Diagnosis of Large Software Systems Based on Colored Petri Nets

Supervised by Tarek Melliti [2] and Philippe Dague [1]

Yingmin Li [1]

[1] LRI IASI, Univ. Paris-Sud, CNRS/INRIA Saclay, France

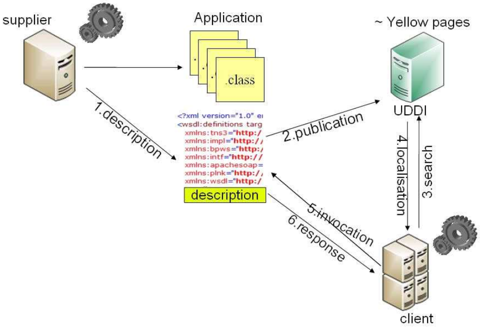
[2] IBISC, Univ. d'Evry Val d'Essonne, CNRS, France

December 9, 2010

Context: basic Web services

Web service:

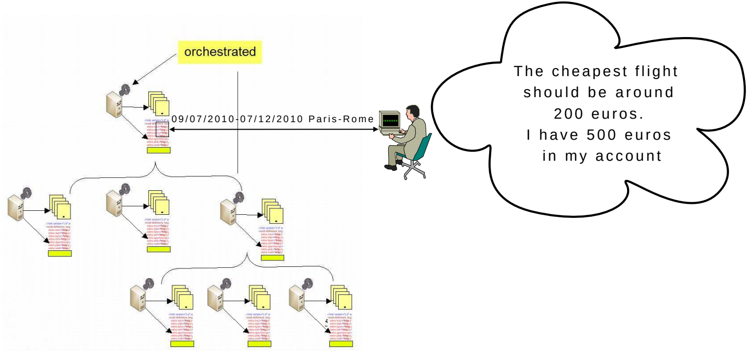
Technology allowing applications to dialogue remotely via Internet independently of the platforms and the languages they rest on. Cheaper and simpler for connection and integration.



Context: composite Web services

A travel agency example

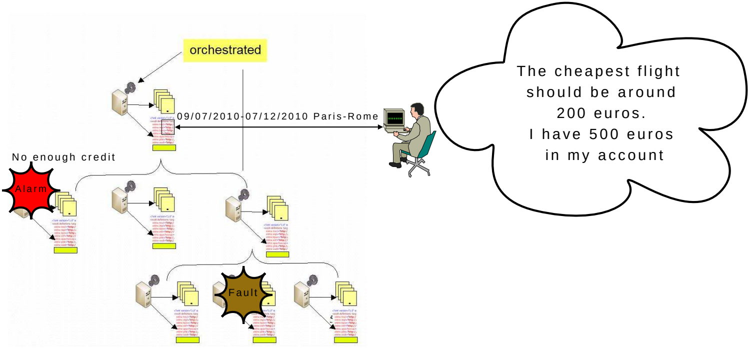
- An orchestrated Web service consists of a travel agency, an airline search engine, and a bank billing system
- On 09/07/2010, a client, reserved a round trip Paris-Rome, 07/11/2010-07/12/2010



Context: diagnosis problems of Web services

A travel agency example

- Reservation failed because of no enough credit?
- Client was confused and asked why the tickets were so expensive

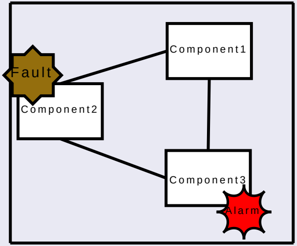


An abstract view

Large software systems

- Consist of several components located on different sites
- Dysfunctions are generated and transformed among the components through data (signals, interface data, etc)

A communicating components system



Observation

Completely ordered: O1, O2, ..., On

Partially ordered: O1, O3 / O2, O4 → On

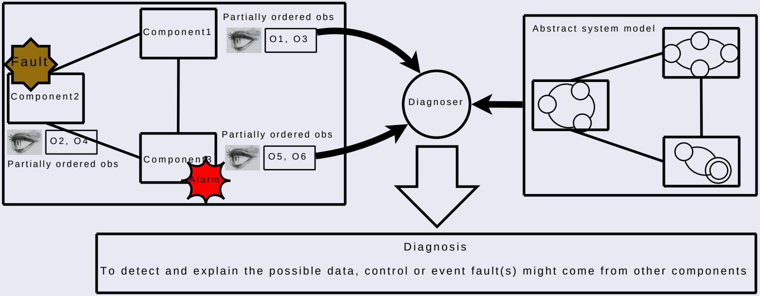
Non-ordered: {O1, O2, ..., On}

Diagnosis

To detect and explain the possible fault(s)

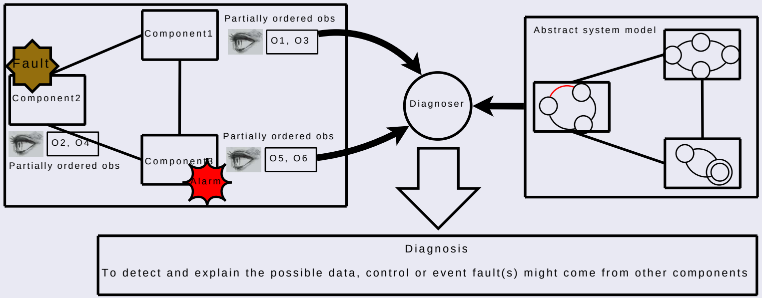
A possible solution: model based diagnosis

A communicating components system diagnosis



A possible solution: model based diagnosis

To model the faulty events as unobservable events



Existing works

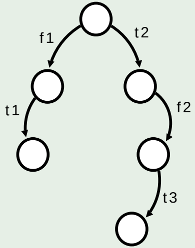
DES models

- Discrete event system (DES) models: Petri net (Petri [1973]), automata (Arto and N. [1969]), process algebra (van Glabbeek [1987]), etc.
- Most of them focus on the state evolution driven by the discrete events
- Faults are modeled as unobservable events (Sampath et al. [1995]) while faulty data is usually not modeled

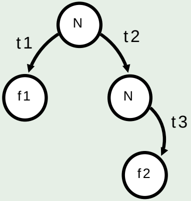
Existing works: diagnoser (Sampath et al. [1995, 1996])

Transform the discrete event model of the system to be diagnosed into a finite state automaton which has only observable events; the historical faulty events are recorded in states.

Example



system



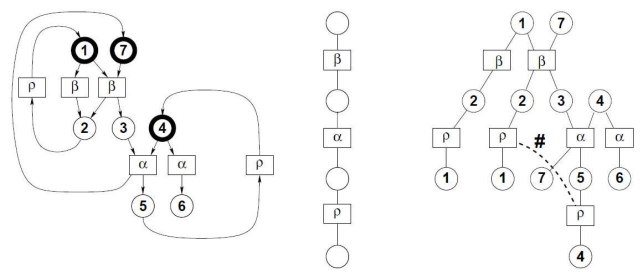
diagnoser

observations: t_2 t_3
Diagnosis=$\{f_2\}$

Existing works: PN unfolding (Benveniste et al. [2003])

Fully describes the concurrent behaviors in a single branching structure, represents all the possible computation steps and their mutual dependencies, as well as all reachable states.

Example

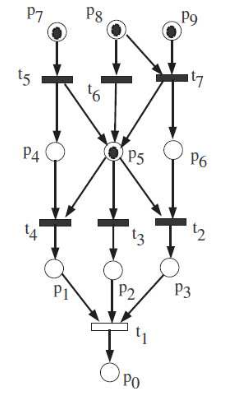


system observations unfolding diagnoser

Existing works: PN backward reasoning (Anglano and Portinale [1994]; Cardoso et al. [1995]; Jiroveanu [2006]; Srinivasan and Jafari [1994])

Starts from the final states which represents a symptom and calculates backwardly according to the backward searching rules to detect all the traces that cover it.

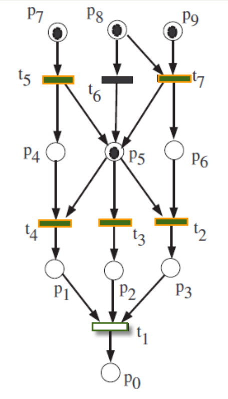
Example



system

t_1

observation



diagnosis

Assumptions

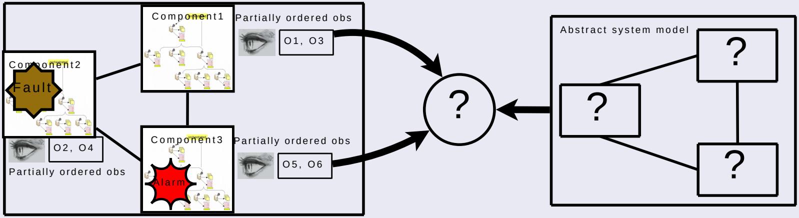
Targets

Locally correct and tested software system with interacting components in a stable network environment, but faulty activities, faulty data and controls (from user, database, interface, etc) that transmitted between the components

Symptoms

Exception(s) thrown on one or more components during the execution

Method: model based diagnosis



Challenges

Existing works

- focus on state evolution
- diagnose by trajectories reconstruction

We need an abstract model

- to represent the correct and fault of data and controls in a unique way
- to represent the correct and fault behaviors of the activities
- to represent the concurrency and partially ordered observations
- allows to handle the loops in an elegant way to avoid unfolding the trajectories
- to diagnose the orchestrated software systems

Our choice

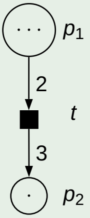
Colored Petri net

Outline

1. Introduction
   - Context: Web service diagnosis problem
   - Model-based diagnosis for Web services
   - Assumptions
2. Abstract model: CPN
   - CPN definition
   - Partially ordered observation
3. Diagnosis
   - CPN as a fault model
   - Diagnosis algorithm
4. Application
   - Architecture of BPEL Monitoring and Diagnosis
   - Decentralized diagnosis architecture
   - Example: travel agency diagnosis problem
5. Conclusions & perspectives
   - Contributions
   - Perspectives

CPN has same structure as PN

Example



Definition (Petri net)

$N = \langle P, T, Pre, Post \rangle$

- P: a set of labeled places
- T: a set of labeled transitions
- $Pre : P \times T \rightarrow \mathcal{N}$, a backward matrix of consumed token number
- $Post : P \times T \rightarrow \mathcal{N}$, a forward matrix of produced token number

Example (Incidence matrix C=Post-Pre)

C	t
p_1	-2
p_2	3

=

Post	t
p_1	
p_2	3

-

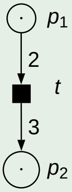
Pre	t
p_1	2
p_2	

Example (State equation)

	M'	=	M_0	+	C	$\times$	T
p_1	1		3		-2		
p_2	4		1		3		1

CPN has same structure as PN

Example



Definition (Petri net)

$N = \langle P, T, Pre, Post \rangle$

- P: a set of labeled places
- T: a set of labeled transitions
- $Pre : P \times T \rightarrow \mathcal{N}$, a backward matrix of consumed token number
- $Post : P \times T \rightarrow \mathcal{N}$, a forward matrix of produced token number

Example (Incidence matrix C=Post-Pre)

C	t
p_1	-2
p_2	3

=

Post	t
p_1	
p_2	3

-

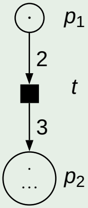
Pre	t
p_1	2
p_2	

Example (State equation)

	M'	=	M_0	+	C	$\times$	T
p_1	1		3		-2		
p_2	4		1		3		1

CPN has same structure as PN

Example



Definition (Petri net)

$N = \langle P, T, Pre, Post \rangle$

- P: a set of labeled places
- T: a set of labeled transitions
- $Pre : P \times T \rightarrow \mathcal{N}$, a backward matrix of consumed token number
- $Post : P \times T \rightarrow \mathcal{N}$, a forward matrix of produced token number

Example (Incidence matrix C=Post-Pre)

C	t
p_1	-2
p_2	3

=

Post	t
p_1	
p_2	3

-

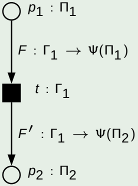
Pre	t
p_1	2
p_2	

Example (State equation)

	M'	=	M_0	+	C	$\times$	T
p_1	1		3		-2		
p_2	4		1		3		1

CPN definition: Structure & dynamic (Li et al. [2009a])

Example

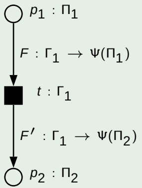


PN to CPN

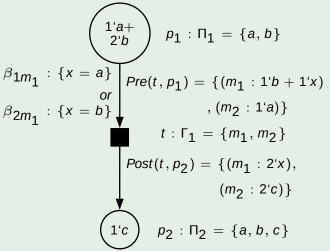
- Places are typed ($p_1 : \Pi_1$)
- Transition are typed ($t : \Gamma_1$)
- Markings are multi-sets of place types ($m(p_1) = \sum_{i \in |\Pi_1|} n_i{`}e_i, n_i \geqslant 0, e_i \in \Pi_1$)
- Weights of an edges are multi-set expressions ($Pre(p_1, t) : \Gamma_1 \to \Psi(\Pi_1)$)

CPN definition: Structure & dynamic (Li et al. [2009a])

Example



Example (Binding β)



PN to CPN

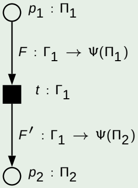
- Places are typed ($p_1 : \Pi_1$)
- Transition are typed ($t : \Gamma_1$)
- Markings are multi-sets of place types ($m(p_1) = \sum_{i \in |\Pi_1|} n_i`e_i, n_i \geqslant 0, e_i \in \Pi_1$)
- Weights of an edges are multi-set expressions ($Pre(p_1, t) : \Gamma_1 \rightarrow \Psi(\Pi_1)$)

Mode firing rule: $M[t_m\rangle^{\beta} M'$

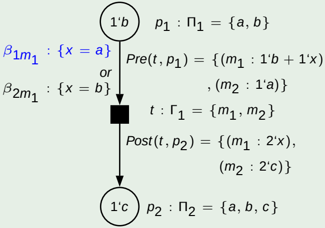
- $M' = M + C(., t)(m_1)^{\beta}$ with $\beta : x = a$
- Marking: M' is reachable from M under β and m
- can be extended to sequence firing

CPN definition: Structure & dynamic (Li et al. [2009a])

Example



Example (Binding β)



PN to CPN

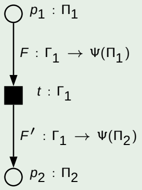
- Places are typed ($p_1 : \Pi_1$)
- Transition are typed ($t : \Gamma_1$)
- Markings are multi-sets of place types ($m(p_1) = \sum_{i \in |\Pi_1|} n_i‘e_i, n_i \geqslant 0, e_i \in \Pi_1$)
- Weights of an edges are multi-set expressions ($Pre(p_1, t) : \Gamma_1 \rightarrow \Psi(\Pi_1)$)

Mode firing rule: $M[t_m\rangle^{\beta} M'$

- $M' = M + C(., t)(m_1)^{\beta}$ with $\beta : x = a$
- Marking: M' is reachable from M under β and m
- can be extended to sequence firing

CPN definition: Structure & dynamic (Li et al. [2009a])

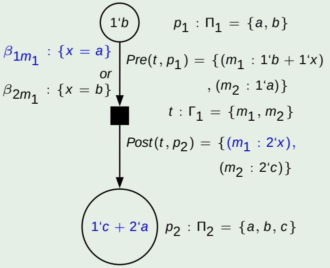
Example



PN to CPN

- Places are typed ($p_1 : \Pi_1$)
- Transition are typed ($t : \Gamma_1$)
- Markings are multi-sets of place types ($m(p_1) = \sum_{i \in |\Pi_1|} n_i`e_i, n_i \geqslant 0, e_i \in \Pi_1$)
- Weights of an edges are multi-set expressions ($Pre(p_1, t) : \Gamma_1 \rightarrow \Psi(\Pi_1)$)

Example (Binding β)



Mode firing rule: $M[t_m\rangle^\beta M'$

- $M' = M + C(., t)(m_1)^\beta$ with $\beta : x = a$
- Marking: M' is reachable from M under β and m
- can be extended to sequence firing

CPN definition: Dynamic

Definition (Characteristic Vector)

