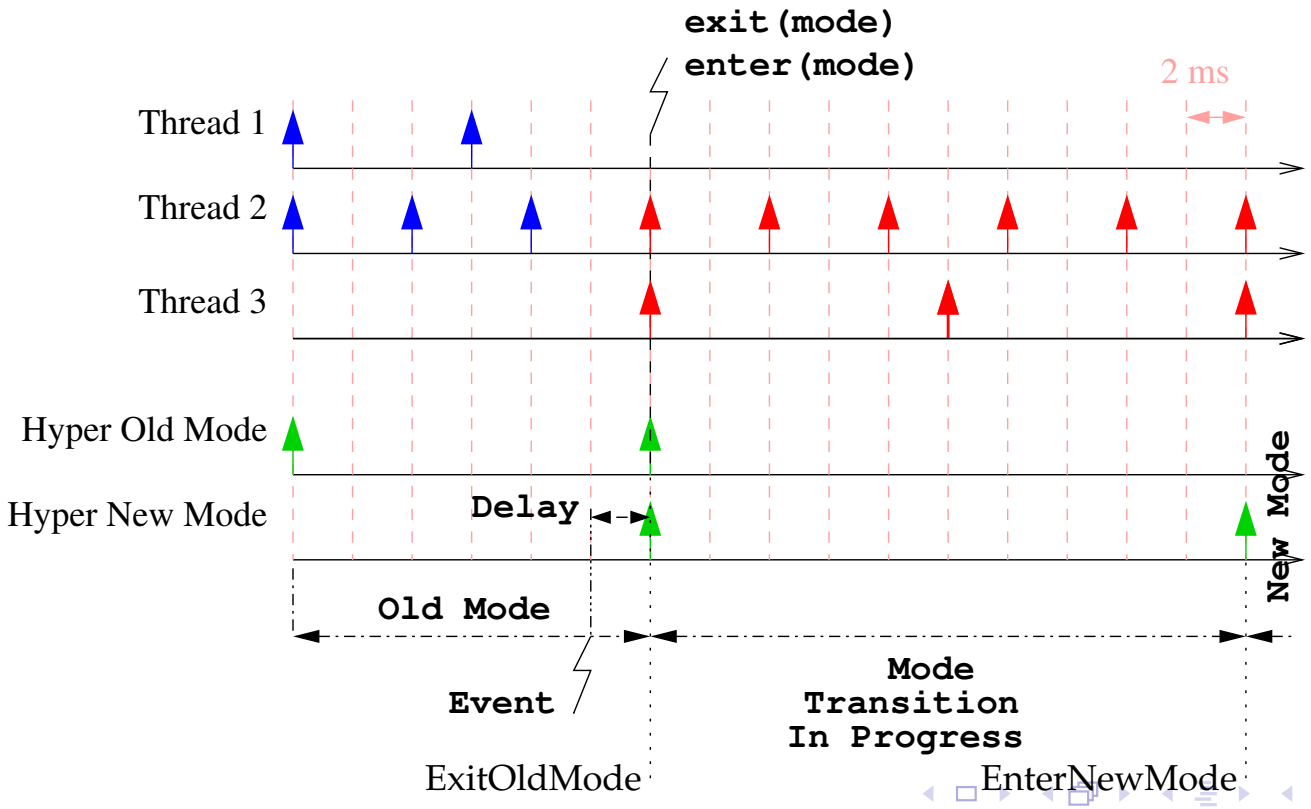
Exemple of SOM change

	active in SOM		
	1	2	3
thread1	x		
thread2	x	x	x
thread3			x

$T_{thread1} = 6ms$, $T_{thread2} = 4ms$, $T_{thread3} = 10ms$

For the mode change from SOM 1 to SOM 2:

- $h_{out}(1,2) = hyperperiod(T_{thread1}, T_{thread2}) = 12ms$
- $h_{in}(1,2) = hyperperiod(T_{thread3}, T_{thread2}) = 20ms$

