Wissensverarbeitung

- Model-Based Diagnosis -

Alexander Felfernig and Gerald Steinbauer

Institute for Software Technology

Inffeldgasse 16b/2

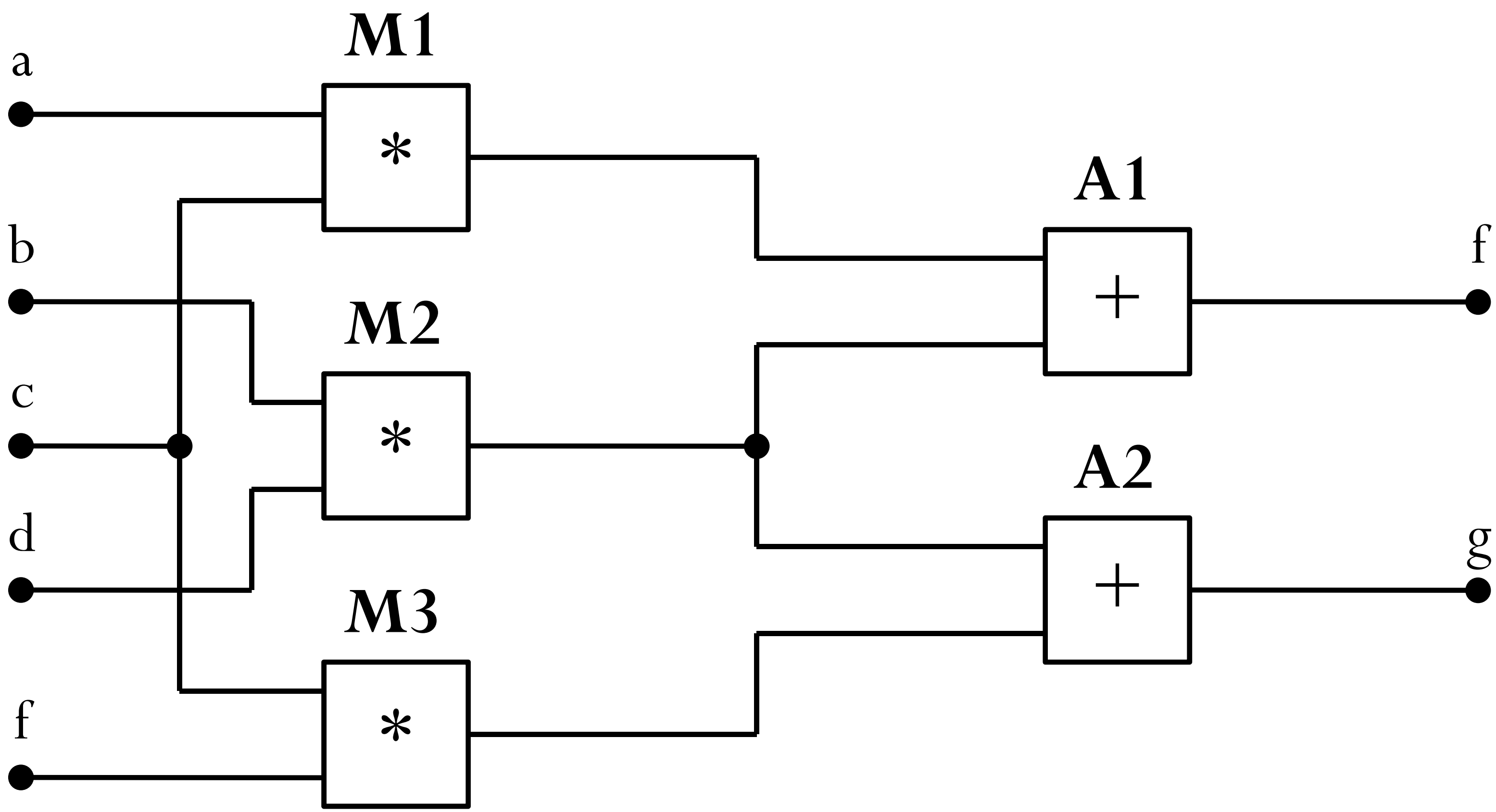
A-8010 Graz

Austria

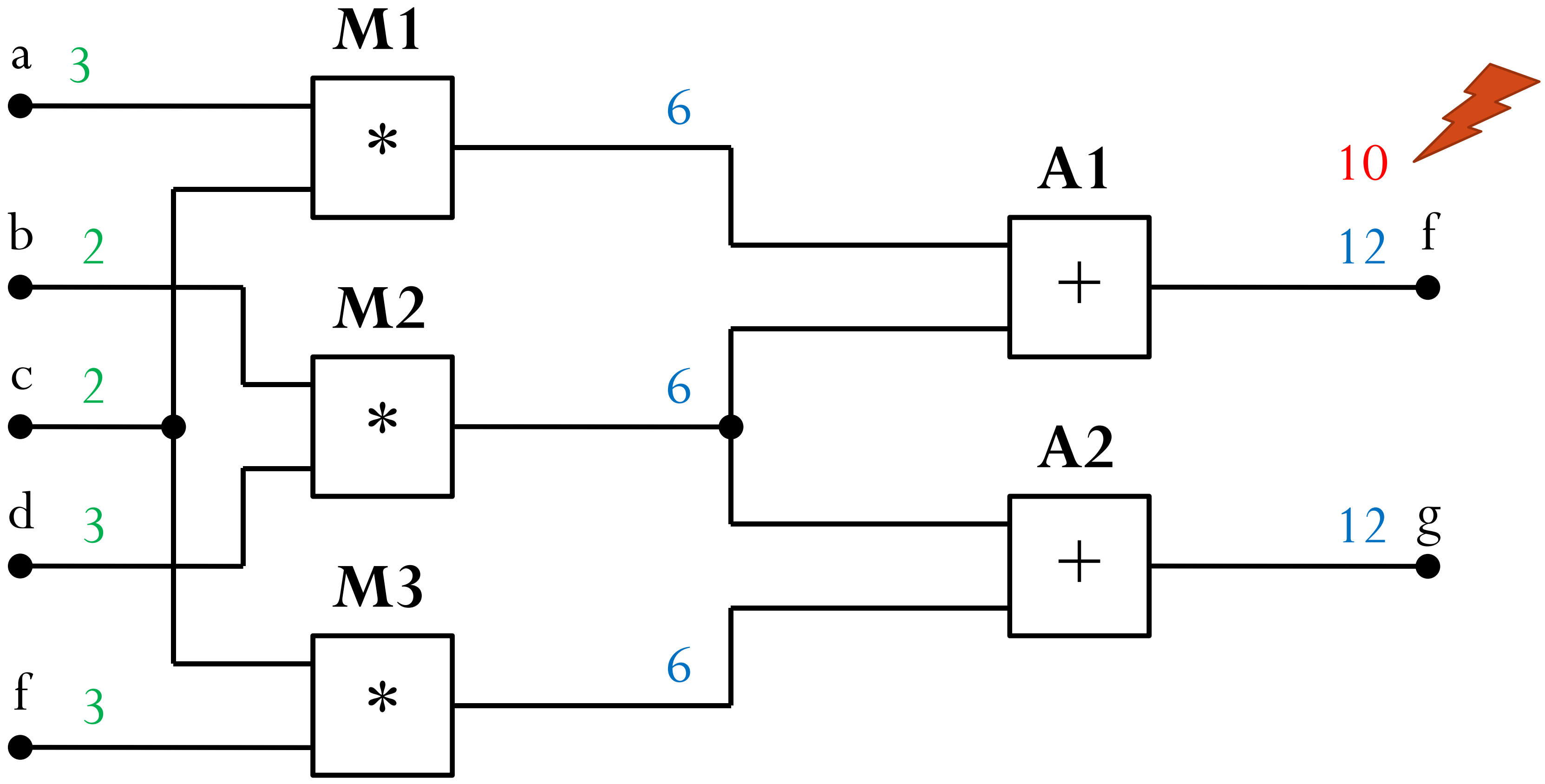
References

- Skriptum (TU Wien, Institut für Informationssysteme, Thomas Eiter et al.) ÖH-Copyshop, Studienzentrum
- Stuart Russell und Peter Norvig. Artificial Intelligence - A Modern Approach. Prentice Hall. 2003.
- Vorlesungsfolien TU Graz (partially based on the slides of TU Wien)

