Rules, Causality and Constraints. Model-Based Reasoning and Structural Knowledge Discovery

Antoni Ligęza

ligeza@agh.edu.pl

AGH

AGH University of Science and Technology
Kraków, Poland

Computational Intelligence and Data Mining
5-th International Workshop
Martinske hole, September 22-25, 2017

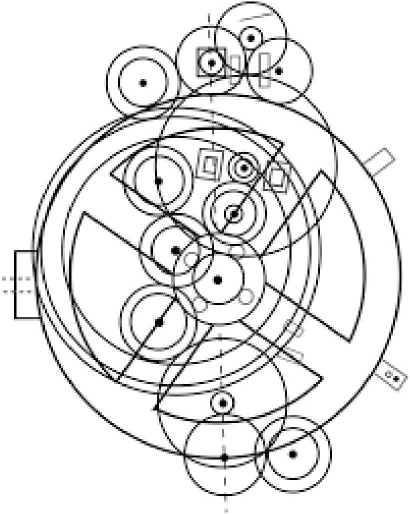
Outline

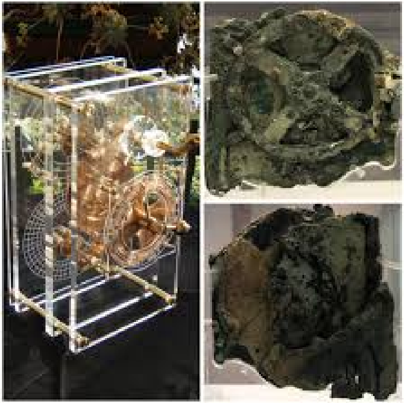
Presentation Outline

An Eternal Question: How Does it Work?

Figure : The Antikythera mechanism; recovered on May 17, 1901. The instrument has been variously dated to about 87 BC, or between 150 and 100 BC, or in 205 BC `https://en.wikipedia.org/wiki/Antikythera_mechanism`

How Does it Work? Model-Based Reasoning





Components + Connections + Causality = Operation

Presentation Outline

A Typical Data Set: Inputs and Outputs

	Attributes			Decision
Case	Temperature	Headache	Nausea	Flu
1	high	yes	no	yes
2	very_high	yes	yes	yes
3	high	no	no	no
4	high	yes	yes	yes
5	high	yes	yes	no
6	normal	yes	no	no
7	normal	no	yes	no
8	normal	yes	no	yes

Hypothesis:

$$Y = f(X_1, X_2, \dots X_k)$$

Shallow Rule Induction – A Naive Example

Car color	Car turns
red	left
red	left
⋮	⋮
black	right
black	right
⋮	⋮

$$Car\_color = red \longrightarrow Car\_turns = left$$

$$Car\_color = black \longrightarrow Car\_turns = right$$

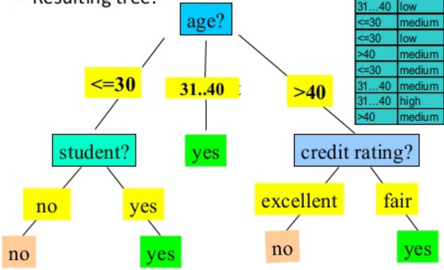
```
car_turns(X,left) :- drives(X,university).
car_turns(X,right) :- drives(X,court).
drives(X,university) :- young(X).
drives(X,court) :- old(X).
young(X) :- write(X), write(' is young and so preferes red cars.').
old(X) :- write(X), write(' is old and so preferes black cars.').
```

Typical Induced Output: Trees or Rules

Decision Tree Induction: An Example

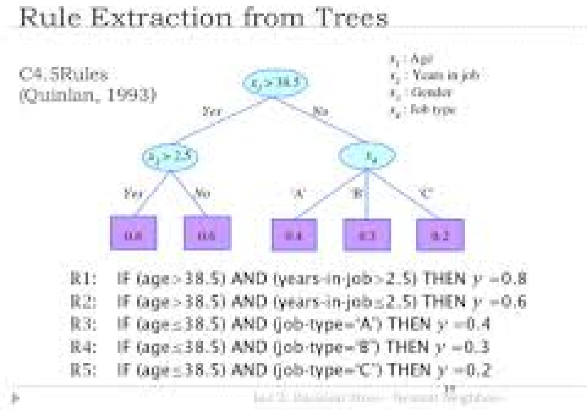
- Training data set: Buys_computer
- The data set follows an example of Quinlan's ID3 (Playing Tennis)
- Resulting tree:

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no



10

Typical Induced Output: Trees or Rules



Problem: shallow knowledge $\Longrightarrow$ Does work — but why?

Bayes Nets: *Causal* Model ?

A further step on...

COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE 126 (2016) 128–142

ELSEVIER

journal homepage: www.intl.elsevierhealth.com/journals/cmpb

Bayesian network modeling: A case study of an epidemiologic system analysis of cardiovascular risk

CrossMark

P. Fuster-Parra [a,b,], P. Tauler [b], M. Bennasar-Veny [b], A. Ligęza [c], A.A. López-González [d], A. Aguiló [b]*

[a] *Department of Mathematics and Computer Science, Universitat Illes Balears, Palma de Mallorca, Baleares E-07122, Spain*
[b] *Research Group on Evidence, Lifestyles & Health, Research Institute on Health Sciences (IUNICS), Universitat Illes Balears, Palma de Mallorca, Baleares E-07122, Spain*
[c] *Department of Applied Computer Science, AGH University of Science and Technology, Kraków PL-30-059, Poland*
[d] *Prevention of Occupational Risks in Health Services, GESMA, Balearic Islands Health Service, Hospital de Manacor, Manacor, Baleares E-07500, Spain*

A R T I C L E I N F O

Article history:
Received 18 August 2015
Received in revised form
28 November 2015
Accepted 11 December 2015

