Model-based Reasoning

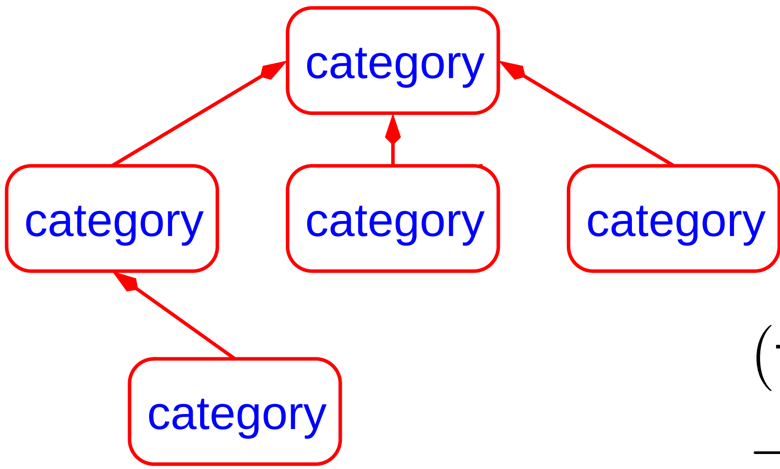
- ‘Traditional’ knowledge systems:
  - Rule based (heuristic rules)

    if conditions then actions/conclusions fi

    ⋮
  - Reasoning:
    - forward chaining: reasoning from facts to conclusions
    - backward chaining: reasoning from goals to facts
  - Recent: business rules
- Model-based systems: reasoning with understandable model, i.e., they have intuitive semantics

Heuristic rules and their disadvantages

Problem solving based on heuristic rules:

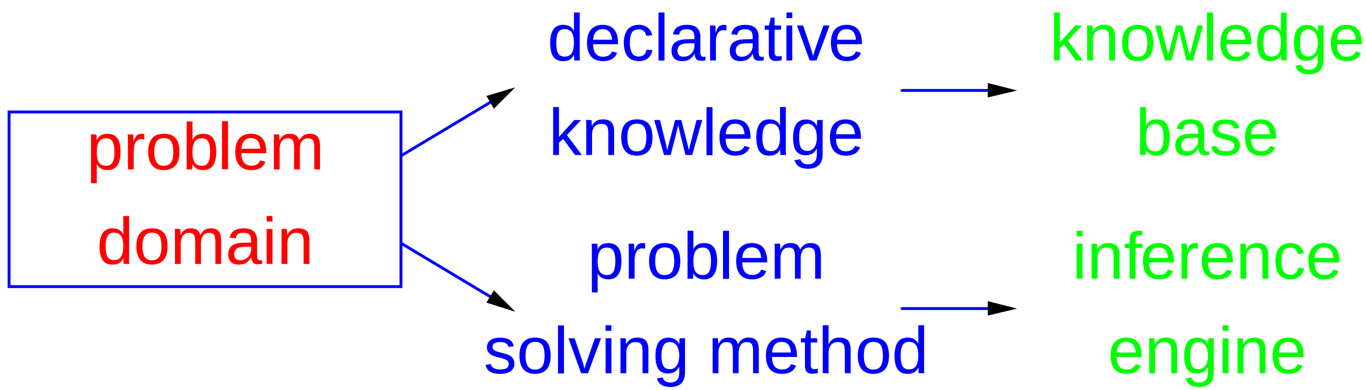


$$(\text{feature}_1 \wedge \cdots \wedge \text{feature}_n)$$
$$\longrightarrow \text{category}$$

Disadvantages:

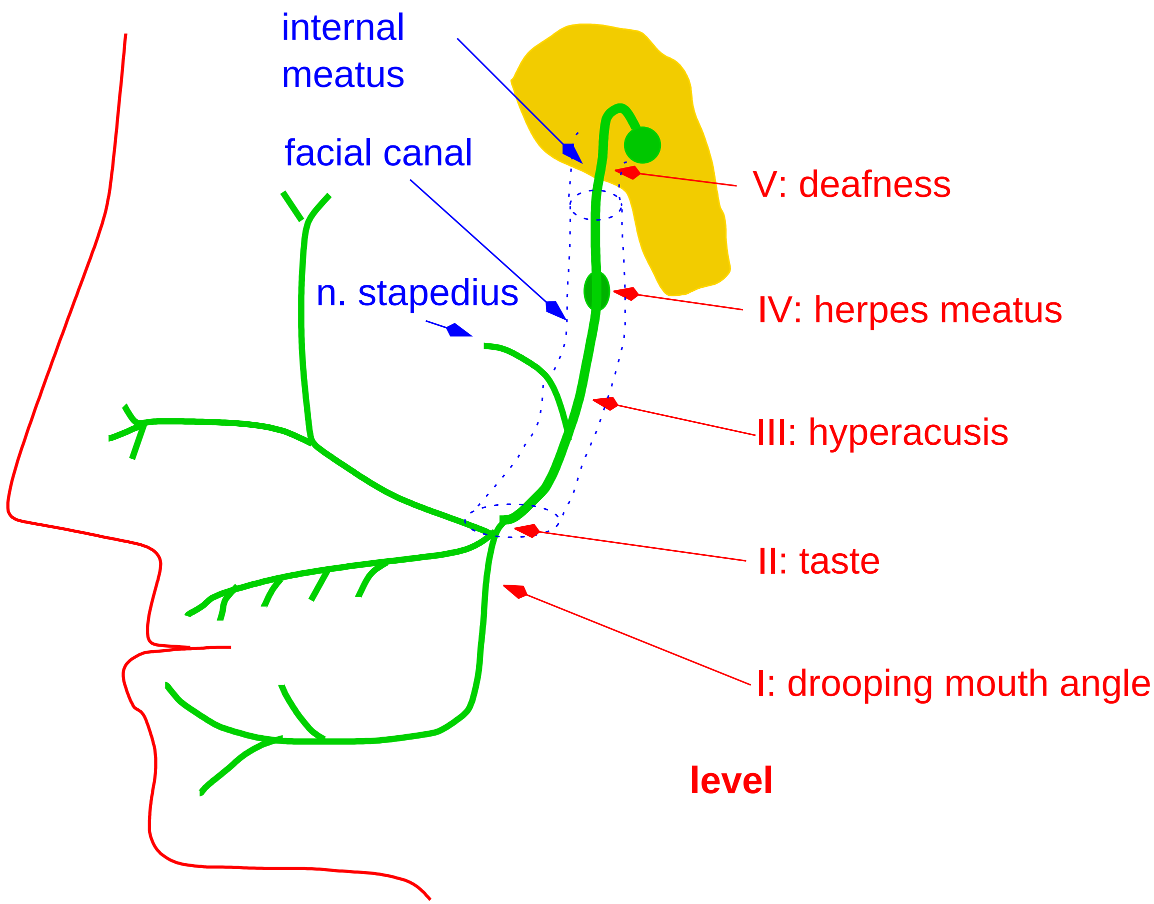
- no use of knowledge about structure and workings
- knowledge maintenance and updating is hard

