26th DX
International Workshop on Principles of Diagnosis
Paris
August 31 – Sept 3, 2015

HYDIAG

Extended Diagnosis and Prognosis for Hybrid Systems

Elodie Chanthery*, **Yannick Pencolé**, Pauline Ribot**, Louise Travé-Massuyès

CNRS-LAAS Toulouse, France

University of Toulouse, *INSA, **UPS Toulouse, France

{elodie.chanthery,yannick.pencole,pauline.ribot,louise}@laas.fr

Introduction

Context: complex dynamic systems

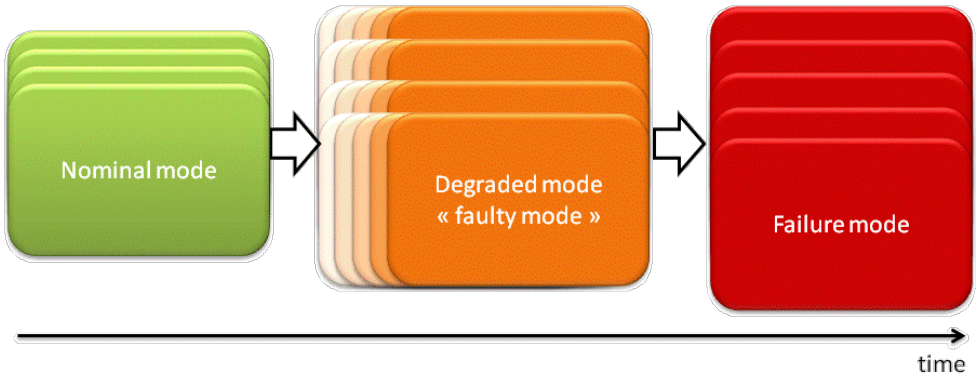
Hybrid Continuous and discrete behaviors

Exposed to failures Maintenance and repair are a part of the total cost of final product

Efficient diagnosis and prognosis techniques to detect, isolate and anticipate faults

Diagnosis

Determines the current health state of the system



Prognosis

More often related to the temporal prediction of the date at which the system is not operationnal anymore and must be repaired

→ RUL prediction

HYDIAG : software developed in Matlab to simulate, diagnose and prognose hybrid systems using model-based techniques

http://projects.laas.fr/hydiag/

Hybrid system modeling for diagnosis

- **Hybrid automaton** [Henzinger 96]

Set of events (discrete control inputs, autonomous mode changes, fault occurrences): observable or unobservable