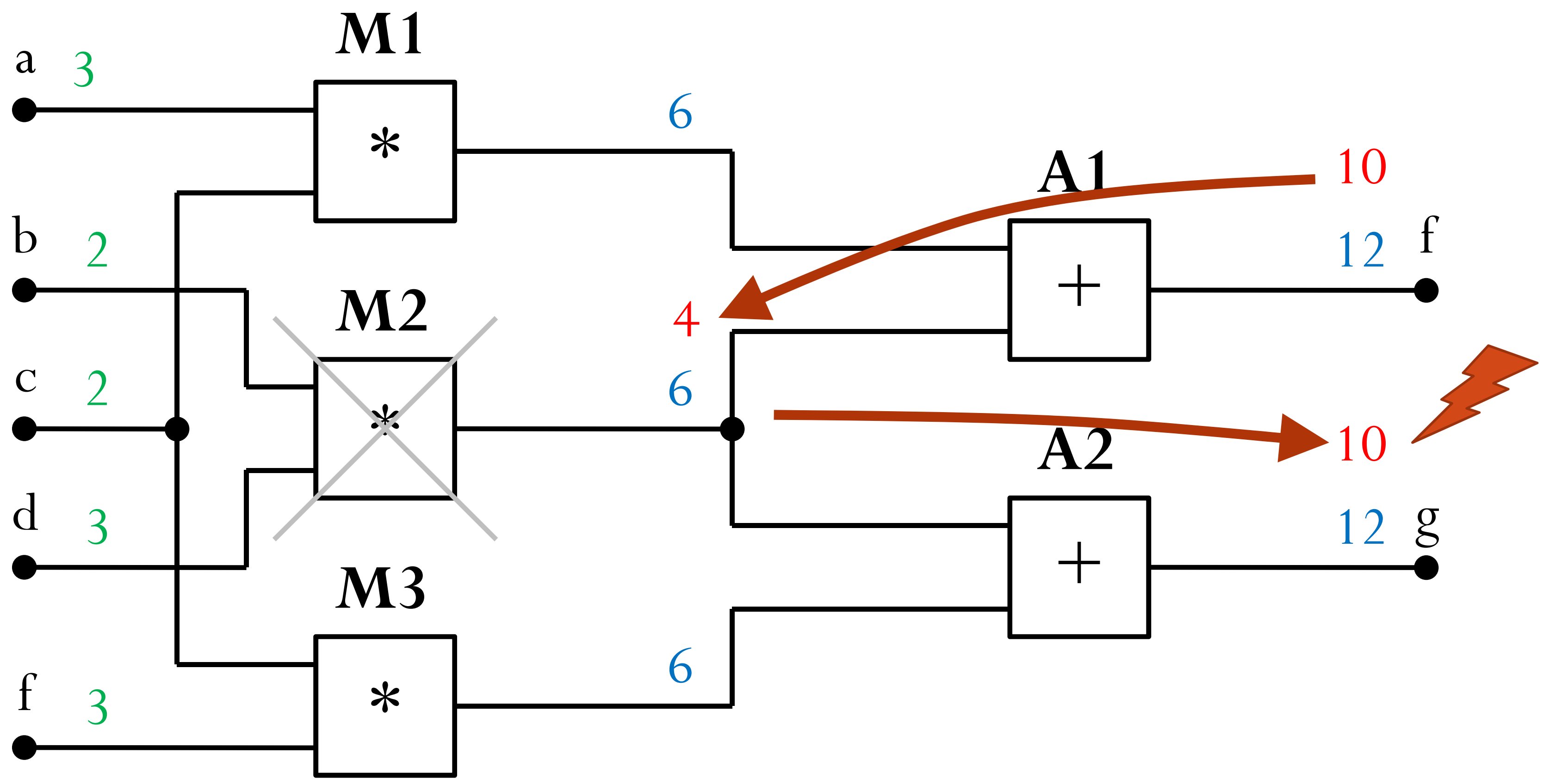
Motivating Example MBD



Motivating Example MBD



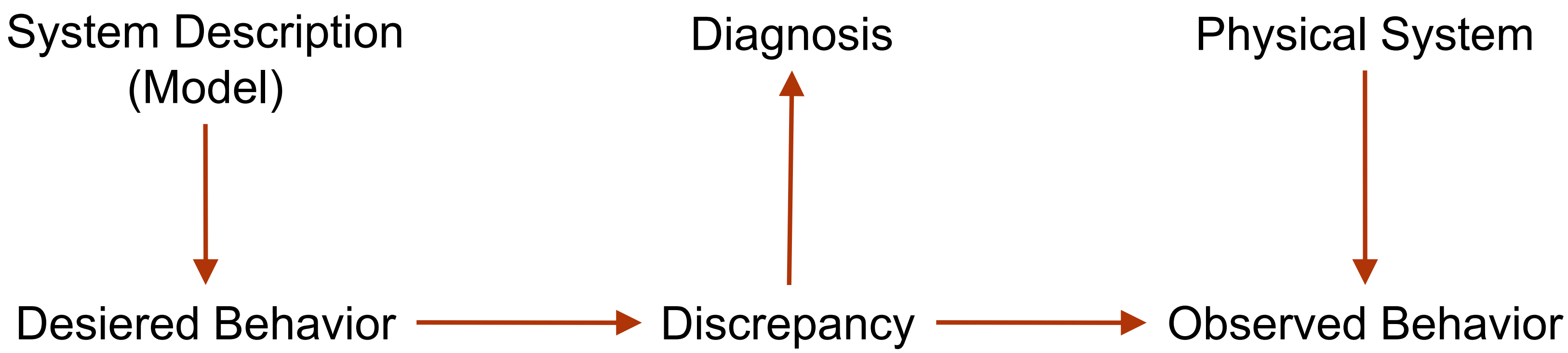
Motivating Example MBD



-5-

Omni-directional Robot

Priciples


System Description
(Model)
Diagnosis
Physical System
Desiered Behavior
Discrepancy
Observed Behavior

Principles

- Requirements
  - Model (Component-Connection-Behavior Model)
  - Powerful Computer
- Benefits
  - General Methodology
  - Easy to maintain
  - Easy adaptable to other problems
  - Cost reduction

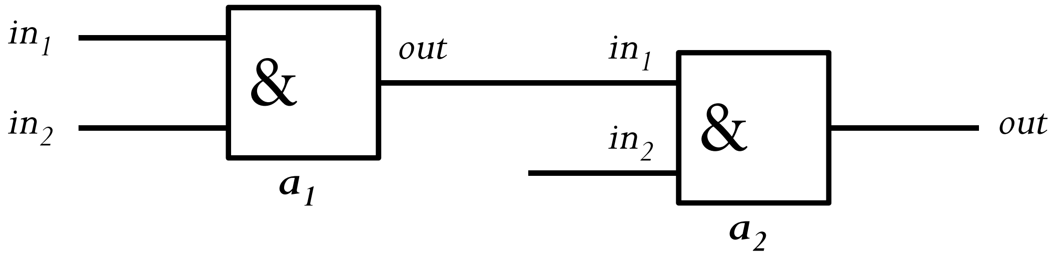
Definitions

1. Diagnosis System: A diagnosis system (*SD*,*COMP*) consists of a system description *SD*, i.e., a set of FOL sentences describing the components behavior and the system structure, and a set of diagnosis components *COMP*.

Example: AND gates

$$and(C) \rightarrow (\neg ab(C) \rightarrow out(C) = in_1(C) \wedge in_2(C))$$

$$and(a_1) \wedge and(a_2) \wedge out(a_1) = in_1(a_2)$$

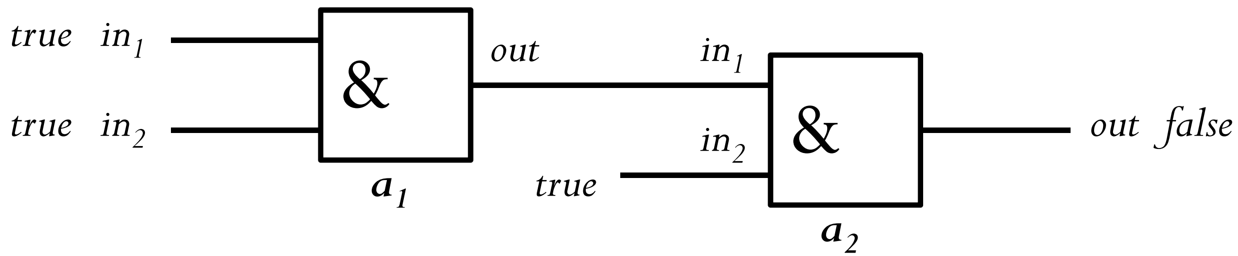


Definitions

2. Diagnosis: Let $(SD, COMP)$ be a diagnosis system and OBS a set of observations. A set $\Delta \subseteq COMP$ is a diagnosis iff
$$SD \cup OBS \cup \{\neg ab(C) \mid C \in COMP \setminus \Delta\} \cup \{ab(C) \mid C \in \Delta\}$$
is consistent.

Example: AND gates

$$OBS=\{in_1(a_1)=true \wedge in_2(a_1)=true \wedge in_2(a_2)=true \wedge out(a_2)=false\}$$



Proposition

1. A diagnosis exists for $(SD, COMP, OBS)$ iff $SD \cup OBS$ is consistent.

Proof: If $SD \cup OBS$ is inconsistent, then obviously it is impossible for all $\Delta \subseteq COMP$ to fulfill the diagnosis condition. So there exists no diagnosis. On the other hand if $SD \cup OBS$ is consistent at least $COMP$ is a diagnosis. ■

Proposition

2. $\{\}$ is a diagnosis for $(SD, COMP, OBS)$ iff $SD \cup OBS \cup \{\neg ab(C) \mid C \in COMP\}$ is consistent.
3. Every superset of a diagnosis is a diagnosis.
4. If Δ is a diagnosis for $(SD, COMP, OBS)$, then for each $C_i \in \Delta$, $SD \cup OBS \cup \{\neg ab(C) \mid C \in COMP \setminus \Delta\} \models ab(C_i)$

Proposition

Proof: If $\Delta = \{\}$ the result is vacuously. Suppose then that $\Delta = \{C_1, \ldots, C_k\}$ and that the proposition is false. Then there exists a C_i such that $SD \cup OBS \cup \{\neg ab(C) \mid C \in COMP \setminus \Delta\} \models ab(C_i)$. From the definition of $\models$ follows that there must be a logical Model M_L with the property $\models^{M_L} SD \cup OBS \cup \{\neg ab(C) \mid C \in COMP \setminus \Delta\} \rightarrow \not\models^{M_L} ab(C_i)$. Now we can conclude $\models^{M_L} \neg ab(C_i)$ which is in contradiction with our initial assumption $C_i \in \Delta$. ■

Proposition

Graz University of Technology

5. Δ is a diagnosis for $(SD, COMP, OBS)$ iff $SD \cup OBS \cup \{\neg ab(C) \mid C \in COMP \setminus \Delta\}$ is consistent.