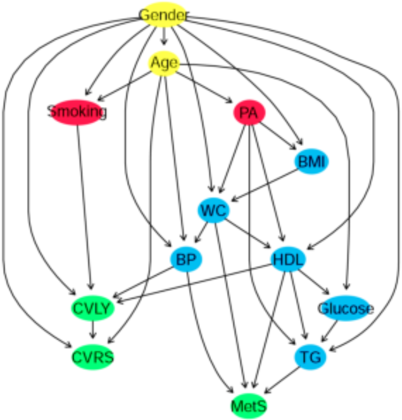
Keywords:
Bayesian networks
Model averaging
Cardiovascular lost years
Cardiovascular risk score
Metabolic syndrome
Causal dependency discovery

A B S T R A C T

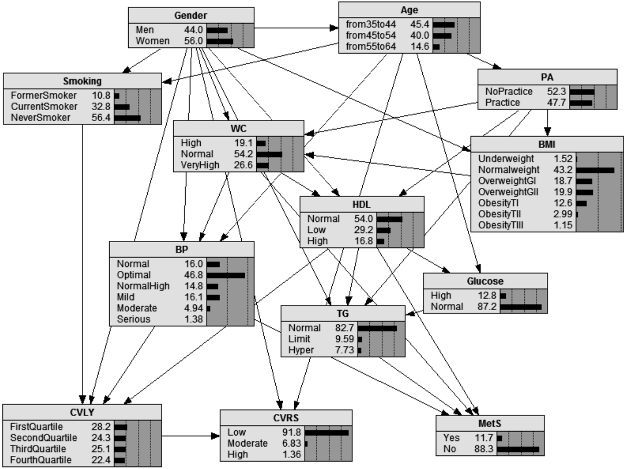
An extensive, in-depth study of cardiovascular risk factors (CVRF) seems to be of crucial importance in the research of cardiovascular disease (CVD) in order to prevent (or reduce) the chance of developing or dying from CVD. The main focus of data analysis is on the use of models able to discover and understand the relationships between different CVRF. In this paper a report on applying Bayesian network (BN) modeling to discover the relationships among thirteen relevant epidemiological features of heart age domain in order to analyze *cardiovascular lost years* (CVLY), *cardiovascular risk score* (CVRS), and *metabolic syndrome* (MetS) is presented. Furthermore, the induced BN was used to make inference taking into account three reasoning patterns: *causal* reasoning, *evidential* reasoning, and *intercausal* reasoning. Application of BN tools has led to discovery of several direct and indirect relationships between different CVRF. The BN analysis showed several interesting results, among them: CVLY was highly influenced by smoking being the group of men the one with highest risk in CVLY; MetS was highly influence by physical activity (PA) being again the group of men the one with highest risk in MetS, and smoking did not show any influence. BNs produce an intuitive, transparent, graphical representation of the relationships between different CVRF. The ability of BNs to predict new scenarios when hypothetical information is introduced makes BN modeling an Artificial Intelligence (AI) tool of special interest in epidemiological studies. As CVD is multifactorial the use of BNs seems to be an adequate modeling tool.

* *Corresponding author at:* Department of Mathematics and Computer Science, Universitat Illes Balears, Palma de Mallorca, Baleares E-07122, Spain. Tel.: +34 971171386.
E-mail address: pilar.fuster@uib.es (P. Fuster-Parra).

http://dx.doi.org/10.1016/j.cmpb.2015.12.010



Bayes Nets: Even More Precise Model

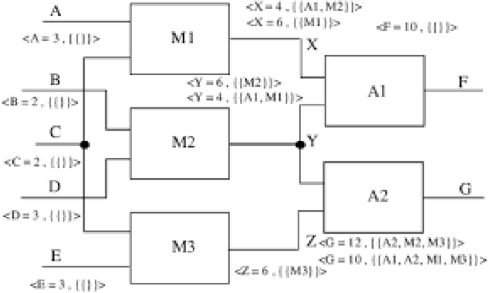


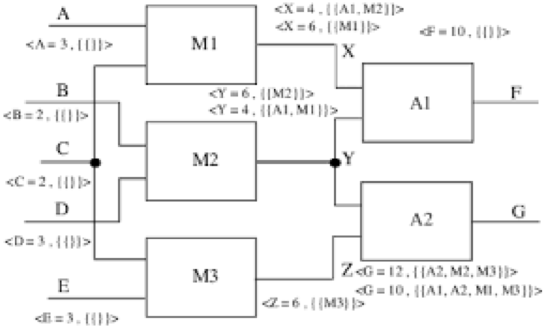
Presentation Outline

Discovering Causal Structure: Motivation:

- majority of ML models cover *shallow* knowledge only,
- most of them are on decision/classification type; no *functional* output,
- often: fuzzy/rough/probabilitic output,
- no investigation of the *guts* — what is inside?
- starting point: diagnostic reasoning.

- variables, values, signals,
- components,
- links,
- internal structure,
- input — internal state — output,
- operation,
- constraints,
- functionality.