- Sequence of modes: $\delta \in Mod^*$
- Characteristic vector of δ: $\overrightarrow{\delta} : Mod \rightarrow \mathbb{N}$
- Transition characteristic vector $\overrightarrow{\delta_T} : T \rightarrow \mathbb{N}$

Example

- $\delta = t_1.m_1\ t_2.m_3\ t_1.m_2\ t_1.m_1\ t_1.m_2$
- $\overrightarrow{\delta}(m_1) = 2$
- $\overrightarrow{\delta_T}(t_1) = 4$

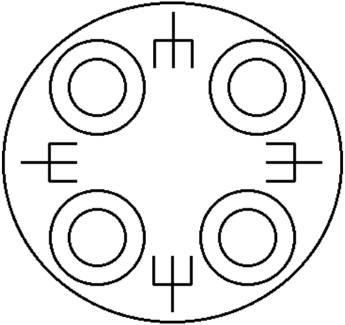
State equation

Given S=$\langle N, M\rangle$ a CPN-S and modes sequence, $\delta \in Mod^*$ with $M[\delta\rangle$ then the reached marking M' after the firing of δ is :

$$M' = M + C \times \overrightarrow{\delta}$$

CPN definition: example (1/2)

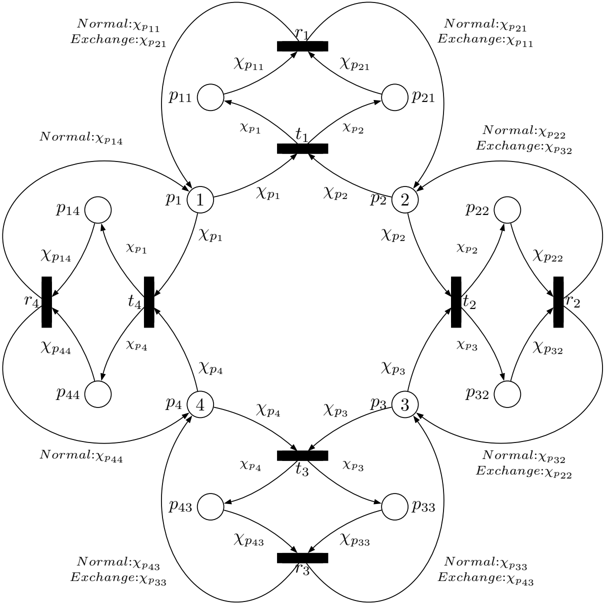
- Dining philosophers (of size four)
- No blocking version (taking both forks concurrently)
- Forks are identified by the serial number 1-4
- One of them (num 4) is well-organized:
  - to take the forks, and before eat, he/she checks the id of the forks
- Three of them (num 1, 2 and 3) are Unorganized:
  - don't verify the forks'id and might exchange forks in hands before restore them.



CPN definition: example (2/2)

Notations

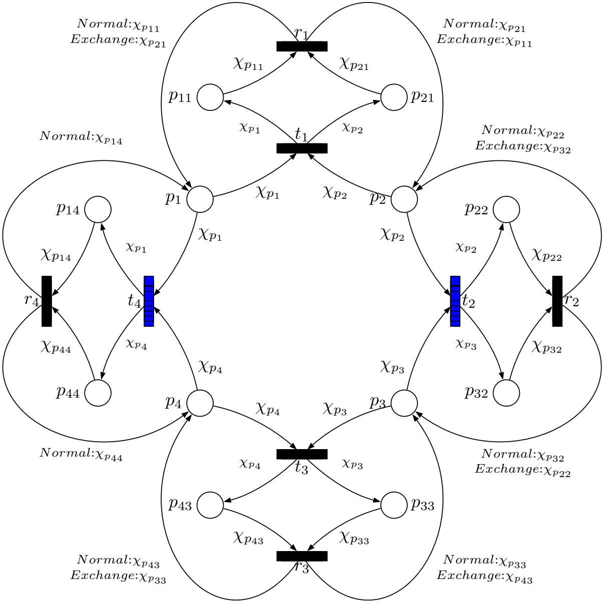
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

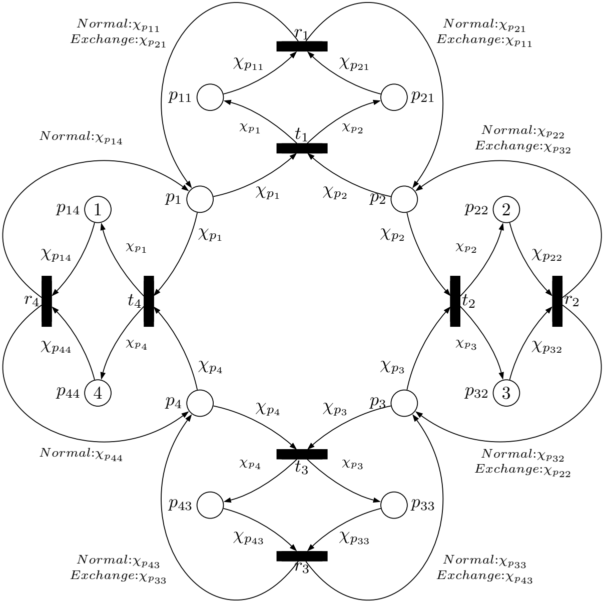
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

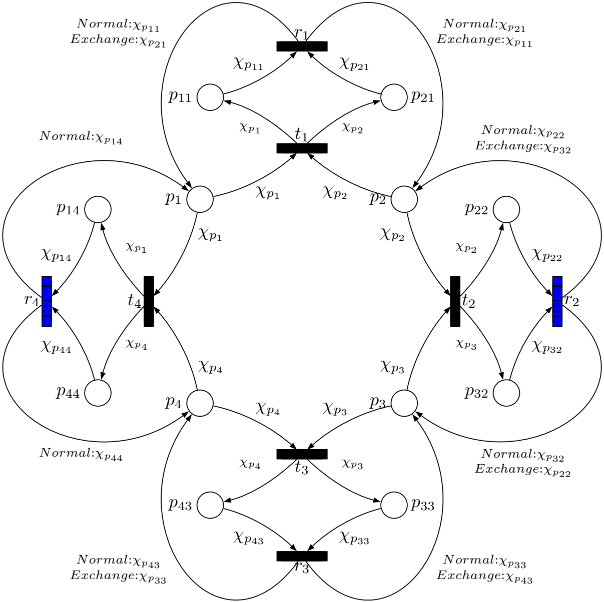
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

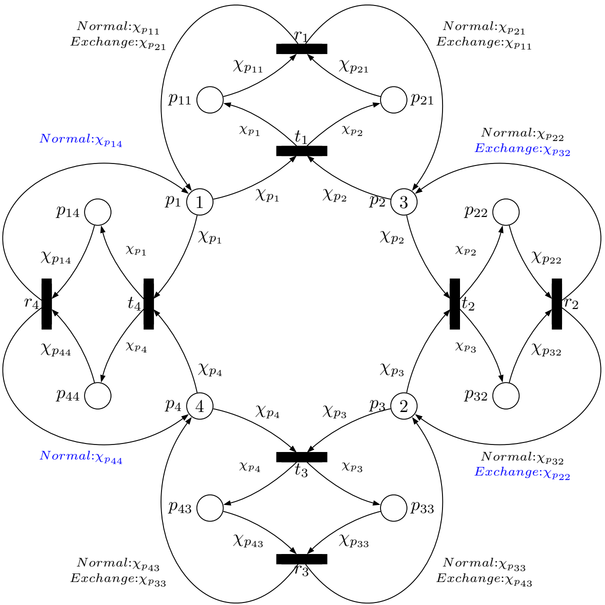
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

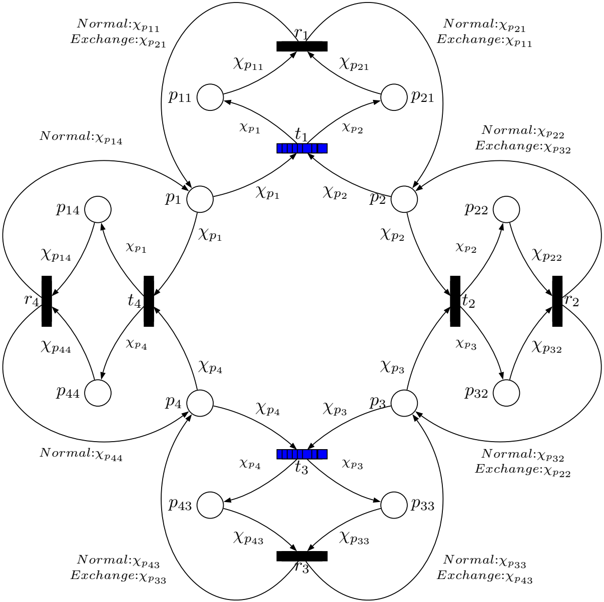
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

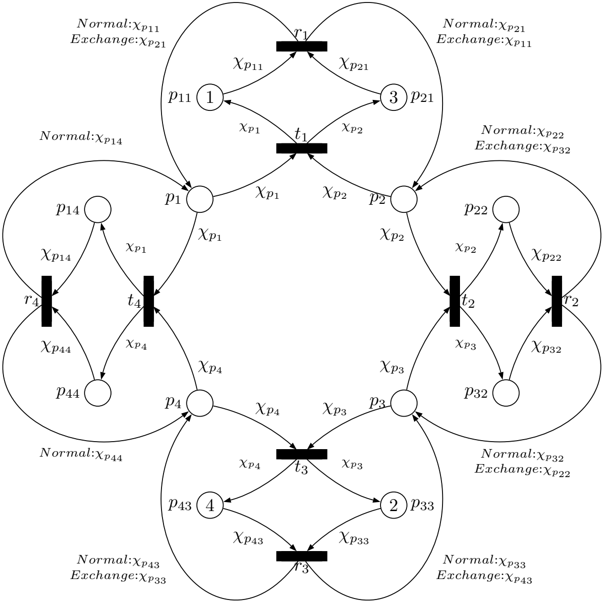
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

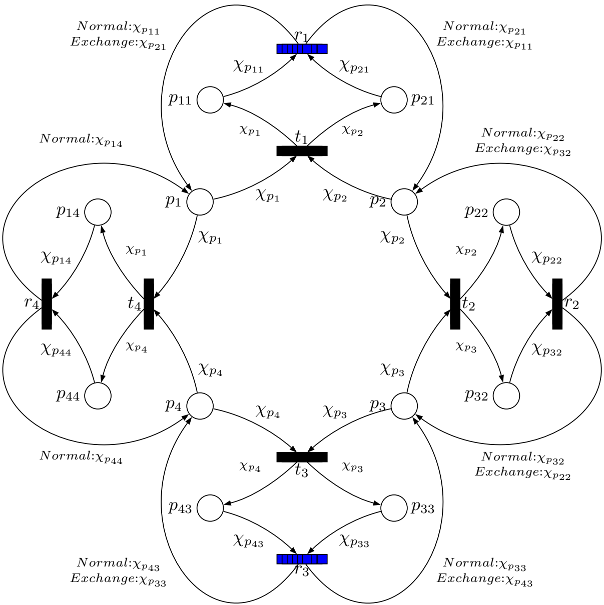
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

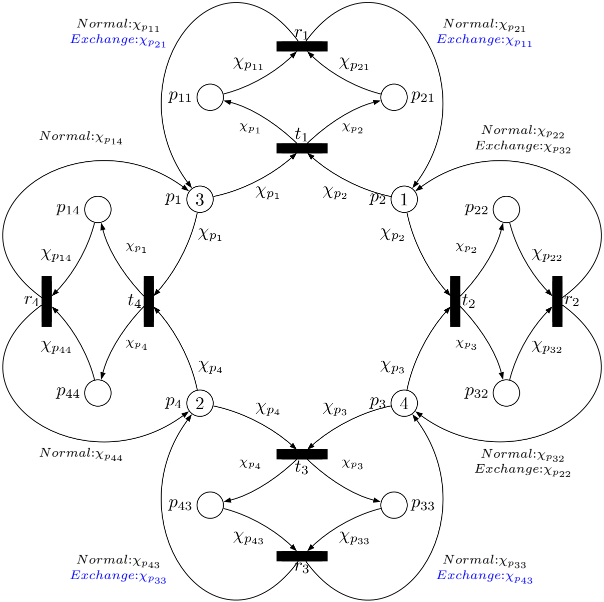
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

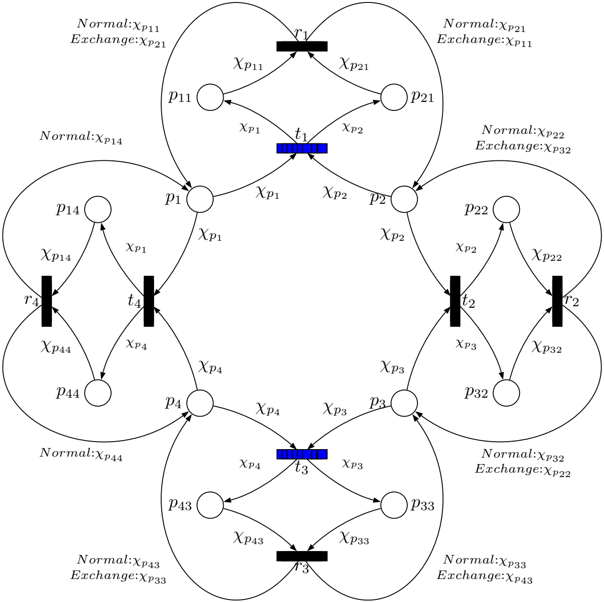
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

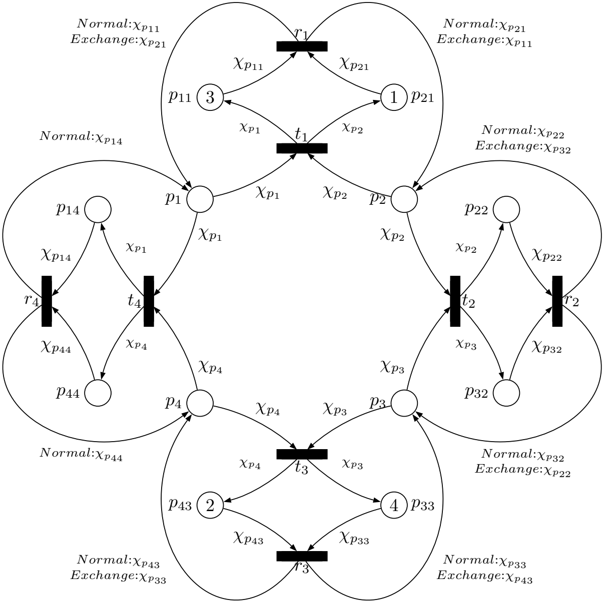
- p_i: the forks on the table
- t_i and r_i: philosopher *i* takes and released the fork
- p_{ii}: the fork in the right hand of philosopher *i*
- p_{ji}: the fork in the left hand of philosopher *i*



CPN definition: example (2/2)

Notations

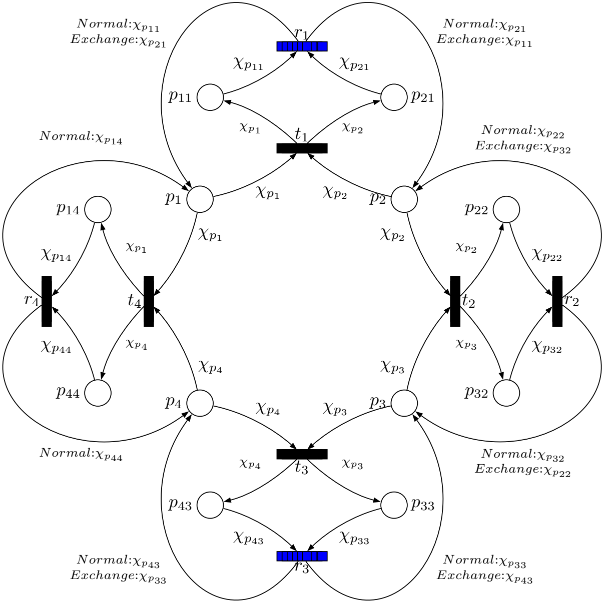
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