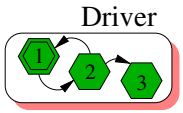


From AADL to TPN

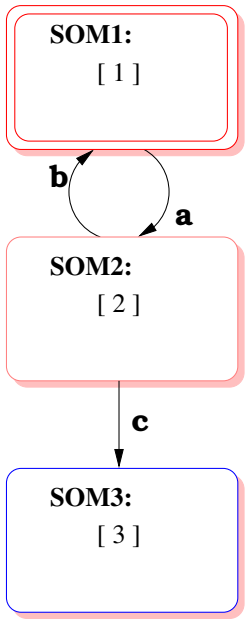
Two steps:

1. Build an untimed SOM state machine by flattening the hierarchical mode state machine
2. Translate this state machine into TPN and add timing informations to model synchronisations

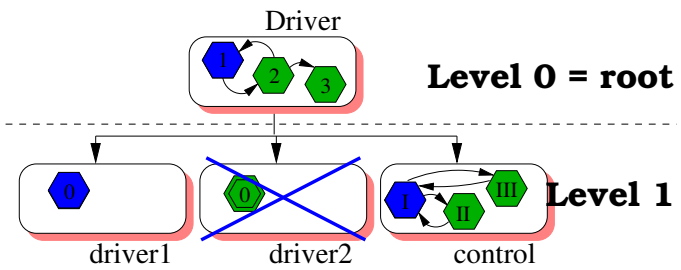
Flattening of the hierarchical mode state machine