Model use



- Models:
  - usually designed for handling multiple problems ⇒ reuse
  - capture instantaneous behaviour, temporal behaviour, structure
- Methods: diagnosis, decision making, prediction, planning

Medical diagnosis of facial palsy

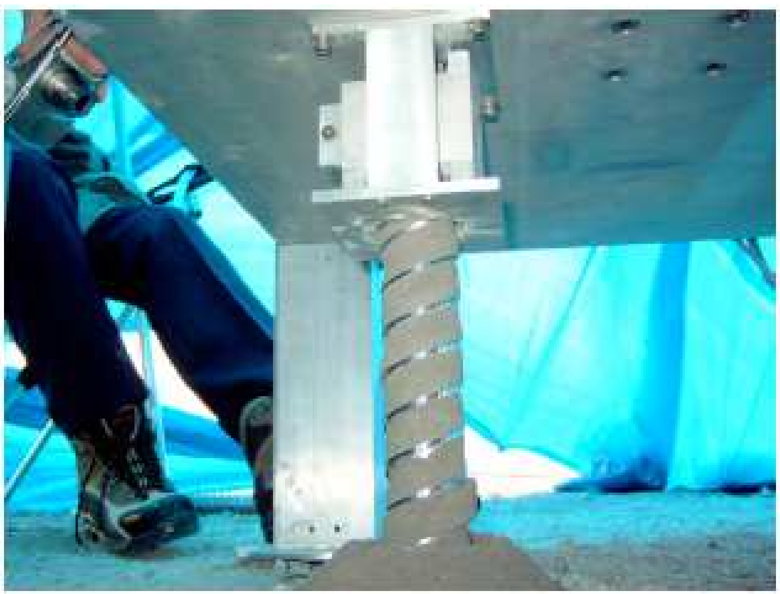


Diagnosis:

Symptoms-level$_{i+1}$ $=$
Symptoms-level$_i$ $\cup$ New-symptoms

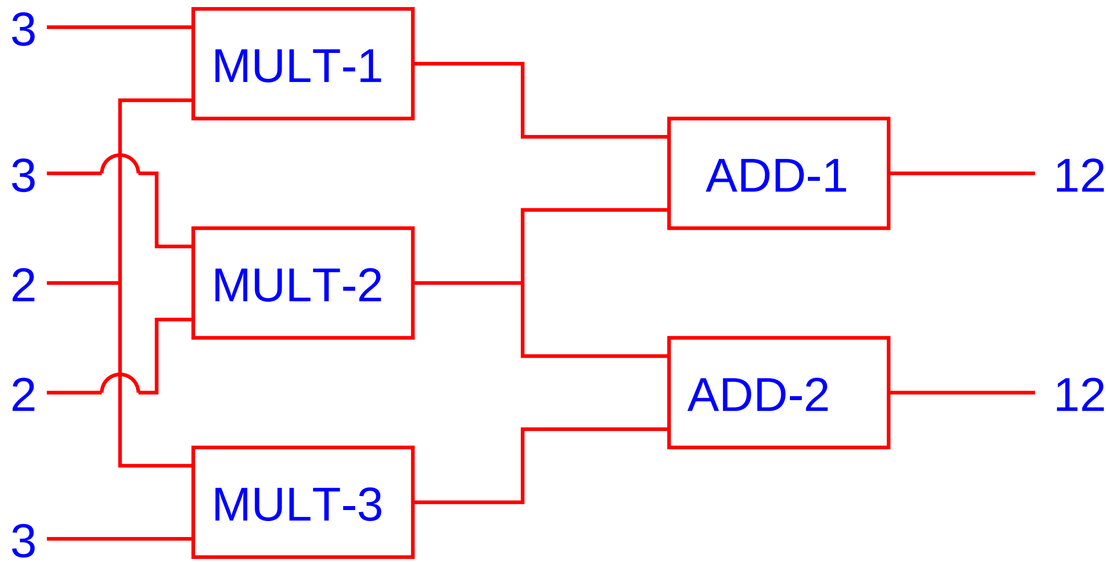
Drilling Automation for Mars

Autonomous drill
with sensors



Fault diagnosis

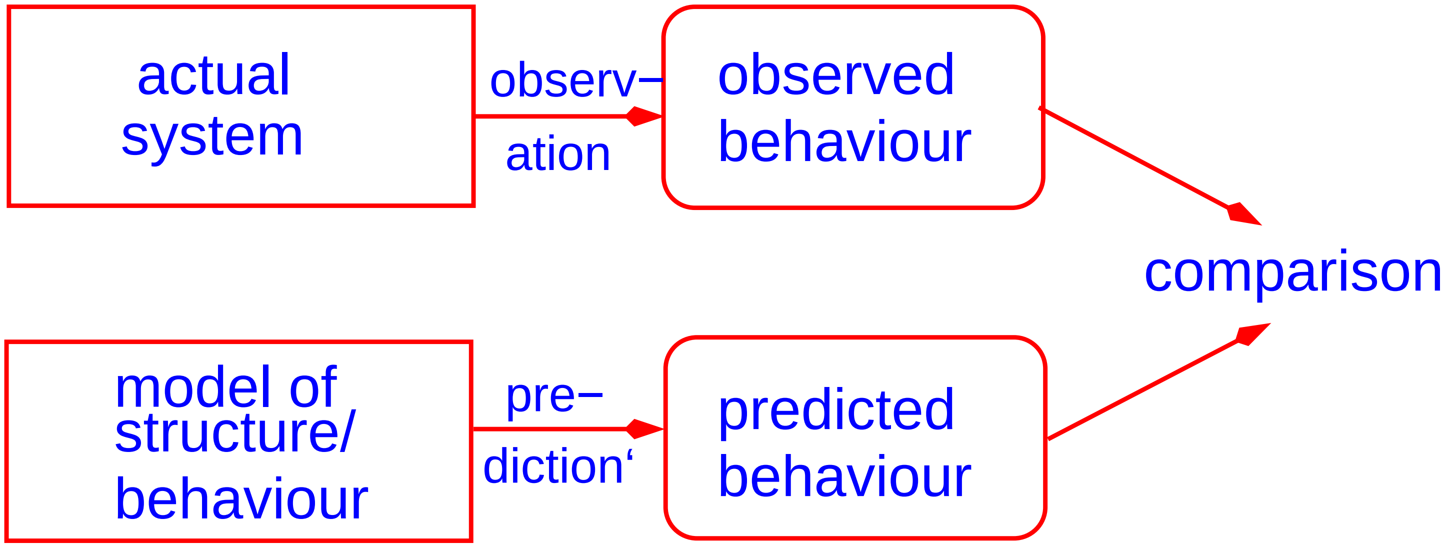
Structure and behaviour



Multiplier-adder

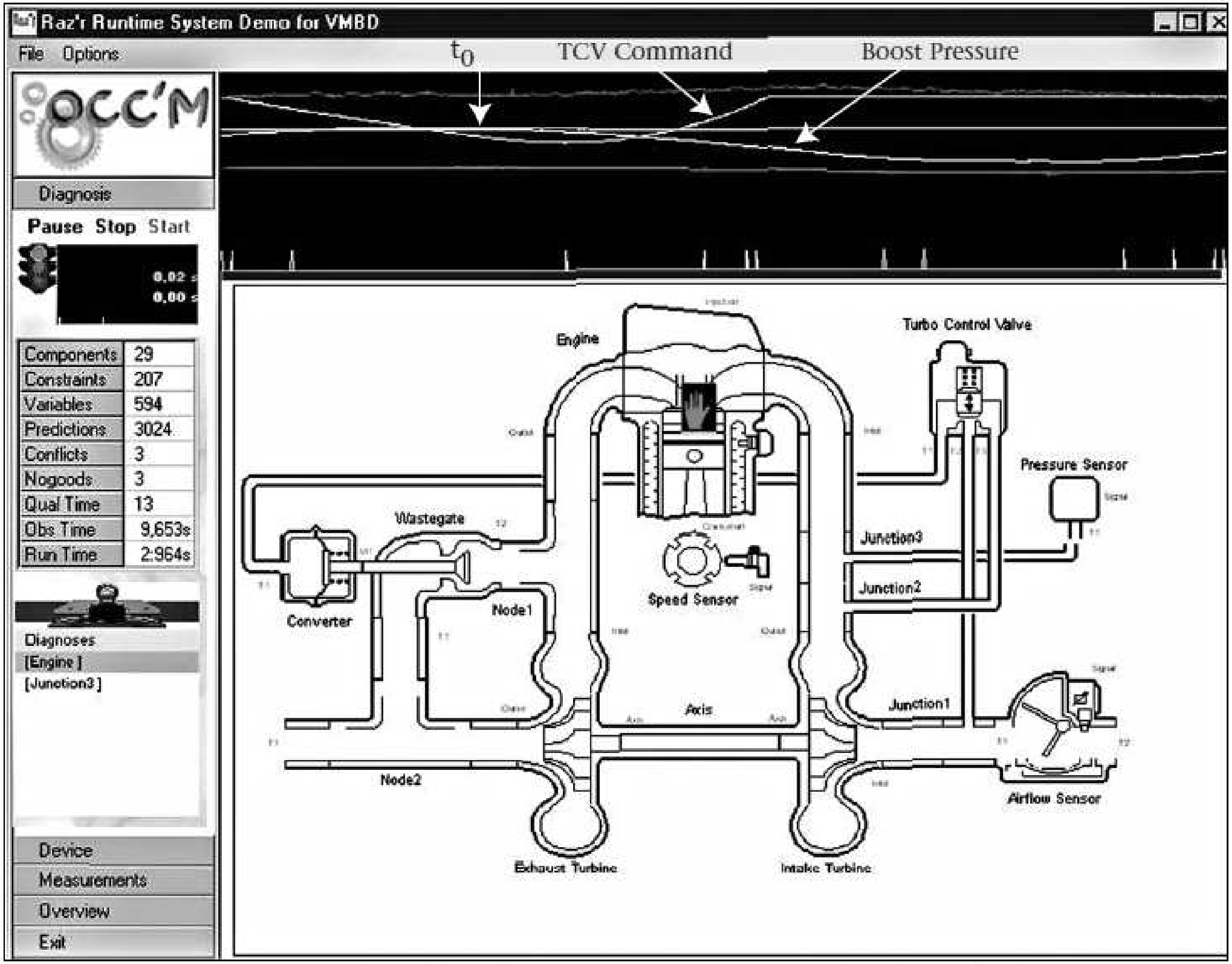
- Structure:
  - components: MULT-1, MULT-2, ...
  - wiring
- Behaviour:
  - behaviour of individual components
  - combined behaviour

Method: model-based diagnosis



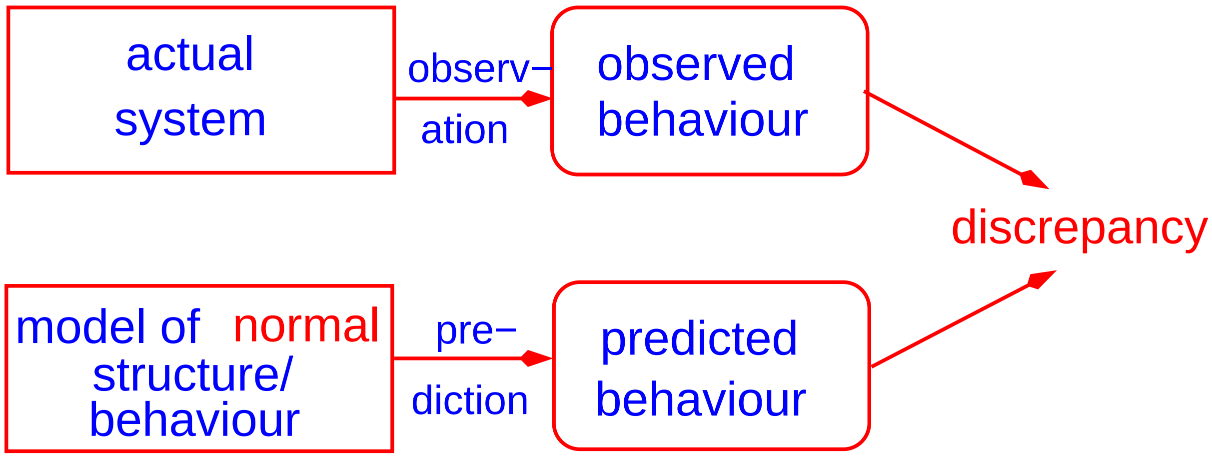
- Model: representation of normal or abnormal behaviour and, possibly, internal structure
- Formalisation:
  - *consistency-based diagnosis*, and
  - *abductive diagnosis*