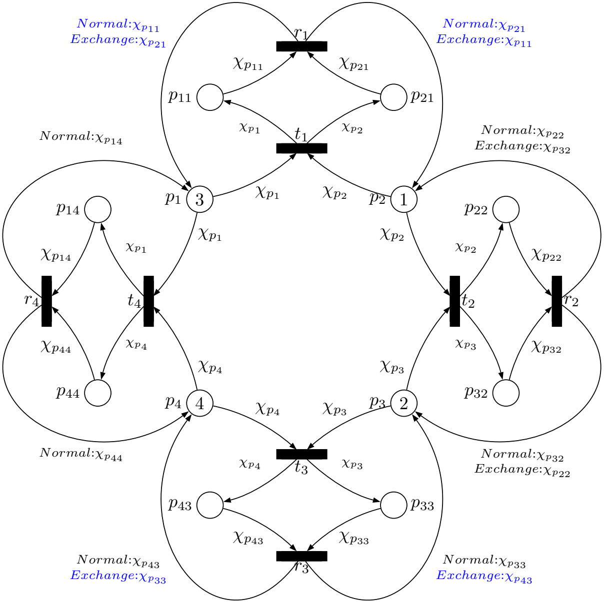
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

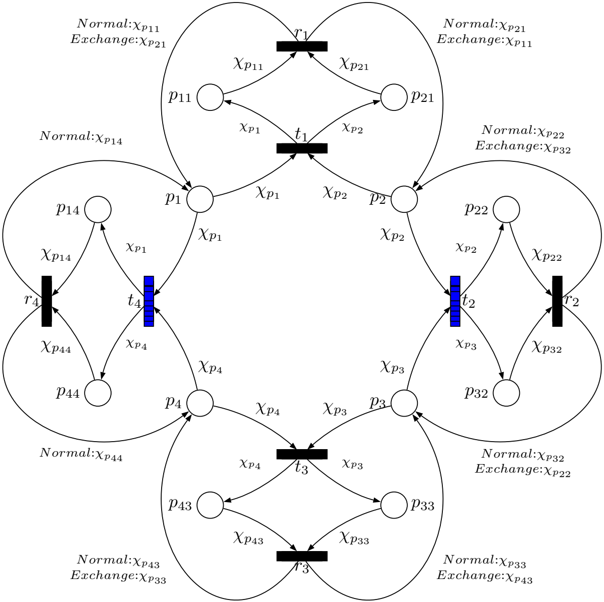
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

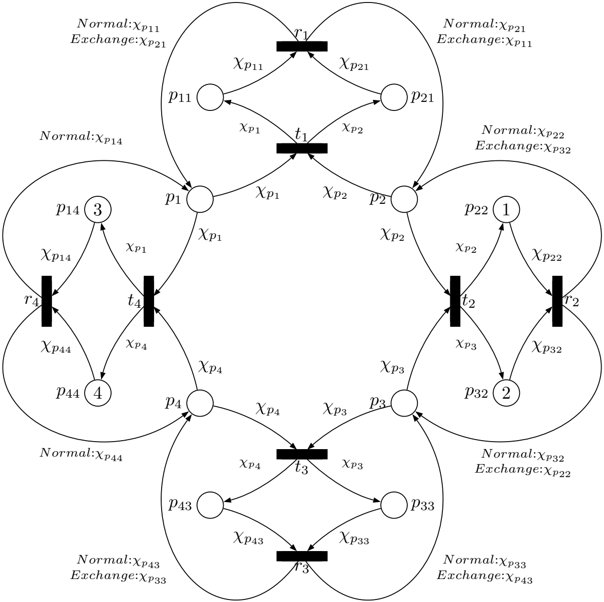
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

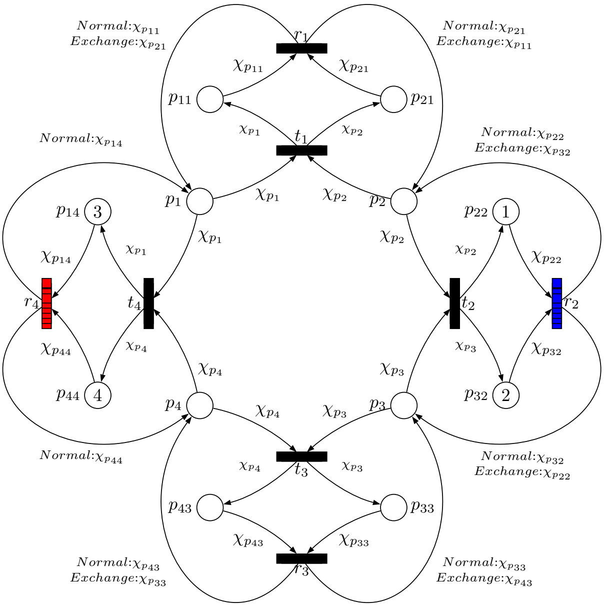
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

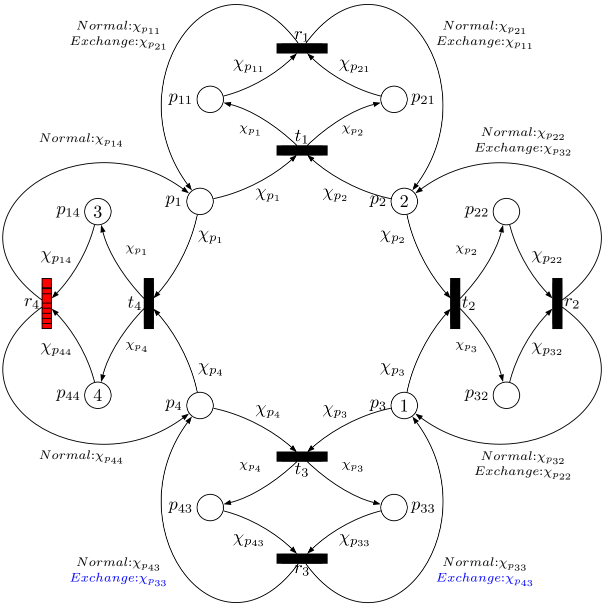
- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



CPN definition: example (2/2)

Notations

- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i



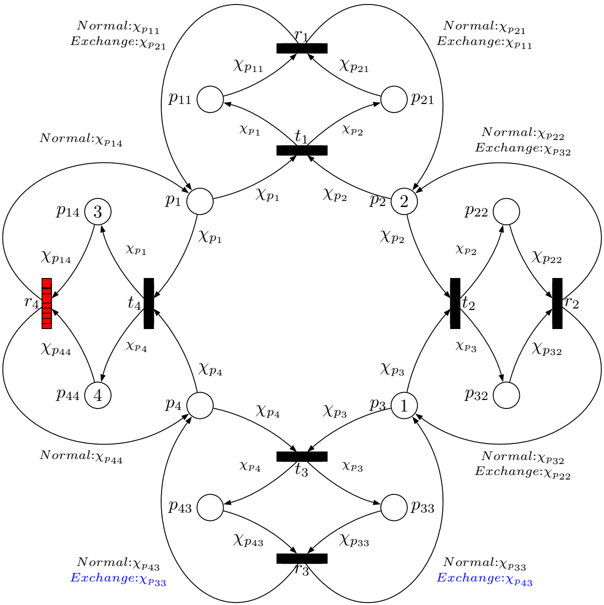
CPN definition: example (2/2)

Notations

- p_i: the forks on the table
- t_i and r_i: philosopher i takes and released the fork
- p_{ii}: the fork in the right hand of philosopher i
- p_{ji}: the fork in the left hand of philosopher i

Observations

- each philosopher ate twice except philosopher 4
- t_i is fired before r_i
- some activities were partially ordered



Partially ordered observation

Definition (Partially ordered observation $(\mathcal{T}, \lhd)$)

- $\mathcal{T} = \{t_i^k | 1 \leqslant k \leqslant |\mathcal{T}(t_i)|\}$: repeated occurred transitions
- $\lhd \subseteq (S(\mathcal{T}) \times S(\mathcal{T}))$: partially ordered relation

Partially ordered observation

Definition (Partially ordered observation $(\mathcal{T}, \lhd)$)

- $\mathcal{T} = \{t_i^k | 1 \leqslant k \leqslant |\mathcal{T}(t_i)|\}$: repeated occurred transitions
- $\lhd \subseteq (S(\mathcal{T}) \times S(\mathcal{T}))$: partially ordered relation

Example (less specific $\lhd$)

- $\lhd_i = \{(t_i^{(k)}, r_i^{(k)}), (r_i^{(k)}, t_i^{(k+1)})\}$
- $\lhd_1$ is less specific than $\lhd_1 \cup \lhd_2$

Partially ordered observation

Definition (Partially ordered observation $(\mathcal{T}, \lhd)$)

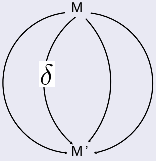
- $\mathcal{T} = \{t_i^k \mid 1 \leqslant k \leqslant |\mathcal{T}(t_i)|\}$: repeated occurred transitions
- $\lhd \subseteq (S(\mathcal{T}) \times S(\mathcal{T}))$: partially ordered relation

Example (less specific $\lhd$)

- $\lhd_i = \{(t_i^{(k)}, r_i^{(k)}), (r_i^{(k)}, t_i^{(k+1)})\}$
- $\lhd_1$ is less specific than $\lhd_1 \cup \lhd_2$

Definition (Minimal partially ordered relation $\lhd_{min}^{M \xrightarrow{\mathcal{T}} M'}$)

$$\mathcal{T} = \begin{cases} t_1 : & 2 \\ r_1 : & 2 \\ t_2 : & 2 \\ r_2 : & 2 \\ t_3 : & 2 \\ r_3 : & 2 \\ t_4 : & 2 \\ r_4 : & 1 \end{cases} \quad \Rightarrow \quad \begin{aligned} S(\mathcal{T}) &= \{t_1^{(1)}, t_1^{(2)}, r_1^{(1)}, r_1^{(2)}, \\ &\quad \ldots, t_4^{(1)}, r_4^{(1)}\} \\ \lhd_i &= \{(t_i^{(k)}, r_i^{(k)}), \\ &\quad (r_i^{(k)}, t_i^{(k+1)})\} \end{aligned}$$



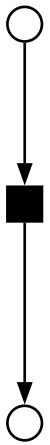
$(t_i, t_j) \in \lhd_{min}^{M \xrightarrow{\mathcal{T}} M'}$, iff $\forall \delta, M[\delta > M' \wedge \overrightarrow{\delta_T} = \mathcal{T} \Rightarrow \delta \vDash t_i \lhd t_j$

Adapt the CPN as a fault model

An abstract fault model

- Define data correctness status $\{b, r, *\}$
- Map all the transition modes into diagnostic transition modes $\{OK, KO\}$
- Abstract transition functionalities as data dependency functions $\{FW, SRC, EL\}$

Data dependencies in transition modes



$p_1 : \Pi_1$ $\qquad$ $p_1 : \Psi_D = status = \{b, r, *\}$

$F : \Gamma_1 \to \Psi(\Pi_1)$ $\qquad$ $\chi p : status$

$t : \Gamma_1$ $\qquad$ $Fault : \Gamma_1 \to \{OK, KO\}$

$F' : \Gamma_1 \to \Psi(\Pi_2) \Rightarrow \forall \chi p \in \Pi_1 \cap \Pi_2, \chi_p' \in \Pi_2, \Pi_1' \subseteq \Pi_1$ $\qquad$ $\forall c, c' \in status, \forall \mathcal{C} \subseteq status,$

$p_2 : \Pi_2$ $\qquad$ $p_2 : status$

Data dependencies in transition modes

$p_1 : \Pi_1$

$F : \Gamma_1 \to \Psi(\Pi_1)$

$t : \Gamma_1$

$F' : \Gamma_1 \to \Psi(\Pi_2) \Rightarrow \forall \chi_p \in \Pi_1 \cap \Pi_2, \chi'_p \in \Pi_2, \Pi'_1 \subseteq \Pi_1$

$$D = \begin{cases} \text{FW:}\Pi_1 \cap \Pi_2 \to \Psi(\Pi_2), FW(\chi_p) = \chi_p \\ \text{SRC:}\emptyset \to \Psi(\Pi_2), SRC = \chi_p \\ \text{EL:}(\Pi_1 \cap \Pi_2)^n \to \Psi(\Pi_2), EL(\Pi'_1) = \chi'_p \end{cases}$$

$p_2 : \Pi_2$

$p_1 : \Psi_D = status = \{b, r, *\}$

$\chi_p : status$

$Fault : \Gamma_1 \to \{OK, KO\}$

$\forall c, c' \in status, \forall \mathcal{C} \subseteq status,$

$$\Rightarrow D^c = \begin{cases} \text{FW:status} \to status, FW^c(c) = c \\ \text{SRC:}\emptyset \to status, SRC^c = * \\ \text{EL :status}^n \to status, EL^c(\mathcal{C}) = c', \text{with } c' = \\ \quad \begin{cases} \text{b, iff } \forall c \in \mathcal{C}, c = b \\ \text{r, iff } \exists c \in \mathcal{C}, c = r \\ *, \textit{iff } \exists c \in \mathcal{C}, c = * \\ \wedge \nexists c'' \in \mathcal{C}, c'' = r \end{cases} \end{cases}$$

$p_2 : status$

Data dependencies in transition modes

$p_1 : \Pi_1$

$F : \Gamma_1 \rightarrow \Psi(\Pi_1)$

$t : \Gamma_1$

$F' : \Gamma_1 \rightarrow \Psi(\Pi_2) \Rightarrow \forall \chi_p \in \Pi_1 \cap \Pi_2, \chi'_p \in \Pi_2, \Pi'_1 \subseteq \Pi_1$

$$D = \begin{cases} \text{FW:}\Pi_1 \cap \Pi_2 \rightarrow \Psi(\Pi_2), FW(\chi_p) = \chi_p \\ \text{SRC:}\emptyset \rightarrow \Psi(\Pi_2), SRC = \chi_p \\ \text{EL:}(\Pi_1 \cap \Pi_2)^n \rightarrow \Psi(\Pi_2), EL(\Pi'_1) = \chi'_p \end{cases} \Rightarrow$$

$p_2 : \Pi_2$

$p_1 : \Psi_D = status = \{b, r, *\}$

$\chi_p : status$

$Fault : \Gamma_1 \rightarrow \{OK, KO\}$

$\forall c, c' \in status, \forall \mathcal{C} \subseteq status,$

$$D^c = \begin{cases} \text{FW:status} \rightarrow status, FW^c(c) = c \\ \text{SRC:}\emptyset \rightarrow status, SRC^c = * \\ \text{EL :status}^n \rightarrow status, EL^c(\mathcal{C}) = c', \text{with } c' = \\ \quad \begin{cases} \text{b, iff } \forall c \in \mathcal{C}, c = b \\ \text{r, iff } \exists c \in \mathcal{C}, c = r \\ *, \textit{iff } \exists c \in \mathcal{C}, c = * \\ \wedge \nexists c'' \in \mathcal{C}, c'' = r \end{cases} \end{cases}$$

$p_2 : status$

Diagnosis properties of data dependency (Ardissono et al. [2005])

EL							
m_t	$c_{\bullet i\, t}$	$c_{\bullet j\, t}$	$c_{t\bullet}$	m_t	$c_{\bullet i\, t}$	$c_{\bullet j\, t}$	$c_{t\bullet}$
OK	b	b	b	KO	b	b	r
OK	r	b	r	KO	r	b	*
OK	*	b	*	KO	*	b	*
OK	*	*	*	KO	*	*	*
OK	*	r	r	KO	*	r	*

FW		
m_t	$c_{\bullet t}$	$c_{t\bullet}$
OK / KO	b	b
OK / KO	r	r
OK / KO	*	*

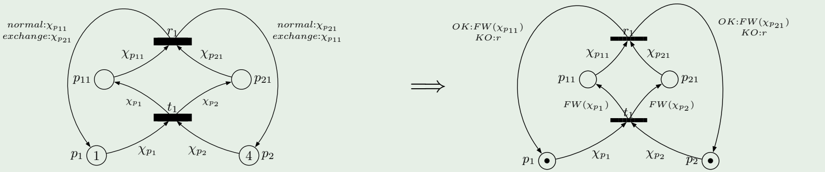
SRC	
m_t	$c_{t\bullet}$
OK	b
KO	r

CPN for diagnosis: fault model

Fault model: $F : \bigcup_{\gamma \in \Gamma} \gamma \rightarrow \{OK, KO\}$

$F(Normal) = OK$, $F(Exchange) = KO$

Example (philosopher 1)

