KEYNOTE @ MBD MEETS RV WORKSHOP 2025

DIAGNOSIS MEETS VERIFICATION: THE PRINCIPLES, APPLICATION AND POTENTIAL OF MBD

Ingo Pill

Graz, 15.9.2025

who‘s talking?

- external lecturer at TU Graz
- until 2024 staff scientist @ Silicon Austria Labs, Graz, Austria
  - deputy head of 2 research units (-2023); trustworthy adaptive computing / collaborative perception & learning
  - management board „SAL Doctoral College“
- 2004 – 2020 (senior) scientist @ Graz Univ. of Tech., Austria
  - *Institute of Software Engineering and AI (SAI, former IST)*
  - *still teaching at TU Graz*
- 2023+ SC chair for „Int. Conference on Principles of Diagnosis and Resilient Systems“

background in AI – diagnosis / model-based diagnosis and reasoning, formal verification (temporal logics, automata, requirements eng.), testing, …

„assistance in the design of intelligent and resilient systems“

what to expect?

- an introduction to model-based diagnosis
  - the why, what and how
- an example of applying MBD to formal models
  - Linear Temporal Logic (LTL)
- potentials and challenges in MBD research
  - design- and run-time

what is diagnosis and what is MBD?

the V&V problem …

project manager: „which guarantees can … car/phone/plant/…"

system operator: „I observed some weird/unexpected behavior, …"

design engineer: „these verification results come unexpected"

automated system: „something went wrong, but what exactly? "

engineer / autonomous sys. : „is there a problem and where is it?"

verification: is there a problem …

diagnosis: … and where is it exactly

diagnosis …

(early) diagnosis systems focused on encoding experience

- we can capture
  - (reversed) cause and effect chains
  - expert knowledge / rules of experience
- some „complex“ computations done before diagnosis time
- hard to maintain – all rules can change with system changes

competing idea

- let‘s use a system model instead
  - employ reasoning from first principles
  - foundations outlined in two seminal papers from ’87

[A theory of diagnosis from first principles, Reiter ‘87]
[Diagnosing multiple Faults, de Kleer and Williams ‘87]

MBD – the concept

employ reason from first principles

break down the complex problem (→ blocks) and reassemble

(1) we describe what we know about the system - SD
(2) we describe what we observed - OBS
(3a) and see whether there's a problem (OBS consistent with SD)

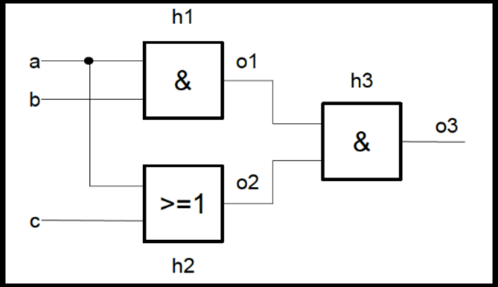
(3b) find maximum sets of SD parts consistent with OBS:
the complement must be faulty = this is a diagnosis

In the literature this concept is called MBD, consistency-based diagnosis, DX approach, …

diagnoses offer explanations

the search for diagnoses resolves conflicts:
what should be (SD) vs. what we saw (OBS)

- we use blocks in SD as basic truths / atoms
  - one health state h_i per block
    - If h_i is true, then the block is correct
    - SD: set of $h_i \rightarrow \text{NominalBehavior}(c_i)$
      (+ some other stuff)



SD
vs.
OBS

natural blocks: physical components, functions, statements, changes in a model, …

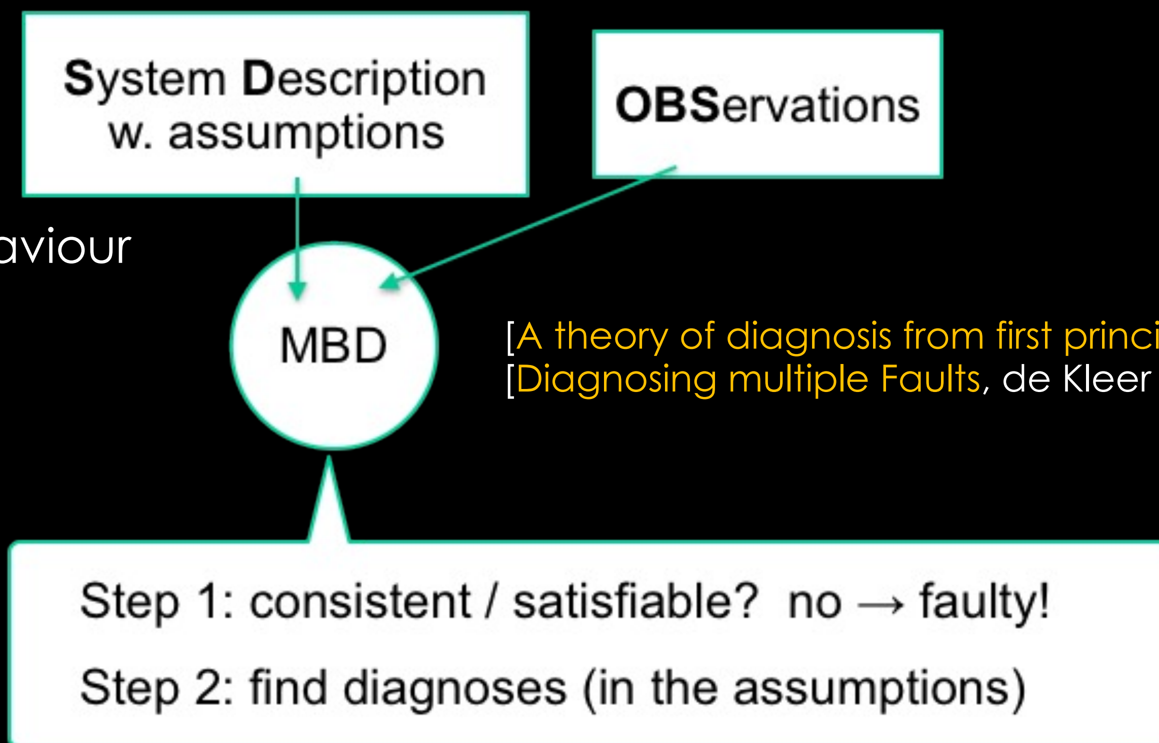
test case	inputs			outputs: nominal			F_1			F_2			F_3		
	a	b	c	o_1	o_2	o_3	o_1	o_2	o_3	o_1	o_2	o_3	o_1	o_2	o_3
1	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
2	⊥	⊤	⊤	⊥	⊤	⊥	⊤	⊤	⊤	⊥	⊤	⊤	⊥	⊤	⊤
3	⊤	⊥	⊤	⊥	⊤	⊥	⊤	⊤	⊤	⊥	⊤	⊤	⊥	⊤	⊤
4	⊤	⊤	⊥	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊥
5	⊥	⊥	⊤	⊥	⊤	⊥	⊥	⊤	⊥	⊥	⊤	⊤	⊥	⊤	⊤
6	⊥	⊤	⊥	⊥	⊥	⊥	⊤	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
7	⊤	⊥	⊥	⊥	⊤	⊥	⊤	⊤	⊤	⊥	⊤	⊤	⊥	⊤	⊤
8	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊤	⊥

