Part I
Diagnosis

1 Types of reasonings

Deduction, Abduction, Learning

Example 1. Famous *syllogism* of Aristotle: "Socrates is a man" "Every man is mortal" *SO* "Socrates is mortal" *Deduction principle* (entailment, prediction, anticipation, planning...)

Elementary! My Dear Watson!

Example 2. Crime scene: Sherlock came and saw Socrates dead: "Socrates is mortal" Sherlock knew one crucial information: "Every man is mortal" Watson said: "what are your conclusions?" Sherlock answered: "Elementary! My Dear Watson! Socrates is a man, that's the reason why he had to die one day" Is it right? Is it deduction? *NO*

This is *abduction*. Main reasoning for *diagnosis*. Sherlock is a master of abduction (and not deduction)...

We cannot deduce that Socrates is a man. Maybe he's a rat, a flower... This is just an hypothesis.

What about learning?

Example 3. A machine knows that "Socrates is a man" and sees that "Socrates is mortal". so it can *learn* a generic rule:

- "Every man is mortal"
- or "Every mortal is a man"
- or "The concept of man has a relationship with the concept of mortality"

2 Diagnosis: an introduction

Diagnosis problem

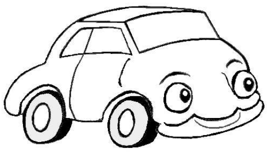
Definition 4. Given:

1. a *system*
2. a set of *observations*

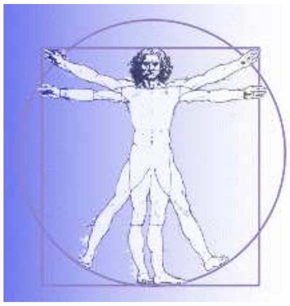
How to:

1. determine the *failures* of the system
2. repair the system

Diagnosis problem: example

Example 5. **System**: 
Observation: The car does not start
Possible diagnoses: The battery does not work, the starter is broken, no petrol...
Repair: test plan to discriminate among the diagnoses (check the battery, ...)

Diagnosis problem: another example

Example 6. **System**: 
Observation: Flu (40 degrees), headache
Possible diagnoses: Cold, Migraine
Repair: Take three pills per day

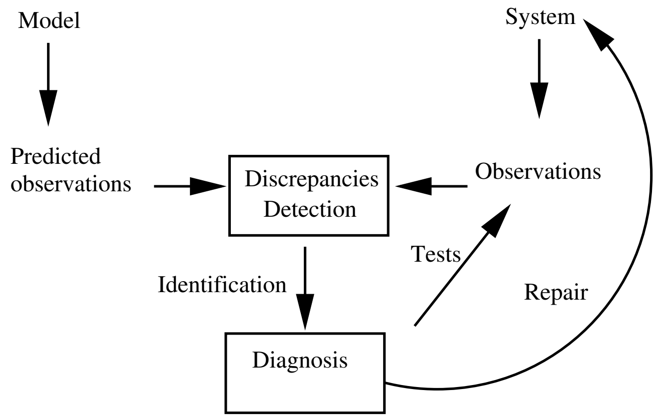
Diagnosis: history

- 70's: heuristic approaches (expert systems)
  - knowledge base = set of abductive rules (need expertises)
  - inference
- 80's: model-based diagnosis (static systems)
- 90's: model-based diagnosis (dynamic systems)

3 Model-based diagnosis

3.1 Knowledge representation

Model-based diagnosis: the idea



Knowledge representation

Definition 7. A *system* is a couple $(SD, COMP)$:

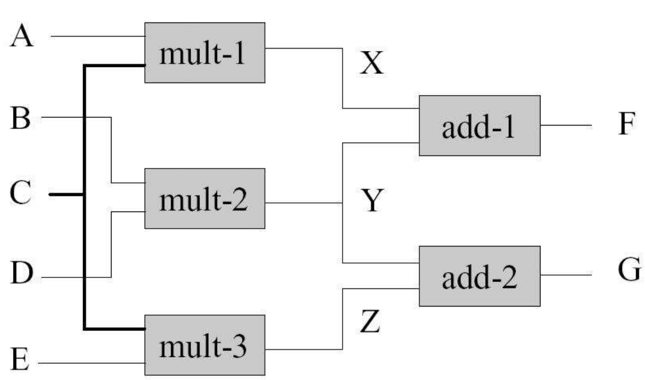
- $COMP$ is a finite set of constants, one constant = one component
- SD is a set of FOL sentences describing the behaviour of the system
  - *Behavioral model* (how a component works)
  - *Structural model* (how components interact)

Definition 8. A *observed system* is a system $(SD, COMP)$ with some observations OBS:

- OBS is a set of atomic sentences.
- Each atomic sentence represents an observation

Knowledge representation: example

Example 9. Davis circuit



Knowledge representation: symbols

Example 10. $COMP = \{a1, a2, m1, m3, m3\}$ SD **predicates**:

- Add additioner
- $Mult$ multiplier
- $In1$ input 1
- $In2$ input 2
- Out output
- Ab abnormal
- Sum sum
- $Prod$ product

Knowledge representation: behavioural model

Example 11. Note: all the variables are universally quantified.

Behavior of an additioner

- $Add(x) \wedge \neg Ab(x) \wedge In1(x, u) \wedge In2(x, v) \wedge Sum(u, v, w) \Rightarrow Out(x, w)$
- $Add(x) \wedge \neg Ab(x) \wedge In1(x, u) \wedge Out(x, w) \wedge Sum(u, v, w) \Rightarrow In2(x, v)$
- $Add(x) \wedge \neg Ab(x) \wedge Out(x, w) \wedge In1(x, u) \wedge Sum(u, v, w) \Rightarrow In1(x, u)$

Behavior of a multiplier

- $Mult(x) \wedge \neg Ab(x) \wedge In1(x, u) \wedge In2(x, v) \wedge Prod(u, v, w) \Rightarrow Out(x, w)$
- $Mult(x) \wedge \neg Ab(x) \wedge In1(x, u) \wedge Out(x, w) \wedge Prod(u, v, w) \Rightarrow In2(x, v)$
- $Mult(x) \wedge \neg Ab(x) \wedge Out(x, w) \wedge In1(x, u) \wedge Prod(u, v, w) \Rightarrow In1(x, u)$

Knowledge representation: structural model

Example 12. Topology, structural model: $COMP = \{a1, a2, m1, m3, m3\}$ $Add(a1)$; $Add(a2)$; $Mult(m1)$; $Mult(m2)$; $Mult(m3)$ Connections: use of the equality

- $Out(m1, u) \wedge In1(a1, v) \Rightarrow u = v$
- $Out(m2, u) \wedge In2(a1, v) \Rightarrow u = v$
- $Out(m2, u) \wedge In1(a1, v) \Rightarrow u = v$
- $Out(m3, u) \wedge In1(a2, v) \Rightarrow u = v$
- $In2(m1, u) \wedge In1(m3, v) \Rightarrow u = v$

Knowledge representation: observations

Example 13. Only the inputs and the output of the circuit are *observable*.

- $In1(m1, 3)$: “The input 1 of the multiplier 1 is 3”
- $In2(m1, 2)$ ....
- $In1(m2, 2)$
- $In2(m2, 3)$
- $In1(m3, 2)$
- $In2(m3, 3)$
- $Out(a1, 10)$
- $Out(a2, 12)$

3.2 Diagnosis: an intuition

Main idea

Definition 14. A *State* of the system $SD, COMP$ is a sentence Φ_Δ with $\Delta \subseteq COMP$ like:

$$\bigwedge_{c\in\Delta} Ab(c) \wedge \bigwedge_{c\notin\Delta} \neg Ab(c)$$

The component of Δ are *abnormal*.

Example 15.