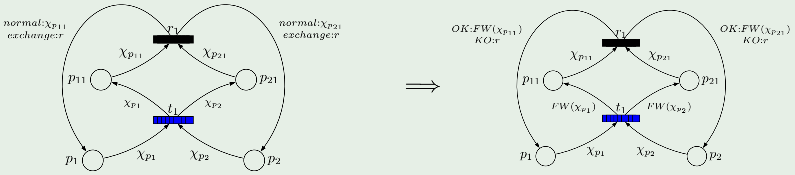


CPN for diagnosis: fault model

Fault model: $F : \bigcup_{\gamma \in \Gamma} \gamma \rightarrow \{OK, KO\}$

$F(Normal) = OK$, $F(Exchange) = KO$

Example (philosopher 1)

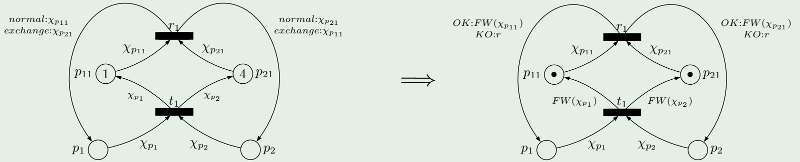


CPN for diagnosis: fault model

Fault model: $F : \bigcup_{\gamma \in \Gamma} \gamma \rightarrow \{OK, KO\}$

$F(Normal) = OK, F(Exchange) = KO$

Example (philosopher 1)

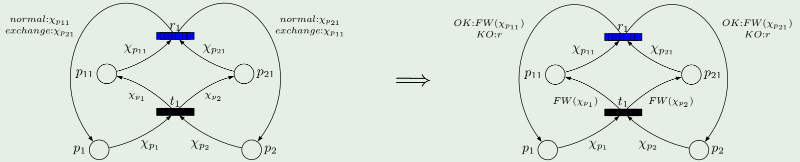


CPN for diagnosis: fault model

Fault model: $F : \bigcup_{\gamma \in \Gamma} \gamma \rightarrow \{OK, KO\}$

$F(Normal) = OK, F(Exchange) = KO$

Example (philosopher 1)

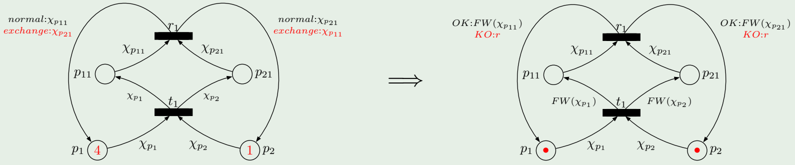


CPN for diagnosis: fault model

Fault model: $F : \bigcup_{\gamma \in \Gamma} \gamma \rightarrow \{OK, KO\}$

$F(Normal) = OK$, $F(Exchange) = KO$

Example (philosopher 1)



Diagnosis problem: $\mathcal{D} = \langle M_0, (S(\mathcal{T}), \lhd), \hat{M} \rangle$ (Li et al. [2009a])

Initial marking M_0

p_1	p_2	p_3	p_4	p_{11}	p_{21}	p_{22}	p_{32}	p_{33}	p_{43}	p_{44}	p_{14}
$\langle *$	$*$	$*$	$*$	0	0	0	0	0	0	0	$0 \rangle$

Symptom $\hat{M}$

p_1	p_2	p_3	p_4	p_{11}	p_{21}	p_{22}	p_{32}	p_{33}	p_{43}	p_{44}	p_{14}
$\langle 0$	$*$	$*$	0	0	0	0	0	0	0	b	r $\rangle$

$(S(\mathcal{T}), \lhd)$

$t_1^{(1)} \lhd r_1^{(1)} \lhd t_1^{(2)} \lhd r_1^{(2)},$

$t_2^{(1)} \lhd r_2^{(1)} \lhd t_2^{(2)} \lhd r_2^{(2)},$

$t_3^{(1)} \lhd r_3^{(1)} \lhd t_3^{(2)} \lhd r_3^{(2)},$

$t_4^{(1)} \lhd r_4^{(1)} \lhd t_4^{(2)}$

Diagnosis problem: $\mathcal{D}= \langle M_0, (S(\mathcal{T}), \lhd), \hat{M}\rangle$ (Li et al. [2009a])

Initial marking M_0

p_1	p_2	p_3	p_4	p_{11}	p_{21}	p_{22}	p_{32}	p_{33}	p_{43}	p_{44}	p_{14}
⟨ *	*	*	*	0	0	0	0	0	0	0	0 ⟩

Symptom $\hat{M}$

p_1	p_2	p_3	p_4	p_{11}	p_{21}	p_{22}	p_{32}	p_{33}	p_{43}	p_{44}	p_{14}
⟨ 0	*	*	0	0	0	0	0	0	0	b	r ⟩

$(S(\mathcal{T}), \lhd)$

$t_1^{(1)} \lhd r_1^{(1)} \lhd t_1^{(2)} \lhd r_1^{(2)}$,

$t_2^{(1)} \lhd r_2^{(1)} \lhd t_2^{(2)} \lhd r_2^{(2)}$,

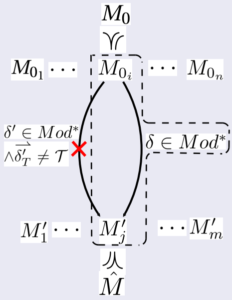
$t_3^{(1)} \lhd r_3^{(1)} \lhd t_3^{(2)} \lhd r_3^{(2)}$,

$t_4^{(1)} \lhd r_4^{(1)} \lhd t_4^{(2)}$

A solution for a CPN diagnosis problem

$\overrightarrow{\delta}_T$	
t_1	:2
r_1	:2
t_2	:2
r_2	:2
t_3	:2
r_3	:2
t_4	:2
r_4	:1

$\overrightarrow{\delta}$	
t_1 :	2
$r_1.OK$	: n_1
$r_1.KO$	: n_2
t_2 :	2
$r_2.OK$	: n_3
$r_2.KO$	: n_4
t_3 :	2
$r_3.OK$	: n_5
$r_3.KO$	: n_6
t_4 :	2
r_4 :	1



Diagnosis problem: $\mathcal{D} = \langle M_0, (S(\mathcal{T}), \lhd), \hat{M} \rangle$ (Li et al. [2009a])

Initial marking M_0

p_1	p_2	p_3	p_4	p_{11}	p_{21}	p_{22}	p_{32}	p_{33}	p_{43}	p_{44}	p_{14}
$*$	$*$	$*$	$*$	0	0	0	0	0	0	0	0

Symptom $\hat{M}$

p_1	p_2	p_3	p_4	p_{11}	p_{21}	p_{22}	p_{32}	p_{33}	p_{43}	p_{44}	p_{14}
0	$*$	$*$	0	0	0	0	0	0	0	b	r

$(S(\mathcal{T}), \lhd)$

$t_1^{(1)} \lhd r_1^{(1)} \lhd t_1^{(2)} \lhd r_1^{(2)},$

$t_2^{(1)} \lhd r_2^{(1)} \lhd t_2^{(2)} \lhd r_2^{(2)},$

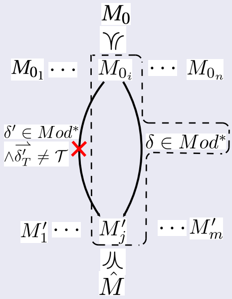
$t_3^{(1)} \lhd r_3^{(1)} \lhd t_3^{(2)} \lhd r_3^{(2)},$

$t_4^{(1)} \lhd r_4^{(1)} \lhd t_4^{(2)}$

A solution for a CPN diagnosis problem

$\overrightarrow{\delta}_T$	
t_1	2
r_1	2
t_2	2
r_2	2
t_3	2
r_3	2
t_4	2
r_4	1

$\overrightarrow{\delta}$	
t_1	2
$r_1.OK$	n_1
$r_1.KO$	n_2
t_2	2
$r_2.OK$	n_3
$r_2.KO$	n_4
t_3	2
$r_3.OK$	n_5
$r_3.KO$	n_6
t_4	2
r_4	1



Covering relation: $* \succcurlyeq b, * \succcurlyeq r$

$$\hat{M} = M_0 + C \times \overrightarrow{\delta} \Longrightarrow \hat{M} \succcurlyeq X_{M_0} + C \times X_{\overrightarrow{\delta}}$$

Inequations system

$\hat{M} \succcurlyeq M_0 + C \times X$

	$\hat{M}$	M_0
p_1	0	*
p_2	*	*
p_3	*	*
p_4	0	*
p_{11}	0	0
p_{21}	0	0
p_{22}	0	0
p_{32}	0	0
p_{33}	0	0
p_{43}	0	0
p_{44}	b	0
p_{14}	r	0

C	t_1	t_2	t_3	t_4	r_1		r_2		r_3		r_4
					OK	KO	OK	KO	OK	KO	
p_1	$-xp_1$			$-xp_1$	$FW(xp_{11})$	r					$FW(xp_{13})$
p_2	$-xp_2$	$-xp_2$			$FW(xp_{21})$	r	$FW(xp_{22})$	r			$FW(xp_{33})$
p_3		$-xp_3$	$-xp_3$				$FW(xp_{32})$	r	$FW(xp_{33})$	r	
p_4			$-xp_4$	$-xp_4$					$FW(xp_{43})$	r	$FW(xp_{44})$
p_{11}	$FW(xp_1)$				$-xp_{11}$	$-xp_{11}$					
p_{21}	$FW(xp_2)$				$-xp_{21}$	$-xp_{21}$					
p_{22}		$FW(xp_2)$					$-xp_{22}$	$-xp_{22}$			
p_{32}		$FW(xp_3)$					$-xp_{32}$	$-xp_{32}$			
p_{33}			$FW(xp_3)$						$-xp_{33}$	$-xp_{33}$	
p_{43}			$FW(xp_4)$						$-xp_{43}$	$-xp_{43}$	
p_{44}				$FW(xp_4)$							$-xp_{44}$
p_{14}				$FW(xp_1)$							$-xp_{14}$

X
t_1 : 2
$r_1.OK$: n_1
$r_1.KO$: n_2
t_2 : 2
$r_2.OK$: n_3
$r_2.KO$: n_4
t_3 : 2
$r_3.OK$: n_5
$r_3.KO$: n_6
t_4 : 2
r_4 : 1
$n_1 + n_2 = 2$
$n_3 + n_4 = 2$
$n_5 + n_6 = 2$

Example (Inequations system)

$$
\begin{cases}
n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1 : 0 \succcurlyeq * - 2`xp_1 - 2`xp_1 + n_1`FW(xp_{11}) + n_2`r + 1`FW(xp_{14}) \\
p_2 : * \succcurlyeq * - 2`xp_2 - 2`xp_2 + n_1`FW(xp_{21}) + n_2`r + n_3`FW(xp_{22}) + n_4`r \\
p_3 : * \succcurlyeq * - 2`xp_3 - 2`xp_3 + n_3`FW(xp_{32}) + n_4`r + n_5`FW(xp_{33}) + n_6`r \\
p_4 : 0 \succcurlyeq * - 2`xp_4 - 2`xp_4 + n_5`FW(xp_{43}) + n_6`r + 1`FW(xp_{44}) \\
p_{11} : 0 \succcurlyeq 0 + 2`FW(xp_1) - n_1`xp_{11} - n_2`xp_{11} \\
p_{21} : 0 \succcurlyeq 0 + 2`FW(xp_2) - n_1`xp_{21} - n_2`xp_{21} \\
p_{22} : 0 \succcurlyeq 0 + 2`FW(xp_2) - n_3`xp_{22} - n_4`xp_{22} \\
p_{32} : 0 \succcurlyeq 0 + 2`FW(xp_3) - n_3`xp_{32} - n_4`xp_{32} \\
p_{33} : 0 \succcurlyeq 0 + 2`FW(xp_3) - n_5`xp_{33} - n_6`xp_{33} \\
p_{43} : 0 \succcurlyeq 0 + 2`FW(xp_4) - n_5`xp_{43} - n_6`xp_{43} \\
p_{44} : b \succcurlyeq 0 + 2`FW(xp_4) - 1`xp_{44} \\
p_{14} : r \succcurlyeq 0 + 2`FW(xp_1) - 1`xp_{14}
\end{cases}
$$

Inequations system

Example (eq_{p_3})

$$p_3 : * \succcurlyeq \; * - 2\text{`}\chi_{p_3} - 2\text{`}\chi_{p_3} + n_3\text{`}FW(\chi_{p_{32}}) + n_4\text{`}r + n_5\text{`}FW(\chi_{p_{33}}) + n_6\text{`}r$$

Example (Inequations system)

$$\begin{cases}
n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1 : 0 \succcurlyeq * - 2\text{`}\chi_{p_1} - 2\text{`}\chi_{p_1} + n_1\text{`}FW(\chi_{p_{11}}) + n_2\text{`}r + 1\text{`}FW(\chi_{p_{14}}) \\
p_2 : * \succcurlyeq * - 2\text{`}\chi_{p_2} - 2\text{`}\chi_{p_2} + n_1\text{`}FW(\chi_{p_{21}}) + n_2\text{`}r + n_3\text{`}FW(\chi_{p_{22}}) + n_4\text{`}r \\
p_3 : * \succcurlyeq * - 2\text{`}\chi_{p_3} - 2\text{`}\chi_{p_3} + n_3\text{`}FW(\chi_{p_{32}}) + n_4\text{`}r + n_5\text{`}FW(\chi_{p_{33}}) + n_6\text{`}r \\
p_4 : 0 \succcurlyeq * - 2\text{`}\chi_{p_4} - 2\text{`}\chi_{p_4} + n_5\text{`}FW(\chi_{p_{43}}) + n_6\text{`}r + 1\text{`}FW(\chi_{p_{44}}) \\
p_{11} : 0 \succcurlyeq 0 + 2\text{`}FW(\chi_{p_1}) - n_1\text{`}\chi_{p_{11}} - n_2\text{`}\chi_{p_{11}} \\
p_{21} : 0 \succcurlyeq 0 + 2\text{`}FW(\chi_{p_2}) - n_1\text{`}\chi_{p_{21}} - n_2\text{`}\chi_{p_{21}} \\
p_{22} : 0 \succcurlyeq 0 + 2\text{`}FW(\chi_{p_2}) - n_3\text{`}\chi_{p_{22}} - n_4\text{`}\chi_{p_{22}} \\
p_{32} : 0 \succcurlyeq 0 + 2\text{`}FW(\chi_{p_3}) - n_3\text{`}\chi_{p_{32}} - n_4\text{`}\chi_{p_{32}} \\
p_{33} : 0 \succcurlyeq 0 + 2\text{`}FW(\chi_{p_3}) - n_5\text{`}\chi_{p_{33}} - n_6\text{`}\chi_{p_{33}} \\
p_{43} : 0 \succcurlyeq 0 + 2\text{`}FW(\chi_{p_4}) - n_5\text{`}\chi_{p_{43}} - n_6\text{`}\chi_{p_{43}} \\
p_{44} : b \succcurlyeq 0 + 2\text{`}FW(\chi_{p_4}) - 1\text{`}\chi_{p_{44}} \\
p_{14} : r \succcurlyeq 0 + 2\text{`}FW(\chi_{p_1}) - 1\text{`}\chi_{p_{14}}
\end{cases}$$

Diagnosis algorithm: single fault

Sketch

1. to propagate with *b* tokens
2. to propagate with *r* tokens
3. to integrate results