Set of continuous variables

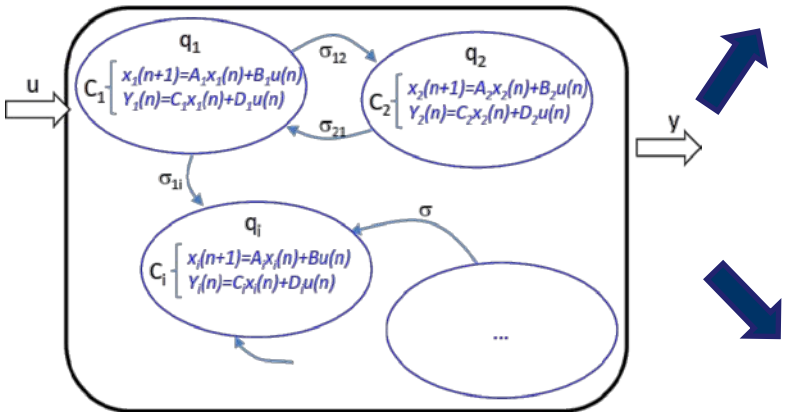
Initial condition

$$S = (\zeta, Q, \Sigma, T, C, (q_0, \zeta_0))$$

Set of the discrete system states

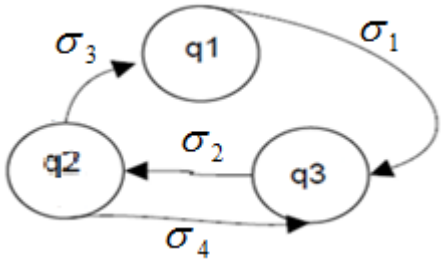
Set of transitions

Set of system constraints linking continuous variables



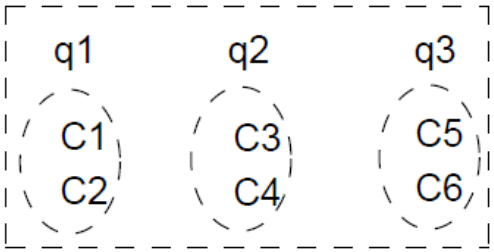
Underlying discrete event system

$$M = (Q, \Sigma, T, q_0)$$



Underlying multimode system

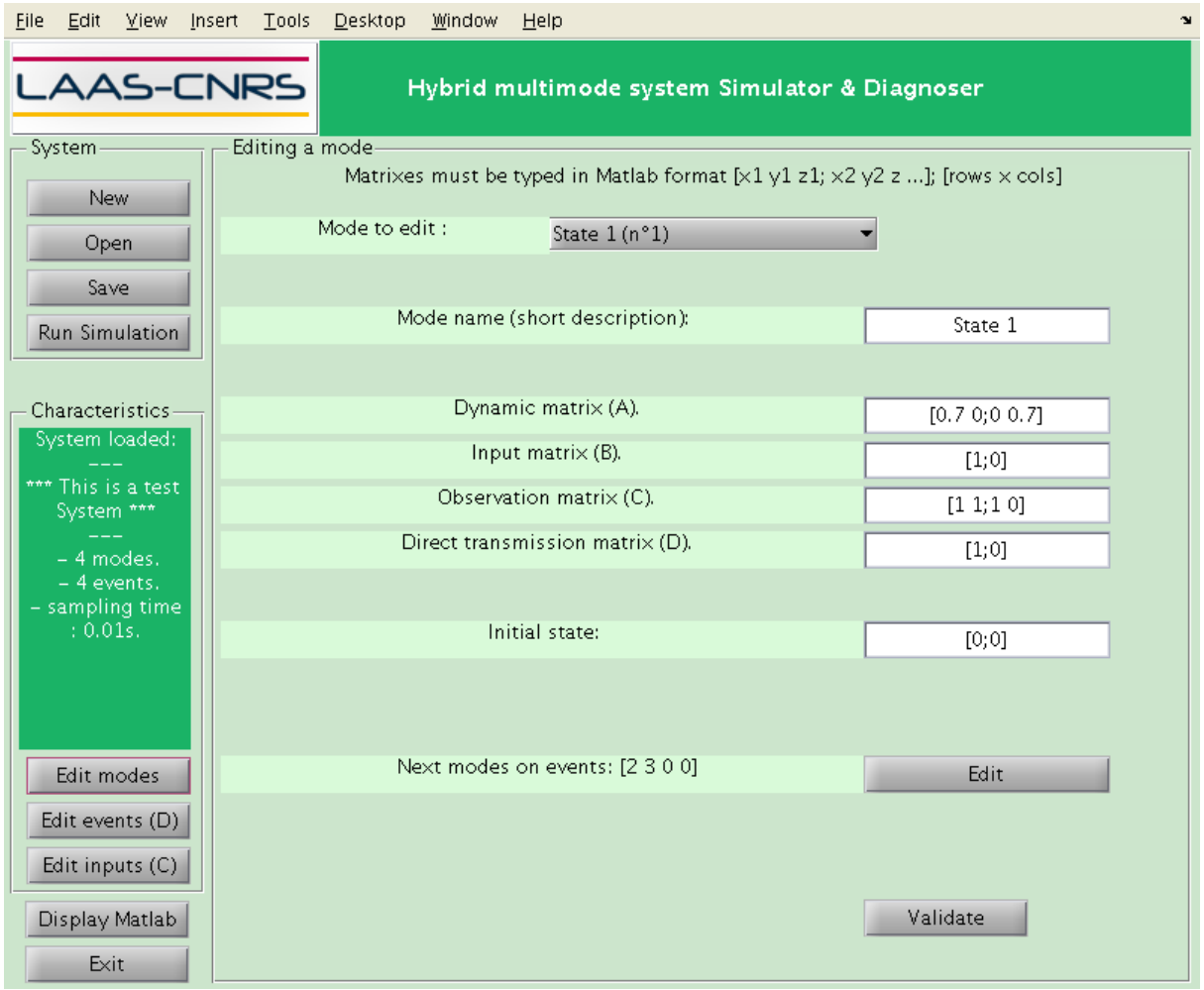
$$\Xi = (\zeta, Q, C, \zeta_0)$$



→ Knowledge required to achieve model-based diagnosis

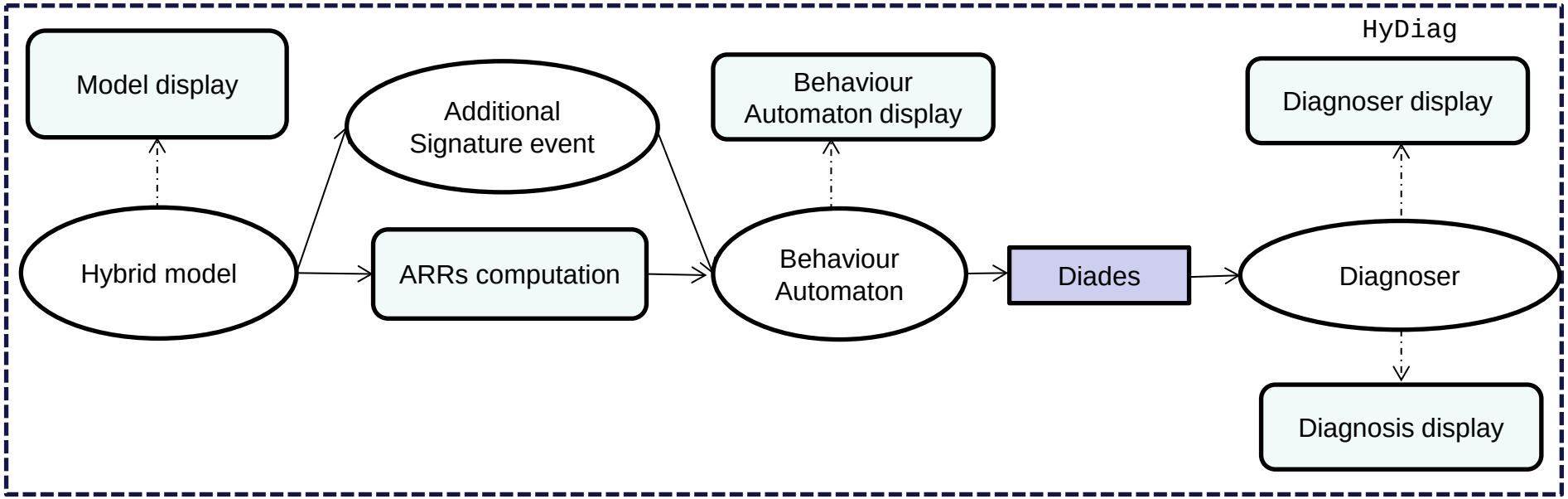
Overview of the native HYDIAG diagnoser (1)