Def: a diagnosis is a subset-minimal set Δ of h_i s.t. SD U OBS U { h_i | h_i not in Δ'} is satisfiable

the search space is $2^{|H|}$

MBD – the traditional scenario

our
knowledge
about the
assumed behaviour



[A theory of diagnosis from first principles, Reiter ‘87]
[Diagnosing multiple Faults, de Kleer and Williams ‘87]

“diagnoses” explain the observed behaviour

computation: two basic concepts

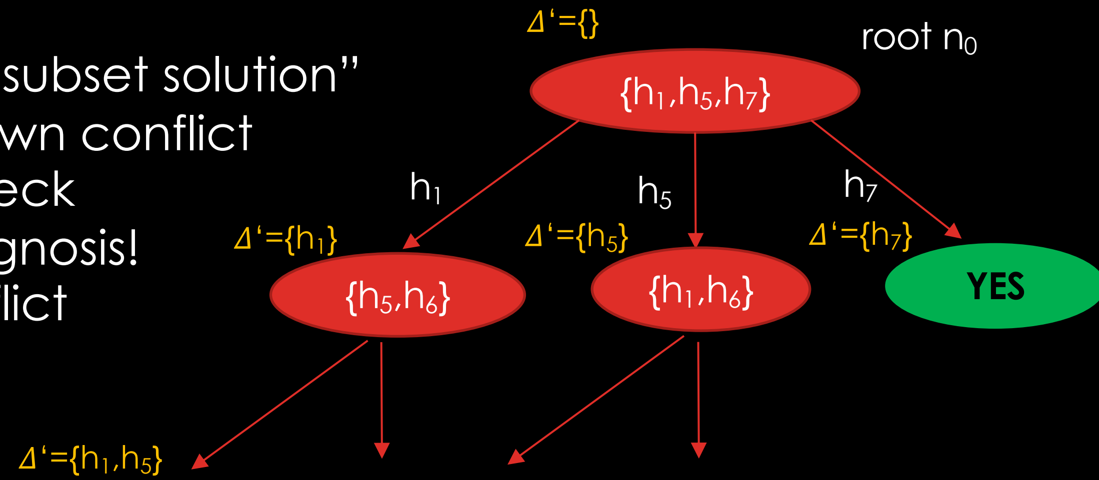
- directly in a solver (basically brute force)
  - iteratively search for a (new) solution
    - limit and increase fault cardinality
    - add blocking clauses for every Δ‘ found
      - at least one h in Δ‘ must be true (not faulty) for other Δs
- conflict-driven
  - conflicts between SD and OBS need to be resolved

computing diagnoses from conflicts

diagnostic search resolves conflicts in H:

- tree/DAG-like exploration
  - create candidates Δ'
  - (1) check if there's a "subset solution"
  - (2) see if there's a known conflict
  - (3) do consistency-check
    - SAT – found a diagnosis!
    - UNSAT – new conflict

node-labels are conflicts!



observation: a Diagnosis Δ is a subset-minimal hitting set of conflicts in H

[A correction to the algorithm in Reiter's theory of diagnosis. Greiner, Smith, Wilkerson, 1989]

some algs. and a comparison

- conflict-driven

  [Diagnosing multiple Faults, de Kleer and Williams '87 (GDE)]
  [RC-Tree: A Variant Avoiding all the Redundancy in Reiter's Minimal Hitting Set Algorithm, I. Pill and T. Quaritsch, 2015]
  [*DynamicHS: Streamlining Reiter's Hitting-Set Tree for Sequential Diagnosis*. *P. Rodler*, *2023]*

- direct

  [ConDiag - Computing minimal diagnoses using a constraint solver, I Nica, F. Wotawa, 2012]
  [Compiling model-based diagnosis to Boolean satisfaction, A. Metodi, R. Stern, M. Kalech, and M. Codish, 2012]

- comparison

  [The Route to Success - A Performance Comparison of Diagnosis Algorithms, I. Nica, I. Pill, T. Quaritsch, F. Wotawa, 2013]
  [Assessing Diagnosis Algorithms: Of Sampling, Baselines, Metrics and Oracles, I. Pill, J. de Kleer, DX 2025 (to appear), best paper award candidate]

and MBD ?