1. $\Delta = \{a1, m2\}$; $\Phi_\Delta = Ab(a1) \wedge Ab(m2) \wedge \neg Ab(a2) \wedge \neg Ab(m1) \wedge \neg Ab(m3)$
2. $\Delta = \emptyset$; $\Phi_\Delta = \neg Ab(a1) \wedge \neg Ab(a2) \wedge \neg Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$: state where every component has a normal behaviour
3. $\Delta = \{a1, a2, m1, m2, m3\}$; $\Phi_\Delta = Ab(a1) \wedge Ab(a2) \wedge Ab(m1) \wedge Ab(m2) \wedge Ab(m3)$: where every component has an abnormal behaviour

Main idea

Definition 16. A *Diagnosis* of the system $SD, COMP$ is a state Φ_Δ such that:

$$SD, OBS, \Phi_\Delta \text{ is satisfiable}$$

The state is *possible* according to SD, OBS (consistency-based).

Definition 17. A diagnosis exists iff:

$$SD, OBS \text{ is satisfiable}$$

If not, the model is not well-designed or incomplete.

Detection of abnormalities

Definition 18. *Normal behaviour* of the system:

$$SD, \Phi_\emptyset$$

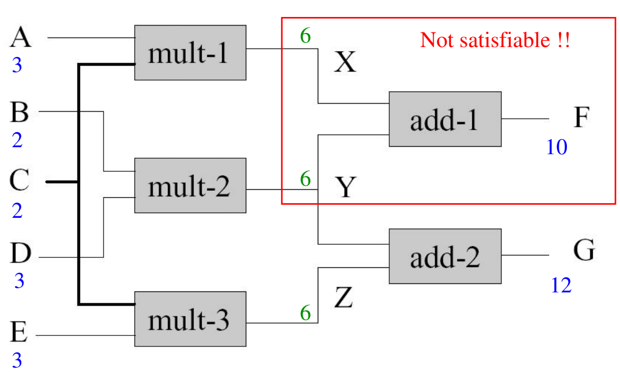
where $\Phi_\emptyset = \bigwedge_{c\in COMP} \neg Ab(c)$.

Definition 19. How to detect *abnormal observations* OBS?
Check the satisfiability of:

$$SD, \Phi_\emptyset, OBS$$

Detection of abnormalities: example

Example 20. In the presented example, $SD, OBS, \Phi_\emptyset$ is unsatisfiable so OBS is an abnormal observations.



Identification of abnormalities

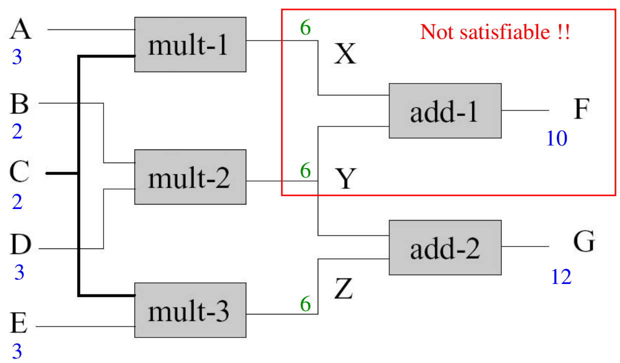
Definition 21. If we have detected that the observations are abnormal, we need to *identify* which components are faulty. We need satisfiability back!!

$$SD, OBS \models Ab(?) \vee ... \vee Ab(?)$$

- Which abnormalities are entailed by SD, OBS?
- Use of inference algorithms to solve that problem.