Definition

3. A conflict set for $(SD, COMP, OBS)$ is a set $CO \subseteq COMP$ such that $SD \cup OBS \cup \{\neg ab(C) \mid C \in CO\}$ is inconsistent. A conflict set is minimal if no proper subset is a conflict set.

Proposition

6. $\Delta \in COMP$ is a diagnosis for $(SD, COMP, OBS)$ iff Δ is a minimal set such that COMP/ Δ is not a conflict set.

Definition

Graz University of Technology

4. Suppose C is a collection of sets. A hitting set for C is a set $H \subseteq \bigcup_{S \in C} S$ such that $H \cap S \neq \emptyset$ for each $S \in C$. A hitting set is minimal if no proper subset is a hitting set.

Theorem

7. $\Delta \subseteq COMP$ is a (minimal) diagnosis for $(SD, COMP, OBS)$ iff Δ is a (minimal) hitting set for the collection of conflicts set.

Proof: (1) By proposition 6 $COMP \backslash \Delta$ is not a conflict set for $(SD, COMP, OBS)$. Hence, every conflict set contains an element of Δ, so that Δ is a hitting set for the collection of conflict sets. (2) We now show that $COMP \backslash \Delta$ is no conflict. If it is a conflict set Δ would not hit it, contradicting the fact that Δ is a hitting set.

Computing Hitting Sets

F … collection of conflicts

1. Let D represent a growing dag. Generate a node which will be the root of the dag.
2. Process the nodes in D in breath-first order. To process a node n:
   a. Define $H(n)$ to be the set of edge labels on the path in D from root to node n.
   b. If for all $x \in F$, $x \cap H(n) \neq \emptyset$ then label n by ✓. Otherwise, label n by Σ where Σ is the first member of F which $x \cap H(n) = \emptyset$.
   c. If n is labeled by a set $\Sigma \in F$, for each $\sigma \in \Sigma$, generate a new downward arc labeled with σ. This arc leads to a new node m with $H(m) = H(n) \cup \{\sigma\}$. The new node m will be processed after all nodes in the same generation as n have been processed.
3. Return the resulting dag D.

Pruning Rules

Graz University of Technology

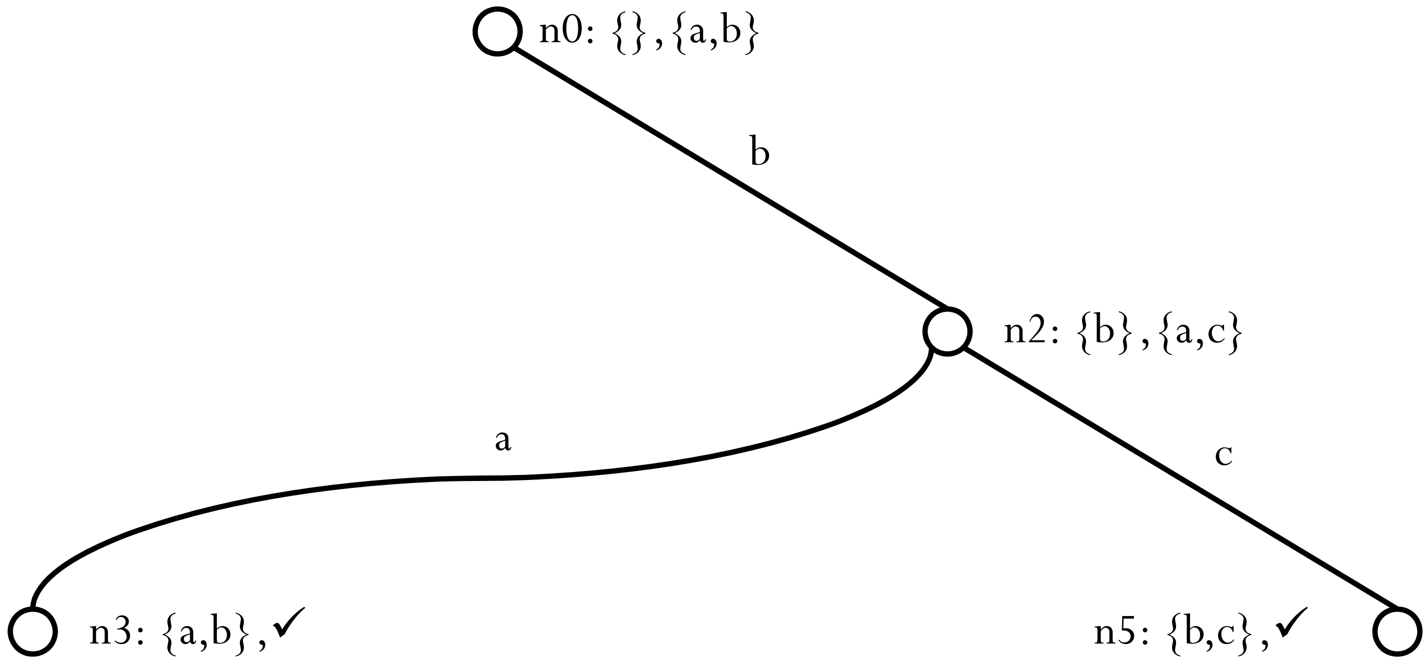
- Reusing nodes: This algorithm will not generate a new *m* as a descendant of node *n*. There are two cases to consider:
  1. If there is a node *n'* in *D* such that $H(n') = H(n) \cup \{\sigma\}$, then let the σ-arc under n point to this exiting node *n'*. Hence, *n'* will have more than one parent.
  2. Otherwise, generate a new node m at the end of this σ-arc as described in the basic HS-DAG algorithm.
- Closing: If there is a node n' in D which is labeled by ✓ and $H(n') \subseteq H(n)$ then close the node n. A label is not computed for n nor any successor nodes are generated.

Pruning Rules

- Pruning: If the set Σ is to label a node n and it has been used previously, then attempt to prune *D* as described in the following:
  1. If there is a node *n'* which has been labeled by the set *S'* of *F* where $\Sigma \subset S'$, then relabel *n'* with Σ. For any α in $S' \setminus \Sigma$, the α–edge under n' is no longer allowed. The node connected by this edge and all its descendants are removed, except those nodes with another ancestor which is not being removed. Note that this step may eliminate the node which is currently processed.
  2. Interchange the sets *S'* and Σ in the collection. Note that this has the same effect as eliminating *S'* from *F*.

Example HS-DAG

- $F = \{\{a,b\},\{b,c\},\{a,c\},\{b,d\},\{b\}\}$



Drawback HS-DAG

- Need to know or compute conflict sets in advance
- Idea: Compute conflict set incrementally when they are required by the HS-DAG algorithm
- Theorem Prover: TP(SD,CH,OBS) denotes a theorem prover call returning a (not necessarily minimal) conflict set if one exists, i.e., $SD \cup OBS \cup \{\neg ab(C) \mid C \in CH\}$ is inconsistent, and ✓ otherwise.