- no restriction in terms of application
- we „only“ need a model and a computation method to do the consistency checks
- can be, *e.g.*, digital, logical, analog, mechanical, cyber-physical, biological, ecological, ethical, economical, and social systems and processes.

[Challenges for Model-based Diagnosis, I. Pill, J. de Kleer, 2024]

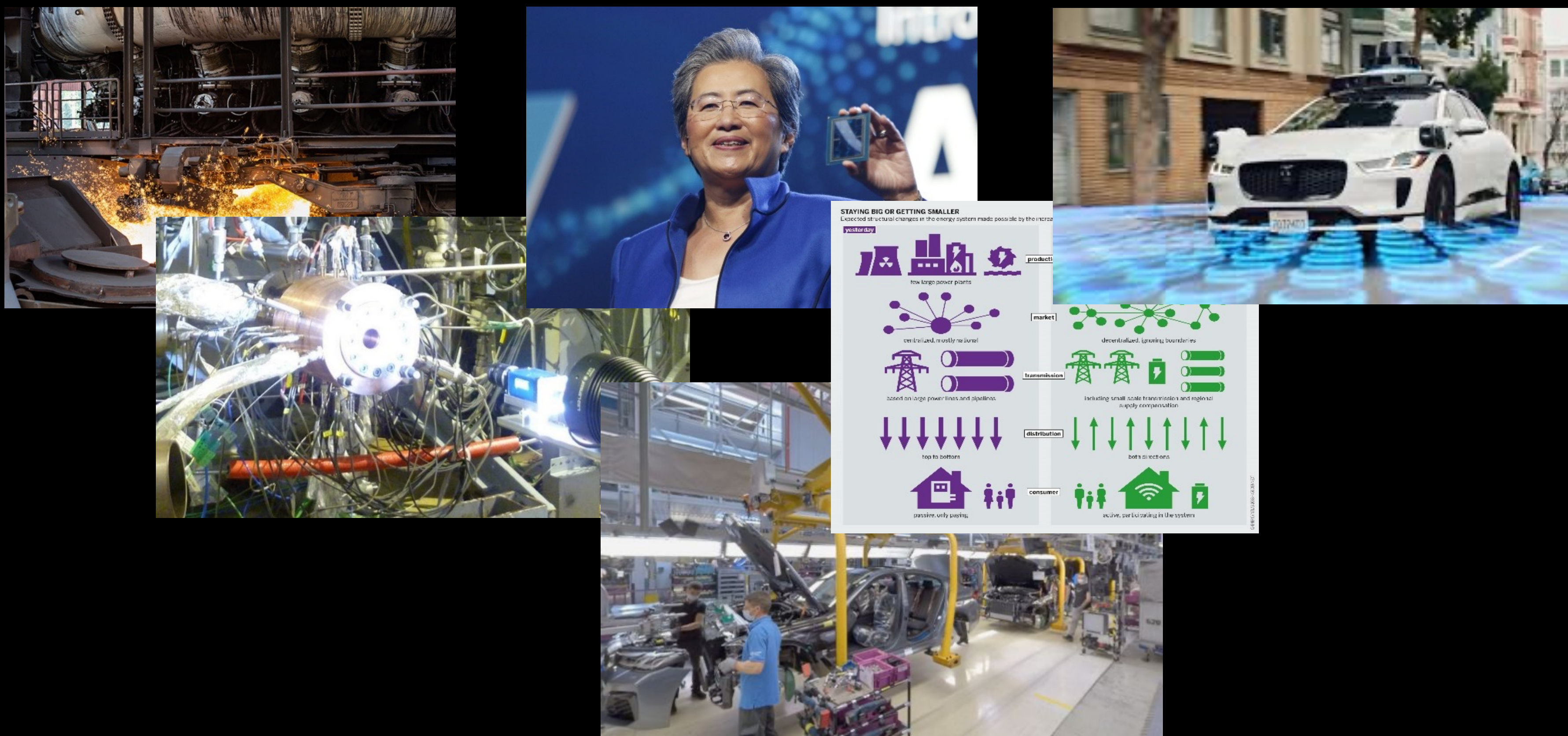
diagnosis is a common task …

extracting a good coffee

- is a challenge
- requires knowledge
    - „expertise“
- there‘s no detailed model
    - general physics known
    - machine model?
    - environment?
    - coffee, water?
- data driven experimentation
    - external data points
    - unknown data quality

source: https://www.delonghi.com/de-at/ec685-m-dedica-espressomaschine/p/EC685.M

much more complex problems

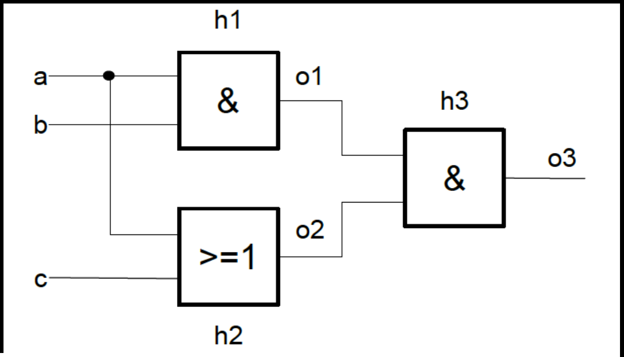


picture sources: VoestAlpine, DLR, AMD, Magna International, Wikipedia, Waymo

from simple comb. circuits

simple circuit – introduce some fault(s)