- **Hybrid model edition**



Overview of the native HYDIAG diagnoser (2)

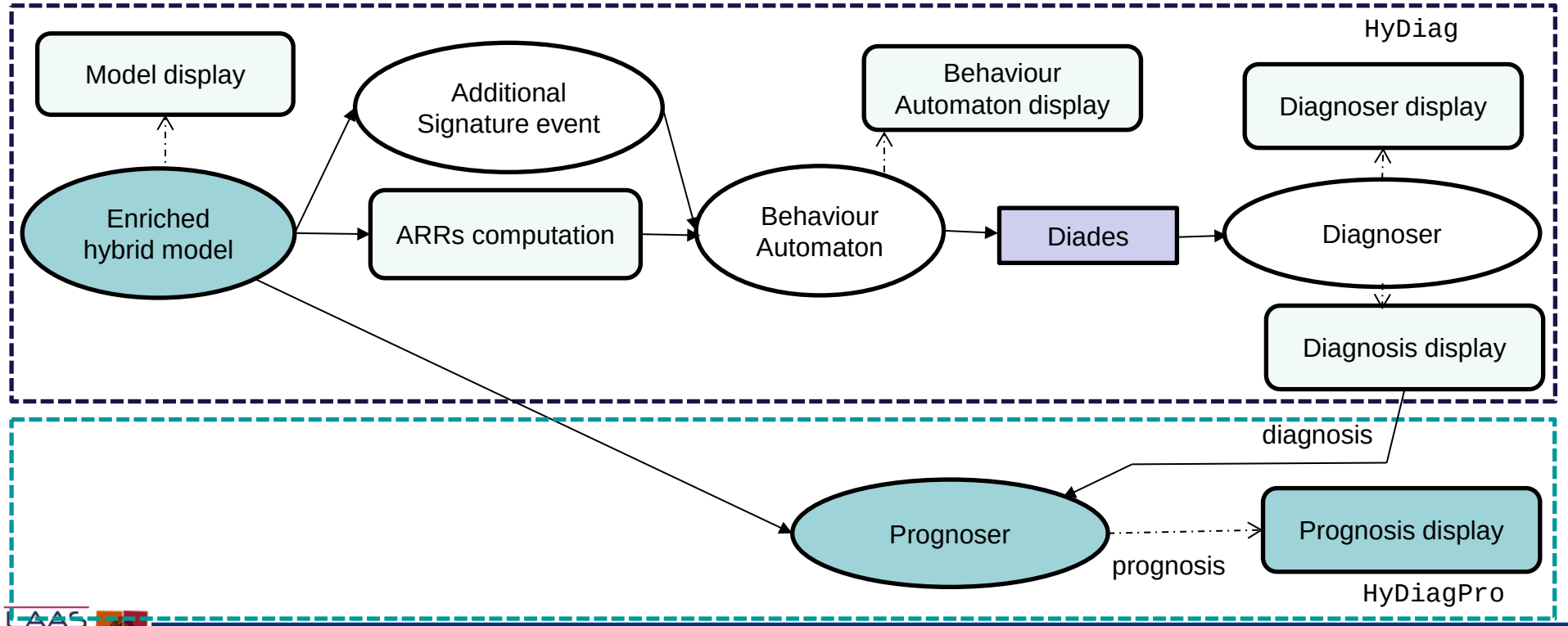
- **HyDiag software architecture**



- **Building the diagnoser** [Bayoudh 2008]
  - Diagnose the continuous part using the parity space method
  - Abstract the continuous part and enrich the discrete part of the model with new observable events
  - Estimate the system mode with the diagnoser method on the resulting DES [Sampath 1996] (DiaDes software, http://homepages.laas.fr/ypencole.DiaDes/)

HyDiagPro : an extension for prognosis (1)

- **Implementation of a prognosis function**
  - Compute the fault probability of the system in each behavioural mode
  - Determine the most likely sequence of dated faulty modes leading to the failure
  - Compute the system remaining useful life
- **HyDiagPro software architecture**



HyDiagPro : an extension for prognosis (2)

- **Enriched hybrid model : associate a set of aging laws to system modes**

$$\mathcal{F} = \left\{ F^{q_i}, i \in \{1,\ldots,card(Q)\} \right\} \Rightarrow \boxed{S^+ = (\zeta, Q, \Sigma, T, C, \mathcal{F}, (q_0, \zeta_0))}$$