Modeling Causal Structure with Rules

$\text{ADD}(x) \wedge \neg\text{AB}(x) \Rightarrow \mathit{Output}(x) = \mathit{Input}1(x) + \mathit{Input}2(x)$,
$\text{MULT}(x) \wedge \neg\text{AB}(x) \Rightarrow \mathit{Output}(x) = \mathit{Input}1(x) * \mathit{Input}2(x)$,
$\text{ADD}(a1)$, $\text{ADD}(a2)$, $\text{MULT}(m1)$, $\text{MULT}(m2)$, $\text{MULT}(m3)$,
$\text{Output}(m1) = \text{Input1}(a1)$, $\text{Output}(m2) = \text{Input2}(a1)$,
$\text{Output}(m2) = \text{Input1}(a2)$, $\text{Output}(m3) = \text{Input2}(a2)$,
$\text{Input2}(m1) = \text{Input1}(m3)$,
$\mathit{Input}1(m1) = A \ldots \mathit{Output}(a2) = G$

Towards Model-Based Reasoning

Modeling Internal/Causal Structure with Rules: The HeKatE/XTT Approach

(?) mth	(->) ssn
in {January,February,December}	:= Summer
in {March,April,May}	:= Autumn
in {June,July,August}	:= Winter
in {September,October,November}	:= Spring

Table id: tab_2 - ms

(?) day	(->) today
in [Tuesday,Friday]	:= workday
in {Saturday,Sunday}	:= weekend

Table id: tab_3 - Table2

(?) today	(?) hour	(->) oper
= workday	in [9,17]	:= during_business_hours
= workday	< 9	:= not_during_business_hours
= workday	> 17	:= not_during_business_hours
= weekend	= any	:= not_during_business_hours

Table id: tab_4 - Table3

(?) ssn	(?) oper	(->) therm
= Spring	= during_business_hours	:= 20
= Spring	= not_during_business_hours	:= 15
= Summer	= during_business_hours	:= 24
= Summer	= not_during_business_hours	:= 27
= Autumn	= during_business_hours	:= 20
= Autumn	= not_during_business_hours	:= 16
= Winter	= during_business_hours	:= 18
= Winter	= not_during_business_hours	:= 14

Table id: tab_5 - Table4

Types of rules

- IF-THEN rules; interpreted forwards (deduction) or backwards (abduction):

$$p_1 \wedge p_2 \wedge \ldots p_k \longrightarrow h$$

$$h : - \; p_1 \wedge p_2 \wedge \ldots p_k.$$

- facts,
- constraints:
  - positive (disjunction; must-hold):

$$q_1 \vee q_2 \vee \ldots q_k$$

  - negative (conjunction; must-not-hold);

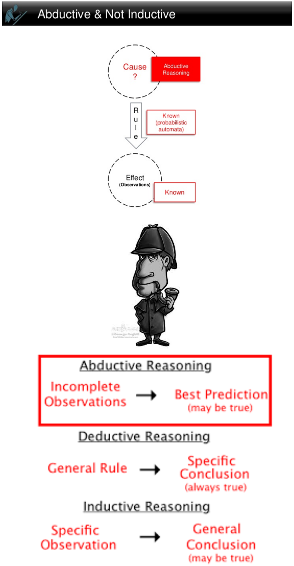
$$\neg q_1 \wedge \neg q_2 \wedge \ldots \neg q_k$$

- functional: calculations or equations (exact numbers),
- functional: qualitative,
- functional: defined with aggregation operators.

Motivation. Towards Exact Model-Based Reasoning

Some loosely provocative questions and statements...

- abduction: what, why and where — what for?
- abduction: investigation of causality,
- abduction: a method of logical inference (but invalid),
- abduction vs. deduction,
- abduction: primary method used by Sherlock Holmes!
- abduction: inevitable ambiguity (potential/admissible solutions; many of them),
- abduction: more constraints — better abduction,
- abduction + constraints + SAT (minimal models).



Motivation. Towards Exact Model-Based Reasoning

Abduction

- Abduction — principal way of problem solving — generation of hypotheses,
- Abduction — performed with backtracking search,
- Abduction — produces numerous, admissible solutions

Abduction: Logical model

$$\frac{\alpha \Longrightarrow \beta, \beta}{\alpha}$$

$$HYP^{+} \cup HYP^{-} \cup KB \models OBS^{+} \cup OBS^{-}$$

$$HYP^{+} \cup HYP^{-} \cup KB \cup OBS^{+} \cup OBS^{-} \not\models \perp$$

An intuitive example: find explanations for *wet_street*

- $rain \longrightarrow water$
- $sprinkler \longrightarrow water$
- $snow \wedge temperature \longrightarrow water$
- $water \longrightarrow wet\_street$,
- $cleaning \longrightarrow wet\_street$
- $oil \longrightarrow wet\_street$

Level of Details in Structure Discovery

Values of variables:

- binary 0/1; true/false,
- ternary -/0/+, qualitative,
- integer numbers.

Connections:

- existence,
- direction,
- breaks,
- shortcuts,
- complex.

Components:

- parametric identification,
- selection one-of,
- function discovery.

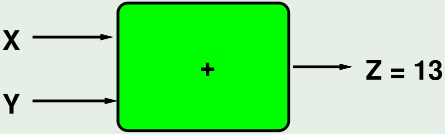
Overall Structure:

- causality,
- structure - causal graph,
- logical and functional dependencies.

Explaining the role of constraints in abduction

Abductive problem without constraints

- X, Y, Z - variables, $X, Y, Z \in \{0, 1, 2, \ldots, 9\}$,
- system: $Z = X + Y$



- Observed: $Z = 13$
- Possible explanations:
  - ($X = 4$ and $Y = 9$),
  - ($X = 5$ and $Y = 8$),
  - ... ,
  - ($X = 9$ and $Y = 4$).
- 6 admissible solutions.

The role of constraints in abduction

Abductive problem with constraints

- X, Y, Z - variables, $X, Y, Z \in \{0, 1, 2, \ldots, 9\}$,

$$Z = X + Y$$

- **Constraint:**

$$Y < X - 3$$

- Observed: $Z = 13$
- Possible explanations: $(X = 9$ and $Y = 4)$,
- 1 admissible solution.