- F_1 : gate g_1 like OR instead of AND
- F_2 : g_3 like OR
- F_3 : g_3 like XOR



test case	inputs			nominal			F_1			F_2			F_3		
	a	b	c	o_1	o_2	o_3	o_1	o_2	o_3	o_1	o_2	o_3	o_1	o_2	o_3
1	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
2	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$
3	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$
4	$\top$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\bot$
5	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$
6	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
7	$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$
8	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\bot$

MBD can explain the failing test cases via comparing

- OBS = observed I/O
- SD = clauses for gates

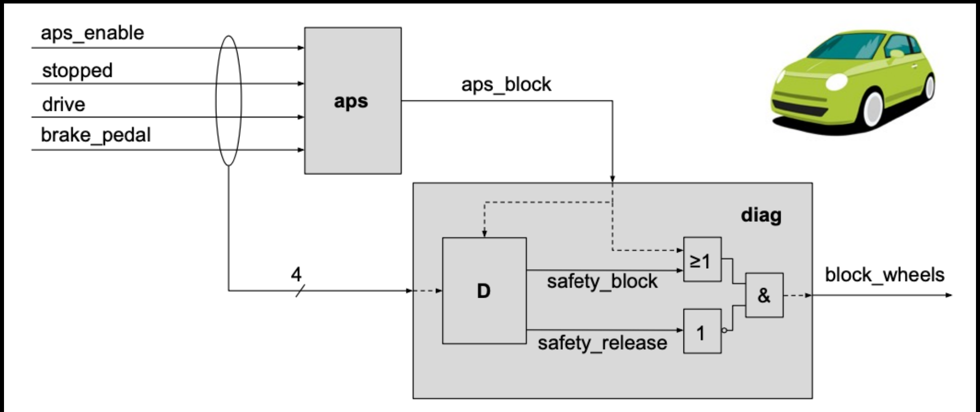
OR: o1 = a || b

$(\neg h_1 \vee \neg o_1 \vee a \vee b)$,

$(\neg h_1 \vee \neg a \vee o_1)$,$(\neg h_1 \vee \neg b_i \vee o_1)$

$h_1 \rightarrow NominalBehavior(g_1)$

to temporal logics & beyond



automated parking brake

R_1: always (block_wheels → stopped)

R_2: always (block_wheels → (block_wheels W (aps_enable → (drive v brake_pedal))))

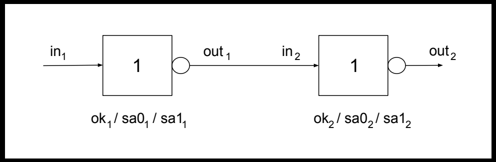
[Behavioral Diagnosis of LTL Specifications at Operator Level, I. Pill, Th. Quaritsch, 2013]
[Extending Automated FLTL Test Oracles With Diagnostic Support, I. Pill, F.Wotawa, 2019]

[Hybrid Systems Diagnosis, S. McIlraith, G. Biswas, D. Clancy, V. Gupta, HSCC 2000]

how about fault models?

weak fault model (WFM) – no assumption on faults

strong fault model (SFM)



alternative behavior

mode set {*corr*, *mode*$_1$, … , *mode*$_{n-1}$}

(e.g. twist operands for subformula δ)

SD: mode → behavior$_{mode}$

h_i → ld(n) bits → add negated minterm to clauses

add negated „unused“ minterms to SD

[Diagnosis with Behavioral Modes, J. de Kleer and B. Williams, 1989]

what are the effects?

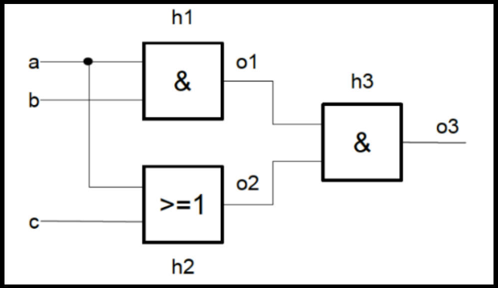
strong fault model diagnosis (SFM)

turns diagnosis into a configuration problem

Δ = assignment for H that makes SD and OBS consistent

supersets of a diagnosis are not a diagnosis by default

diagnoses sometimes offer repairs (example will come)



search space grows from $2^{|H|}$ to $O(\max(n)^{|H|})$

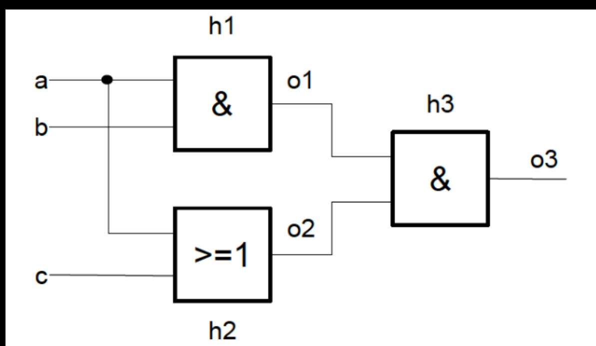
what if I have multiple scenarios?