Identification of abnormalities: example

Example 22. $SD, OBS \models Ab(a1) \vee Ab(m1) \vee (Ab(m2) \wedge Ab(a2)) \vee (Ab(m2) \wedge Ab(m3))$



Identification of abnormalities: example

Example 23. $SD, OBS \models Ab(a1) \vee Ab(m1) \vee (Ab(m2) \wedge Ab(a2)) \vee (Ab(m2) \wedge Ab(m3))$
From that, we guess the following set of states are diagnoses:

1. $Ab(a1) \wedge \neg Ab(a2) \wedge Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$
   - only a1 and m1 are faulty
2. $\neg Ab(a1) \wedge Ab(a2) \wedge \neg Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$
   - only a2 is faulty
3. $Ab(a1) \wedge Ab(a2) \wedge Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$
   - a1, a2, and m1 are faulty
4. $Ab(a1) \wedge Ab(a2) \wedge Ab(m1) \wedge Ab(m2) \wedge Ab(m3)$
   - everything can be faulty!!!
5. ...

But the following state is not a diagnosis state:

1. $\neg Ab(a1) \wedge \neg Ab(a2) \wedge \neg Ab(m1) \wedge \neg Ab(m2) \wedge Ab(m3)$
   - if m3 is faulty there must another faulty component (m2 at least)

Failure knowledge

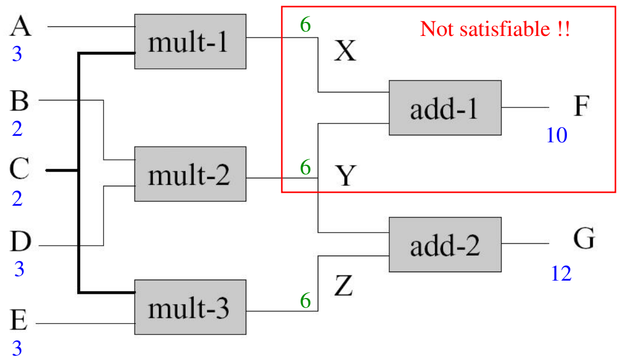
Definition 24. *Failure knowledge*: piece of knowledge about the behaviour of components when they are faulty

Example 25.

- "When faulty, the output of the additioner 2 is always 0"
- $Ab(a2) \Rightarrow Out(a2, 0)$
- "Faulty additioners behave like substracters"
- $Add(x) \wedge Ab(x) \wedge In1(x, u) \wedge In2(x, v) \wedge Substract(u, v, w) \Rightarrow Out(x, w)$

Identification of abnormalities: example 2

Example 26. $SD, \{Ab(a2) \Rightarrow Out(a2, 0)\}, OBS \vDash (Ab(a1) \wedge \neg Ab(a2)) \vee (\neg Ab(a2) \wedge Ab(m1)) \vee (\neg Ab(a2) \wedge Ab(m2) \wedge Ab(m3))$



Identification of abnormalities: example 2

Example 27. $SD, \{Ab(a2) \Rightarrow Out(a2, 0)\}, OBS \vDash (Ab(a1) \wedge \neg Ab(a2)) \vee (\neg Ab(a2) \wedge Ab(m1)) \vee (\neg Ab(a2) \wedge Ab(m2) \wedge Ab(m3))$

From that, we guess the following set of states are diagnoses:

1. $Ab(a1) \wedge \neg Ab(a2) \wedge Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$
   - only a1 and m1 are faulty
2. $\neg Ab(a1) \wedge \neg Ab(a2) \wedge \neg Ab(m1) \wedge Ab(m2) \wedge Ab(m3)$
   - m1 and m2 are faulty
3. $Ab(a1) \wedge \neg Ab(a2) \wedge Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$
   - m1 and a1 are faulty
4. ...

But the following state is not a diagnosis state:

1. $\neg Ab(a1) \wedge Ab(a2) \wedge \neg Ab(m1) \wedge Ab(m2) \wedge Ab(m3)$
   - any hypothesis where a2 is faulty is not a diagnosis any more

Diagnosis representation: Partial Diagnosis

For n components, the number of potential diagnoses is 2^n. We need a clever representation.

Definition 28. A *Partial Diagnosis* is a conjunction Φ of Ab literals such that every state Φ' *covered* by Φ is a diagnosis.

Example 29.