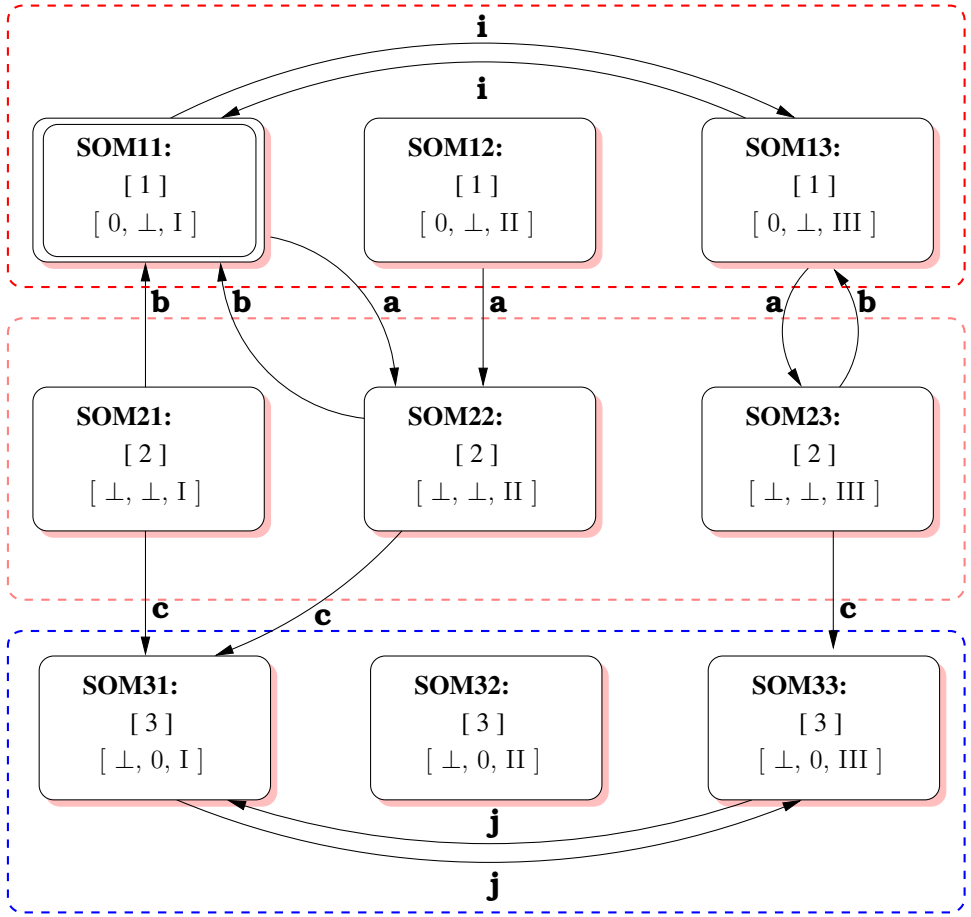


Level 0 = root

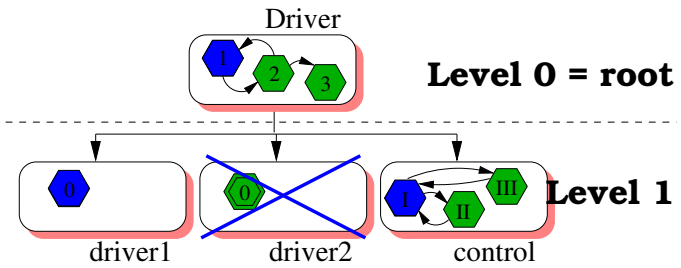


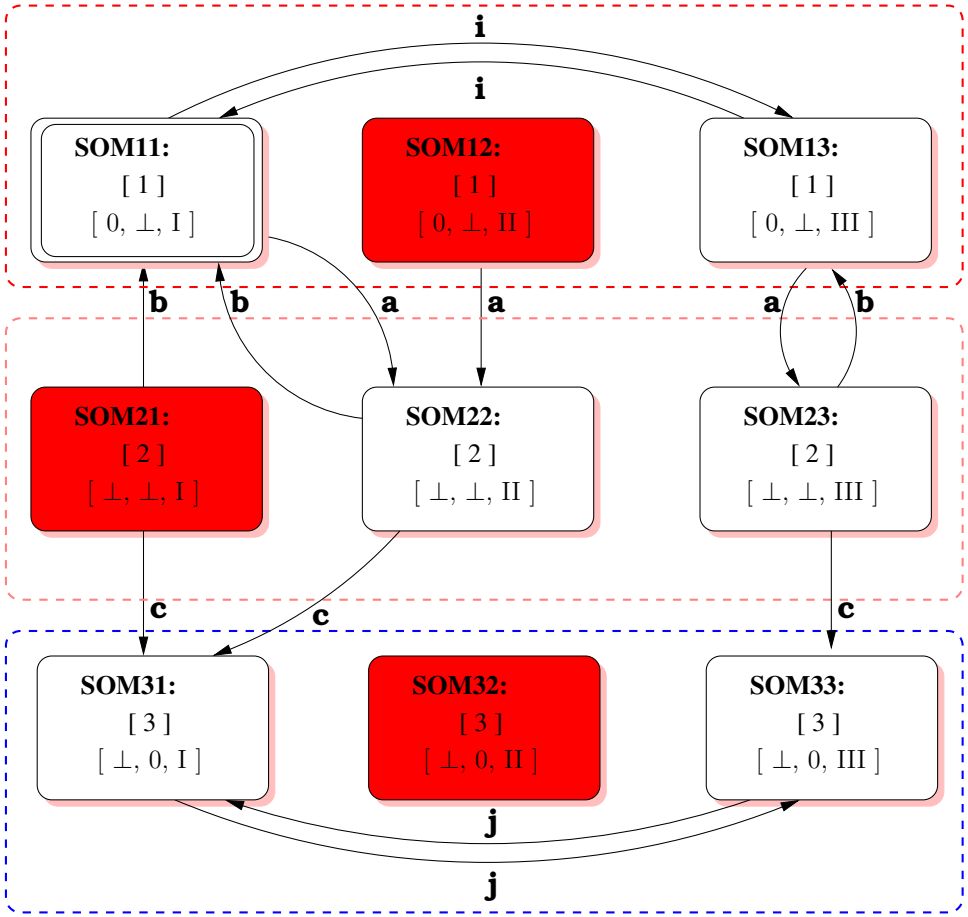
Flattening of the hierarchical mode state machine