  - An aging law based on the parametrized Weibull model gives at any time the probability that the fault f_j occurs from a mode q_i [Ribot et al. 2011]

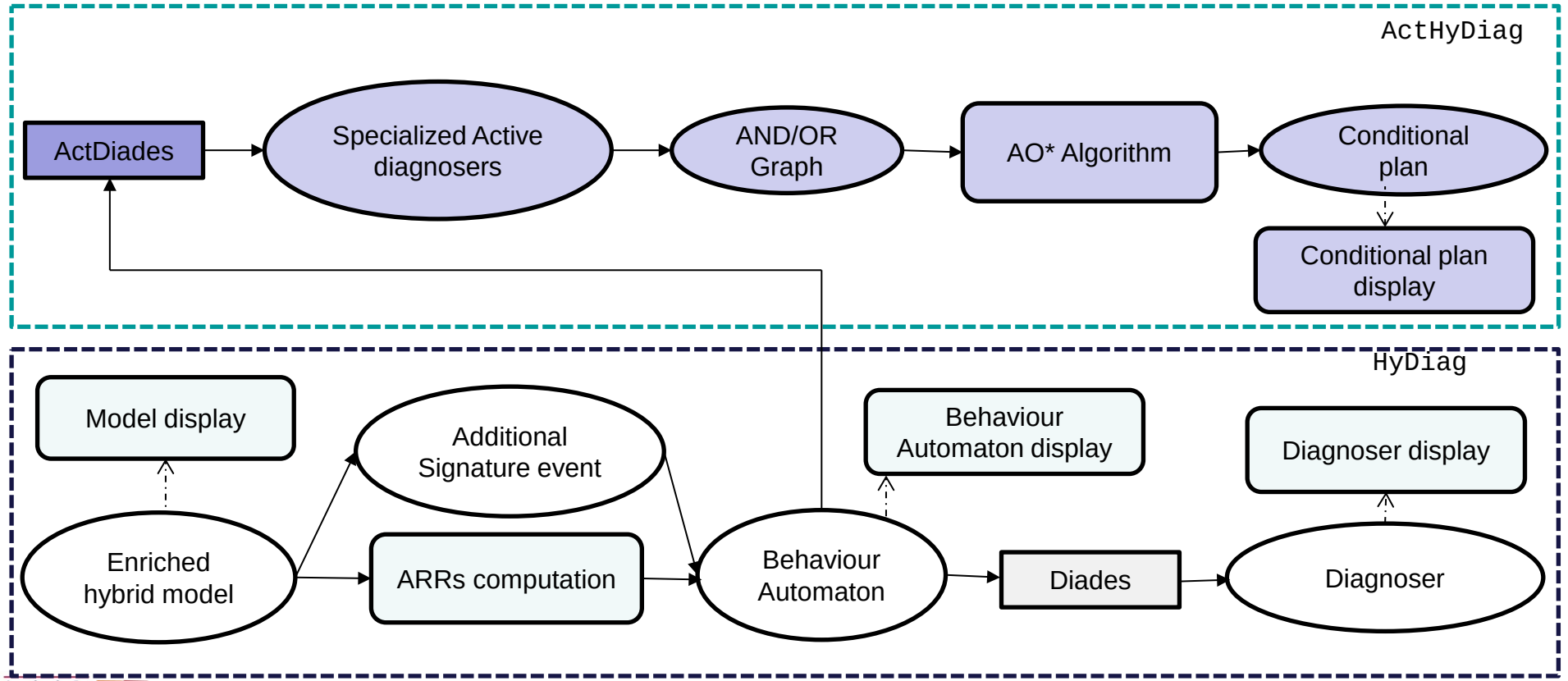
$$\boxed{f_j^{q_i}(t) = \int_0^t W(t, \beta_j^{q_i}, \eta_j^{q_i}, \gamma_j^{q_i}) dt} \qquad W(t, \beta_j^{q_i}, \eta_j^{q_i}, \gamma_j^{q_i}) = \frac{\beta_j^{q_i}}{\eta_j^{q_i}} \left(\frac{t - \gamma_j^{q_i}}{\eta_j^{q_i}} \right)^{(\beta_j^{q_i} - 1)} e^{-\left(\frac{t - \lambda_j^{q_i}}{\eta_j^{q_i}} \right)^{\beta_j^{q_i}}}$$

    - η, β: fixed by the system mode q_i → characterize degradation in mode q_i that leads to the fault f_j
    - γ : set at run time → it memorizes the overall degradation evolution accumulated in the past modes

- **Building the prognoser** [Zabi et al. 2013]
  - Predict fault occurrences using the aging laws in S^+ and the result of current diagnosis
  - Update on-line the aging laws (parameter γ) according to the operation time in each mode
  - For each new diagnosis result, computes the most likely sequence of dated faults that leads to a system failure → the RUL is estimated from this sequence