- $\Phi = Ab(a1) \wedge \neg Ab(a2) \wedge Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$ is a diagnosis so it is a partial diagnosis
- $\Phi = Ab(a1) \wedge Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$ is a partial diagnosis because it covers the two diagnoses
  - $\Phi' = Ab(a1) \wedge \neg Ab(a2) \wedge Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$
  - $\Phi' = Ab(a1) \wedge Ab(a2) \wedge Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$

Diagnosis representation: Kernel Diagnosis

Definition 30. A *Kernel Diagnosis* is a partial diagnosis that is covered by only itself. Kernel diagnoses provide a very economical way to implicitly represent all the diagnoses.

Example 31. Example 1:

- $Ab(a1)$ is a kernel diagnosis for example 1. Every conjunction covered by $Ab(a1)$ is a partial diagnosis. The empty clause $\emptyset$ is not a kernel diagnosis because it covers $\neg Ab(a1)$ which is not a partial diagnosis.
- $Ab(m1)$, $Ab(m2) \wedge Ab(a2)$, $Ab(m2) \wedge Ab(m3)$ are the other kernel diagnoses of example 1.

Example 2:

- $Ab(a1) \wedge \neg Ab(a2)$, $\neg Ab(a2) \wedge Ab(m1)$, $\neg Ab(a2) \wedge Ab(m2) \wedge Ab(m3)$

Diagnosis representation: Preferences

A diagnosis is an *hypothesis* (it may be true) and not a *conclusion*. So we may decide to *prefer* some of these diagnoses.

- Diagnoses with a minimal number of abnormal components
- Diagnoses with a set of abnormal components that is minimal: *minimal diagnoses*
  - i.e. if I remove one component from this set (it becomes normal) the corresponding state is not a diagnosis anymore
- Diagnoses that "explain in the best way" the observations: *explanation*

Diagnosis representation: Preferences

Example 32. Example 1:

- **2 diagnoses with minimal cardinality**
  1. $Ab(a1) \wedge \neg Ab(a2) \wedge \neg Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$
  2. $\neg Ab(a1) \wedge \neg Ab(a2) \wedge Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$
- **4 minimal diagnoses**
  1. $Ab(a1) \wedge \neg Ab(a2) \wedge \neg Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$ (same as above)
  2. $\neg Ab(a1) \wedge \neg Ab(a2) \wedge Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$ (same as above)
  3. $\neg Ab(a1) \wedge Ab(a2) \wedge Ab(m1) \wedge Ab(m2) \wedge \neg Ab(m3)$
  4. $\neg Ab(a1) \wedge \neg Ab(a2) \wedge \neg Ab(m1) \wedge Ab(m2) \wedge Ab(m3)$

Explanation

Definition 33. A diagnosis Φ_Δ for an observed system $(SD, COMP, OBS)$ is an *explanation* for an elementary observation $o \in OBS$ iff

$$SD, \Phi_\Delta \models o$$

1. select diagnoses that explain all the observations of OBS
2. select diagnoses that explain a biggest subset of OBS
3. select diagnoses that explain the biggest subset of OBS

Explanation: example

Example 34. Example 1:

All the diagnoses that cover the following sentence (which is not a partial diagnosis) are explanations of $Out(a2, 12)$

$$\neg Ab(m2) \wedge \neg Ab(m3) \wedge \neg Ab(a2)$$

for instance:

$$Ab(a1) \wedge \neg Ab(a2) \wedge \neg Ab(m1) \wedge \neg Ab(m2) \wedge \neg Ab(m3)$$

Summary

- Be careful between *Deduction* and *Abduction*
- *Diagnosis* reasoning is generally close to *Abduction*
- Model-based diagnosis for static systems
  - Description of a model with FOL (structural/behavioural model)
  - Use of Failure knowledge in the model
- Diagnosis:
  - Detection is satisfiability problem
  - Identification consists in retrieving the satisfiability
- Diagnosis representation:
  - Kernel diagnosis: an efficient way to represent all the diagnoses.
- Diagnosis preference:
  - Minimal diagnoses, Explanations