Conclusion

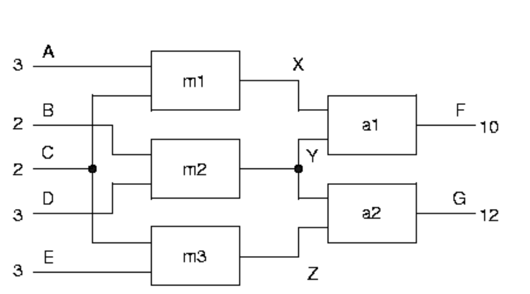
- CONSTRAINTS can refine results of ABDUCTION; less models generated,
- propagation of CONSTRAINTS can reduce computational effort,
- ABDUCTION + CONSTRAINTS = CONSTRUCTIVE ABDUCTION

The example problem

The Paradigm to be Explored Further on

MODEL-BASED REASONING =
COMPONENTS + STRUCTURE + CONSTRAINTS + CAUSALITY

The Multiplier-Adder System




```
[...]
A #= 3,
B #= 2,
C #= 2,
D #= 3,
E #= 3,
F #= 10,
G #=12,
[...]
```

Presentation Outline

Abduction, Diagnosis, Constraints: a Recapitulation

- Abductive Model-Based Diagnosis/Consistency-Based Diagnosis — main ideas:
  - SYSTEM vs. MODEL: discrepancy $\Rightarrow$ *misbehavior*,
  - CONFLICTS: find all *conflict sets*,
  - DIAGNOSES: minimal *hitting sets*.
- Abductive Consistency-Based Diagnosis — output:
  - multiple-fault diagnoses,
  - minimal diagnoses,
  - binary fault evaluation (no further evaluation of fault type),
  - *numerous* potential diagnoses,
- CSP — Constraint Satisfaction Problem for diagnosis:
  - multiple modes of component behavior
  - more precise diagnoses,
  - elimination of spurious behavior models.
- Qualitative vs. numerical models:
  - *modes* of faulty behavior: binary, qualitative, numerical,
  - more efficient elimination of inconsistency and spurious models.

Presentation Outline

Constraint Satisfaction Problem

CSP statement

- $X = \{X_1, X_2, \ldots, X_k\}$ — variables, $D = \{D_1, D_2, \ldots, D_k\}$ — their domains,
- $C = \{(S_i, R_i) \colon i = 1, 2, \ldots, n\}$ — constraints; S_i — scope; R_i — relation.

CSP solution

A solution to CSP: (X, D, C) — any assignment of values to variables of X:

$$\{X_1 = d_1, X_2 = d_2, \ldots, X_k = d_k\},$$

where $d_i \in D_i$, and for any constraint in $(S_i, R_i) \in C$, R_i is satisfied.

A CSP Example

CSP vs. DP

All vs. first solution

- DP: *all* potential solutions,
- CSP: a *single* solution.

Binary vs. finite domains

- DP: binary domains (i.e. component is OK or faulty),
- CSP: finite discrete domains.

general vs. specific models

- DP: domain specific models,
- CSP: generic models.

Presentation Outline

An Example System: Multiplier-Adder

The multiplier-adder system to be diagnosed

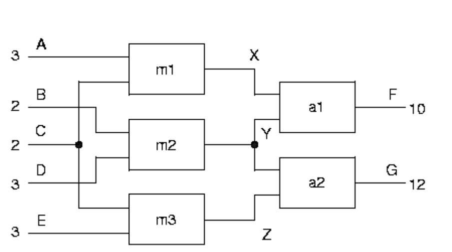


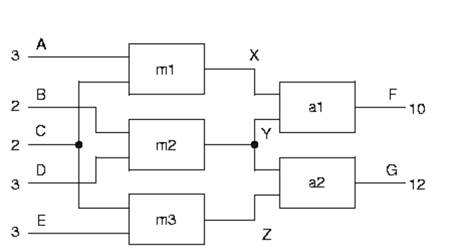
Figure : An example arithmetic system

The basic diagnostic matrix

M/F	$m1$	$m2$	$m3$	$a1$	$a2$
F	1	1		1	
G		1	1		1

Abductive Consistency-based Diagnosis

The multiplier-adder system to be diagnosed

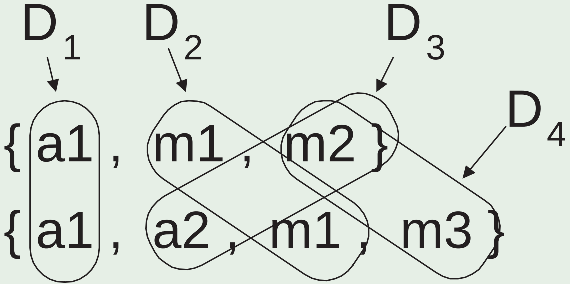


Consistency-Based Diagnosis

- MISBEHAVIOR: F=10 (should be 12); note that G=12 is O.K.
- ABDUCTION — CONFLICTS: $\{a1, m1, m2\}$,$\{a1, m1, a2, m3\}$,
- REPAIR — DIAGNOSES: $\{a1\}, \{m1\}, \{m2, m3\}, \{a2, m2\}$.

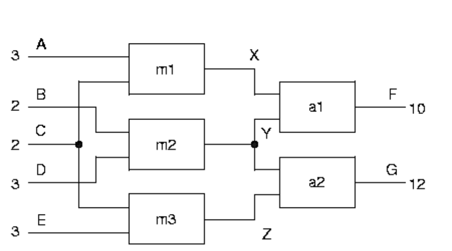
Diagnoses as Hitting Sets

Calculating the Diagnoses



An Example System: Multiplier-Adder

The multiplier-adder system to be diagnosed



The complete diagnostic matrix

M/F	$m1$	$m2$	$m3$	$a1$	$a2$
F	1	1		1	
G		1	1		1
$F-G$	1		1	1	1

Presentation Outline

Multiple-Fault Diagnosis with Diagnostic Matrices: New Ideas

Principles of multiple-fault diagnostic approach

- providing a **new interpretation** of the matrices with rules – the new rules should follow the **causal direction of inference** (i.e. from faults which are the initial causes to manifestations),
- introducing **two types of diagnostic matrices**, each of them having different logical interpretation, one with logical **OR-type** meaning and another one with logical **AND-type meaning**,
- as a consequence, introducing **two types of causal rules**, each of them having different logical interpretation, one with logical **OR-type** meaning and another one with logical **AND-type meaning**,
- introducing a **two-level knowledge representation** with OR matrices at the lower level and AND matrices in the upper one,