OCC'M



Raz'r Runtime System Demo for VMBD
File Options
t_0
TCV Command
Boost Pressure
Diagnosis
Pause Stop Start
Components 29
Constraints 207
Variables 594
Predictions 3024
Conflicts 3
Nogoods 3
Qual Time 13
Obs Time 9,653s
Run Time 2:964s
Diagnoses
[Engine]
[Junction3]
Device
Measurements
Overview
Exit
Engine
Turbo Control Valve
Pressure Sensor
Wastegate
Converter
Node1
Node2
Speed Sensor
Junction3
Junction2
Junction1
Axis
Airflow Sensor
Exhaust Turbine
Intake Turbine

Consistency-based diagnosis

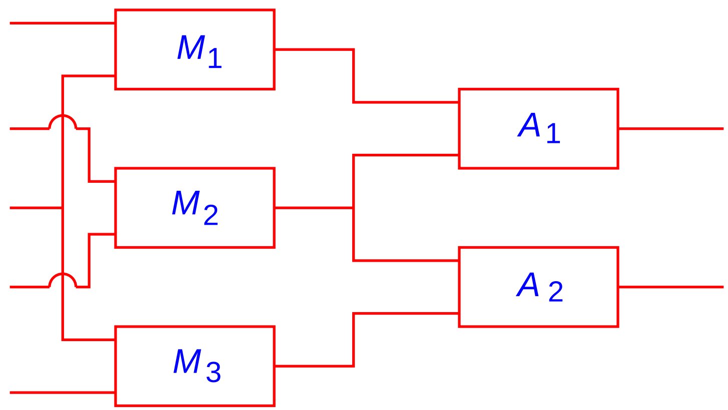


Difference between predicted behaviour and observed behaviour $\Rightarrow$ defect!

Originators:

- R. Reiter, "A Theory of diagnosis from first principles", *Artificial Intelligence*, vol. 32, 57–95, 1987.
- J. de Kleer, A.K. Macworth, and R. Reiter, "Characterising diagnoses and systems", *Artificial Intelligence*, vol. 52, 197–222, 1992.

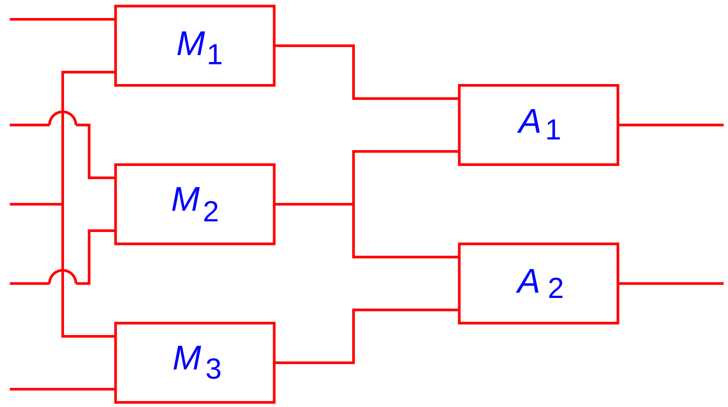
Normal behaviour



SYStem specification $\mathrm{SYS} = (\mathrm{SD}, \mathrm{COMPS})$:

- Components that may be defective (faulty):
  $\mathrm{COMPS} = \{M_1, M_2, M_3, A_1, A_2\}$
- SD (System Description):
  - *generic* description of component behaviour (what the component does)
  - declaration of components: $\mathrm{MUL}(M_1)$, $\mathrm{ADD}(A_1)$
  - connection between components

Normal behaviour: formal



SYStem specification $\mathrm{SYS} = (\mathrm{SD}, \mathrm{COMPS})$:

- SD (System Description):

  $\forall x(\mathsf{MUL}(x) \rightarrow \mathrm{in}_1(x) \times \mathrm{in}_2(x) = \mathrm{out}(x))$
  $\forall x(\mathsf{ADD}(x) \rightarrow \mathrm{in}_1(x) + \mathrm{in}_2(x) = \mathrm{out}(x))$

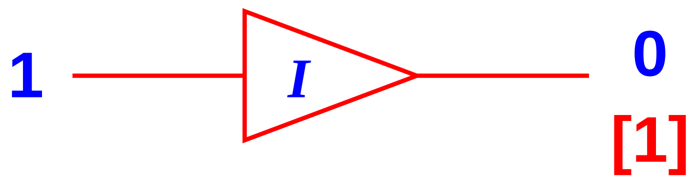
  $\mathrm{MUL}(M_1), \mathrm{MUL}(M_2), \mathrm{MUL}(M_3), \mathrm{ADD}(A_1), \mathrm{ADD}(A_2)$
  $\mathrm{in}_1(A_1) = \mathrm{out}(M_1), \mathrm{in}_2(A_1) = \mathrm{out}(M_2)$
  $\mathrm{in}_1(A_2) = \mathrm{out}(M_2), \mathrm{in}_2(A_2) = \mathrm{out}(M_3)$

- $\mathrm{COMPS} = \{M_1, M_2, M_3, A_1, A_2\}$

Ab predicate

- Ab(c): component c is *abnormal*
- $\neg$Ab(c): component c is not abnormal, i.e. *normal*

Example (Inverter I):



$$\mathrm{SD} = \{\forall x((\mathrm{INV}(x) \wedge \neg \mathrm{Ab}(x)) \rightarrow \neg(\mathrm{out}(x) = \mathrm{in}(x))),\ \mathrm{INV}(I)\}$$

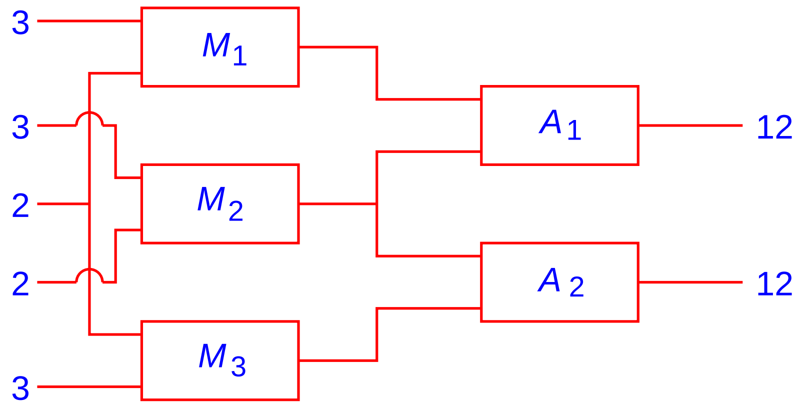
- Input: $\mathrm{in}(I) = 1$; observed output: $\mathrm{out}(I) = 1$

$$\mathrm{SD} \cup \{\mathrm{in}(I) = 1, \mathrm{out}(I) = 1\} \cup \{\neg \mathrm{Ab}(I)\} \vDash \bot$$

$$\mathrm{SD} \cup \{\mathrm{in}(I) = 1, \mathrm{out}(I) = 1\} \cup \{\mathrm{Ab}(I)\} \nvDash \bot$$