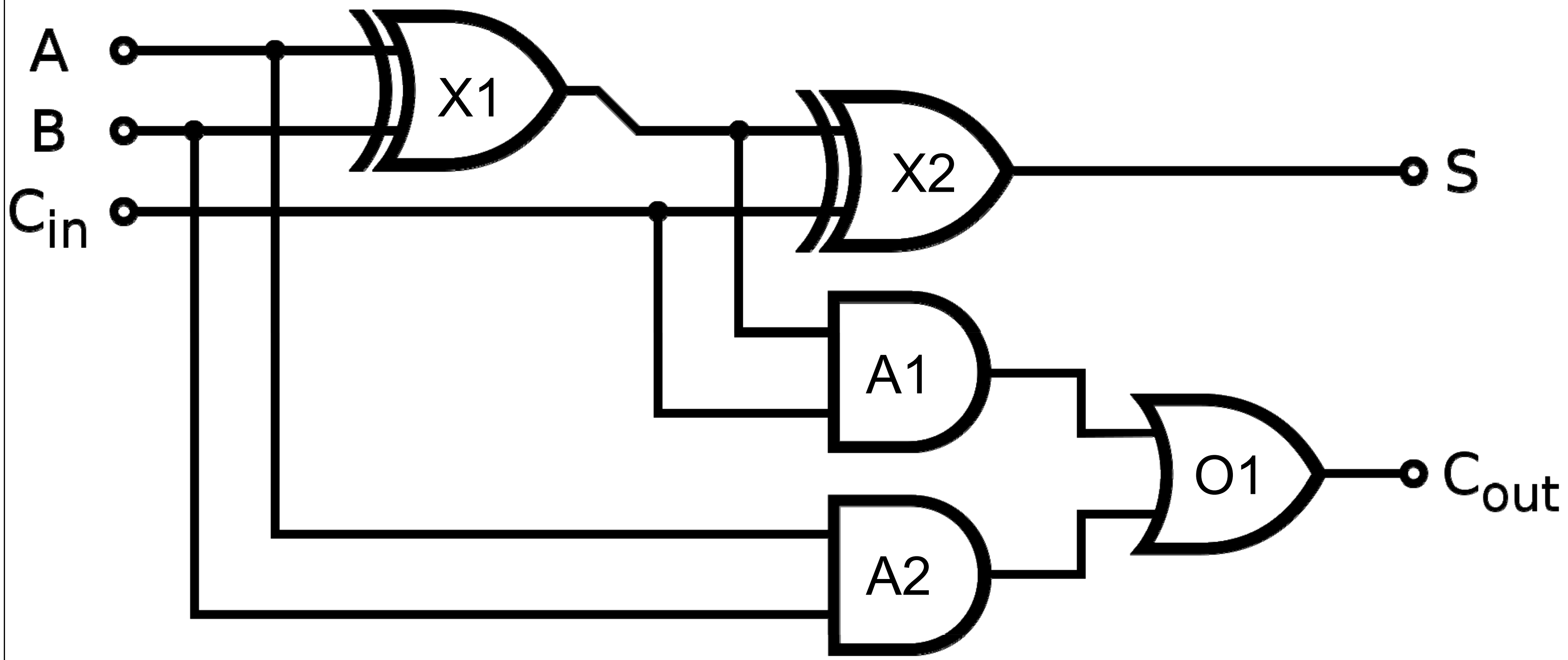
Computing Diagnoses

Graz University of Technology

$Diagnose(SD, COMP, OBS)$

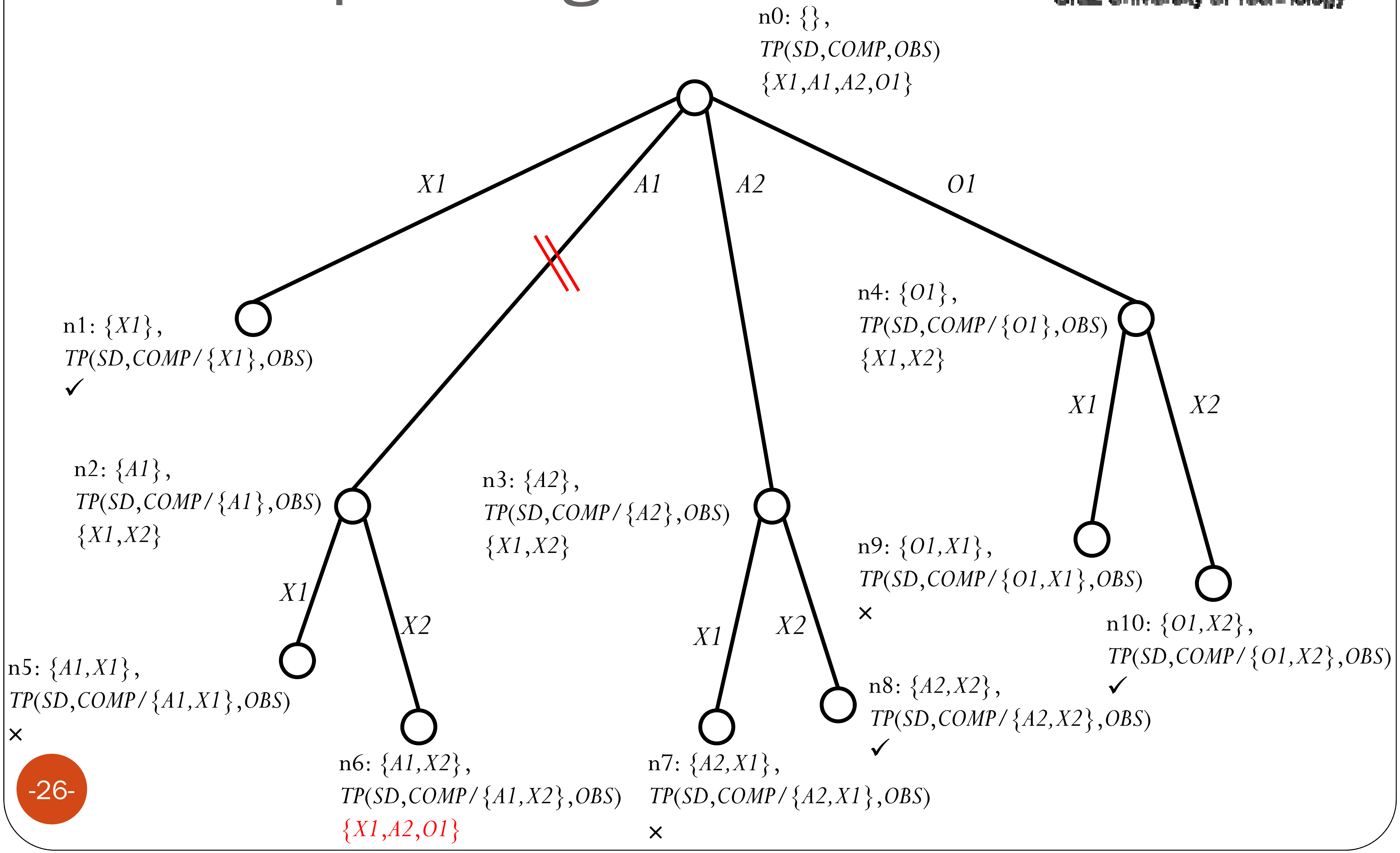
1. Generate a pruned hs-dag D for the collection F of conflict sets for $(SD, COMP; OBS)$ as described previously, except that whenever, in the process of generating D a node n of D needs access to F to compute its label, label that node by $TP(SD, COMP \setminus H(n), OBS)$.
2. Return $\{H(n) \mid \text{n is a node of } D \text{ labeld by } \checkmark\}$.

Example 1 Bit Full Adder

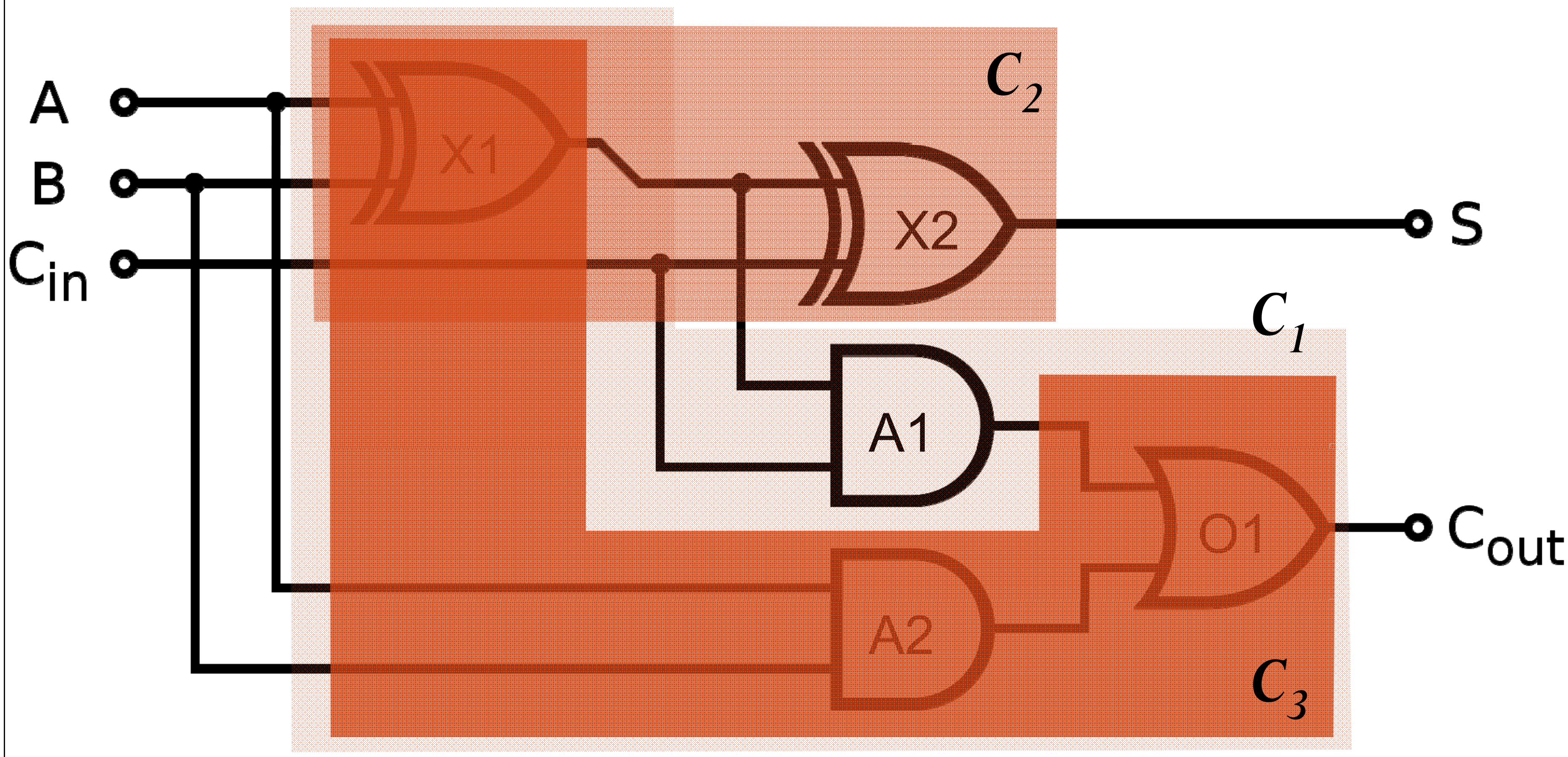


$$OBS: A=1, B=0, C_{in}=1, S=1, C_{out}=0$$

Example Diagnose



Conflicts 1 Bit Full Adder



Multiple Diag. Candidates

- Problem: How to distinguish between several diagnoses candidates (discrimination)?
- Idea: Use additional measurements?
- Additional measurements are i.e. costly. How to select the most valuable additional measurement?