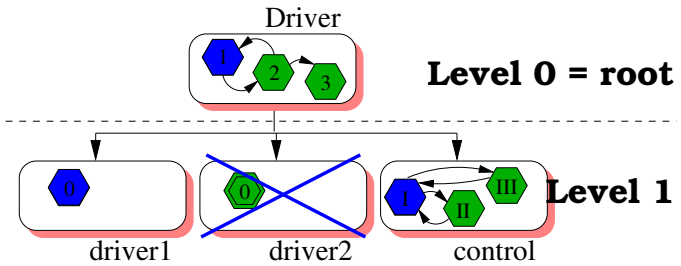


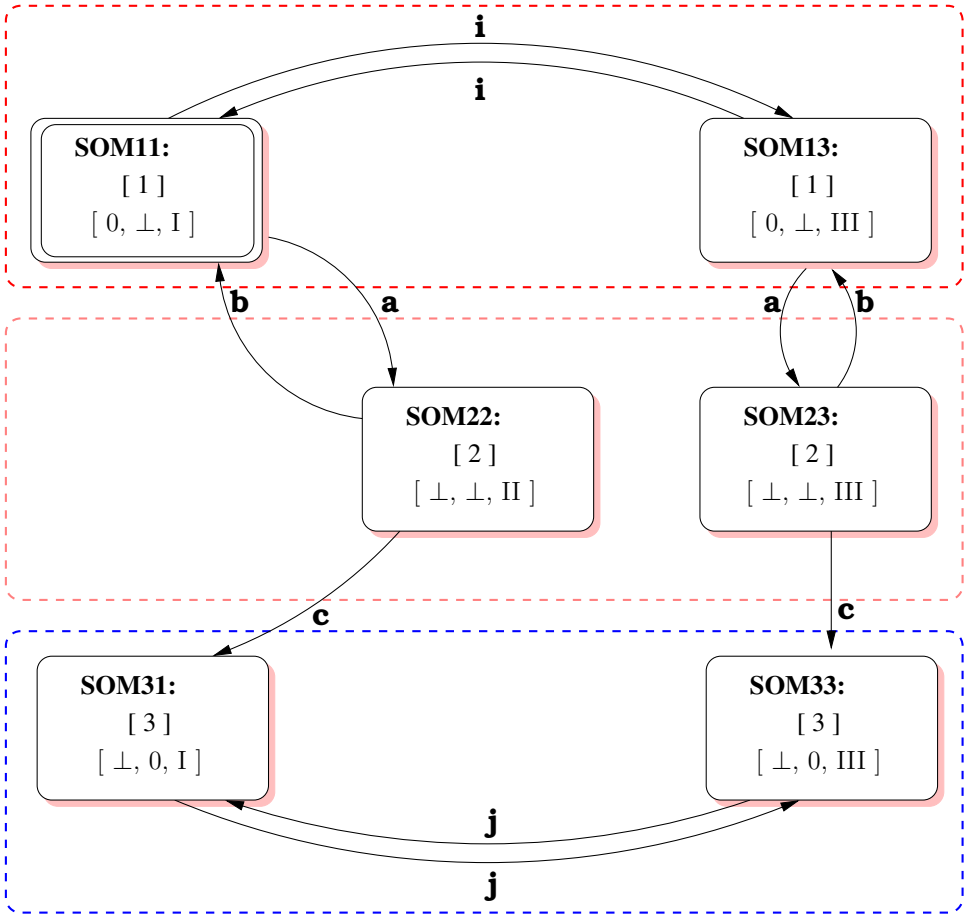
Flattening of the hierarchical mode state machine