(assumption that I is (ab)normal is (in)consistent)

Normal behaviour formal



SYStem specification SYS = (SD, COMPS):

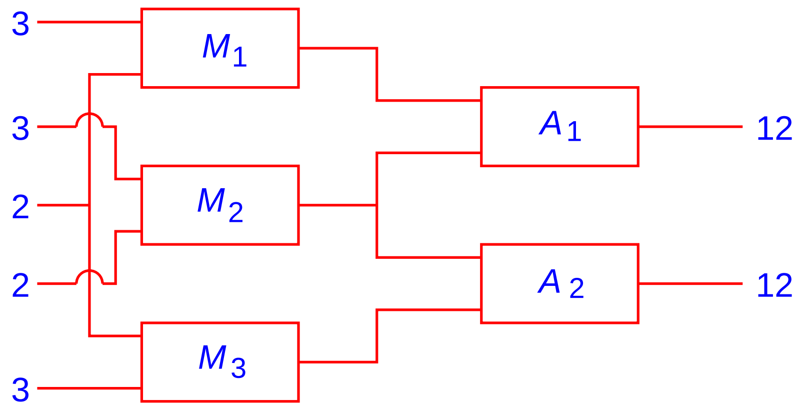
- SD (System Description):

$$\forall x((\text{MUL}(x) \wedge \neg \text{Ab}(x)) \rightarrow \text{in}_1(x) \times \text{in}_2(x) = \text{out}(x))$$
$$\forall x((\text{ADD}(x) \wedge \neg \text{Ab}(x)) \rightarrow \text{in}_1(x) + \text{in}_2(x) = \text{out}(x))$$

$$\text{MUL}(M_1), \text{MUL}(M_2), \text{MUL}(M_3), \text{ADD}(A_1), \text{ADD}(A_2)$$
$$\text{in}_1(A_1) = \text{out}(M_1), \text{in}_2(A_1) = \text{out}(M_2)$$
$$\text{in}_1(A_2) = \text{out}(M_2), \text{in}_2(A_2) = \text{out}(M_3)$$

- $\text{COMPS} = \{M_1, M_2, M_3, A_1, A_2\}$

Prediction of normal behaviour

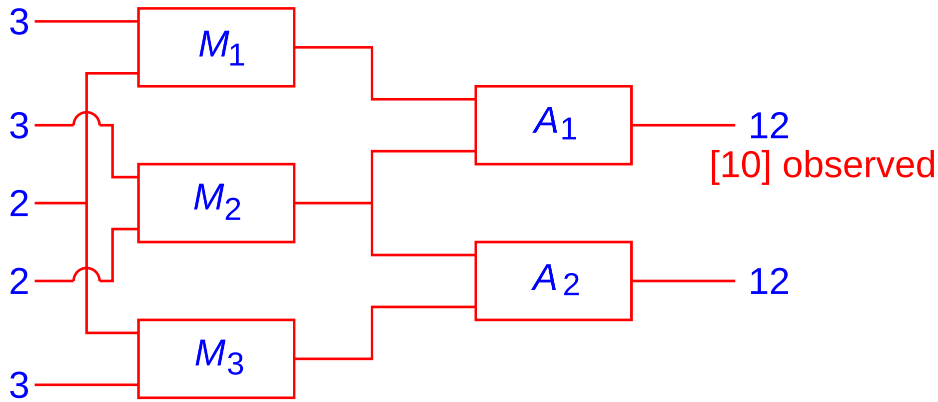


- System specification SYS = (SD, COMPS)
- A prediction:

  $\mathrm{SD} \cup \{\neg \mathrm{Ab}(c) \mid c \in \mathrm{COMPS}$ is not defective$\} \cup \mathrm{Inputs} \models \mathrm{Behaviour}$ (Inputs stands for all observations)
- Example:

  $\mathrm{SD} \cup \{\neg \mathrm{Ab}(M_1), \neg \mathrm{Ab}(M_2), \neg \mathrm{Ab}(A_1)\} \cup \{\mathrm{in}_1(M_1) = 3, \mathrm{in}_2(M_1) = 2, \mathrm{in}_1(M_2) = 3, \mathrm{in}_2(M_2) = 2\} \models \mathrm{out}(A_1) = 12$

There is a fault!



Let $(\text{out}(A_1) = 10) \in \text{Inputs}$, then:

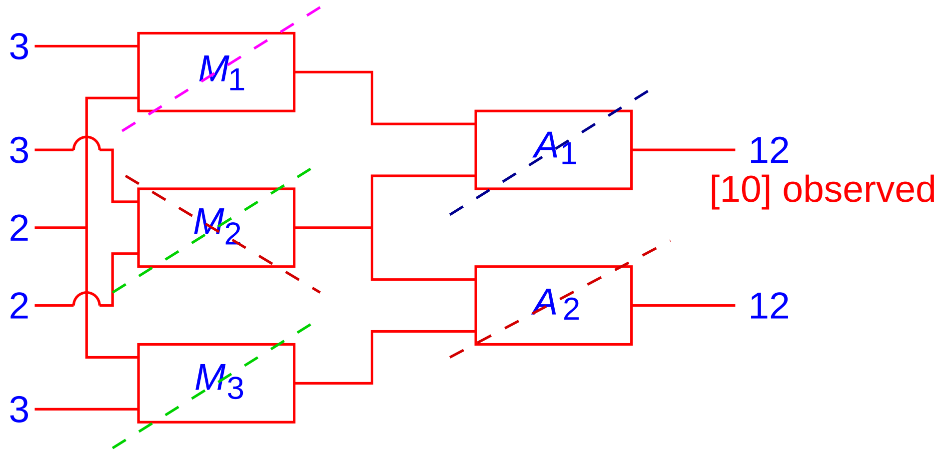
$$\text{SD} \cup \{\neg\text{Ab}(c) \mid c \in \text{COMPS}\} \cup \text{Inputs} \models \bot$$

because:

- $\text{SD} \cup \{\neg\text{Ab}(c) \mid c \in \text{COMPS}\} \cup \text{Inputs} \models \text{out}(A_1) = 12$, and
- $\text{SD} \cup \{\neg\text{Ab}(c) \mid c \in \text{COMPS}\} \cup \text{Inputs} \models \text{out}(A_1) = 10$

$\Rightarrow$ faulty component

Which components are faulty?