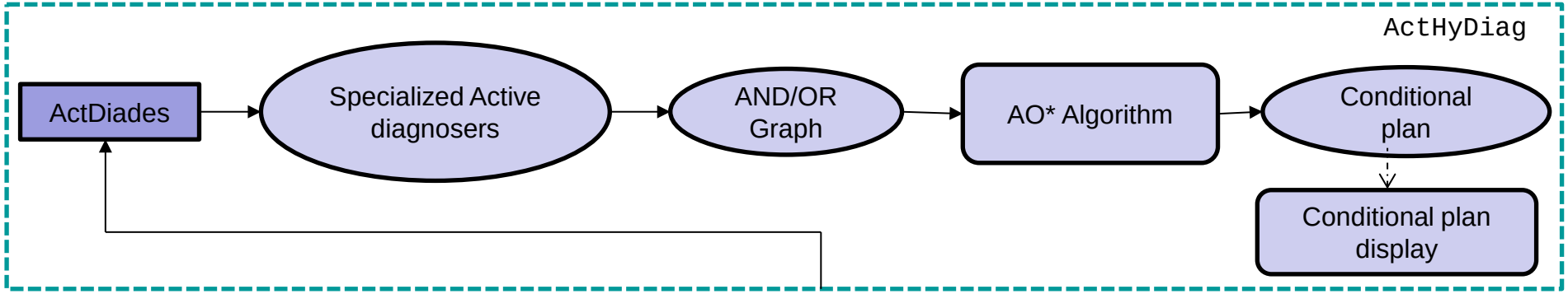
ActHyDiag : an extension for active diagnosis (1)

- **Implementation of an active diagnosis function** [Chanthery et al. 2010]
  - Compute a set of active diagnosers to predict if a fault can be diagnosed with certainty by applying an action plan in case of a ambiguous diagnosis provided by HyDiag
- **ActHyDiag software architecture**



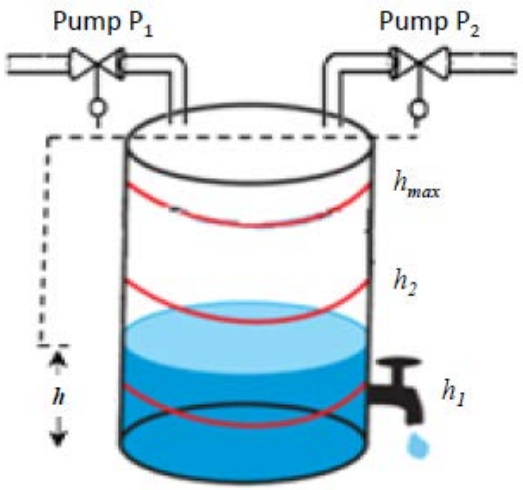
ActHyDiag : an extension for active diagnosis (2)

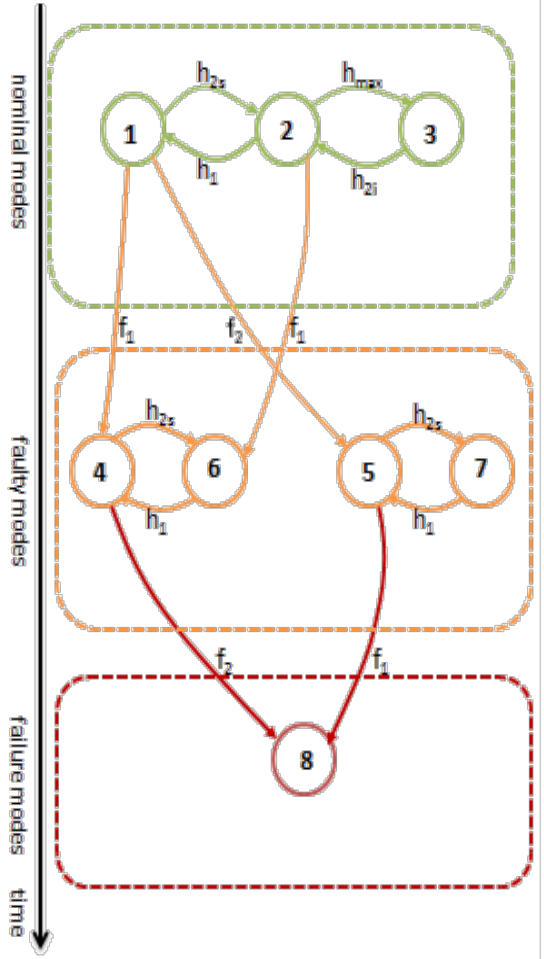
- **Computing the action plan to diagnose a fault** [Chanthery et al. 2010]
  - Indicate discrete actions and their costs in an additional file
  - Compute an active diagnoser for each anticipated fault to extract a planning domain as a AND/OR graph
  - Use an AO* algorithm to compute a conditional plan that optimizes action cost criterion
  - At runtime, launch an active diagnosis session in case of an ambiguous diagnosis to analyse if the faulty situation is discriminable by applying some actions



HyDiag/HyDiagPro demonstration (1)

- **Water tank system model**





3 sensors: h_1, h_2, h_{max}
Pump control signals : ON, OFF

3 nominal modes: q_1 q_2 , q_3
2 faults (unobservable): f_1 f_2

Failure mode when (f_1 AND f_2) have occurred.