Measurement Selection

- Definition 5: A diagnosis Δ for $(SD, COMP, OBS)$ predicts Π iff
  $SD \cup OBS \cup \{ab(C) \mid C \in \Delta\} \cup \{\neg ab(C) \mid C \in COMP \setminus \Delta\} \models \Pi$
  i.e., on the assumption that the components of Δ are all faulty, and the remaining components are all functioning normally, the system behavior Π must hold.
- Proposition 8: A diagnosis Δ for $(SD, COMP, OBS)$ predicts Π iff
  $SD \cup OBS \cup \{\neg ab(C) \mid C \in COMP \setminus \Delta\} \models \Pi$

Measurement Selection

- Theorem 9: Suppose every diagnosis of $(SD, COMP, OBS)$ predicts one of $\Pi, \neg\Pi$. Then:
  1. Every diagnosis which predicts Π is a diagnosis for $(SD, COMP, OBS \cup \{\Pi\})$.
  2. No diagnosis which predicts $\neg\Pi$ is a diagnosis for $(SD, COMP, OBS \cup \{\Pi\})$.
  3. Any diagnosis for $(SD, COMP, OBS \cup \{\Pi\})$ which is not a diagnosis for $(SD, COMP, OBS)$ is a strict superset of some diagnosis for $(SD, COMP, OBS)$ which predicts $\neg\Pi$. Any new diagnosis resulting from the new measurement Π will be a strict superset of some old diagnosis which predicted $\neg\Pi$.

Measurement Selection

Graz University of Technology

- Corollary 10: Suppose that {} is not a diagnosis for $(SD, COMP, OBS)$. Then under the assumption of theorem 9, any new diagnosis arising from the new measurement Π will be a multiple fault diagnosis.
- Corollary 11: Suppose that {} is not a diagnosis for $(SD, COMP, OBS)$. Then under the assumption of theorem 9, the single fault diagnoses for $(SD, COMP, OBS \cup \{\Pi\})$ are precisely those of $(SD, COMP, OBS)$ which predict Π.

Next Measurement Point

- Given: diagnosis candidates (minimal diagnoses and their superset), fault probabilities for each component $p(C)$, possible measurements $x_i = v_{ik}$ where x_i denotes the quantity and v_{ik} a value.
  - R_{ik} … candidates which remain if x_i is measured to be v_{ik}
  - S_{ik} … candidates which x_i must be v_{ik}
  - U_i … candidates which do not predict a value for x_i
  - $R_{ik} = S_{ik} \cup U_i$ and $S_{ik} \cap U_i = \emptyset$

Next Measurement Point

- The best measurement is one which minimizes the expected entropy of candidate probabilities resulting form measurement:

$$H_e(x_i) = \sum_{k=1}^{m} p(x_i = v_{ik}) \cdot H(x_i = v_{ik})$$

where $v_{i1}, \ldots, v_{im}$ are possible values.

Next Measurement Point

$$p(x_i = v_{ik}) = p(S_{ik}) + \varepsilon_{ik}, 0 < \varepsilon_{ik} < p(U_i)$$

$$\sum_{k=1}^{m} \varepsilon_{ik} = p(U_i), p(S_{ik}) = \sum_{\Delta \in S_{ik}} p_d(\Delta), p(U_i) = \sum_{\Delta \in U_{ik}} p_d(\Delta)$$

$$p_d(\Delta) = \prod_{C \in \Delta} p(C) \cdot \prod_{C \in COMP \setminus \Delta} (1 - p(C))$$

Assume: Each v_{ik} is equal likely iff a candidate does not predict a value x_i, i.e., $\varepsilon_{ik} = p(U_i)/m$

Next Measurement Point

$$p(x_i = v_{ik}) = p(S_{ik}) + p(U_i)/m$$

$$H_e(x_i) = H + \Delta H_e(x_i)$$

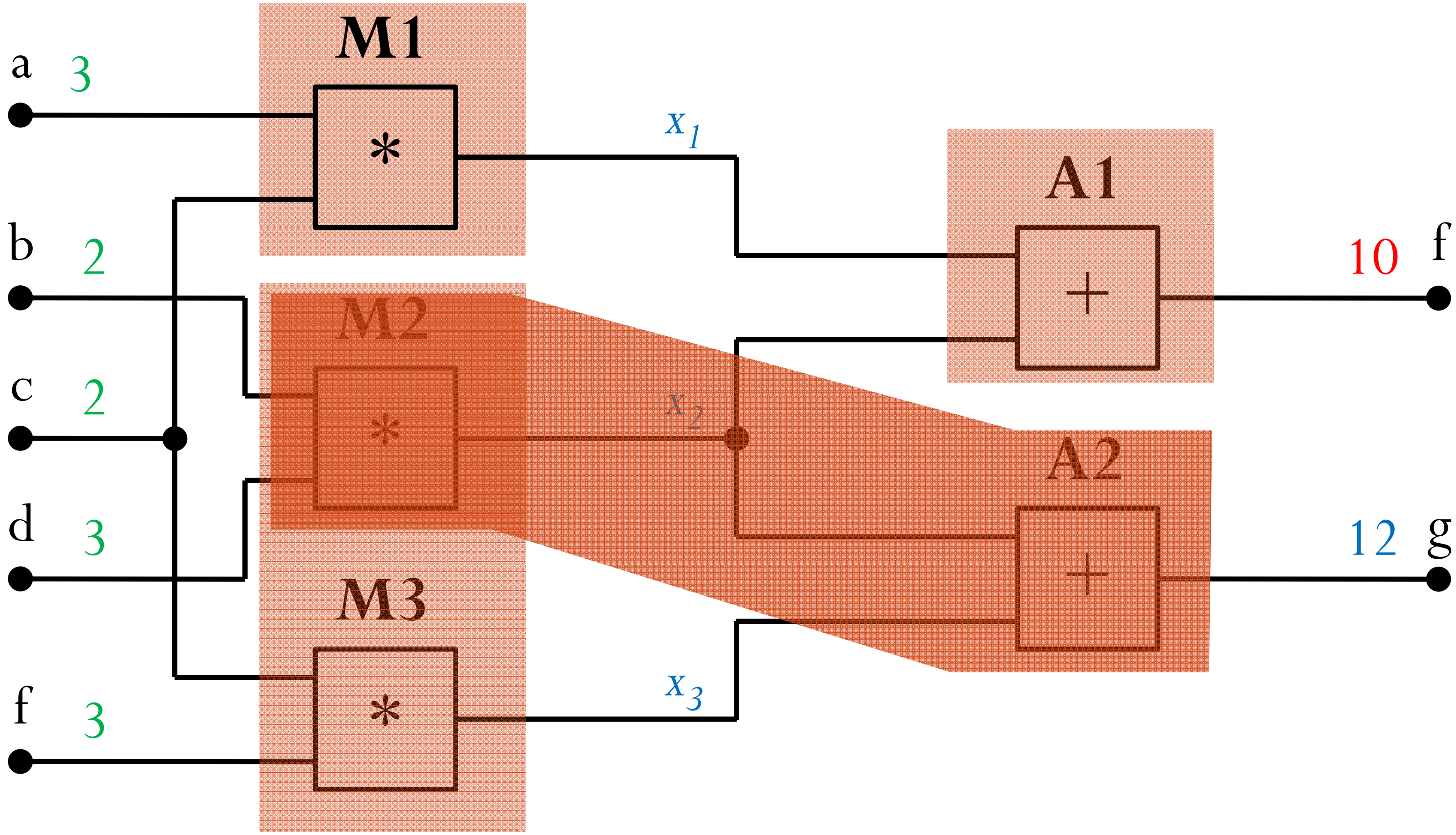
$$\Delta H_e(x_i) = \sum_{k=1}^{n} p(x_i = v_{ik}) \cdot \ln(p(x_i = v_{ik})) +$$

$$+ p(U_i) \cdot \ln(p(U_i)) - \frac{n \cdot p(U_i)}{m} \ln\left(\frac{p(U_i)}{m}\right)$$

$$n = |S_{ik}|$$

$$\min_i(\Delta H_e(x_i)) \Rightarrow \min_i(H_e(x_i))$$

Example Measur. Select.



p(M1)=p(M2)=p(M3)=p(A1)=p(A2)=0.1

Diagnosis	$p(\Delta)$	x_1	x_2	x_3
M1	0.06561	4	6	6
M1,M2	0.00729	4	6	6
M1,M2,M3	0.00081	-	-	-
M1,M2,M3,A1	0.00009	-	-	-
M1,M2,M3,A1,A2	0.00001	-	-	-
M1,M2,M3,A2	0.00009	-	-	-
M1,M2,A1	0.00081	-	6	6
M1,M2,A1,A2	0.00009	-	-	6
M1,M2,A2	0.00081	-	-	6
M1,M3	0.00729	4	6	6
M1,M3,A1	0.00081	-	6	6
M1,M3,A1,A2	0.00009	-	6	-
M1,M3,A2	0.00081	4	6	-
M1,A1	0.00729	-	6	6
M1,A1,A2	0.00081	-	6	6
M1,A2	0.00729	4	6	6

Diagnosis	$p(\Delta)$	x_1	x_2	x_3
A1	0.06561	6	6	6
A1,M2	0.00729	6	6	6
A1,M2,M3	0.00081	6	-	-
A1,M2,M3,A2	0.00009	6	-	-
A1,M2,A2	0.00081	6	-	6
A1,M3	0.00729	6	6	6
A1,M3,A2	0.00081	6	6	-
A1,A2	0.00729	6	6	6
M2,M3	0.00729	6	4	8
M2,M3,A2	0.00081	6	4	-
M2,A2	0.00729	6	4	6

Line	X	$p(X)$
X_1	$S_{1[4]}$	0.08829
	$S_{1[6]}$	0.10539
	U_1	0.01171
	$X_1=4$	0.094145
	$X_1=6$	0.111245
X_2	$S_{2[4]}$	0.01539
	$S_{2[6]}$	0.18639
	U_2	0.00361
	$X_2=4$	0.017195
	$X_2=6$	0.188195
X_3	$S_{3[6]}$	0.19368
	$S_{3[8]}$	0.00729
	U_3	0.00442
	$X_3=6$	0.19589
	$X_3=8$	0.00950

Example Measur. Select.