Example (Inequations system)

$$
\begin{cases}
n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1:\ 0 \succcurlyeq * - 2`xp_1 - 2`xp_1 + n_1`FW(xp_{11}) + n_2`r + 1`FW(xp_{14}) \\
p_2:\ * \succcurlyeq * - 2`xp_2 - 2`xp_2 + n_1`FW(xp_{21}) + n_2`r + n_3`FW(xp_{22}) + n_4`r \\
p_3:\ * \succcurlyeq * - 2`xp_3 - 2`xp_3 + n_3`FW(xp_{32}) + n_4`r + n_5`FW(xp_{33}) + n_6`r \\
p_4:\ 0 \succcurlyeq * - 2`xp_4 - 2`xp_4 + n_5`FW(xp_{43}) + n_6`r + 1`FW(xp_{44}) \\
p_{11}:\ 0 \succcurlyeq 0 + 2`FW(xp_1) - n_1`xp_{11} - n_2`xp_{11} \\
p_{21}:\ 0 \succcurlyeq 0 + 2`FW(xp_2) - n_1`xp_{21} - n_2`xp_{21} \\
p_{22}:\ 0 \succcurlyeq 0 + 2`FW(xp_2) - n_3`xp_{22} - n_4`xp_{22} \\
p_{32}:\ 0 \succcurlyeq 0 + 2`FW(xp_3) - n_3`xp_{32} - n_4`xp_{32} \\
p_{33}:\ 0 \succcurlyeq 0 + 2`FW(xp_3) - n_5`xp_{33} - n_6`xp_{33} \\
p_{43}:\ 0 \succcurlyeq 0 + 2`FW(xp_4) - n_5`xp_{43} - n_6`xp_{43} \\
p_{44}:\ b \succcurlyeq 0 + 2`FW(xp_4) - 1`xp_{44} \\
p_{14}:\ r \succcurlyeq 0 + 2`FW(xp_1) - 1`xp_{14}
\end{cases}
$$

Diagnosis algorithm: single fault

Sketch

1. to propagate with *b* tokens
2. to propagate with *r* tokens
3. to integrate results

Example (Inequations system)

$$
\begin{cases}
n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1:\ 0 \succcurlyeq * - 2`xp_1 - 2`xp_1 + n_1`FW(xp_{11}) + n_2`r + 1`FW(xp_{14}) \\
p_2:\ * \succcurlyeq * - 2`xp_2 - 2`xp_2 + n_1`FW(xp_{21}) + n_2`r + n_3`FW(xp_{22}) + n_4`r \\
p_3:\ * \succcurlyeq * - 2`xp_3 - 2`xp_3 + n_3`FW(xp_{32}) + n_4`r + n_5`FW(xp_{33}) + n_6`r \\
p_4:\ 0 \succcurlyeq * - 2`xp_4 - 2`xp_4 + n_5`FW(xp_{43}) + n_6`r + 1`FW(xp_{44}) \\
p_{11}:\ 0 \succcurlyeq 0 + 2`FW(xp_1) - n_1`xp_{11} - n_2`xp_{11} \\
p_{21}:\ 0 \succcurlyeq 0 + 2`FW(xp_2) - n_1`xp_{21} - n_2`xp_{21} \\
p_{22}:\ 0 \succcurlyeq 0 + 2`FW(xp_2) - n_3`xp_{22} - n_4`xp_{22} \\
p_{32}:\ 0 \succcurlyeq 0 + 2`FW(xp_3) - n_3`xp_{32} - n_4`xp_{32} \\
p_{33}:\ 0 \succcurlyeq 0 + 2`FW(xp_3) - n_5`xp_{33} - n_6`xp_{33} \\
p_{43}:\ 0 \succcurlyeq 0 + 2`FW(xp_4) - n_5`xp_{43} - n_6`xp_{43} \\
p_{44}:\ b \succcurlyeq 0 + 2`FW(xp_4) - 1`xp_{44} \\
p_{14}:\ r \succcurlyeq 0 + 2`FW(xp_1) - 1`xp_{14}
\end{cases}
$$

Example ($xp_{44} = b$ as a constraint)

$$p_{44} = b$$

Diagnosis algorithm: single fault

Sketch

1. to propagate with *b* tokens
2. to propagate with *r* tokens
3. to integrate results

Example (Inequations system)

$$
\begin{cases}
n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1: \; 0 \succcurlyeq * - 2'\chi_{p_1} - 2'\chi_{p_1} + n_1' FW(\chi_{p_{11}}) + n_2' r + 1' FW(\chi_{p_{14}}) \\
p_2: \; * \succcurlyeq * - 2'\chi_{p_2} - 2'\chi_{p_2} + n_1' FW(\chi_{p_{21}}) + n_2' r + n_3' FW(\chi_{p_{22}}) + n_4' r \\
p_3: \; * \succcurlyeq * - 2'\chi_{p_3} - 2'\chi_{p_3} + n_3' FW(\chi_{p_{32}}) + n_4' r + n_5' FW(\chi_{p_{33}}) + n_6' r \\
p_4: \; 0 \succcurlyeq * - 2'\chi_{p_4} - 2'\chi_{p_4} + n_5' FW(\chi_{p_{43}}) + n_6' r + 1' FW(\chi_{p_{44}}) \\
p_{11}: \; 0 \succcurlyeq 0 + 2' FW(\chi_{p_1}) - n_1' \chi_{p_{11}} - n_2' \chi_{p_{11}} \\
p_{21}: \; 0 \succcurlyeq 0 + 2' FW(\chi_{p_2}) - n_1' \chi_{p_{21}} - n_2' \chi_{p_{21}} \\
p_{22}: \; 0 \succcurlyeq 0 + 2' FW(\chi_{p_2}) - n_3' \chi_{p_{22}} - n_4' \chi_{p_{22}} \\
p_{32}: \; 0 \succcurlyeq 0 + 2' FW(\chi_{p_3}) - n_3' \chi_{p_{32}} - n_4' \chi_{p_{32}} \\
p_{33}: \; 0 \succcurlyeq 0 + 2' FW(\chi_{p_3}) - n_5' \chi_{p_{33}} - n_6' \chi_{p_{33}} \\
p_{43}: \; 0 \succcurlyeq 0 + 2' FW(\chi_{p_4}) - n_5' \chi_{p_{43}} - n_6' \chi_{p_{43}} \\
p_{44}: \; b \succcurlyeq 0 + 2' FW(\chi_{p_4}) - 1' \chi_{p_{44}} \\
p_{14}: \; r \succcurlyeq 0 + 2' FW(\chi_{p_1}) - 1' \chi_{p_{14}}
\end{cases}
$$

Example ($\chi_{p_{44}} = b$ as a constraint)

$$p_{44} = b$$

$$\downarrow$$

$$p_4 = b$$

Diagnosis algorithm: single fault

Sketch

1. to propagate with *b* tokens
2. to propagate with *r* tokens
3. to integrate results

Example (Inequations system)

$$
\begin{cases}
n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1:\ 0 \succcurlyeq * - 2`\chi p_1 - 2`\chi p_1 + n_1`FW(\chi p_{11}) + n_2`r + 1`FW(\chi p_{14}) \\
p_2:\ * \succcurlyeq * - 2`\chi p_2 - 2`\chi p_2 + n_1`FW(\chi p_{21}) + n_2`r + n_3`FW(\chi p_{22}) + n_4`r \\
p_3:\ * \succcurlyeq * - 2`\chi p_3 - 2`\chi p_3 + n_3`FW(\chi p_{32}) + n_4`r + n_5`FW(\chi p_{33}) + n_6`r \\
p_4:\ 0 \succcurlyeq * - 2`\chi p_4 - 2`\chi p_4 + n_5`FW(\chi p_{43}) + n_6`r + 1`FW(\chi p_{44}) \\
p_{11}:\ 0 \succcurlyeq 0 + 2`FW(\chi p_1) - n_1`\chi p_{11} - n_2`\chi p_{11} \\
p_{21}:\ 0 \succcurlyeq 0 + 2`FW(\chi p_2) - n_1`\chi p_{21} - n_2`\chi p_{21} \\
p_{22}:\ 0 \succcurlyeq 0 + 2`FW(\chi p_2) - n_3`\chi p_{22} - n_4`\chi p_{22} \\
p_{32}:\ 0 \succcurlyeq 0 + 2`FW(\chi p_3) - n_3`\chi p_{32} - n_4`\chi p_{32} \\
p_{33}:\ 0 \succcurlyeq 0 + 2`FW(\chi p_3) - n_5`\chi p_{33} - n_6`\chi p_{33} \\
p_{43}:\ 0 \succcurlyeq 0 + 2`FW(\chi p_4) - n_5`\chi p_{43} - n_6`\chi p_{43} \\
p_{44}:\ b \succcurlyeq 0 + 2`FW(\chi p_4) - 1`\chi p_{44} \\
p_{14}:\ r \succcurlyeq 0 + 2`FW(\chi p_1) - 1`\chi p_{14}
\end{cases}
$$

Example ($\chi p_{44} = b$ as a constraint)

$p_{44} = b$

↓

$p_4 = b$

↓

$p_{43} = b \wedge n_6 = 0$

Diagnosis algorithm: single fault

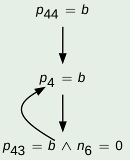
Sketch

1. to propagate with *b* tokens
2. to propagate with *r* tokens
3. to integrate results

Example (Inequations system)

$$
\left\{
\begin{array}{ll}
 & n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1: & 0 \succcurlyeq * - 2`\chi p_1 - 2`\chi p_1 + n_1`FW(\chi p_{11}) + n_2`r + 1`FW(\chi p_{14}) \\
p_2: & * \succcurlyeq * - 2`\chi p_2 - 2`\chi p_2 + n_1`FW(\chi p_{21}) + n_2`r + n_3`FW(\chi p_{22}) + n_4`r \\
p_3: & * \succcurlyeq * - 2`\chi p_3 - 2`\chi p_3 + n_3`FW(\chi p_{32}) + n_4`r + n_5`FW(\chi p_{33}) + n_6`r \\
p_4: & 0 \succcurlyeq * - 2`\chi p_4 - 2`\chi p_4 + n_5`FW(\chi p_{43}) + n_6`r + 1`FW(\chi p_{44}) \\
p_{11}: & 0 \succcurlyeq 0 + 2`FW(\chi p_1) - n_1`\chi p_{11} - n_2`\chi p_{11} \\
p_{21}: & 0 \succcurlyeq 0 + 2`FW(\chi p_2) - n_1`\chi p_{21} - n_2`\chi p_{21} \\
p_{22}: & 0 \succcurlyeq 0 + 2`FW(\chi p_2) - n_3`\chi p_{22} - n_4`\chi p_{22} \\
p_{32}: & 0 \succcurlyeq 0 + 2`FW(\chi p_3) - n_3`\chi p_{32} - n_4`\chi p_{32} \\
p_{33}: & 0 \succcurlyeq 0 + 2`FW(\chi p_3) - n_5`\chi p_{33} - n_6`\chi p_{33} \\
p_{43}: & 0 \succcurlyeq 0 + 2`FW(\chi p_4) - n_5`\chi p_{43} - n_6`\chi p_{43} \\
p_{44}: & b \succcurlyeq 0 + 2`FW(\chi p_4) - 1`\chi p_{44} \\
p_{14}: & r \succcurlyeq 0 + 2`FW(\chi p_1) - 1`\chi p_{14}
\end{array}
\right.
$$

Example ($\chi p_{44} = b$ as a constraint)



Diagnosis algorithm: single fault

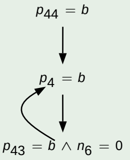
Sketch

1. to propagate with *b* tokens
2. to propagate with *r* tokens
3. to integrate results

Example (Inequations system)

$$
\begin{cases}
n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1:\ 0 \succcurlyeq * - 2\text{`}\chi p_1 - 2\text{`}\chi p_1 + n_1\text{`}FW(\chi p_{11}) + n_2\text{`}r + 1\text{`}FW(\chi p_{14}) \\
p_2:\ * \succcurlyeq * - 2\text{`}\chi p_2 - 2\text{`}\chi p_2 + n_1\text{`}FW(\chi p_{21}) + n_2\text{`}r + n_3\text{`}FW(\chi p_{22}) + n_4\text{`}r \\
p_3:\ * \succcurlyeq * - 2\text{`}\chi p_3 - 2\text{`}\chi p_3 + n_3\text{`}FW(\chi p_{32}) + n_4\text{`}r + n_5\text{`}FW(\chi p_{33}) + n_6\text{`}r \\
p_4:\ 0 \succcurlyeq * - 2\text{`}\chi p_4 - 2\text{`}\chi p_4 + n_5\text{`}FW(\chi p_{43}) + n_6\text{`}r + 1\text{`}FW(\chi p_{44}) \\
p_{11}:\ 0 \succcurlyeq 0 + 2\text{`}FW(\chi p_1) - n_1\text{`}\chi p_{11} - n_2\text{`}\chi p_{11} \\
p_{21}:\ 0 \succcurlyeq 0 + 2\text{`}FW(\chi p_2) - n_1\text{`}\chi p_{21} - n_2\text{`}\chi p_{21} \\
p_{22}:\ 0 \succcurlyeq 0 + 2\text{`}FW(\chi p_2) - n_3\text{`}\chi p_{22} - n_4\text{`}\chi p_{22} \\
p_{32}:\ 0 \succcurlyeq 0 + 2\text{`}FW(\chi p_3) - n_3\text{`}\chi p_{32} - n_4\text{`}\chi p_{32} \\
p_{33}:\ 0 \succcurlyeq 0 + 2\text{`}FW(\chi p_3) - n_5\text{`}\chi p_{33} - n_6\text{`}\chi p_{33} \\
p_{43}:\ 0 \succcurlyeq 0 + 2\text{`}FW(\chi p_4) - n_5\text{`}\chi p_{43} - n_6\text{`}\chi p_{43} \\
p_{44}:\ b \succcurlyeq 0 + 2\text{`}FW(\chi p_4) - 1\text{`}\chi p_{44} \\
p_{14}:\ r \succcurlyeq 0 + 2\text{`}FW(\chi p_1) - 1\text{`}\chi p_{14}
\end{cases}
$$

Example ($\chi p_{44} = b$ as a constraint)



Example ($\chi p_{14} = r$)

$p_{14} = r$

Diagnosis algorithm: single fault

Sketch

1. to propagate with *b* tokens
2. to propagate with *r* tokens
3. to integrate results

Example (Inequations system)

$$
\begin{cases}
n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1: \; 0 \succcurlyeq * - 2`\chi_{p_1} - 2`\chi_{p_1} + n_1`FW(\chi_{p_{11}}) + n_2`r + 1`FW(\chi_{p_{14}}) \\
p_2: \; * \succcurlyeq * - 2`\chi_{p_2} - 2`\chi_{p_2} + n_1`FW(\chi_{p_{21}}) + n_2`r + n_3`FW(\chi_{p_{22}}) + n_4`r \\
p_3: \; * \succcurlyeq * - 2`\chi_{p_3} - 2`\chi_{p_3} + n_3`FW(\chi_{p_{32}}) + n_4`r + n_5`FW(\chi_{p_{33}}) + n_6`r \\
p_4: \; 0 \succcurlyeq * - 2`\chi_{p_4} - 2`\chi_{p_4} + n_5`FW(\chi_{p_{43}}) + n_6`r + 1`FW(\chi_{p_{44}}) \\
p_{11}: \; 0 \succcurlyeq 0 + 2`FW(\chi_{p_1}) - n_1`\chi_{p_{11}} - n_2`\chi_{p_{11}} \\
p_{21}: \; 0 \succcurlyeq 0 + 2`FW(\chi_{p_2}) - n_1`\chi_{p_{21}} - n_2`\chi_{p_{21}} \\
p_{22}: \; 0 \succcurlyeq 0 + 2`FW(\chi_{p_2}) - n_3`\chi_{p_{22}} - n_4`\chi_{p_{22}} \\
p_{32}: \; 0 \succcurlyeq 0 + 2`FW(\chi_{p_3}) - n_3`\chi_{p_{32}} - n_4`\chi_{p_{32}} \\
p_{33}: \; 0 \succcurlyeq 0 + 2`FW(\chi_{p_3}) - n_5`\chi_{p_{33}} - n_6`\chi_{p_{33}} \\
p_{43}: \; 0 \succcurlyeq 0 + 2`FW(\chi_{p_4}) - n_5`\chi_{p_{43}} - n_6`\chi_{p_{43}} \\
p_{44}: \; b \succcurlyeq 0 + 2`FW(\chi_{p_4}) - 1`\chi_{p_{44}} \\
p_{14}: \; r \succcurlyeq 0 + 2`FW(\chi_{p_1}) - 1`\chi_{p_{14}}
\end{cases}
$$

Example ($\chi_{p_{44}} = b$ as a constraint)