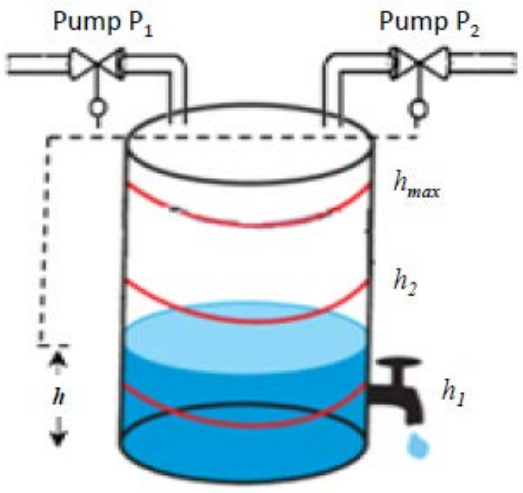
For each mode:

$$F^{q_i} = \begin{bmatrix} f_1^{q_i}(t) \\ f_2^{q_i}(t) \end{bmatrix}$$

Assumption : the pumps may fail only if they are on.

HyDiag/HyDiagPro demonstration (2)

- **Water tank system model**

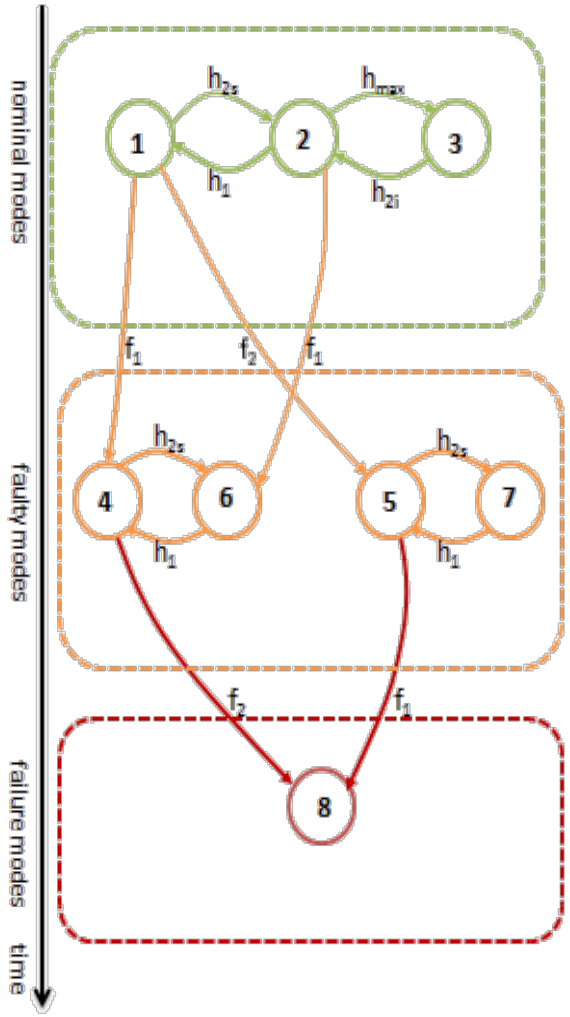


Underlying multi model system

$$C_i \begin{cases} X(k+1) = AX(k) + BU(k) \\ Y(k) = CX(k) + DU(k) \end{cases}$$

$$A = (1), B = \begin{pmatrix} e_i T_e / S \\ e_i T_e / S \\ e_i T_e / S \end{pmatrix}, C = (1), D = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Underlying DES



Mode \ Pump	Pump1	Pump2
1	ON	ON
2	ON	OFF
3	OFF	OFF
4	Fail	ON
5	ON	Fail
6	Fail	OFF
7	OFF	Fail
8	Fail	Fail

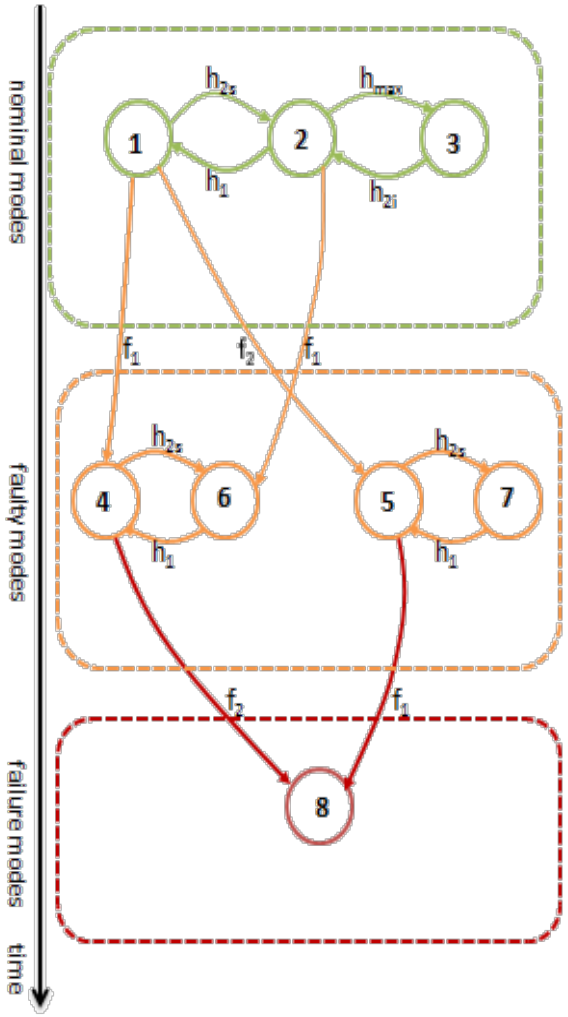
HyDiag/HyDiagPro demonstration (3)