Possible diagnoses (faulty components) D:

- $D = \{A_1\}, \{M_1\}, \{M_2, M_3\}, \{A_2, M_2\}$, because $\mathrm{SD} \cup \{\neg \mathrm{Ab}(c) \mid c \in \mathrm{COMPS} - D\}$ $\cup \{\mathrm{Ab}(c) \mid c \in D\} \cup \mathrm{Inputs} \not\models \bot$
- D must be the *smallest* set because, $D = \mathrm{COMPS}$ would also be a diagnosis otherwise

$\Rightarrow$ multiple diagnoses

Diagnostic problem

- System specification $\mathsf{SYS} = (\mathsf{SD}, \mathsf{COMPS})$
- Diagnostic problem $\mathsf{DP} = (\mathsf{SYS}, \mathsf{OBS})$, with OBS a set of observations
- A diagnosis D: smallest (subset minimal) set of components, such that

$$\mathsf{SD} \cup \mathsf{OBS} \cup \{\mathsf{Ab}(c) \mid c \in D\} \cup \{\neg\mathsf{Ab}(c) \mid c \in \mathsf{COMPS} - D\}$$

is consistent

Example:

$$\begin{aligned}\mathsf{OBS} = \{&\mathrm{in}_1(M_1) = 3, \mathrm{in}_2(M_1) = 2, \mathrm{in}_1(M_2) = 3, \mathrm{in}_2(M_2) = 2,\\ &\mathrm{in}_1(M_3) = 2, \mathrm{in}_2(M_3) = 3, \mathrm{out}(A_1) = 10, \mathrm{out}(A_2) = 12\}\end{aligned}$$

Algorithms

- Enumerate all diagnoses: #P complete (NP hard for enumeration):

$$\mathrm{Ab}(A_1) \wedge \neg\mathrm{Ab}(A_2) \wedge \neg\mathrm{Ab}(M_1) \wedge \neg\mathrm{Ab}(M_2) \wedge \neg\mathrm{Ab}(M_3)$$
$$\mathrm{Ab}(A_1) \wedge \mathrm{Ab}(A_2) \wedge \neg\mathrm{Ab}(M_1) \wedge \neg\mathrm{Ab}(M_2) \wedge \neg\mathrm{Ab}(M_3)$$
$$\mathrm{Ab}(A_1) \wedge \mathrm{Ab}(A_2) \wedge \mathrm{Ab}(M_1) \wedge \neg\mathrm{Ab}(M_2) \wedge \neg\mathrm{Ab}(M_3)$$
$$\vdots$$

- Heuristic methods:
  - hitting set algoritme (Reiter)
  - assumption-based truth maintenance system (ATMS, De Kleer)
- Restrictions: for example, only maximally 2 defects, then complexity upperbound

Basic problem: which idea should underly such algorithms?

Conflict set

Let CS $\subseteq$ COMPS be a set of components, then CS is called a conflict set iff

$$\text{SD} \cup \text{OBS} \cup \{\neg\text{Ab}(c) \mid c \in \text{CS}\}$$

is inconsistent ($\text{SD} \cup \text{OBS} \cup \{\neg\text{Ab}(c) \mid c \in \text{CS}\} \vDash \bot$)

Proposition: For each $D \subseteq$ COMPS that is a diagnosis and each conflict set CS it holds that: $D \cap \text{CS} \neq \varnothing$

Proof: $\text{SD} \cup \text{OBS} \cup \{\neg\text{Ab}(c) \mid c \in \text{COMPS} - D\} \nvDash \bot$, with D subset minimal $\Rightarrow$ COMPS $- D$ is subset maximal, hence

$$\text{SD} \cup \text{OBS} \cup \underbrace{\{\neg\text{Ab}(c) \mid c \in \text{COMPS} - D\} \cup \{\neg\text{Ab}(c')\}}_{\text{CS}} \vDash \bot$$

for c' in CS

Basic ideas: hitting sets

- Determine conflict sets of the diagnostic problem DP
- Each conflict set CS has at least one element in common with a diagnosis D:

$$
\begin{array}{ccc}
D = \{c_1, c_2, & \ldots & , c_m\} \\
\quad / & & \backslash \\
\text{CS}_1 = \{\ldots, c_1, \ldots\} & & \text{CS}_m = \{\ldots, c_m, \ldots\}
\end{array}
$$

- Compute so-called hitting sets:
  - Let F be a set of sets, and
  - $H \subseteq \bigcup_{S \in F} S$,
  - H is a hitting set if for all $S \in F : H \cap S \neq \varnothing$
- Example: $F = \{\{1,2\},\{3,4\}\}$, then $H = \{2,4\}$ is a (non-unique) hitting set ($H = \{1,3\}$ is also a hitting set)

Diagnosis as hitting set

Theorem: D is a diagnosis for diagnostic problem $\text{DP} = (\text{SYS}, \text{OBS})$ iff D is a minimal hitting set for all conflict sets of DP

Proof (sketch):

- Prop. page 19: for all conflict sets CS: $\text{CS} \cap D \neq \varnothing$. Thus, D is a hitting set
- D is also a *minimal* hitting set, as $\text{COMPS} - D$ is no conflict set, whereas $\{c\} \cup (\text{COMPS} - D)$ is a conflict set for any $c \in D$

Hitting-set tree

We construct a tree structure for computing diagnoses

Some definitions:

- Let F be a set of sets
- Let $T = (V, E, l_V, l_E)$ be a labelled tree, with V a set of nodes and $E \subseteq V \times V$ a set of edges, and
  - l_V a node label function:

$$l_V : V \rightarrow F \cup \{\checkmark\}$$

  - l_E an edge label function:

$$l_E : E \rightarrow \bigcup_{S \in F} S$$

Node label function

Let F be a set of sets:

- l_V a node label function:

$$l_V : V \to F \cup \{\checkmark\}$$

  with

$$l_V(v) = \begin{cases} S & \text{if } S \in F,\ S \neq \varnothing \\ \checkmark & \text{otherwise} \end{cases}$$