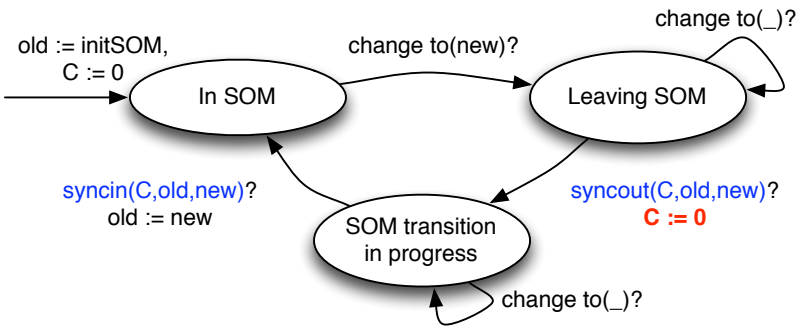


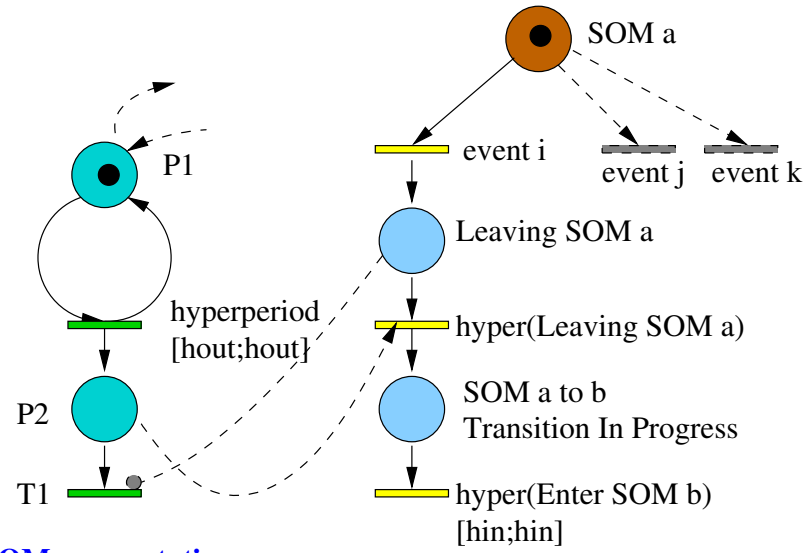
Flattening of the hierarchical mode state machine