- **Water tank system model**

Weibull parameters of aging models

Aging laws		β	η
F^{q_1}	$f_1^{q_1}$	1.5	3000
	$f_2^{q_1}$	1.5	4000
F^{q_2}	$f_1^{q_2}$	2	3000
	$f_2^{q_2}$	1	7000
F^{q_3}	$f_1^{q_3}$	1	8000
	$f_2^{q_3}$	1	7000
F^{q_4}	$f_1^{q_4}$	NaN	NaN
	$f_2^{q_4}$	2	4000
F^{q_5}	$f_1^{q_5}$	2	3000
	$f_2^{q_5}$	NaN	NaN
F^{q_6}	$f_1^{q_6}$	NaN	NaN
	$f_2^{q_6}$	1	7000
F^{q_7}	$f_1^{q_7}$	1	8000
	$f_2^{q_7}$	NaN	NaN
F^{q_8}	$f_1^{q_8}$	NaN	NaN
	$f_2^{q_8}$	NaN	NaN

Underlying DES

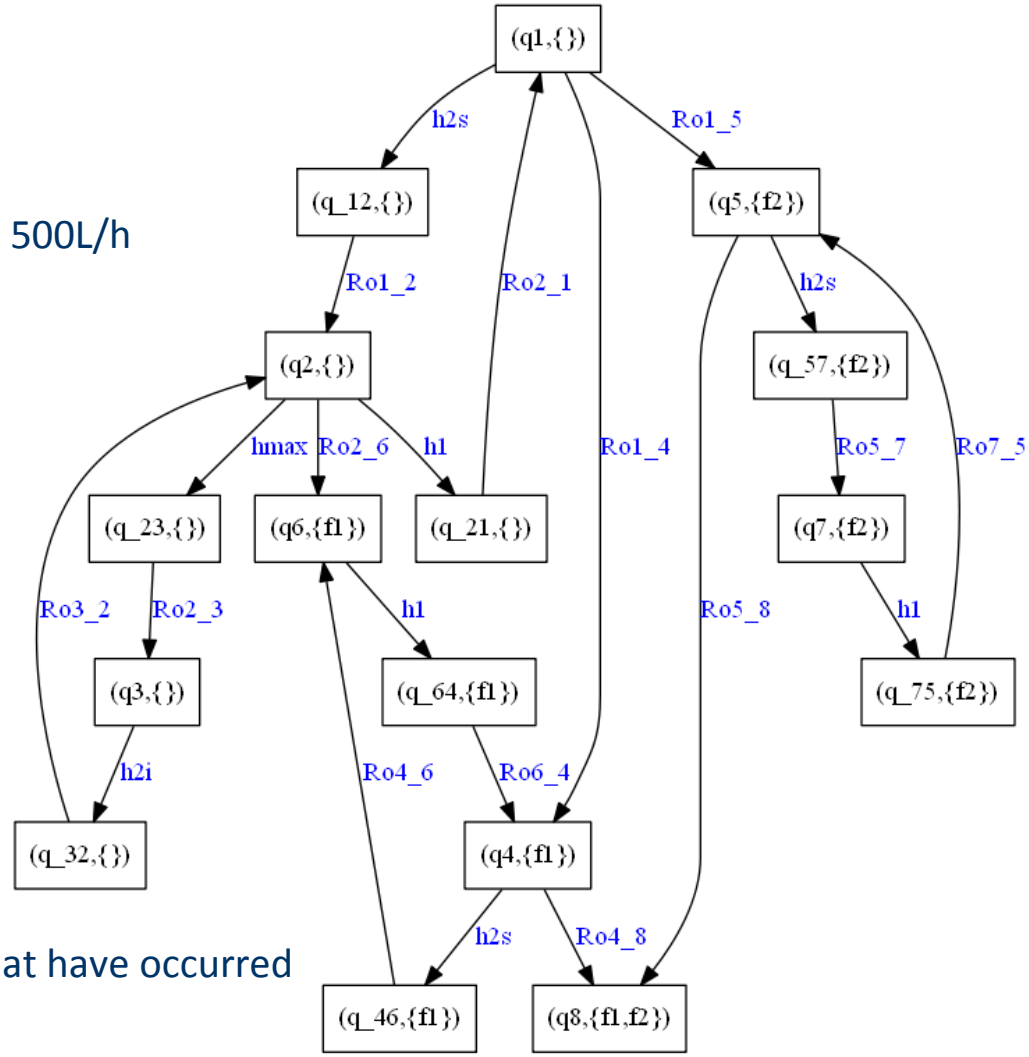


Pump / Mode	Pump1	Pump2
1	ON	ON
2	ON	OFF
3	OFF	OFF
4	Fail	ON
5	ON	Fail
6	Fail	OFF
7	OFF	Fail
8	Fail	Fail

HyDiag/HyDiagPro demonstration (4)

- **Simulation parameters**
  - Time horizon : 400h
  - Sampling period : 36s
  - Flow rates of P1 and P2 : 750L/h and 500L/h
  - Scenario :
    - fault f_1 is injected after 3500h
    - fault f_2 not injected