- long-term observations
  - temporal behavior
  - multiple scenarios / plans
- results from a test suite
- observations from multiple system instances

this is different …

explaining ***a*** scenario → characterizing a system

- e.g. use combinatorial testing for circuits



test case	inputs			outputs: nominal			F_1			F_2			F_3		
	a	b	c	o_1	o_2	o_3	o_1	o_2	o_3	o_1	o_2	o_3	o_1	o_2	o_3
1	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
2	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$
3	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$
4	$\top$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\bot$
5	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$
6	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
7	$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$
8	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\bot$

[Exploiting Observations from Combinatorial Testing for Diagnostic Reasoning,
I. Pill and F. Wotawa, 2019]

multiple scenarios - how to?

- **a multi-scenario diagnosis** for a set T of failed test cases (failed scenarios) is a subset-minimal set Δ s.t. SD U OBS_j U { h_i | h_i not in Δ'} is satisfiable for each OBS_j
- **all scenarios** OBS_j are investigated in a global search space
  - **global conflict buffer** (try to use known conflicts first)

[Exploiting Observations from Combinatorial Testing for Diagnostic Reasoning,

I. Pill and F. Wotawa, 2019]

[Computing Multi-Scenario Diagnoses, I. Pill and F. Wotawa, 2020 (MSRC-Tree)]
[Model-based diagnosis with multiple observations, A. Ignatiev et al., 2019]

multiple scenarios - how to?

- RC-Tree → MSRC-Tree
  - when checking diagnosis candidates, loop over scenarios
  - multiple strategies

- compute only a set of conflicts to describe *global* Δ s.t. |Δ|≤ bound l

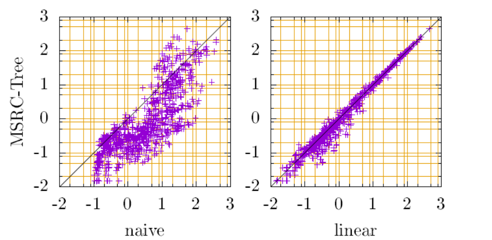


Figure 2: Scatter plots comparing MSRC-Tree's average run-time (in 10^y seconds) against those of the naive algorithm and the linear MSRC-Tree variant (C17/C432/C499).

[Computing Multi-Scenario Diagnoses (MSRC-Tree), I. Pill and F. Wotawa, 2020]
[Model-based diagnosis with multiple observations, A. Ignatiev et al., 2019]

Part II – an example: LTL

(2) Linear Temporal Logic

[Temporal Logic of Programs, Pnueli, 1977]

- **we can describe programs and seq. circuits**
  - **specifications AND implementations**
- **clocked, discrete time steps**
- **initially for infinite computations**
  - **finite semantics as well (later)**
  - contained e.g. in PSL (IEEE Std. 1850)
  - easy extension for further operators / purposes

MBD of LTL Descriptions

focus on operator occurrences in a formula φ

did we use the right operator for subformula δ?

system description SD with „assumptions“ on ops

$h_\delta \rightarrow$ NominalBehavior(δ)

observations **OBS = trace values**

SD U OBS U { h_δ | δ in φ} inconsistent

→ faulty specification / LTL description

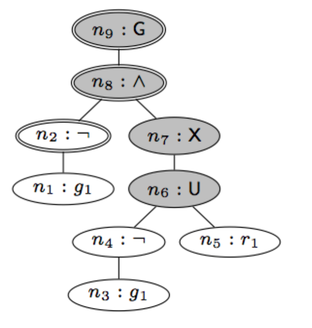
create a SAT encoding for MBD

SAT model for φ,τ

- basic ingredients:
  - encode operator semantics directly
  - add variables for all subformulae
  - temporal instantiation
- similar to encodings for model-checking, e.g.
  [Symbolic Model Checking without BDDs, Biere, Cimatti et al., 1999]
- structure-preserving CNF
- polynomial (linear growth with length of τ or spec)
- use with any diagnosis algorithm
  - HS-DAG / RC-Tree / direct ones

CNF encoding for LTL

„collect“ clauses traversing the parse tree and τ for all t_i:



$\varphi = a \vee b$: unfolding rationale $\varphi_i \leftrightarrow a_i \vee b_i$

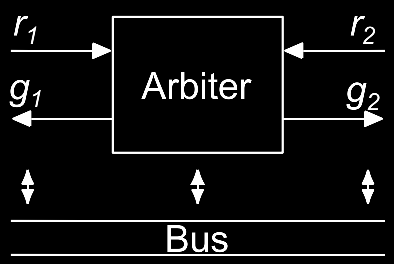
3 clauses: $(\neg\varphi_i \vee a_i \vee b_i)$,$(\neg a_i \vee \varphi_i)$,),$(\neg b_i \vee \varphi_i)$