$p_{44} = b$

↓

$p_4 = b$

↓

$p_{43} = b \wedge n_6 = 0$

Example ($\chi_{p_{14}} = r$)

$p_{14} = r$

↓

$p_1 = r$

Diagnosis algorithm: single fault

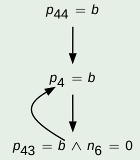
Sketch

1. to propagate with *b* tokens
2. to propagate with *r* tokens
3. to integrate results

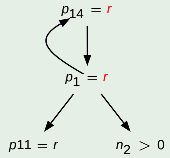
Example (Inequations system)

$$
\left\{
\begin{array}{ll}
 & n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1 : & 0 \succcurlyeq * - 2^{\backprime}\chi p_1 - 2^{\backprime}\chi p_1 + n_1^{\backprime}FW(\chi p_{11}) + n_2^{\backprime}r + 1^{\backprime}FW(\chi p_{14}) \\
p_2 : & * \succcurlyeq * - 2^{\backprime}\chi p_2 - 2^{\backprime}\chi p_2 + n_1^{\backprime}FW(\chi p_{21}) + n_2^{\backprime}r + n_3^{\backprime}FW(\chi p_{22}) + n_4^{\backprime}r \\
p_3 : & * \succcurlyeq * - 2^{\backprime}\chi p_3 - 2^{\backprime}\chi p_3 + n_3^{\backprime}FW(\chi p_{32}) + n_4^{\backprime}r + n_5^{\backprime}FW(\chi p_{33}) + n_6^{\backprime}r \\
p_4 : & 0 \succcurlyeq * - 2^{\backprime}\chi p_4 - 2^{\backprime}\chi p_4 + n_5^{\backprime}FW(\chi p_{43}) + n_6^{\backprime}r + 1^{\backprime}FW(\chi p_{44}) \\
p_{11} : & 0 \succcurlyeq 0 + 2^{\backprime}FW(\chi p_1) - n_1^{\backprime}\chi p_{11} - n_2^{\backprime}\chi p_{11} \\
p_{21} : & 0 \succcurlyeq 0 + 2^{\backprime}FW(\chi p_2) - n_1^{\backprime}\chi p_{21} - n_2^{\backprime}\chi p_{21} \\
p_{22} : & 0 \succcurlyeq 0 + 2^{\backprime}FW(\chi p_2) - n_3^{\backprime}\chi p_{22} - n_4^{\backprime}\chi p_{22} \\
p_{32} : & 0 \succcurlyeq 0 + 2^{\backprime}FW(\chi p_3) - n_3^{\backprime}\chi p_{32} - n_4^{\backprime}\chi p_{32} \\
p_{33} : & 0 \succcurlyeq 0 + 2^{\backprime}FW(\chi p_3) - n_5^{\backprime}\chi p_{33} - n_6^{\backprime}\chi p_{33} \\
p_{43} : & 0 \succcurlyeq 0 + 2^{\backprime}FW(\chi p_4) - n_5^{\backprime}\chi p_{43} - n_6^{\backprime}\chi p_{43} \\
p_{44} : & b \succcurlyeq 0 + 2^{\backprime}FW(\chi p_4) - 1^{\backprime}\chi p_{44} \\
p_{14} : & r \succcurlyeq 0 + 2^{\backprime}FW(\chi p_1) - 1^{\backprime}\chi p_{14}
\end{array}
\right.
$$

Example ($\chi p_{44} = b$ as a constraint)



Example ($\chi p_{14} = r$)



Diagnosis algorithm: single fault

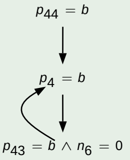
Sketch

1. to propagate with *b* tokens
2. to propagate with *r* tokens
3. to integrate results

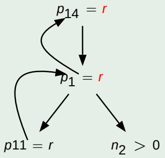
Example (Inequations system)

$$
\begin{cases}
n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1 : 0 \succcurlyeq * - 2`\chi p_1 - 2`\chi p_1 + n_1`FW(\chi p_{11}) + n_2`r + 1`FW(\chi p_{14}) \\
p_2 : * \succcurlyeq * - 2`\chi p_2 - 2`\chi p_2 + n_1`FW(\chi p_{21}) + n_2`r + n_3`FW(\chi p_{22}) + n_4`r \\
p_3 : * \succcurlyeq * - 2`\chi p_3 - 2`\chi p_3 + n_3`FW(\chi p_{32}) + n_4`r + n_5`FW(\chi p_{33}) + n_6`r \\
p_4 : 0 \succcurlyeq * - 2`\chi p_4 - 2`\chi p_4 + n_5`FW(\chi p_{43}) + n_6`r + 1`FW(\chi p_{44}) \\
p_{11} : 0 \succcurlyeq 0 + 2`FW(\chi p_1) - n_1`\chi p_{11} - n_2`\chi p_{11} \\
p_{21} : 0 \succcurlyeq 0 + 2`FW(\chi p_2) - n_1`\chi p_{21} - n_2`\chi p_{21} \\
p_{22} : 0 \succcurlyeq 0 + 2`FW(\chi p_2) - n_3`\chi p_{22} - n_4`\chi p_{22} \\
p_{32} : 0 \succcurlyeq 0 + 2`FW(\chi p_3) - n_3`\chi p_{32} - n_4`\chi p_{32} \\
p_{33} : 0 \succcurlyeq 0 + 2`FW(\chi p_3) - n_5`\chi p_{33} - n_6`\chi p_{33} \\
p_{43} : 0 \succcurlyeq 0 + 2`FW(\chi p_4) - n_5`\chi p_{43} - n_6`\chi p_{43} \\
p_{44} : b \succcurlyeq 0 + 2`FW(\chi p_4) - 1`\chi p_{44} \\
p_{14} : r \succcurlyeq 0 + 2`FW(\chi p_1) - 1`\chi p_{14}
\end{cases}
$$

Example ($\chi p_{44} = b$ as a constraint)



Example ($\chi p_{14} = r$)



Diagnosis algorithm: single fault

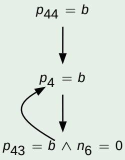
Sketch

1. to propagate with b tokens
2. to propagate with r tokens
3. to integrate results

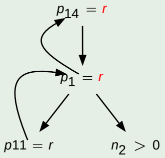
Example (Inequations system)

$$
\begin{cases}
n_1 + n_2 = 2 \quad n_3 + n_4 = 2 \quad n_5 + n_6 = 1 \\
p_1 : 0 \succcurlyeq * - 2^{\backprime}\chi p_1 - 2^{\backprime}\chi p_1 + n_1^{\backprime} FW(\chi p_{11}) + n_2^{\backprime} r + 1^{\backprime} FW(\chi p_{14}) \\
p_2 : * \succcurlyeq * - 2^{\backprime}\chi p_2 - 2^{\backprime}\chi p_2 + n_1^{\backprime} FW(\chi p_{21}) + n_2^{\backprime} r + n_3^{\backprime} FW(\chi p_{22}) + n_4^{\backprime} r \\
p_3 : * \succcurlyeq * - 2^{\backprime}\chi p_3 - 2^{\backprime}\chi p_3 + n_3^{\backprime} FW(\chi p_{32}) + n_4^{\backprime} r + n_5^{\backprime} FW(\chi p_{33}) + n_6^{\backprime} r \\
p_4 : 0 \succcurlyeq * - 2^{\backprime}\chi p_4 - 2^{\backprime}\chi p_4 + n_5^{\backprime} FW(\chi p_{43}) + n_6^{\backprime} r + 1^{\backprime} FW(\chi p_{44}) \\
p_{11} : 0 \succcurlyeq 0 + 2^{\backprime} FW(\chi p_1) - n_1^{\backprime}\chi p_{11} - n_2^{\backprime}\chi p_{11} \\
p_{21} : 0 \succcurlyeq 0 + 2^{\backprime} FW(\chi p_2) - n_1^{\backprime}\chi p_{21} - n_2^{\backprime}\chi p_{21} \\
p_{22} : 0 \succcurlyeq 0 + 2^{\backprime} FW(\chi p_2) - n_3^{\backprime}\chi p_{22} - n_4^{\backprime}\chi p_{22} \\
p_{32} : 0 \succcurlyeq 0 + 2^{\backprime} FW(\chi p_3) - n_3^{\backprime}\chi p_{32} - n_4^{\backprime}\chi p_{32} \\
p_{33} : 0 \succcurlyeq 0 + 2^{\backprime} FW(\chi p_3) - n_5^{\backprime}\chi p_{33} - n_6^{\backprime}\chi p_{33} \\
p_{43} : 0 \succcurlyeq 0 + 2^{\backprime} FW(\chi p_4) - n_5^{\backprime}\chi p_{43} - n_6^{\backprime}\chi p_{43} \\
p_{44} : b \succcurlyeq 0 + 2^{\backprime} FW(\chi p_4) - 1^{\backprime}\chi p_{44} \\
p_{14} : r \succcurlyeq 0 + 2^{\backprime} FW(\chi p_1) - 1^{\backprime}\chi p_{14}
\end{cases}
$$

Example ($\chi p_{44} = b$ as a constraint)



Example ($\chi p_{14} = r$)



Example (Diagnosis)

$D_1 = \{\{p_1\}, \{r_1.KO\}\}$

Advantages: cyclic observations handling

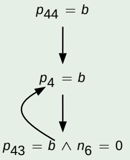
Sketch

1. to propagate with b tokens
2. to propagate with r tokens
3. to integrate result

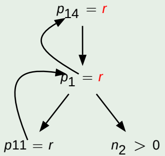
Example (Inequations system)

$$
\left\{
\begin{array}{l}
n_1 + n_2 = 1000 \quad n_3 + n_4 = 1000 \quad n_5 + n_6 = 999 \\
p_1 : 0 \succcurlyeq * - 1000^{\backprime} \chi p_1 - 1000^{\backprime} \chi p_1 + n_1^{\backprime} FW(\chi p_{11}) + n_2^{\backprime} r + 999^{\backprime} FW(\chi p_{14}) \\
p_2 : * \succcurlyeq * - 1000^{\backprime} \chi p_2 - 1000^{\backprime} \chi p_2 + n_1^{\backprime} FW(\chi p_{21}) + n_2^{\backprime} r + n_3^{\backprime} FW(\chi p_{22}) + n_4^{\backprime} r \\
p_3 : * \succcurlyeq * - 1000^{\backprime} \chi p_3 - 1000^{\backprime} \chi p_3 + n_3^{\backprime} FW(\chi p_{32}) + n_4^{\backprime} r + n_5^{\backprime} FW(\chi p_{33}) + n_6^{\backprime} r \\
p_4 : 0 \succcurlyeq * - 1000^{\backprime} \chi p_4 - 1000^{\backprime} \chi p_4 + n_5^{\backprime} FW(\chi p_{43}) + n_6^{\backprime} r + 999^{\backprime} FW(\chi p_{44}) \\
p_{11} : 0 \succcurlyeq 0 + 1000^{\backprime} FW(\chi p_1) - n_1^{\backprime} \chi p_{11} - n_2^{\backprime} \chi p_{11} \\
p_{21} : 0 \succcurlyeq 0 + 1000^{\backprime} FW(\chi p_2) - n_1^{\backprime} \chi p_{21} - n_1^{\backprime} \chi p_{21} \\
p_{22} : 0 \succcurlyeq 0 + 1000^{\backprime} FW(\chi p_2) - n_3^{\backprime} \chi p_{22} - n_4^{\backprime} \chi p_{22} \\
p_{32} : 0 \succcurlyeq 0 + 1000^{\backprime} FW(\chi p_3) - n_3^{\backprime} \chi p_{32} - n_4^{\backprime} \chi p_{32} \\
p_{33} : 0 \succcurlyeq 0 + 1000^{\backprime} FW(\chi p_3) - n_5^{\backprime} \chi p_{33} - n_6^{\backprime} \chi p_{33} \\
p_{43} : 0 \succcurlyeq 0 + 1000^{\backprime} FW(\chi p_4) - n_5^{\backprime} \chi p_{43} - n_6^{\backprime} \chi p_{43} \\
p_{44} : b \succcurlyeq 0 + 1000^{\backprime} FW(\chi p_4) - 999^{\backprime} \chi p_{44} \\
p_{14} : r \succcurlyeq 0 + 1000^{\backprime} FW(\chi p_1) - 999^{\backprime} \chi p_{14}
\end{array}
\right.
$$

Example ($\chi p_{44} = b$ as a constraint)



Example ($\chi p_{14} = r$)



Example (Diagnosis)

$D_1 = \{\{p_1\}, \{r_1.KO\}\}$

Multi faults diagnosis (Li et al. [2009a])

Cartesian-union operator $\overset{\cup}{\times}$

$\overset{\cup}{\times}$ is an operator that calculates the Cartesian product and then keeps the minimal subsets.

Multi faults diagnosis (Li et al. [2009a])

Cartesian-union operator $\overset{\cup}{\times}$

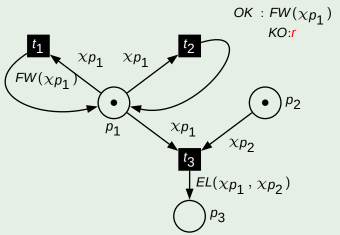
$\overset{\cup}{\times}$ is an operator that calculates the Cartesian product and then keeps the minimal subsets.

Example

- D_1=$\{\{p_1\}, \{r_1.KO\}\}$
- Suppose the 4th philosopher finds faulty cutlery on both sides: a new symptom on $p_{44} : \chi_{p_{44}} = r \Rightarrow D_2 = \{\{p_4\}, \{r_3.KO\}\}$
- D=$D_1 \overset{\cup}{\times} \mathrm{D}_2$=$\{\{p_1, p_4\}, \{p_1, r_3.KO\}, \{r_1.KO, p_4\}, \{r_1.KO, r_3.KO\}\}$

Opposite example for minimal diagnosis

CPN model

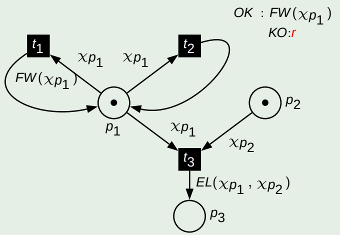


Opposite example for minimal diagnosis

Example $(S(\mathcal{T}), \lhd)$

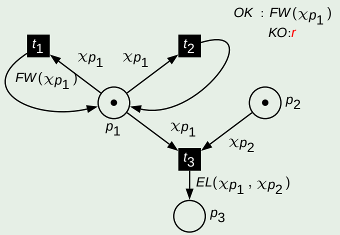
- $S(\mathcal{T}) = \{t_1, t_2, t_2\}$
- $t_1 \lhd t_3$, $t_2 \lhd t_3$

CPN model



Opposite example for minimal diagnosis

CPN model



Example $(S(\mathcal{T}), \lhd)$

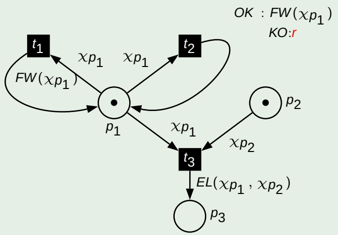
- $S(\mathcal{T}) = \{t_1, t_2, t_2\}$
- $t_1 \lhd t_3$, $t_2 \lhd t_3$

Example $(\overrightarrow{\delta}_T)$

t_1	t_2	t_3
1	1	1

Opposite example for minimal diagnosis

CPN model



Example $(S(\mathcal{T}), \lhd)$

- $S(\mathcal{T}) = \{t_1, t_2, t_2\}$
- $t_1 \lhd t_3, t_2 \lhd t_3$

Example $(\overrightarrow{\delta}_T)$

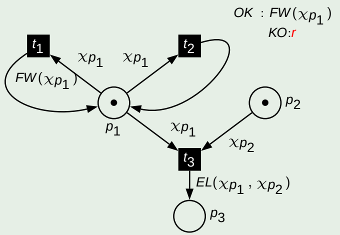
t_1	t_2	t_3
1	1	1

Example (Two possible observation traces)

1. $t_1\ t_2\ t_3$: $t_2.KO$ can transmit the fault by r token
2. $t_2\ t_1\ t_3$: the effect of $t_2.KO$ is overwriten by t_1

Opposite example for minimal diagnosis

CPN model



Example ($(S(\mathcal{T}), \lhd)$)