	X_1	X_2	X_3
Entropy	-0.458637	-0.381701	-0.360562

Computing Measurements

- Problem
  - Previous algorithm fits not for large systems, use of supersets
- Practical Solution
  - Use only computed diagnose candidates, no subersets

Revised Algorithm

- D … set of diagnoses for $(SD, COMP, OBS)$

$$p(x_i = v_{ik}) = \sum_{\Delta \in D \wedge cond(\Delta)} p_d(\Delta)$$

$$where \quad SD \cup OBS \cup \{\neg ab(C) \mid C \in COMP \setminus \Delta\} \succ (x_i = v_{ik}) \Rightarrow cond(\Delta)$$

$$v_{ik} \in \{v_{i1}, \ldots, v_{ik}\} \cup \{undef\}$$

$$H(x_i) = \sum_{v_{ik}} p(x_i = v_{ik}) \cdot \ln(p(x_i = v_{ik}))$$

- Search for $min_i \mid H(x_i) \mid$

Example Measur. Select.

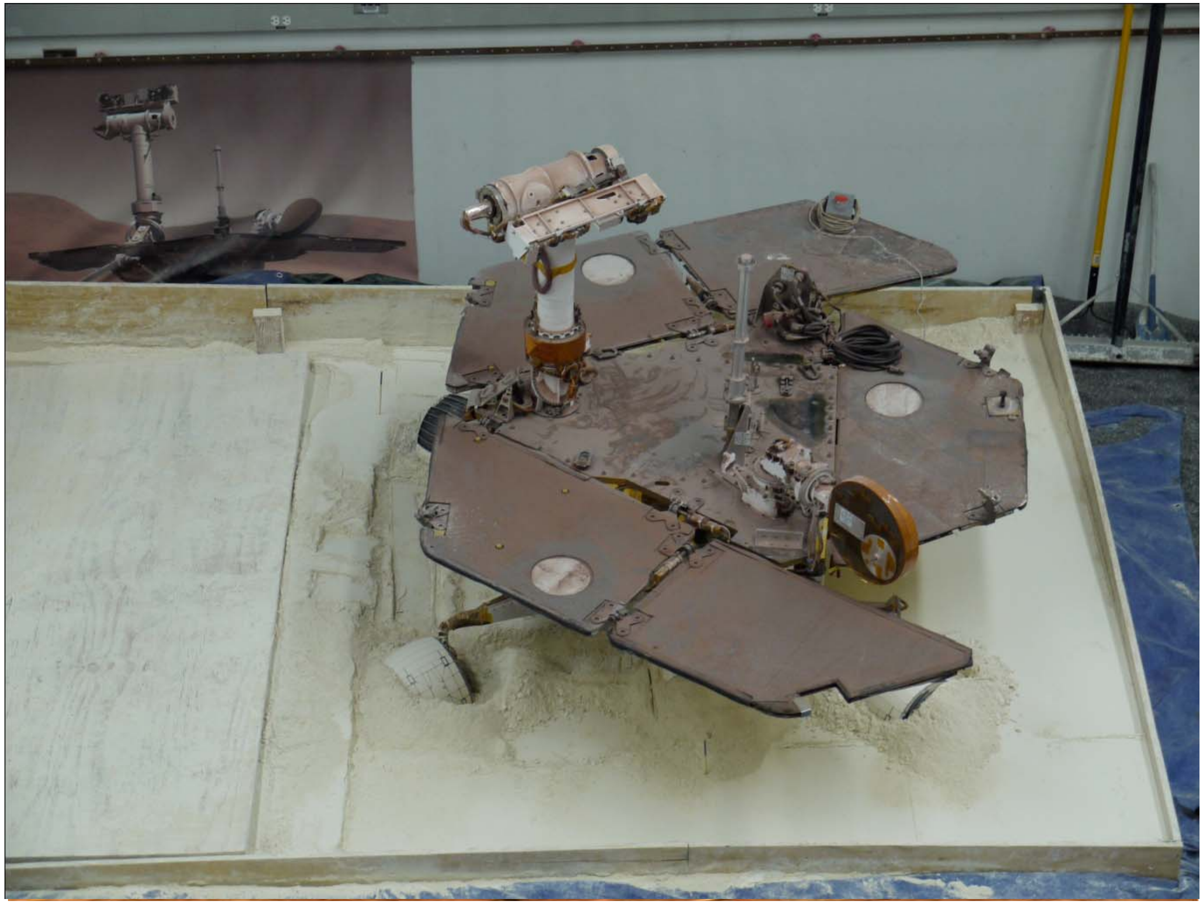
Line	X	$p(X)$
X_1	$X_1=4$	0.06561
	$X_1=6$	0.08019
X_2	$X_2=4$	0.01458
	$X_2=6$	0.13122
X_3	$X_3=6$	0.13851
	$X_3=8$	0.06561

	X_1	X_2	X_3
Entropy	-0.381071	-0.328138	-0.452532

Literature

- Raymound Reiter, *A theory of diagnosis from first principles*, Artificial Intelligence, Volume 32, Issue 1 (April 1987)
- Russel Greiner, Babara Smith and Ralph Wilkerson, *A correction to the algorithm in Reiter's theory of diagnosis*, Artificial Intelligence, Volume 41 , Issue 1 (November 1989)
- Johan de Kleer and Brian Williams, *Diagnosing multiple faults*, Artificial Intelligence Volume 32, Issue 1 (April 1987)

Who will replace faulty components?



© NASA/JPL

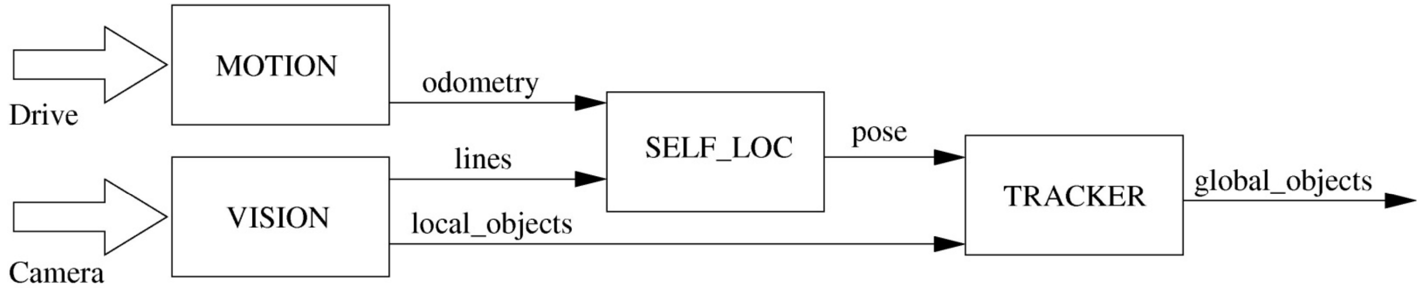
The need for model-based reasoning

- faults at runtime in robots are not totally avoidable
  - bad design, bad implementation, exogenous events, wear or damage, uncertainty
  - also military and commercial system fail frequently [Carlson & Murphy 2005]
- automatic detection, localization and repair desired for systems with no or limited possible intervention
- general methods for a wide range of systems needed
- divers properties of the systems (qualitative or quantitative)
- model-based techniques fit perfectly

Qualitative Diagnosis

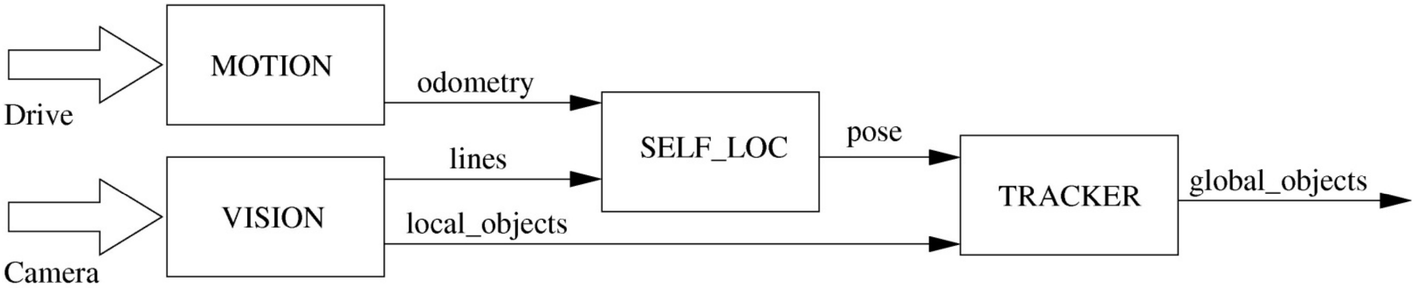
- modeling and monitoring
  - models and observations as logical clauses [Reiter 1987]
  - Horn clauses for efficiency reasons
  - component-based modeling schema
- fault detection
  - inconsistency in the logical theory
- fault localization
  - systematic resolving of the inconsistencies (retract assumptions)
- properties
  - needs discrete models and observations
  - general reasoning possible
  - usually more intuitive