Conjunctive and Disjunctive Faults

Disjunctive conceptual faults = Conflicts

A **Disjunctive Conceptual Faults** or **Intermediate Conceptual Fault** (a DCF or an ICF, for short), is a hypothesis that a certain set of components must contain a faulty component under certain set of manifestations observed. A particular DCF_i can be expressed as a set of faults, $DCF_i = \{f^1, f^2, \ldots, f^{j_i}\}$ or logically, as a disjunction $DCF_i = f^1 \vee f^2 \vee \ldots \vee f^{j_i}$. Disjunctive rules is:

$$rule_{i\_or} : f^1 \vee f^2 \vee \ldots \vee f^{j_i} \longrightarrow m_i \tag{1}$$

Conjunctive conceptual faults = Diagnoses

A **Conjunctive Conceptual Fault** (a CCF, for short) is the hypothesis that several faults occur at the same time. A particular CCF_i can be expressed as a set of faults, $CCF_i = \{f^1, f^2, \ldots, f^{j_i}\}$ or logically, as a conjunction $CCF_i = f^1 \wedge f^2 \wedge \ldots \wedge f^{j_i}$. Conjunctive rules is:

$$rule_{i\_and} : f^1 \wedge f^2 \wedge \ldots \wedge f^{j_i} \longrightarrow m_i \tag{2}$$

Disjunctive Matrix and Rules

Disjunctive diagnostic matrix

Table : An OR binary diagnostic matrix for the adder system (the lower level)

DCF	$m1$	$m2$	$m3$	$a1$	$a2$
DCF_1 (F)	1	1		1	
DCF_2 (F-G)	1		1	1	1
DCF_3 (G)		1	1		1

Disjunctive causal rules

$$
\begin{aligned}
rule_{1\_or} &: m1 \vee m2 \vee a1 \longrightarrow DCF_1 \\
rule_{2\_or} &: m1 \vee m3 \vee a1 \vee a2 \longrightarrow DCF_2 \\
rule_{3\_or} &: m2 \vee m3 \vee a2 \longrightarrow DCF_3
\end{aligned} \tag{3}
$$

Conjunctive Matrix and Rules

Conjunctive diagnostic matrix

Table : An AND binary diagnostic matrix for the adder system (the upper level)

M	DCF_1	DCF_2	DCF_3
F*, G, (F-G)*	1	1	
F, G*, (F-G)*		1	1
F*, G*, F-G	1		1
F*, G*, (F-G)*	1	1	1

Conjunctive causal rules

$$
\begin{aligned}
&rule_{1\_and}: DCF_1 = 1 \land DCF_2 = 1 \longrightarrow F^*, G, (F-G)^* \\
&rule_{2\_and}: DCF_2 = 1 \land DCF_3 = 1 \longrightarrow F, G^*, (F-G)^* \\
&rule_{3\_and}: DCF_1 = 1 \land DCF_3 = 1 \longrightarrow F^*, G^* \\
&rule_{4\_and}: DCF_1 = 1 \land DCF_2 = 1 \land DCF_3 = 1 \longrightarrow F^*, G^*, (F-G)^*
\end{aligned}
\tag{4}
$$

Presentation Outline

The Two-Layer Approach: Causal Graph

Multiplier-adder: causal graph for multiple-fault diagnoses

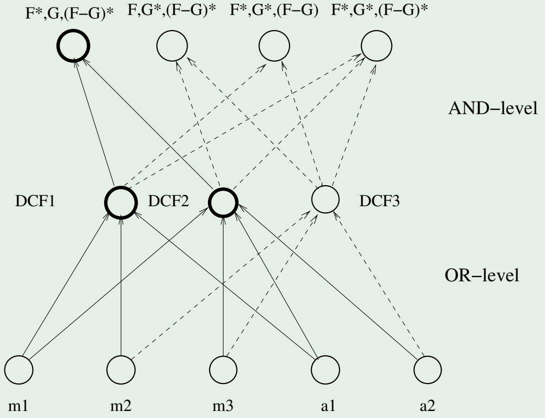


Figure : An AND/OR causal graph for the example multiplier-adder system

Final Diagnoses

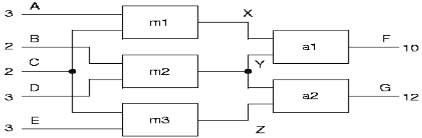
Multiplier-adder: final multiple-fault diagnoses

Table : Final possible diagnoses

Manifestations	Diagnoses
F^*,G, $(F\text{-}G)^*$	$\{a1\}$, $\{m1\}$, $\{a2, m2\}$, $\{m2, m3\}$
F, G^*, $(F\text{-}G)^*$	$\{a2\}$, $\{m3\}$, $\{a3, m2\}$, $\{m1, m2\}$,
F^*, G^*, (F-G)	$\{m2\}$, $\{a1, a2\}$, $\{a1, m3\}$, $\{a2, m1\}$, $\{m1, m3\}$
F^*, G^*, $(F\text{-}G)^*$	$\{a1, a2\}$, $\{a1, m2\}$, $\{a1, m3\}$, $\{a2, m1\}$, $\{a2, m2\}$,$\{m1, m2\}$, $\{m2, m3\}$, $\{m1, m3\}$

Presentation Outline

A CSP Framework for Extended DP



A CSP like diagnostic problem statement

- $O = \{A, B, C, D, E, F, G\}$ — observable variables,
- $H = \{X, Y, Z\}$ — hidden variables,
- $D = \{m1, m2, m3, a1, a2\}$ — diagnostic variables,
- $V = O \cup H \cup D$ — all variables,
- $\{-, 0, +\}$ — extended domains of diagnostic variables,
- M — model (the set of equations),
- OBS — current observations,
- **MORE PRECISE DIAGNOSES — Qualitative Diagnoses**,
- **ADDITIONAL CONSTRAINTS — Elimination of Spurious Diagnoses**.