- $S(\mathcal{T}) = \{t_1, t_2, t_2\}$
- $t_1 \lhd t_3$, $t_2 \lhd t_3$

Example ($\overrightarrow{\delta}_T$)

t_1	t_2	t_3
1	1	1

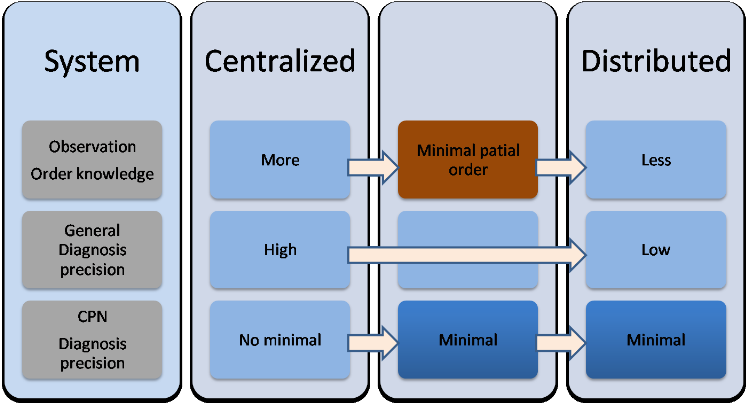
Example (Two possible observation traces)

1. $t_1\ t_2\ t_3$: $t_2.KO$ can transmit the fault by r token
2. $t_2\ t_1\ t_3$: the effect of $t_2.KO$ is overwriten by t_1

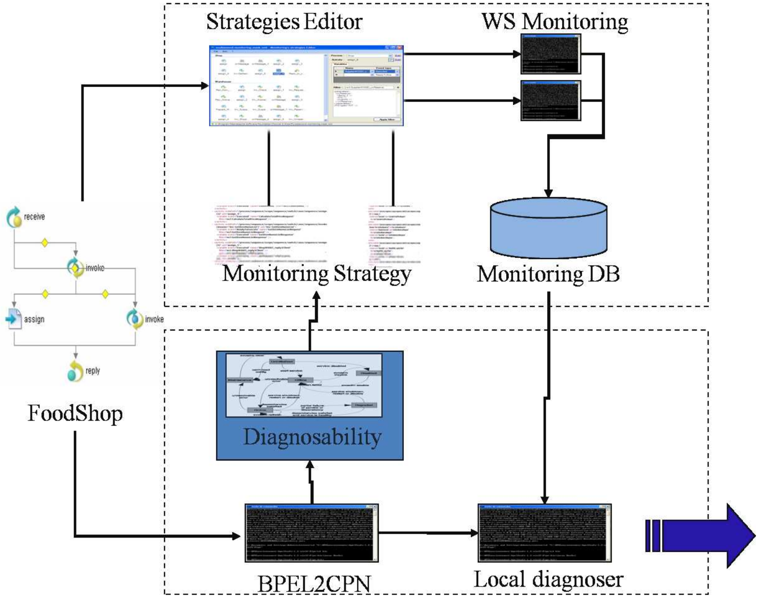
Diagnosis D

$D = \{\{p_2\}, \{t_2.KO\}\}$ is only minimal only for trace 1

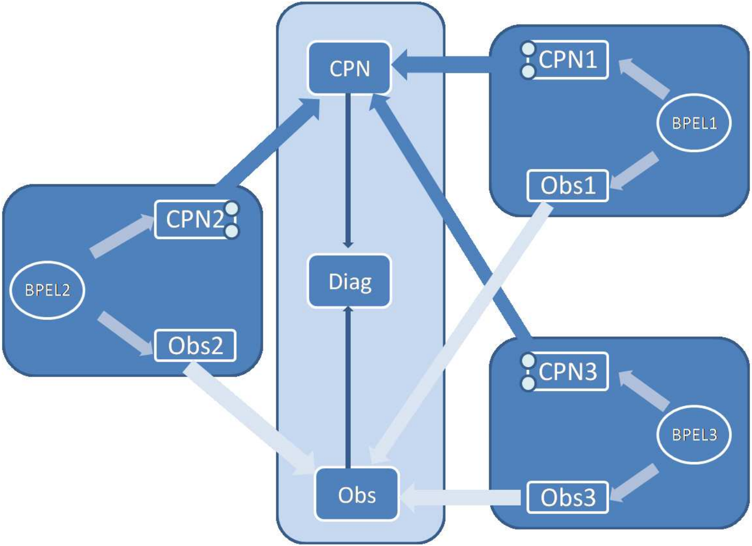
Diagnosis minimality vs. precision



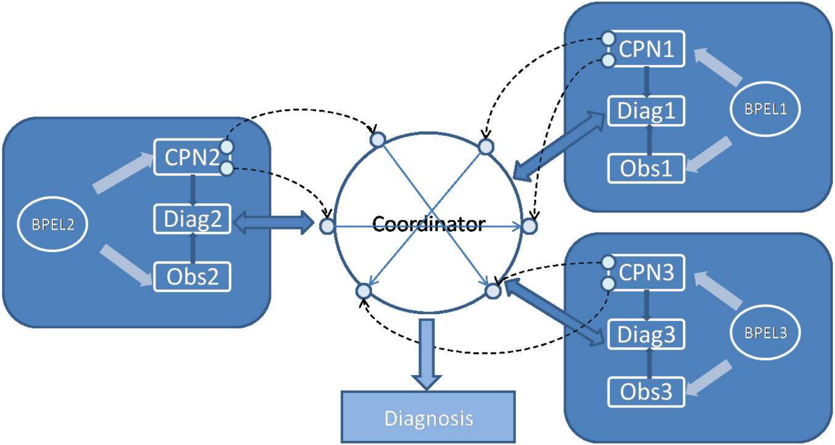
Architecture of BPEL Monitoring and Diagnosis (WSDIAMOND)



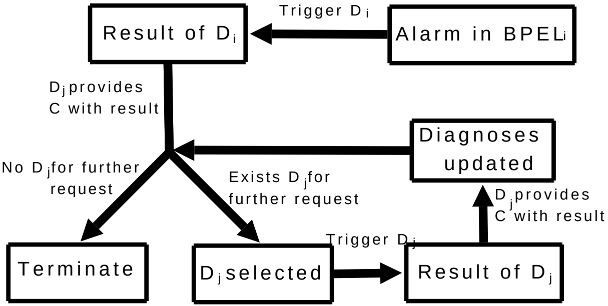
Centralized diagnosis solution



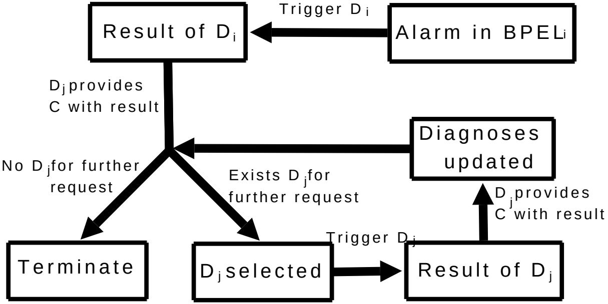
Decentralized diagnosis architecture (Li et al. [2009b])



Decentralized coordination protocol (Zaitsev [2005])



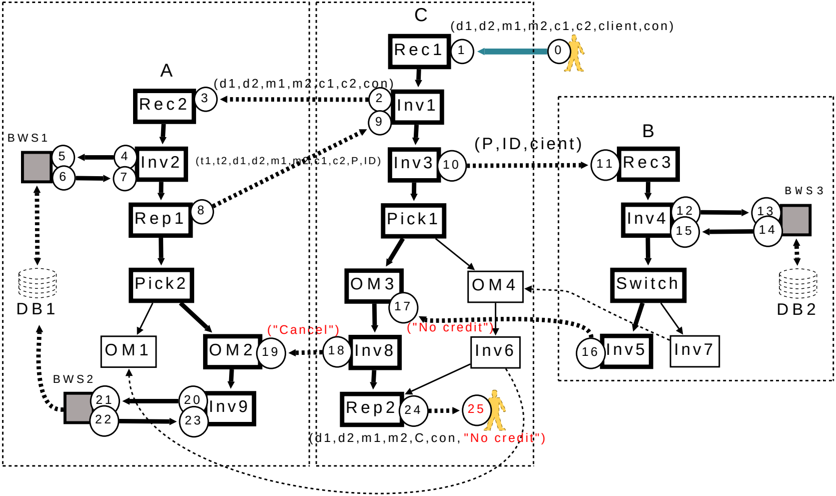
Decentralized coordination protocol (Zaitsev [2005])



Equivalence of global and decentralized diagnosis (theorem 2)

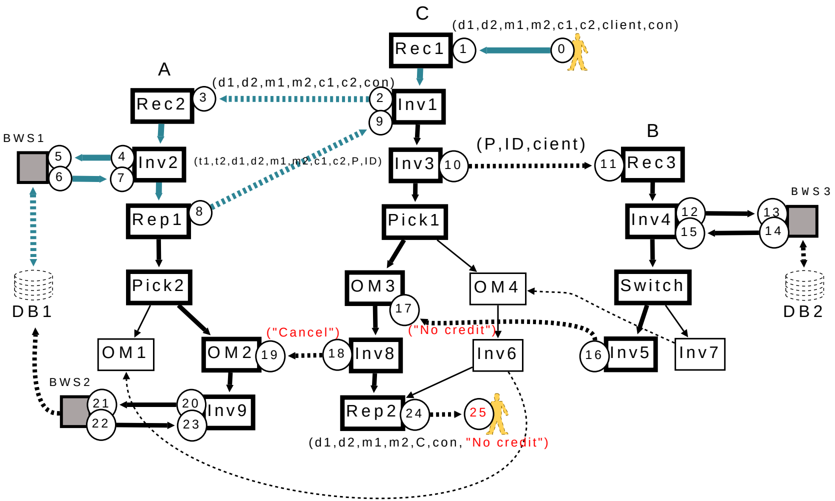
$$Diag = \bigcup_{\mathbb{C}^l_{j\to l}\neq\emptyset} Diag^l \overset{\cup}{\underset{\mathbb{C}}{\times}} Diag^j$$: the solution of the global inequations system is the union of the solutions of the local inequations systems if they are solved according to the affecting order of the bordered places.

Diagnosis problem



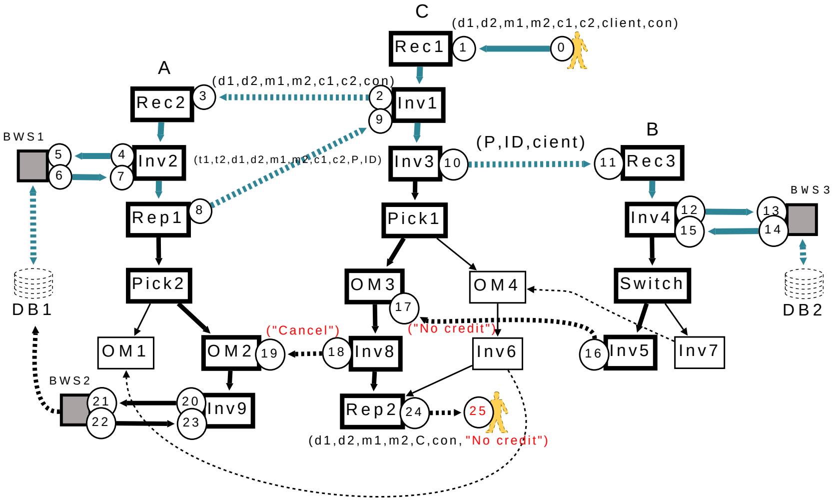
Client sent the departure/arrive dates and cities, client info and reservation condition

Diagnosis problem



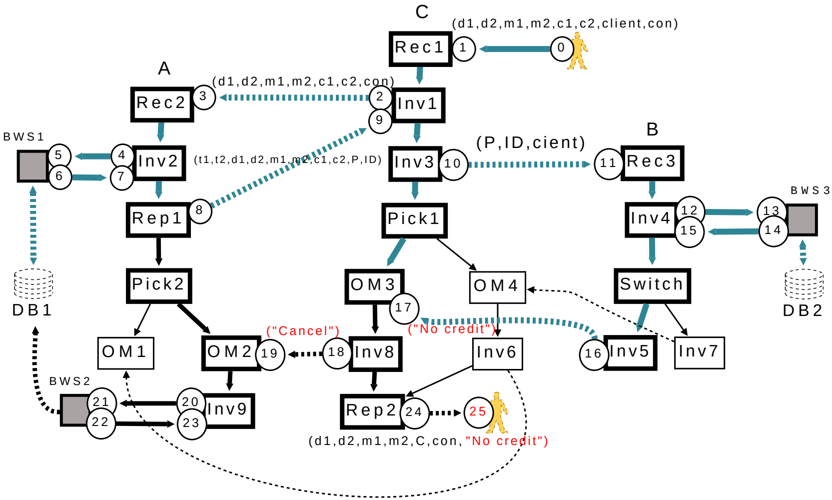
Travel agency (C) requested the airline company (A) to search for the flights and reserved the most satisfying one

Diagnosis problem



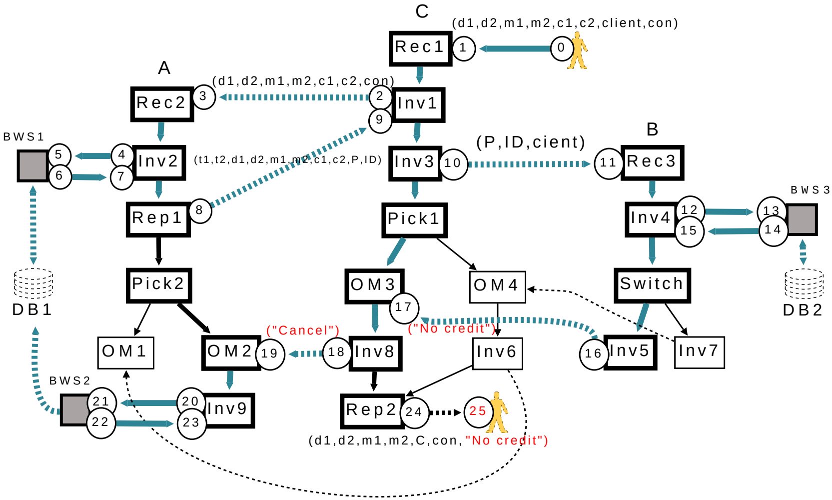
Travel agency (C) requested the payment with bank (B) account

Diagnosis problem



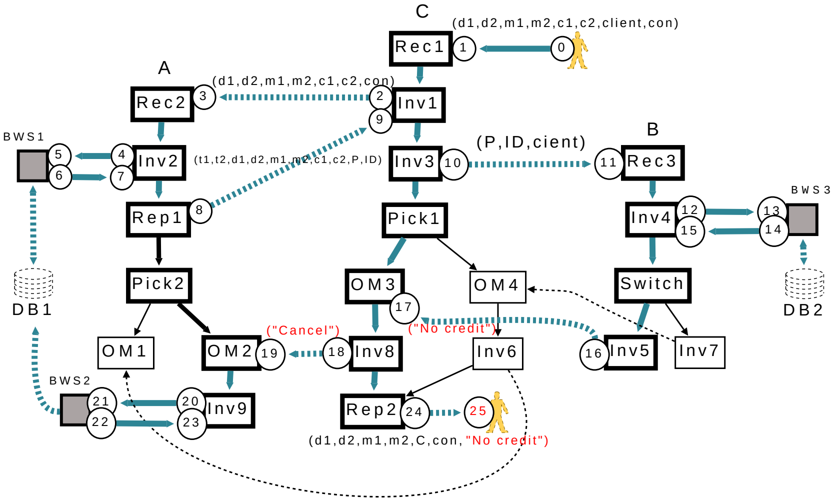
Bank (B) returned "no credit" fault

Diagnosis problem



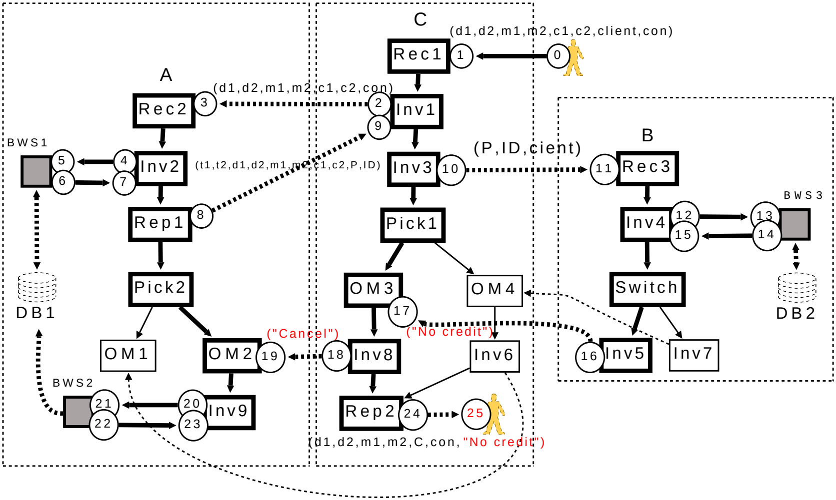
Travel agency (C) canceled the reservation