Example Robot Control Software



- control software of our soccer and service robots
- based on Miro framework [Utz 2005]
- independent software modules
- communication via method calls or events (CORBA)
- diagnosis is based on the communication between modules
- component based model [Friedrich 1999]

Software Model



- ¬AB(MOTION)→ok(odometry)
- ¬AB(TRACKER)∧ok(pose)∧ok(local_objects)→ok(global_objects)

◀ ok(pose) → ¬AB(SELF_LOC)

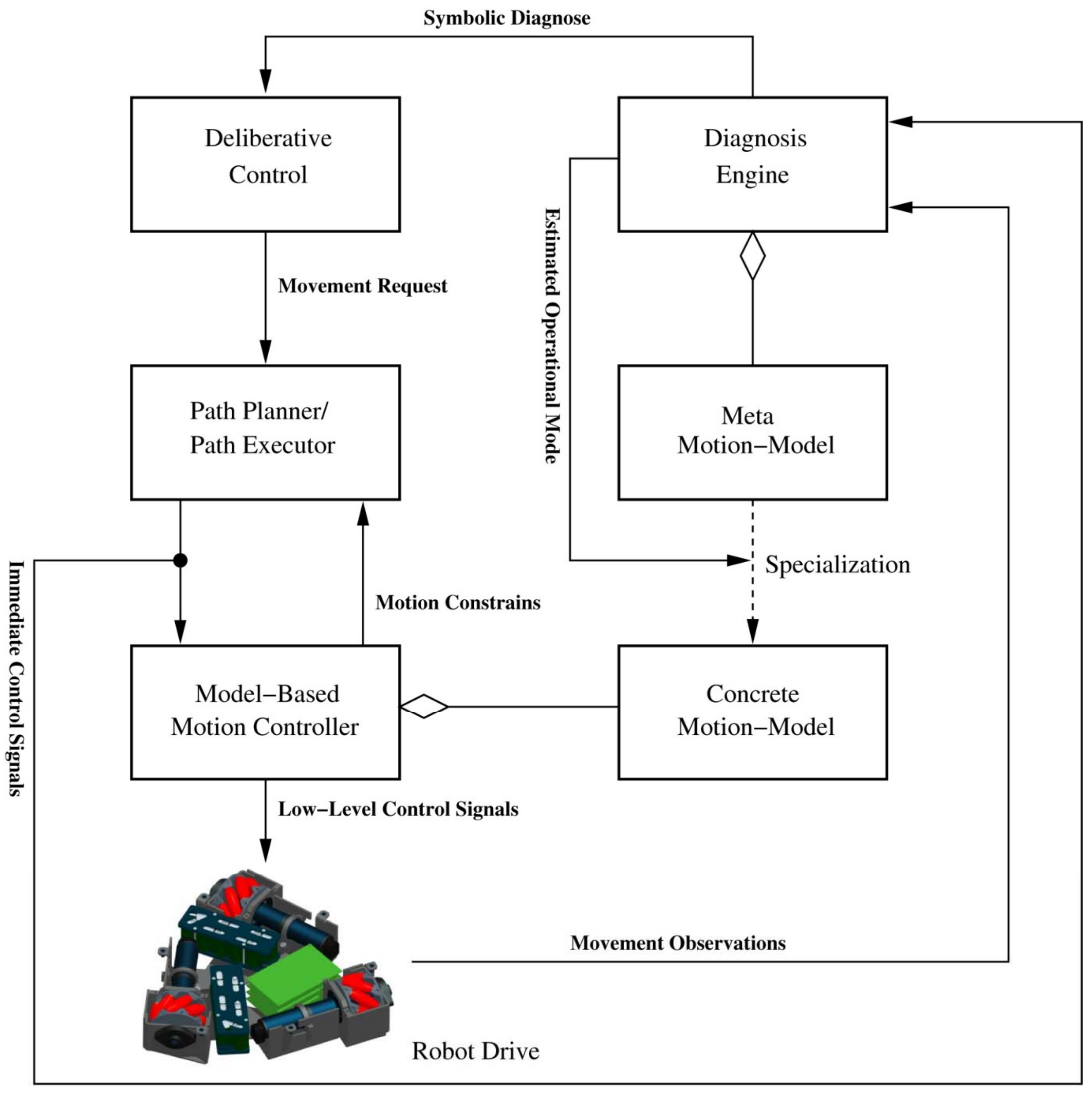
Monitoring, Diagnosis & Repair

- Monitoring connections by observers
  - periodic event production
  - conditional event production
  - periodic method call
  - observer generate the observations
- Diagnosis
  - triggered if a observer recognized a violation
  - model-based diagnosis (Reiter + LTUR [Minoux 1988])
- Repair
  - planned restart of the effected modules (direct or indirect)
- Experiments
  - successful automated recover from deadlocks and crashes

Quantitative Diagnosis

- modeling and monitoring
  - probabilistic hybrid automata [Hofbaur 2005]
- fault detection and localization
  - multi-hypotheses tracking
  - find the most probable operation mode (nominal or faulty)
- properties
  - capable to deal with continuous observations and uncertainty

Framework


Symbolic Diagnose
Deliberative Control
Diagnosis Engine
Movement Request
Estimated Operational Mode
Path Planner/ Path Executor
Meta Motion–Model
Specialization
Immediate Control Signals
Motion Constrains
Model–Based Motion Controller
Concrete Motion–Model
Low–Level Control Signals
Movement Observations
Robot Drive

Fault Scenarios

1. transparent re-configuration

   re-configuration retains full functionality (redundancy necessary)

2. controlled degrading of the functionality

   reduction to a limited but known functionality

   report of the new functionality to higher control layers

3. safe state

   fault is too bad

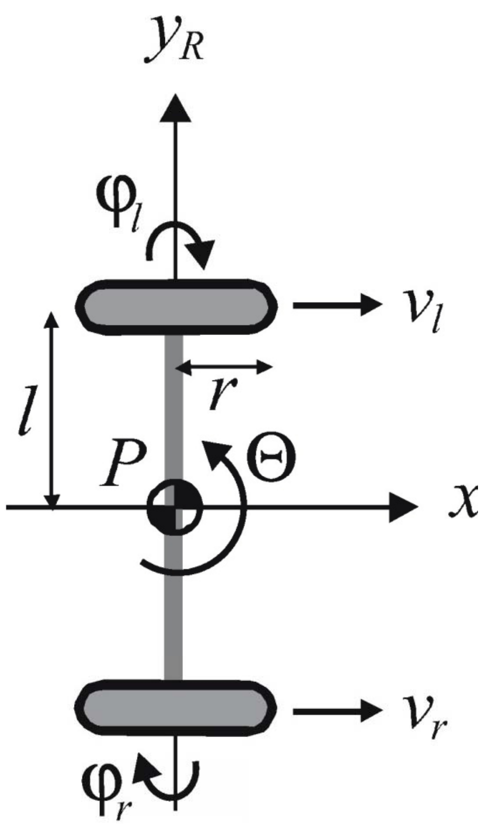
   report of that circumstance to higher control layers

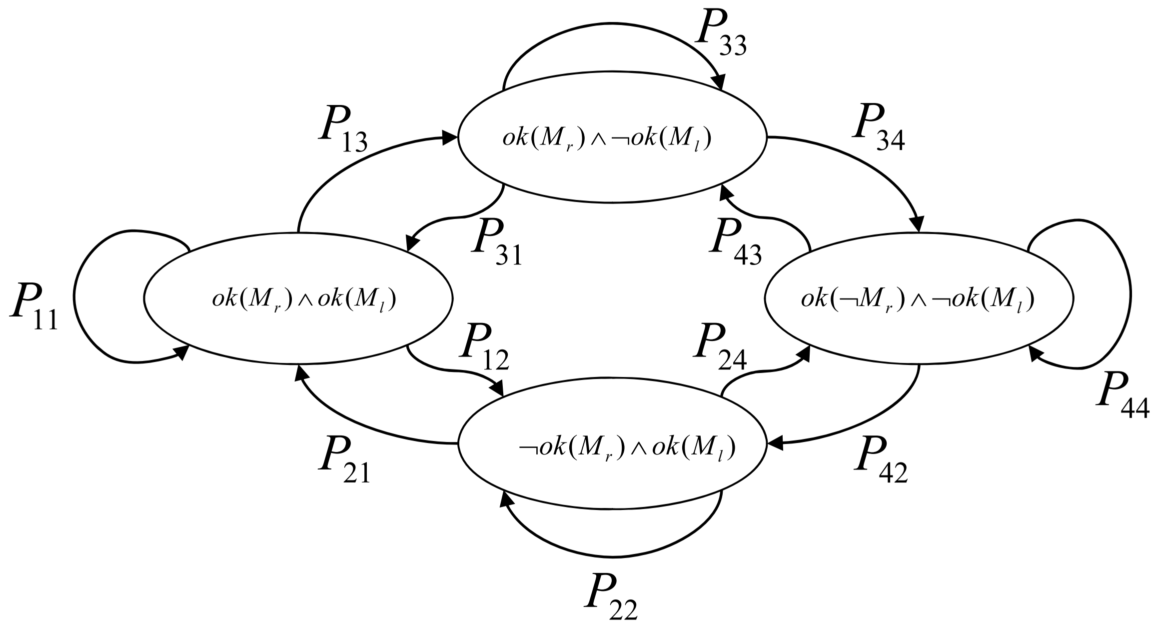
System Model

- qualitative models too coarse
- describes discrete and continuous behaviors
- handles uncertainty and noise