Presentation Outline

Qualitative Notation

Component behavior

- $c(0)$ — component c is correct; for intuition, 0 stands for *nominal behavior*,
- $c(-)$ — component c is incorrect, it lowers down the signal,
- $c(+)$ — component c is incorrect, it increases the signal.

Shorthand Notation

$$c(0|+) = c(0) \vee c(+)$$

$$c(0|-) = c(0) \vee c(-)$$

$$c(-|+) = c(-) \vee c(+)$$

Inconsistency Detection

$$c(0) \wedge c(-)$$

$$c(0) \wedge c(+)$$

$$c(-) \wedge c(+)$$

Qualitative Diagnostic Approach

Qualitative Conflict

A **Qualitative Conflict** (QC for short) or a *Qualitative Disjunctive Conceptual Fault* (QDCF) is any set of the form

$$QDCF = \{c_1(\#), c_2(\#), \ldots, c_k(\#)\}$$

such that under the current observations all the elements $c_1, c_2, \ldots, c_k$ cannot be working together correctly, and for $\# \in \{-, +, -/+\}$ the specification covers possible explanations of the observed behavior.

Example Qualitative Conflicts

$$QDCF_1 = \{m1(-), m2(-), a1(-)\}$$

$$QDCF_2 = \{m1(-|+), m3(-|+), a1(-|+), a2(-|+)\}$$

Qualitative Diagnostic Approach

Qualitative Diagnosis

A (minimal) **Qualitative Diagnosis**

$$D = \{d_1(\#\}, d_2(\#), \ldots, d_n(\#)$$

is any minimal hitting set for all the *QDCF*-s, satisfying the following conditions:

- D is internally consistent (i.e. it does not contain a pair $d(-)$ and $d(+)$),
- D is consistent with observations, i.e.

$$SD \cup OBS \cup \{d(-|+)|d \in D\} \cup \{d(0)|d \in (COMP \setminus D)\}$$

is consistent.

Example Qualitative Diagnoses

$$\{a1(-)\}$$

$$\{m2(-), m3(+)\}$$

New: Qualitative Signal Composition

Extended Table

	I1(–)		I1(0)	I1(+)	
I2(-)	O'(–)	(1)	O(–)	O'(?)	(2)
I2(0)	O(–)		O(0)	O(+)	
I2(+)	O'(?)	(3)	O(+)	O'(+)	(4)

Table : Definition of composition of qualitative values

Explanation of the 4 Specific Cases

1. $O'(-)$ denotes the output variable value in case of **simultaneous decrease of both of the inputs**; hence, plausibly:

$$O'(-) \leq O(-), \tag{5}$$

2. $O(?)$ can be $O(-)$, $O(0)$, and $O(+)$; in the first case plausibly:

$$O'(-) \geq O(-), \tag{6}$$

   while in the third case plausibly:

$$O'(+) \leq O(+), \tag{7}$$

3. $O(?)$ can be $O(-)$, $O(0)$, and $O(+)$; in the first case refer to (6), while in the third case refer to (7).
4. $O(+)$ denotes the output variable value in case of **simultaneous increase of both of the inputs**; hence, plausibly:

$$O'(+) \geq O(+), \tag{8}$$

Rules as additional constraints: I

Type 1 rules: normal inputs, faulty component rules

Assumption: $input1(Comp, 0)$ and $input2(Comp, 0)$

$$d(Comp, Mode) \longrightarrow output(Comp, Mode)$$

Example rules

$$d(m1, -) \longrightarrow output(m1, -)$$

$$d(m1, +) \longrightarrow output(m1, +)$$

$$d(a1, -) \longrightarrow output(a1, -)$$

$$d(a1, +) \longrightarrow output(a1, +)$$

There are 10 rules (2 for each component)

Rules as additional constraints: II

Type 2 rules: deviated inputs, normal component

Assumption: $d(Comp, 0)$

$$input1(Comp, Mode1) \wedge input2(Comp, Mode2) \longrightarrow output(Comp, Mode)$$

Example rules

Table : Behavior of correct component with deviated inputs.

inputs	-	0	+
-	-	-	?
0	-	0	+
+	?	+	+

$$input1(a1, -) \wedge input2(a1, 0) \longrightarrow output(a1, -)$$

$$input1(a1, -) \wedge input2(a1, -) \longrightarrow output(a1, -)$$

$$input1(a1, 0) \wedge input2(a1, +) \longrightarrow output(a1, +)$$

Rules as additional constraints: III

Type 3 rules: deviated inputs, faulty component rules

$$input1(Comp, M1) \wedge input2(Comp, M2) \wedge d(Comp, M3) \longrightarrow output(Comp, Mode)$$

Example rules

Table : Behavior of incorrect component with deviated inputs.