- Example: $F = \{\{1,2\},\{4,5\}\}$
  - Nodes $V = \{u, v, w\}$
  - $l_V(u) = \{1,2\}$, $l_V(v) = \{4,5\}$, and $l_V(w) = \checkmark$

Edge label function

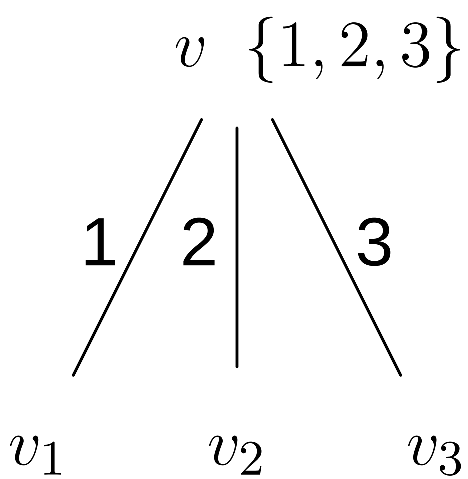
Let F be a set of sets:

- l_E an edge label function:

$$l_E : E \to \bigcup_{S \in F} S$$

with if $l_V(v) = S$ and $\forall s \in S$: $l_E(v, v_s) = s$

Example:



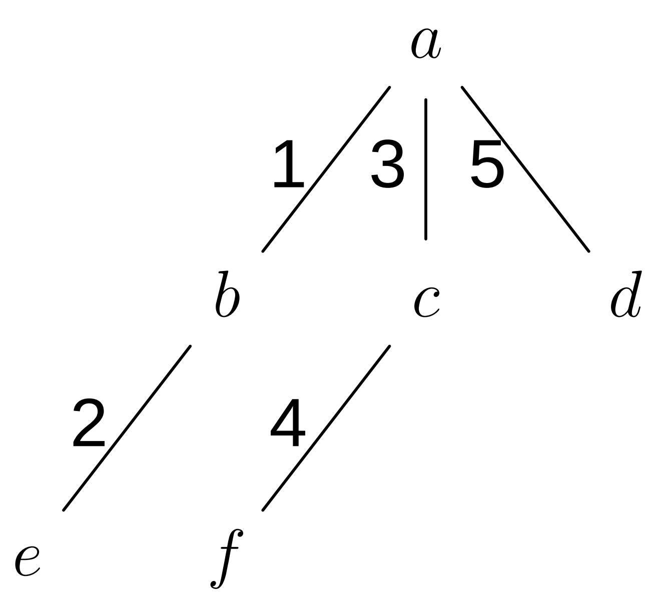
- $F = \{\{4, 5\}, \{1, 2, 3\}\}$, and $S = \{1, 2, 3\}$
- $l_V(v) = \{1, 2, 3\}$
- $l_E(v, v_1) = 1$, $l_E(v, v_2) = 2$, $l_E(v, v_3) = 3$

Construction of hitting sets

Let $T = (V, E, l_V, l_E)$ be a labelled tree, then the hitting set $H(v)$ for node v is defined as:

$$H(v) = \{l_E(u, w) \mid (u, w) \text{ is on the path from the root to } v\}$$

Example:



- $H(a) = \varnothing$
- $H(b) = \{1\}$
- $H(e) = \{1, 2\}$
- $H(f) = \{3, 4\}$

Hitting-set algorithm

- F is the set of conflict sets, which is initially empty
- Let node v_s be a child of node v, then $l_V(v_s) = \text{CS}$ if there is exists a $\text{CS} \in F$ with

$$\text{CS} \cap H(v_s) = \varnothing$$

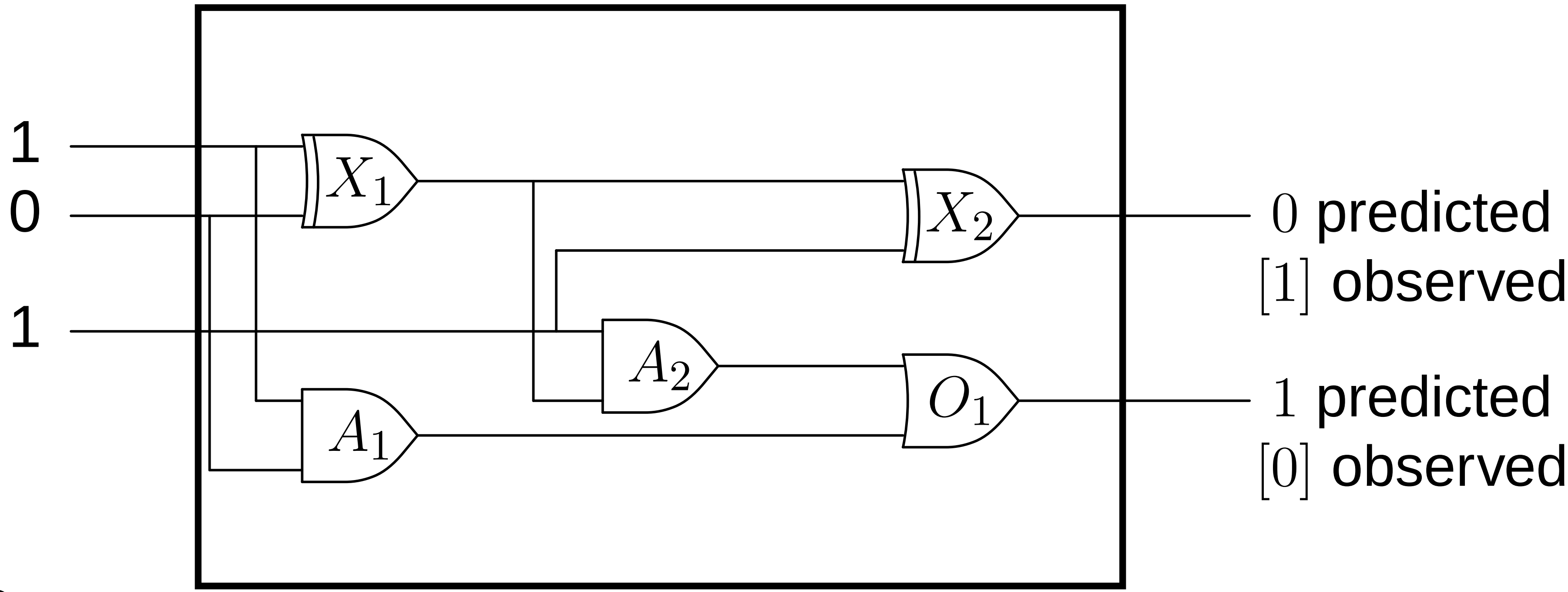
  (CS is not yet covered by $H(v_s)$ and we have to extend the path)
- If no suitable $\text{CS} \in F$, call logical reasoning program TP:
  - Call: TP($\text{SD}, \text{COMPS} - H(v_s), \text{OBS}$)
  - Returns: conflict set CS if
    $\text{SD} \cup \text{OBS} \cup \{\neg\text{Ab}(c) \mid c \in \text{COMPS} - H(v_s)\} \vDash \bot$, with
    $\text{CS} \cap H(v_s) = \varnothing$ and $\text{CS} \subseteq \text{COMPS} - H(v_s)$
    otherwise, ✓ (consistent)