$\varphi = \delta \, U \, \psi$ („delta is true until psi becomes true“)

rationales:	clauses:
(f) $\varphi_i \rightarrow (\psi_i \vee (\delta_i \wedge \varphi_{i+1})$	(f_1) $\neg\varphi_i \vee \psi_i \vee \delta_i$ (f_2) $\neg\varphi_i \vee \psi_i \vee \varphi_{i+1}$
(g) $\psi_i \rightarrow \varphi_i$	(g_1) $\neg\psi_i \vee \varphi_i$
(h) $\delta_i \wedge \varphi_{i+1} \rightarrow \varphi_i$	(h_1) $\neg\delta_i \vee \neg\varphi_{i+1} \vee \varphi_i$
(i) $\varphi_k \rightarrow (\psi_l \vee \ldots \vee \psi_k)$	(i_1) $\neg\varphi_k \vee \psi_l \vee \ldots \vee \psi_k$

for MBD: just add $\neg h_\delta$ to each clause of operator δ $(h_\delta \rightarrow SD_\delta)$

some example: arbiter

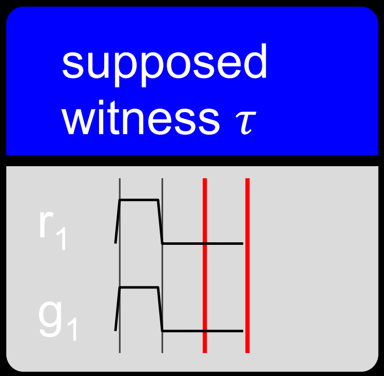


R1: „*any request granted eventually*“

R2: „*no simultaneous grants*“

R3: „*no initial spurious grants*“

R4: „*no further grants until new request*“



R4 in LTL: $G(g_i \rightarrow X(\neg g_i \; U \; r_i))$

globally($g_i \rightarrow$ *next*((*not* g_i) *until* r_i))

[Formal Analysis of Hardw. Requirements, I. Pill, A. Cimatti et al., 2006]

arbiter example: (WFM) diagnoses

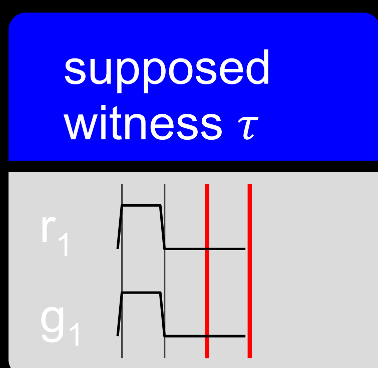
R1: „*any request granted eventually*"

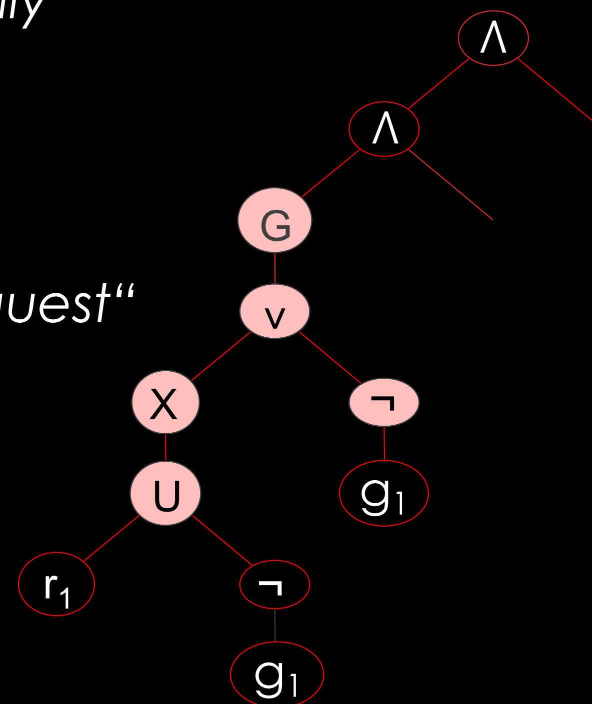
R2: „*no simultaneous grants*"

R3: „*no initial spurious grants*"

R4: „*no further grants until new request*"

$$G(g_i \rightarrow X(\neg g_i \; U \; r_i))$$



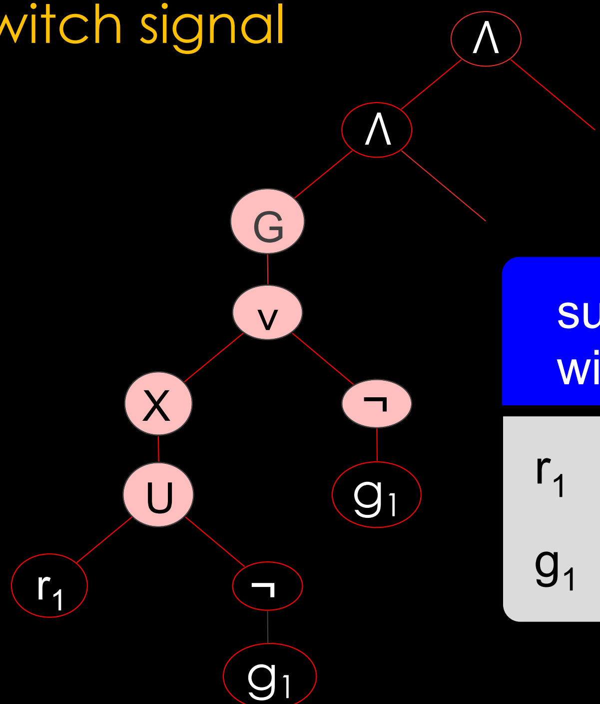


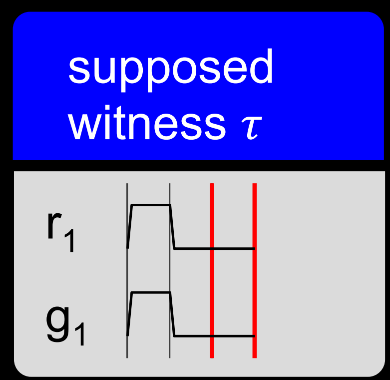
you said SFM can offer repairs …

- other Boolean operator
- other temporal operator
- twist operands
- switch signal