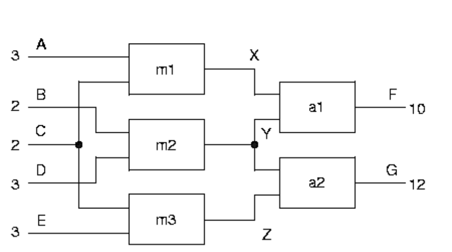
input1	input2	Component Mode	Output
-	-	-	-
-	0	-	-
0	-	-	-
0	0	-	-
+	+	+	+
+	0	+	+
0	+	+	+
0	0	+	+

$$input1(a2, +) \wedge input2(a2, 0) \wedge d(a2, +) \longrightarrow output(a2, +)$$

Presentation Outline

Qualitative Diagnoses: Back to Example

The multiplier-adder system to be diagnosed



$$QDCF_1 = \{m1(-), m2(-), a1(-)\}$$

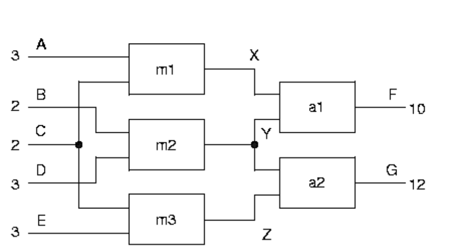
$$QDCF_2 = \{m1(-|+), m3(-|+), a1(-|+), a2(-|+)\}$$

Case: $\{m1(-)\}$

$$D = \{m1-\}$$

Qualitative Diagnoses: Back to Example

The multiplier-adder system to be diagnosed

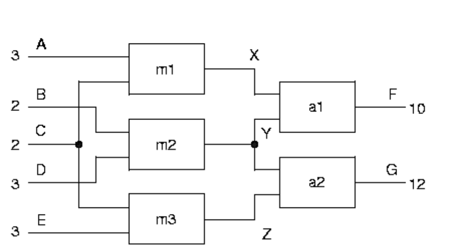


$$QDCF_1 = \{m1(-), m2(-), a1(-)\}$$
$$QDCF_2 = \{m1(-|+), m3(-|+), a1(-|+), a2(-|+)\}$$

Case: $\{a1(-)\}$

$$D = \{a1-\}$$

Qualitative Diagnoses: Back to Example



$$QDCF_1 = \{m1(-), m2(-), a1(-)\}$$
$$QDCF_2 = \{m1(-|+), m3(-|+), a1(-|+), a2(-|+)\}$$

Case: $\{m2(-)\}$

$$D = \{m2(-), m3(+)\}$$
$$D = \{m2(-), a2(+)\}$$

Presentation Outline

Towards Knowledge Compilation

Assumptions and procedure outline

- input = 27 qualitative cases (3 values for F times 3 values of G times 3 values for the comparison of F vs. G),
- the pattern (F(0), G(0), F=G) represents correct behavior (no conflicts observed),
- 8 other patterns where F=G, are internally inconsistent.
- other 6 patterns (F(-),G(+),F>G), (F(+),G(-),F<G), (F(0),G(-),F<G), (F(0),G(+),F>G), (F(-),G(0),F>G), (F(+),G(0),F<G) are also inconsistent;
- there are 27 - (1+8+6) = 12 potential feasible input combinations of F, G, F-G.

Knowledge Compilation Idea

- select feasible inputs (all vs. most likely),
- calculate qualitative conflicts and diagnoses (off-line),
- in case of multiple-element potential diagnoses design additional tests.

Towards Knowledge Compilation

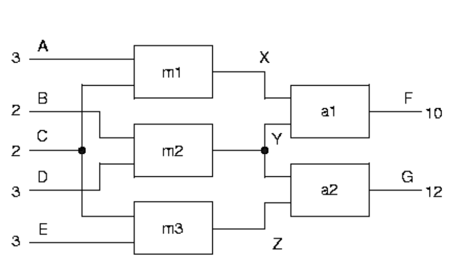
All Possible Failure States

No.	F	G	F∼ G	Comment
1	-	0	$F < G$	F – not-ok; G –ok; F∼G – not-ok
2	+	0	$F > G$	F – not-ok; G –ok; F∼G – not-ok
3	0	-	$F > G$	F –ok; G – not-ok; F∼G – not-ok
4	0	+	$F < G$	F –ok; G – not-ok; F∼G – not-ok
5	-	-	$F < G$	F – not-ok; G – not-ok; F∼G – not-ok
6	-	-	$F > G$	F – not-ok; G – not-ok; F∼G – not-ok
7	-	+	$F < G$	F – not-ok; G – not-ok; F∼G – not-ok
8	+	-	$F > G$	F – not-ok; G – not-ok; F∼G – not-ok
9	+	+	$F > G$	F – not-ok; G – not-ok; F∼G – not-ok
10	+	+	$F < G$	F – not-ok; G – not-ok; F∼G – not-ok
11	-	-	$F = G$	F – not-ok; G – not-ok; F∼G –ok
12	+	+	$F = G$	F – not-ok; G – not-ok; F∼G –ok

Presentation Outline

An example problem: faulty components parametrization

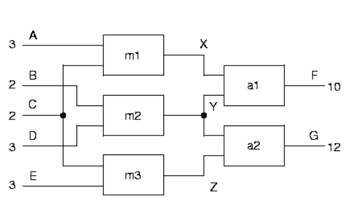
The Multiplier-Adder System




```
[...]
A #= 3,
B #= 2,
C #= 2,
D #= 3,
E #= 3,
F #= 10,
G #=12,
[...]
```

An example problem continued

Model for diagnosis: m1



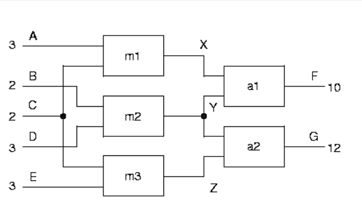

```
% K1/M1 = multiplier error

A * C * K1 #= X *M1,
B * D #= Y,
C * E #= Z,
X + Y #= F,
Y + Z #= G,
K1 #> 0, M1 #> 0.
```

Solution:

```
X=4, Y=6, Z=6, K1=2, M1=3
```

An example problem continued

Model for diagnosis: a1



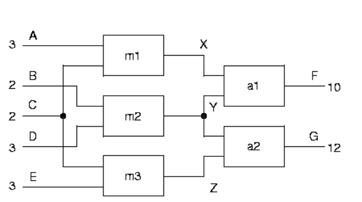

```
% A1 = addition  error
A * C #= X,
B * D #= Y,
C * E #= Z,
X + Y - A1 #= F,
Y + Z #= G.
```