Hitting-set algorithm

```
Diagnose(SD, COMPS, OBS)
{
   generate HS tree by calling
        CS <- TP(SD, COMPS - H(v), OBS);
   (F is build op from these CS’s)
   leaves v with ✓
      determine diagnosis H(v)
   determine subset-minimal H(v)
        in the HS tree
}

TP(A, B, C)
{
   use resolution on A U B U C
}
```

Example: full-adder



SD:

$\forall x((\text{ANDG}(x) \wedge \neg\text{Ab}(x)) \rightarrow (\text{out}(x) = \text{in}_1(x) \wedge \text{in}_2(x)))$

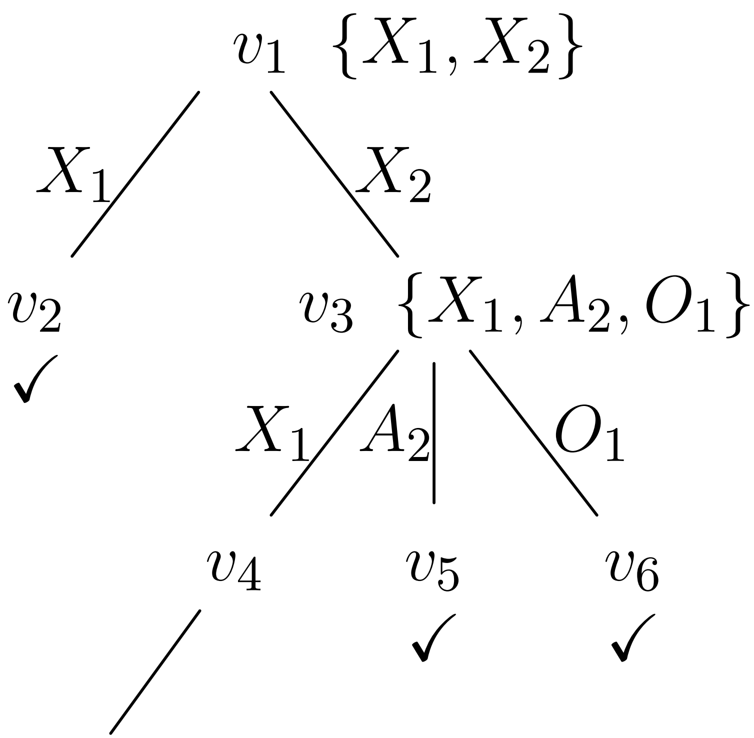
$\forall x((\text{ORG}(x) \wedge \neg\text{Ab}(x)) \rightarrow (\text{out}(x) = \text{in}_1(x) \vee \text{in}_2(x)))$

$\vdots$

$\text{ORG}(O_1), \text{ANDG}(A_1), \text{XORG}(X_1), \ldots$

$\text{COMPS} = \{A_1, A_2, X_1, X_2, O_1\}$

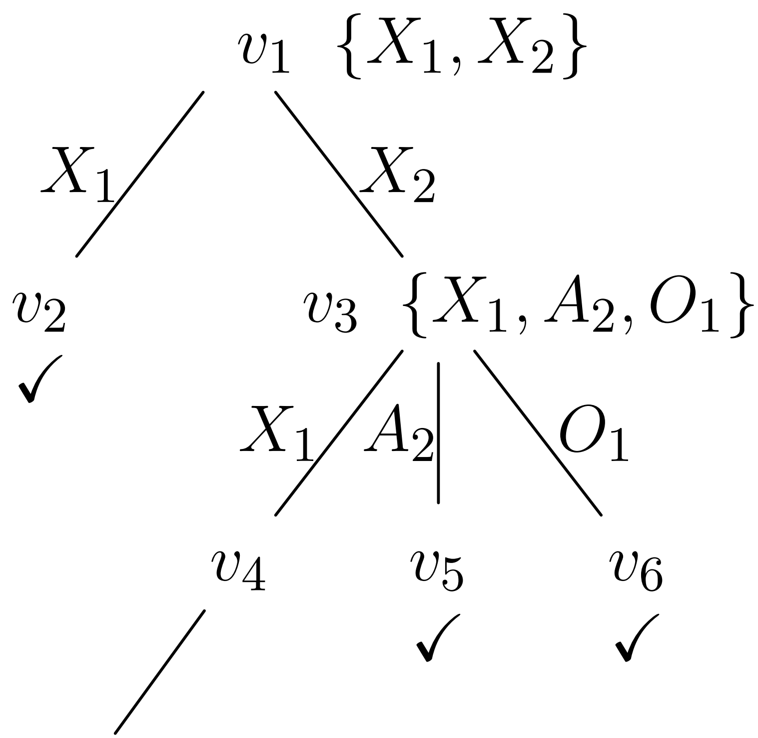
Example HS tree



1. $\text{CS}_1 \leftarrow \text{TP}(\text{SD}, \text{COMPS}, \text{OBS})$;
   $\text{CS}_1 \leftarrow \{X_1, X_2\}$
2. $\text{CS}_2 \leftarrow \text{TP}(\text{SD}, \text{COMPS} - \{X_1\}, \text{OBS})$;
   $\text{CS}_2 \leftarrow \checkmark$ (diagnosis found)
3. $\text{CS}_3 \leftarrow \text{TP}(\text{SD}, \text{COMPS} - \{X_2\}, \text{OBS})$;
   $\text{CS}_3 \leftarrow \{X_1, A_2, O_1\}$
4. $\vdots$

Diagnoses D: $\{X_1\}$, $\{X_2, A_2\}$, $\{X_2, O_1\}$
(note that $\{X_2, X_1\}$ not subset minimal)

Pruning of the HS tree



Note that there is no need to extend the hitting set $H(v_4)$ (as $\{X_2, X_1\}$ is not subset minimal)

$\Rightarrow$ pruning of the hitting-set tree