R4: $G(g_1 \rightarrow X\,(\neg g_1\ U\ r_1))$

1: $G(g_1 \rightarrow X\,(\neg g_1\ W\ r_1))$
2: $X(g_1 \rightarrow X\,(\neg g_1\ U\ r_1))$
3: $G(g_1 \rightarrow X\,(r_1\ R\ \neg g_1))$
4: $G(g_1 \rightarrow X\,(\neg g_1\ U\ r_2))$
5: $G(g_1 \rightarrow F\,(\neg g_1\ U\ r_1))$
6: $F(g_1 \rightarrow X\,(\neg g_1\ U\ r_1))$
7: $G(g_1 \rightarrow X\,(r_1\ U\ \neg g_1))$
8: $G(g_1 \rightarrow X\,(\neg g_1\ U\ g_2))$
9: $G(g_1 \rightarrow X\,(r_1 W\ \neg g_1))$





you mentioned finite semantics

- infinite examples come from model-checkers, documents, tools like RAT ...
- testing and RV give you finite examples though
  - finite LTL semantics are slightly different (e.g. X)
  - encoding for diagnosis and oracle
    - oracle needs Boolean propagation only

[**Extending Automated FLTL Test Oracles With Diagnostic Support**, I. Pill, F.Wotawa, IDEAR@ISSRE'19]

[**Automated generation of (F)LTL oracles for testing and debugging**, I. Pill, Franz Wotawa, J. of Systems and Software, Volume 139]

Part III – challenges and potentials

MBD is

- good at explaining – diagnoses offer justified explanations
- sound – a computed solution is correct
- complete – we can find the entire set of solutions
- intuitive, flexible (algorithms, domain)
- sometimes offers repairs (SFM of spec or design)

- depends on a „white-box“ model + engine

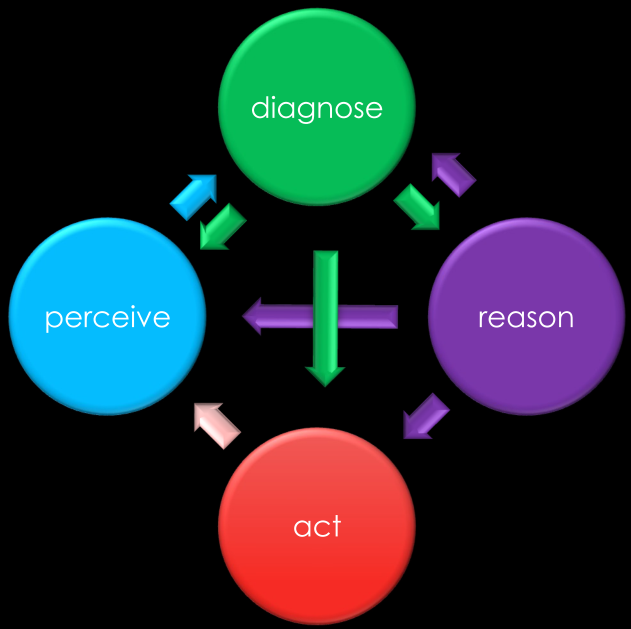
sometimes …

- … reasoning with MBD is not fast enough
- think about a resilient agent
  - but do we need (all) explanations then?
  - reliability of actions might suffice as first info
    - focus on reliable actions in the planning
    - use SFL to derive reliability of individual actions
  - use that in the (re-)planning

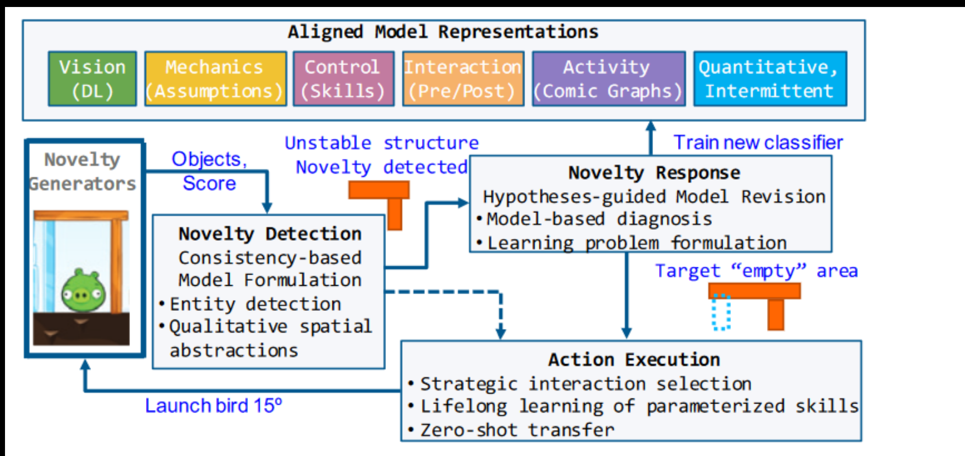
Resilience is the intrinsic ability of a system to sustain its required operations when impacted by expected and unexpected contingencies that were potentially not considered at design time

that is, in an ideal world …

- a resilient system reasons about options and decide
- we have a lot of resources to reason about options
  - derive the most promising/efficient action sequences
    - perceive, diagnose, reason + act
  - no Markov property restrictions (history is relevant)