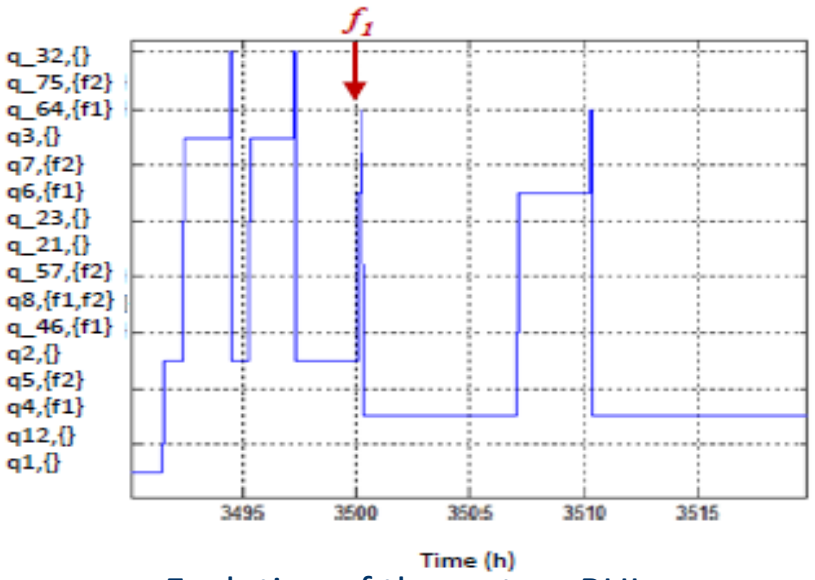
- **Diagnoser computed by HYDIAG**



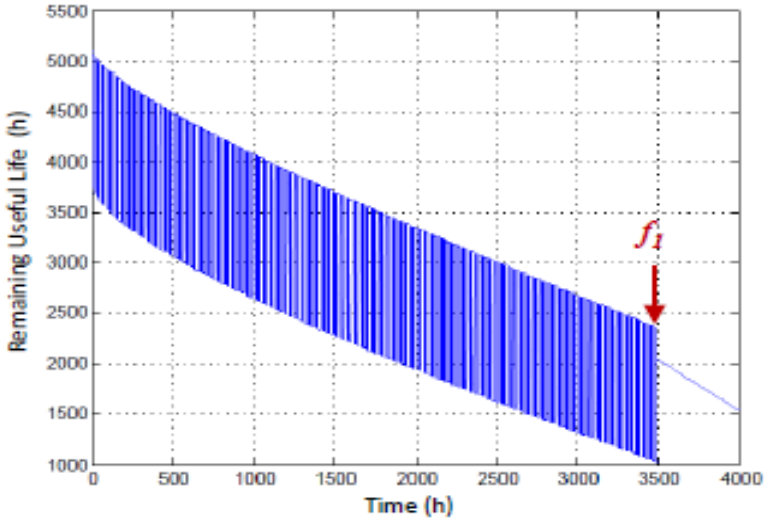
→ Diagnoser state : a belief state of the system model labelled with faults that have occurred

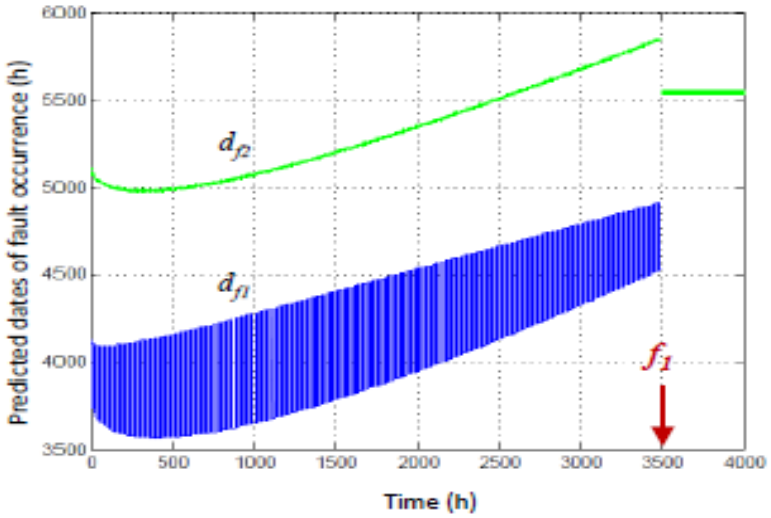
HyDiag/HyDiagPro Demonstration (5)

- **Simulation results**
  - Evolution of the diagnoser belief state
    - Before 3500h : states are tagged with a nominal diagnosis
    - After 3500h : states are tagged with t f_1
  - Evolution of the predicted date of fault occurrences $\mathrm{d}f_1$ and $\mathrm{d}f_2$
    - the system oscillates between stressfull modes and less stressfull modes before 3500h



- Evolution of the system RUL





Conclusions

- **HyDiag**
  - Software developed in Matlab by the DISCO team at LAAS-CNRS
  - Simulate and diagnose hybrid systems using model-based techniques
- **HyDiagPro : extension to prognosis**
  - Demonstration of HyDiagPro on an academic example
- **ActHyDiag : extension to active diagnosis**
  - Active diagnosis algorithm is currently tested on a concrete industrial case