Adding time to the model

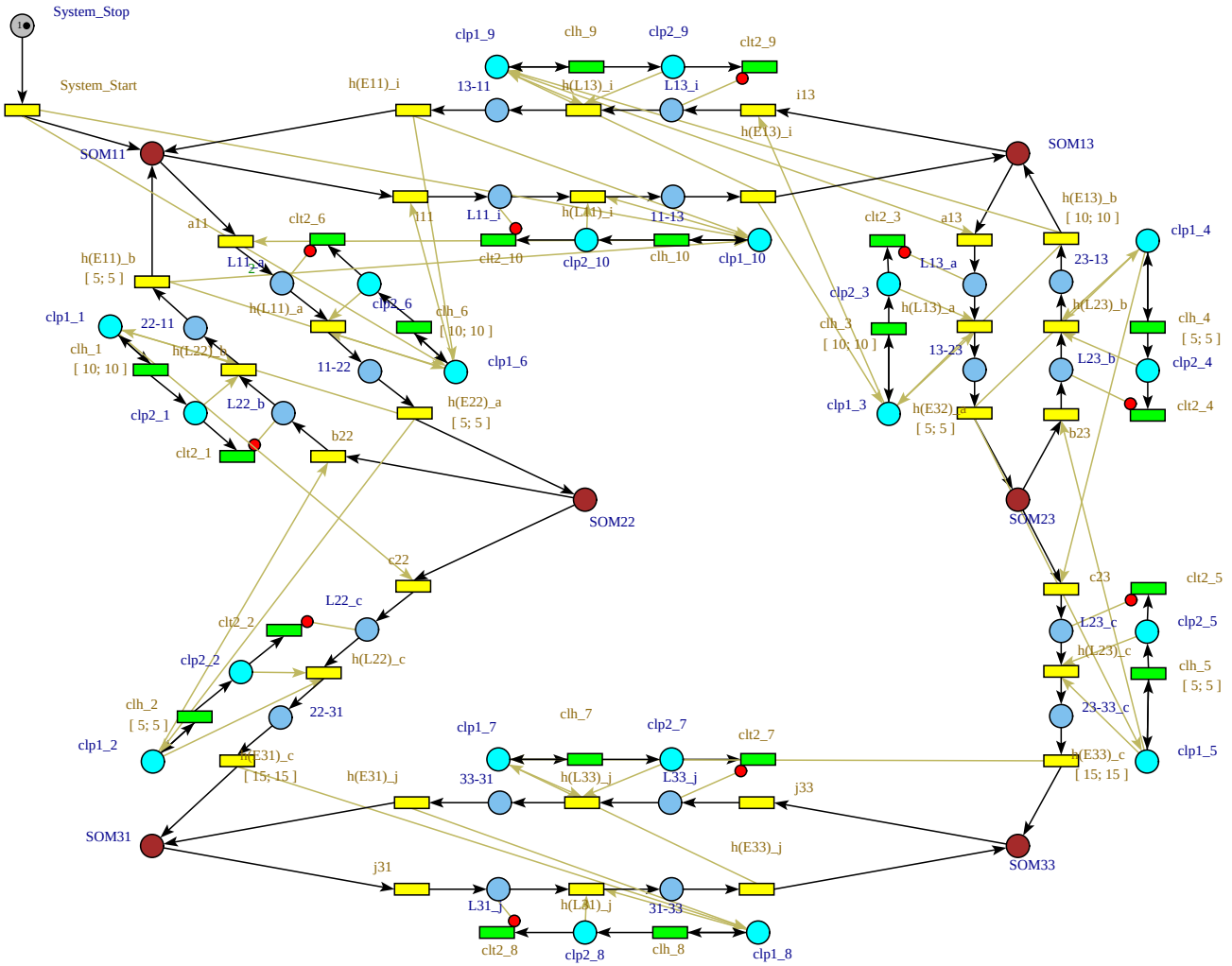




SOM commutation synchronisation pattern

SOM transition pattern

Adding time to the model



Formalisation of the specification

Two examples of formalisation using TPN-TCTL

- **All the SOM i are reachable from initial state**
  $(M(P_{SOM_{11}}) = 1) \rightsquigarrow_{[0,inf[} (M(P_{SOM_i}) = 1)$

Formalisation of the specification

Two examples of formalisation using TPN-TCTL

- **All the SOM i are reachable from initial state** $(M(P_{SOM_{11}}) = 1) \rightsquigarrow_{[0,inf[} (M(P_{SOM_i}) = 1)$

- **Verification of the worst-case response time of the system to a SOM change request ($SOM_{11} \xrightarrow{a} SOM_{22}$):** $(M(P_{event_a}) = 1) \rightsquigarrow_{[0,max]} (M(P_{SOM_{22}}) = 1)$ (where max is the specified bound);

- Formalisation of the modal behaviour of AADL specifications
  - unambiguous intepretation formulated.

- Formalisation of the modal behaviour of AADL specifications
  - unambiguous intepretation formulated.
- Translation from a subset of AADL to TPN (timed behaviour of the AADL model abstracted to mode switches)
  - flattening of the AADL description;
  - adding time to the model by using patterns.

- Formalisation of the modal behaviour of AADL specifications
  - unambiguous intepretation formulated.
- Translation from a subset of AADL to TPN (timed behaviour of the AADL model abstracted to mode switches)
  - flattening of the AADL description;
  - adding time to the model by using patterns.
- Possibility of simulation and formal verification using the tool Romeo

Work-in-progress

- Development of a prototype (AADL to Romeo)
- Proof of the algorithm AADL $\rightarrow$ TPN
- Closing the TPN with a model of the environment

Work-in-progress

- Development of a prototype (AADL to Romeo)
- Proof of the algorithm AADL $\rightarrow$ TPN
- Closing the TPN with a model of the environment

Other concerns

- Obstacles concerning the exploitation of the TPN (in an industrial design process): scalability + traceability

Thank you for your attention

Any question ?