- probabilistic hybrid automata
- $A = \langle x, u, y, F, T, N \rangle$
  - *x* .. continuous and discrete state variables (operation mode)
  - *u* .. continuous and discrete input variables (control signals)
  - *y* .. continuous and discrete output variables (observations)
  - *F* .. equations of the systems dynamic in different mode
  - *T* .. topology of the automata, mode transitions and probabilities
  - *N* .. noise

Example - Differential Drive





$$\begin{pmatrix} v_x \\ \omega \end{pmatrix} = \begin{pmatrix} \frac{r}{2} & \frac{r}{2} \\ \frac{r}{2l} & -\frac{r}{2l} \end{pmatrix} \begin{pmatrix} \omega_r \\ \omega_l \end{pmatrix} \quad ok(M_r) \wedge ok(M_l)$$

$$\begin{pmatrix} v_x \\ \omega \end{pmatrix} = \begin{pmatrix} 0 & \frac{r}{2} \\ 0 & -\frac{r}{2l} \end{pmatrix} \begin{pmatrix} \omega_r \\ \omega_l \end{pmatrix} \quad \neg ok(M_r) \wedge ok(M_l)$$

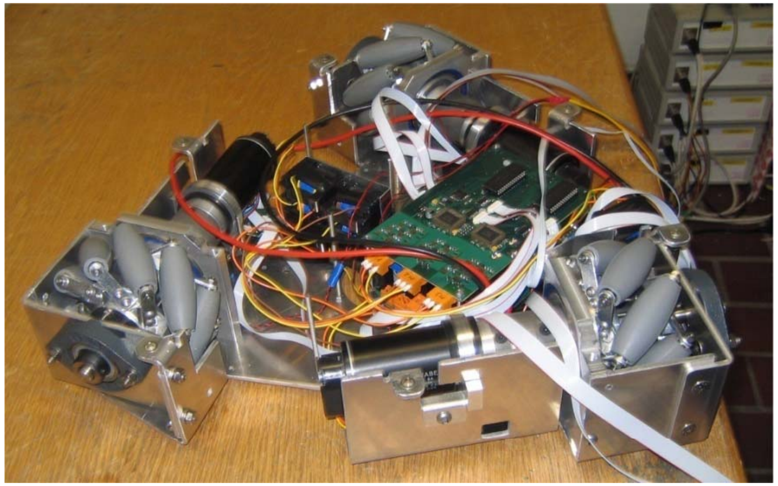
Quantitative Diagnosis Process

- faults are modeled as operation mode
- fault detection & localization as mode estimation
- problems are uncertainty and noise

- inputs for diagnosis are input/output sequences
- find the most probable mode (ok or ¬ok)
- multi-hypotheses tracking as a solution
  - filter for continuous values
  - hypotheses-tree for discrete states
- problem state explosion
  - pruning techniques

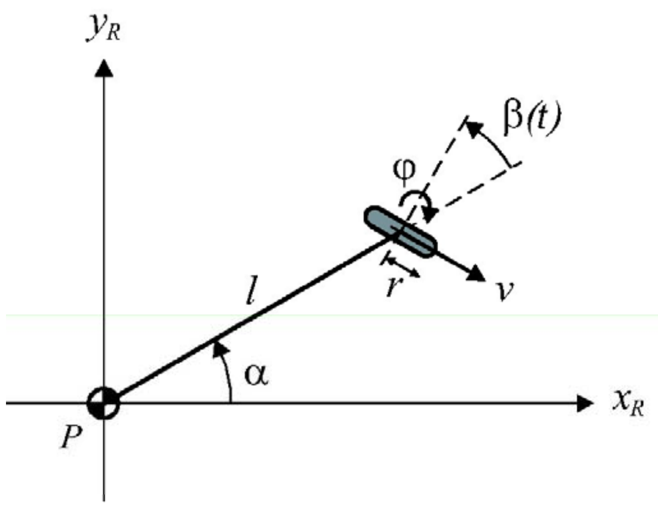
Re-Configuration in Case of a Fault

- by the model-based controller
- maps the desired movements to low-level actuator commands
- on-line re-configuration of the control laws (according the detected faults)
- oracle about the space of the possible movements
- gives up DOFs if needed
- informs higher layers (path-planner, mission planner)



Modeling of the Kinematics

- uses rolling und sliding constraints
- supports all types of wheels
  - castor wheels
  - omni-wheels
  - steered wheels
  - standard wheels
- combination of constraints of all wheels models the kinematics
- allows to determine the space of admissible and controllable movements Σ



Information in the Matrix Σ

rank of the matrix Σ is equivalent to the DOFs of the robot

$rank(\Sigma) = 3$	omni-directional drive
$rank(\Sigma) = 2$	position of the ICR limited to a single line
$rank(\Sigma) = 1$	rotation around one point
$rank(\Sigma) = 0$	no movement possible

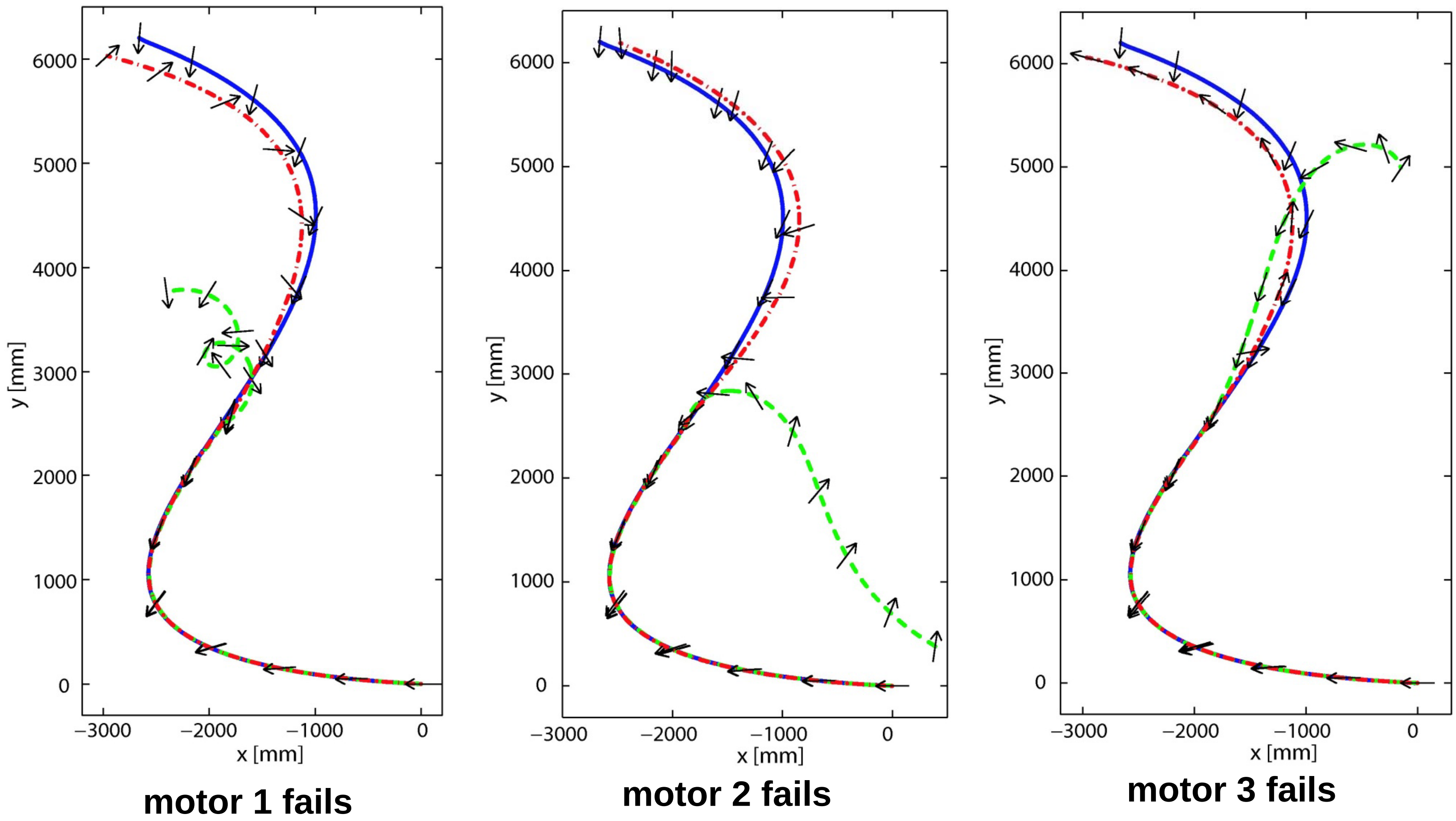
limitations of movements can be directly derived from the Matrix Σ

ICR on the plane, a single line or a point

Example Omni-Drive

Diagnosis Experiments
Omni-Drive

Graz University of Technology



Conclusion

- automated reaction to faults are desired for truly autonomous systems
- model-based reasoning can help
  - fault detection, localization and repair
  - general method
- different modeling schema
  - qualitative
  - quantitative
- successful applications
  - control software
  - drive hardware
  - robot belief (future research)

Graz University of Technology

Thank You!