Solution:

```
X=6, Y=6, Z=6, A1=2
```

An example problem continued

Model for diagnosis: $\{a2, m2\}$



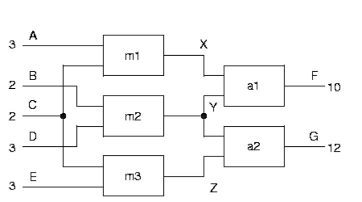

```
% A2 = addition  error
% K2/M2 = multiplier error

A * C #= X,
B * D * K2 #= Y * M2,
C * E  #= Z,
X + Y  #= F,
Y + Z +A2 #= G,
K2 #> 0, M2 #> 0.
```

Solution:

```
X=6, Y=4, Z=6, K2=2, M2=3, A2=2
```

An example problem continued

Model for diagnosis: $\{m2, m3\}$



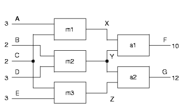

```
% K2/M2 = multiplier error
% K3/M3 = multiplier error

A * C #= X,
B * D * K2 #= Y * M2,
C * E * K3 #= Z * M3,
X + Y #= F,
Y + Z #= G,
K2 #> 0, M2 #> 0,
K3 #> 0, M3 #> 0.
```

Solution:

```
X=6, Y=4, Z=8, K2=2, M2=3,
K3=4, M3=3
```

An example problem: structure discovery

Connection discovery



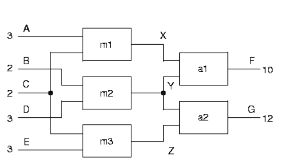

```
Vars = [XA1,YA1,YA2,ZA2],
Vars ins 0..100,
Conns = [X_XA1,X_YA1,X_YA2,X_ZA2,
         Y_XA1,Y_YA1,Y_YA2,Y_ZA2,
         Z_XA1,Z_YA1,Z_YA2,Z_ZA2],
Conns ins 0..1,
```

Reification: Modeling existence of connections

```
X_XA1 #==> X #= XA1, Y_XA1 #==> Y #= XA1, Z_XA1 #==> Z #= XA1,
X_YA1 #==> X #= YA1, Y_YA1 #==> Y #= YA1, Z_YA1 #==> Z #= YA1,
X_YA2 #==> X #= YA2, Y_YA2 #==> Y #= YA2, Z_YA2 #==> Z #= YA2,
X_ZA2 #==> X #= ZA2, Y_ZA2 #==> Y #= ZA2, Z_ZA2 #==> Z #= ZA2,
```

Example: structure discovery continued

Connection discovery continued



Each adder input must be connected:

```
X_XA1 #\/ Y_XA1 #\/ Z_XA1,
X_YA1 #\/ Y_YA1 #\/ Z_YA1,
X_YA2 #\/ Y_YA2 #\/ Z_YA2,
X_ZA2 #\/ Y_ZA2 #\/ Z_ZA2,
```

Modeling uniqueness of connections

```
#\(X_XA1 #/\ Y_XA1), #\(X_XA1 #/\ Z_XA1), #\(Y_XA1 #/\ Z_XA1),
#\(X_YA1 #/\ Y_YA1), #\(X_YA1 #/\ Z_YA1), #\(Y_YA1 #/\ Z_YA1),
#\(X_YA2 #/\ Y_YA2), #\(X_YA2 #/\ Z_YA2), #\(Y_YA2 #/\ Z_YA2),
#\(X_ZA2 #/\ Y_ZA2), #\(X_ZA2 #/\ Z_ZA2), #\(Y_ZA2 #/\ Z_ZA2).
```

Symmetry breaking

```
XA1 #=< YA1, YA2 #=< ZA2,
```

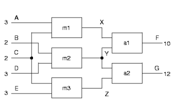
Connection discovery: some results

Table : Example results of internal connections discovery

(A,B,C,D,E)	(F,G)	Symmetry Breaking	No. of models
(3,2,2,3,3)	(12,12)	No	81
(3,2,2,3,3)	(12,12)	Yes	81
(1,3,5,7,11)	(26,76)	No	4
(1,3,5,7,11)	(26,76)	Yes	1
(1,2,3,4,5)	(11,23)	No	4
(1,2,3,4,5)	(11,23)	Yes	1

An example problem: component function identification

Function discovery




```
Funs = [M1M,M2M,M3M,A1A,A2A],
Funs ins 0..1,
M1M #==> X #= A*C, %If M1M=1 then operation is multiplication
M2M #==> Y #= B*D,
M3M #==> Z #= C*E,
#\M1M #==> X #= A+C,  %If M1M=0 then operation is addition
#\M2M #==> Y #= B+D,
#\M3M #==> Z #= C+E,
A1A #==> F #= XA1 + YA1, %If A1A=1 then operation is addition
A2A #==> G #= YA2 + ZA2,
#\A1A #==> F #= XA1*YA1,  %If A1A=0 then the operation is multiplication
#\A2A #==> G #= YA2*ZA2,
```

Function discovery: some results

Table : Example results of functionality and internal connections discovery

(A,B,C,D,E)	(F,G)	No. of models
(3,2,2,3,3)	(12,12)	132
(1,3,5,7,11)	(26,76)	1
(1,2,3,4,5)	(11,23)	1

Presentation Outline

Concluding Remarks

Conclusions:

- Exploring mutual interplay of Rules, Causality and Constraints seems inspiring, especially in modeling deep knowledge,
- Model-Based Reasoning can be based on Rules, Abduction and supported with Constraint Programming,
- Both qualitative and exact numerical models can be investigated,
- Structural knowledge can be discovered with Constraint Programming,
- Rules + Causality + Constraints = Operation.

Further Issues and Work Directions:

- typical ML-repository data is insufficient for causal/structural investigation,
- data extensions: causal, logical, functional and temporal aspects,
- knowledge extensions: components, connections, causality, constraints,...