Diagnosis problem



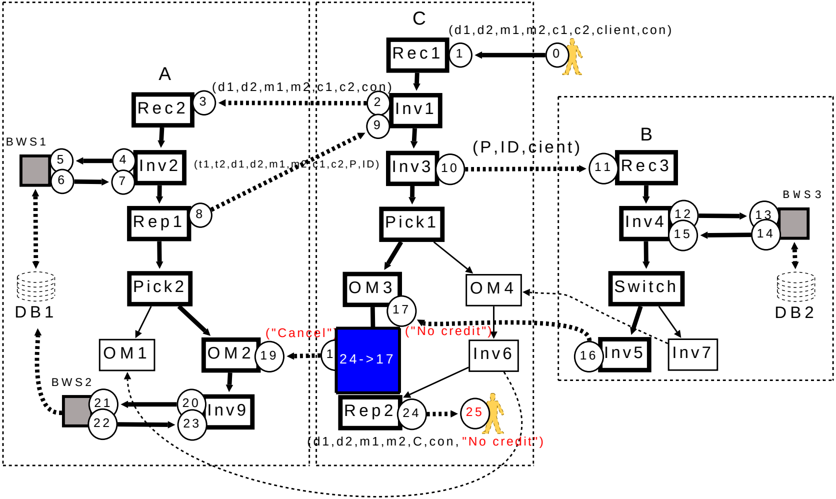
Travel agency (C) return a "no credit" fault to client

Diagnosis coordination



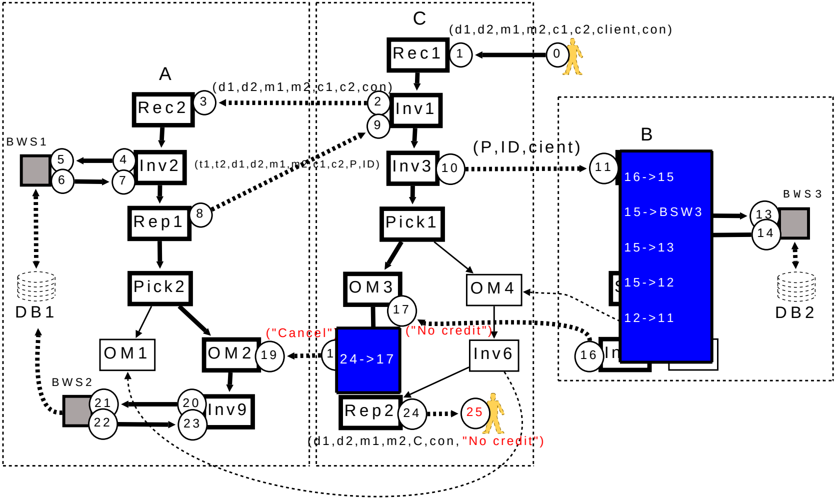
Mapping:
customer:25->C:24, C:17->B:16, B:15->BWS3:14, BWS3:13->B:12,
B:11->C:10, C:9->A:8, A:7->BWS1:6, BWS1:5->A:4, A:3->C:2, C:1->customer:0

Diagnosis coordination



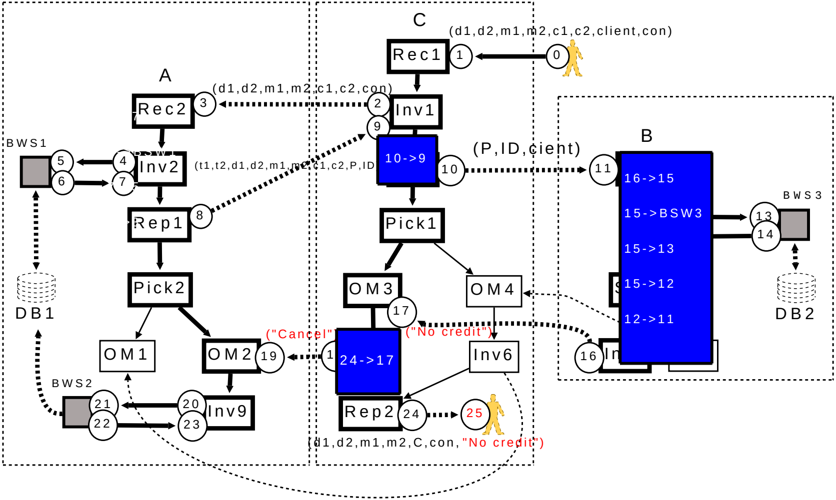
Mapping:
customer:25->C:24, C:17->B:16, B:15->BWS3:14, BWS3:13->B:12,
B:11->C:10, C:9->A:8, A:7->BWS1:6, BWS1:5->A:4, A:3->C:2, C:1->customer:0

Diagnosis coordination



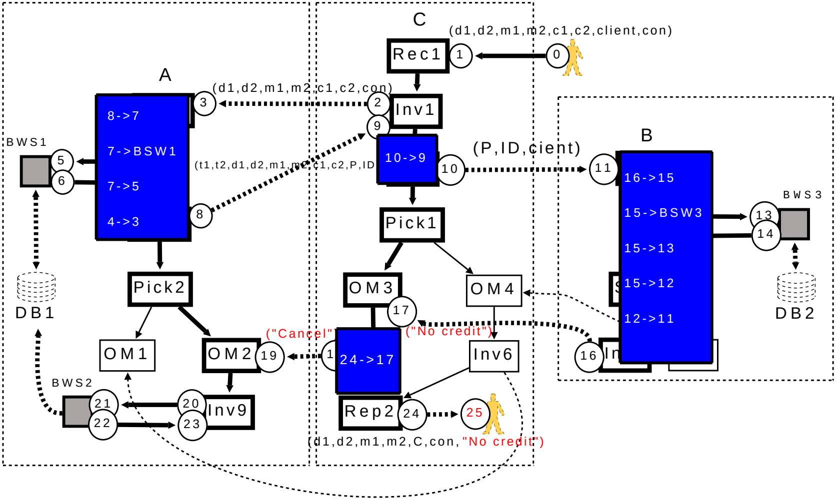
Mapping:
customer:25->C:24, C:17->B:16, B:15->BWS3:14, BWS3:13->B:12,
B:11->C:10, C:9->A:8, A:7->BWS1:6, BWS1:5->A:4, A:3->C:2, C:1->customer:0

Diagnosis coordination



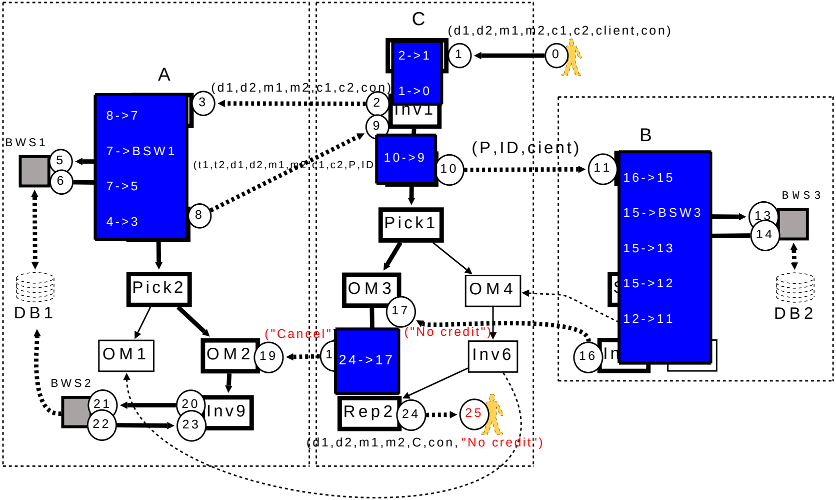
Mapping:
customer:25->C:24, C:17->B:16, B:15->BWS3:14, BWS3:13->B:12,
B:11->C:10, C:9->A:8, A:7->BWS1:6, BWS1:5->A:4, A:3->C:2, C:1->customer:0

Diagnosis coordination



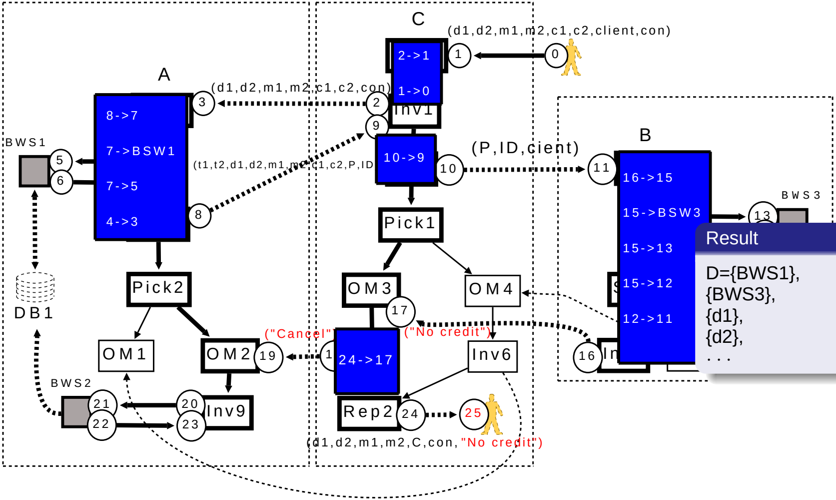
Mapping:
customer:25->C:24, C:17->B:16, B:15->BWS3:14, BWS3:13->B:12,
B:11->C:10, C:9->A:8, A:7->BWS1:6, BWS1:5->A:4, A:3->C:2, C:1->customer:0

Diagnosis coordination



Mapping:
customer:25->C:24, C:17->B:16, B:15->BWS3:14, BWS3:13->B:12,
B:11->C:10, C:9->A:8, A:7->BWS1:6, BWS1:5->A:4, A:3->C:2, C:1->customer:0

Diagnosis coordination



Result

D={BWS1},
{BWS3},
{d1},
{d2},
. . .

Mapping:
customer:25->C:24, C:17->B:16, B:15->BWS3:14, BWS3:13->B:12,
B:11->C:10, C:9->A:8, A:7->BWS1:6, BWS1:5->A:4, A:3->C:2, C:1->customer:0

A possible explanation

Date format interpreting fault: dd/mm/yyyy vs. mm/dd/yyyy

- Reservation failed because of no enough credit
- Input on agency service: on Jul 9, reserved a ticket on Nov 7 and Dec 7
- Input on airline service: reserved a ticket on Jul 11 and Jul 12
- Tickets were much more expensive on Jul 9 for Jul 11 and Jul 12!

Contributions

CPN model (Li et al. [2009a])

- Places represent the data and control
- Transition modes represent the correct and faulty system behavior
- Characteristic vector represents the partially ordered observation
- Token colors represent the data correctness status: correct (b), faulty (r), unknown ($*$)

Automatic decentralized diagnosis application

- Local monitoring components (C#, Java, MySQL by Omar Aaouatif, intern)
- BPEL2CPN transilator (Java Li et al. [2007],Li et al. [2009a])
  - Dependency relationships between I/O data for each basic activity: illustrate the faults transmission between data and controls in BPEL
  - Complex variables (XPath) $\Rightarrow$ CPN places
  - Basic activities and structural operators translation $\Rightarrow$ CPN transitions
- Local diagnosers and decentralized coordinator (Java)

Diagnosis algorithm (Li et al. [2009a],Li et al. [2009b])

- Effective off-line diagnosis based on algebraic symbolic calculation
- Handling cyclic observations in an elegant way
- Decentralized coordinator protocols

Perspectives

Diagnosis for choreographed BPEL services

- to study the diagnosis protocol for the distributed architecture that each local diagnoser recognize its own neighbors and updates its diagnosis according to the diagnosis requests of its neighborhoods.
- to extend the fault prediction based on the data fault diagnosis. Once the faulty input data is confirmed, the prediction for the following places that are not reported to be faulty is reliable. The approach performs the forward reasoning on the CPNs model.

Generalize CPN model for diagnosis

- to study the problem of diagnosability of the CPNs model by checking the discriminative properties of the columns of the CPNs incidence matrix. If in two transitions modes are not discriminative in the incidence matrix, these two transitions cannot be discriminated in diagnosis.
- to introduce the time as the transition "guard" and probability as a new color into the CPNs model. The time stamps of the monitoring log and the probabilistic information can be used to improve the diagnosis precision.

Bibliography I

Cosimo Anglano and Luigi Portinale. B-W analysis: a backward reachability analysis for diagnostic problem solving suitable to parallel implementation. In Valette, R., editor, *Lecture Notes in Computer Science; Application and Theory of Petri Nets 1994, Proceedings 15th International Conference, Zaragoza, Spain*, volume 815, pages 39–58. Springer-Verlag, 1994.

Liliana Ardissono, Luca Console, Anna Goy, Giovanna Petrone, Claudia Picardi, Marino Segnan, and Daniele Theseider Dupré. Enhancing web services with diagnostic capabilities. In *European Conference on Web Services*, pages 182–191, Växjö, Sweden, 2005. IEEE CS.

Salomaa Arto and Sneddon Ian N. *Theory of Automata*. Pergamon Press Reprint, 1969. ISBN 0080133762.

A. Benveniste, E. Fabre, C. Jard, and S. Haar. Diagnosis of asynchronous discrete event systems, a net unfolding approach. *IEEE Trans. on Automatic Control*, 48:714–727, 2003.

J. Cardoso, L.A. Kunzle, and R. Valette. Petri net based reasoning for the diagnosis of dynamic discrete event systems. In *In 6th International Fuzzy Systemes Association World Congress*, page 333¨C336, July 1995.

George Jiroveanu. *Fault diagnosis for large Petri nets*. PhD thesis, Belguim, sep 2006.

Y. Li, T. Melliti, and P. Dague. Modeling bpel ws for diagnosis: towards self-healing ws. In *International Conference on Web Information Systems and Technologies*, pages 795–803. IEEE C.S., 2007.

Y. Li, T. Melliti, and P. Dague. A colored petri nets model for diagnosing data faults of bpel services. In *International Workshop on Principles of Diagnosis*, 2009a.

Y. Li, T. Melliti, and P. Dague. A decentralized model-based diagnosis for bpel services. In *annual IEEE International Conference on Tools with Artificial Intelligence*. IEEE C.S., 2009b.

C. A. Petri. Concepts of net theory. In *MFCS*, pages 137–146, 1973.

Bibliography II

M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis. Diagnosability of discrete event systems. *IEEE Transactions on Automatic Control*, 40(9):1555–1575, sep 1995.

M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D.C. Teneketzis. Failure diagnosis using discrete-event models. *Control Systems Technology, IEEE Transactions on*, 4(2):105–124, Mar 1996. ISSN 1063-6536. doi: 10.1109/87.486338.

V.S. Srinivasan and M.A. Jafari. Fault detection/monitoring using timed petri nets. *IEEE Transactions On Systems Managment and Cybernetics*, 23(4):1155¨C1162, 1994.

R. J. van Glabbeek. Bounded nondeterminism and the approximation induction principle in process algebra. In *4th Annual Symposium on Theoretical Aspects of Computer Sciences on STACS 87*, pages 336–347, London, UK, 1987. Springer-Verlag. ISBN 0-387-17219-X. URL `http://portal.acm.org/citation.cfm?id=28220.28247`.

D. A. Zaitsev. Solving the fundamental equation of petri net in the process of composition of functional subnet. *Artificial Intelligence, no. 1, 2005*, pages 59–68, 2005. In Russian.

Q&A

Thank you!