… and then we would do …

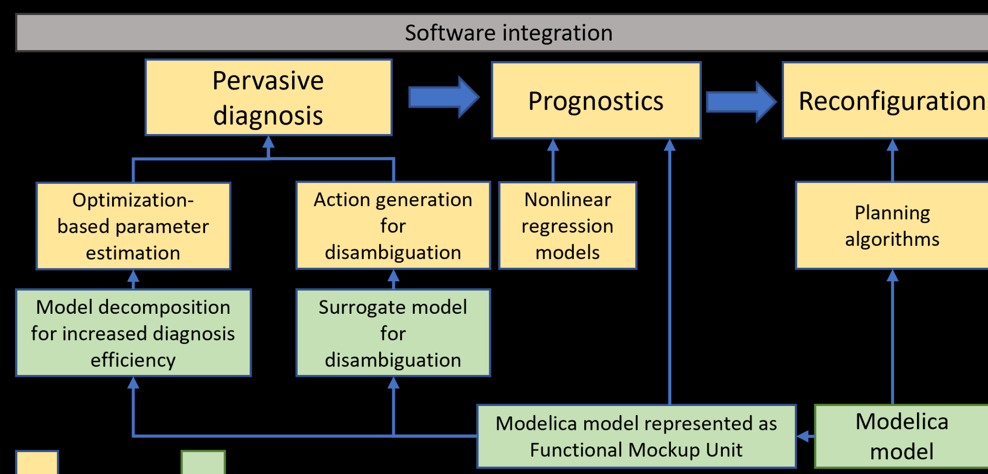


Novelty detection

[Model-based Novelty Detection for Open-World AI M. Klenk, W. Piotrowski, R. Stern, S. Mohan, and J. de Kleer, DX Workshop 2020]

Fusing Diagnosis and Prognosis

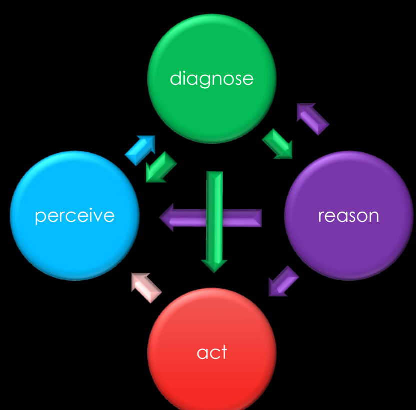
[System Resilience through Health Monitoring and Reconfiguration, I. Matei, W. Piotrowski, A Perez, J. de Kleer, J. Tierno, W. Mungowan,V. Turnewitsch, ACM Trans. on Cyber-Phys. Sys., 2024



in the real world …

limited resources, but still need to make informed decisions

- approximate a real *decision* via a “reaction”-policy
- do reasoning, but have to improve runtime performance/resources
    - scale down single steps / concept



(1) PLAN
(2) EXECUTE
    monitor + stop/repair/“reset“
(3) UPDATE
    belief: what works or not

[**Drawing on SFL for Making Intelligent Decisions in RBL,**
M. Zimmermann, I. Pill, F. Wotawa, DX Workshop 2020]

computation times are not the only challenge for MBD

[Challenges for Model-based Diagnosis, I. Pill, J. de Kleer, DX conference 2024]

failure of function vs. components

- a human considers the observed problem
  - exploits common sense reasoning and expertise
    - at various abstraction levels
    - hierarchical view / „divide et impera“
- how to capture this in MBD models / algorithms?
  - dependency graphs / dependent failure descriptions
  - learn and maintain abstract representations

models are approximations

- MBD is often sound and complete w.r.t. the model
  - not everything is modeled (e.g., radiation)
    - capacitors might get heated by resistors
  - hidden assumptions/simplifications might change
- currently we have no means to
  - assess an MBD model and its consequences
  - express confidence in the model and its consequences

component degradation

- WFM theory considers a component healthy/unhealthy
- fault models capture problematic behavior only
- we can't capture degradation
    - how well does a system still work?

- the PHM community has models, but incompatible with MBD
- for monitoring and logics like STL, a notion was introduced

Topic: Resilient Autonomy Supported by Continuous Tracking of Component Degradation via Model-Based Diagnosis

considering synergies and levels

- many systems are massively replicated
  - cars, screws, copiers, mobile phones, …
  - inefficient to rediscover faults (design, …)
    - problem in a plane/drone – instant report in the fleet
  - use data from other copies for discrimination
    - is the problem local in time/space/system/…?
      - collaborating robots – more knowledge
    - exploit digital twins
- different levels of time and scope
  - immediate/intermediate/LT

please take home that

there‘s a huge potential for research in combining RV and MBD for driving the resilient systems of tomorrow …

[Challenges for Model-based Diagnosis, I. Pill, J. de Kleer, DX conference 2024] get in touch

ingo.pill@gmail.com

If this was interesting to you, consider joining us at the next
International Conference on Principles of Diagnosis and Resilient Systems