Debugging Unrealizable Specifications with Model-Based Diagnosis*

Robert Könighofer, Georg Hofferek, and Roderick Bloem

IAIK – Graz University of Technology

robert.koenighofer@student.tugraz.at

www.iaik.tugraz.at

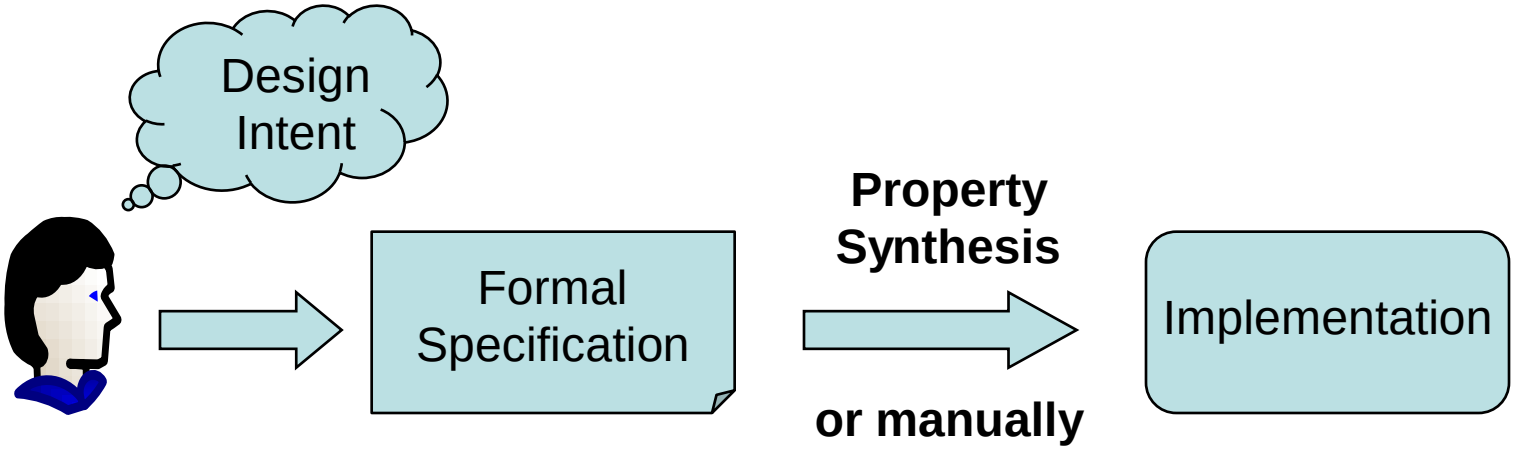
* This work was supported in part by the European Commission through project DIAMOND (FP7-2009-IST-4-248613)

Contents

- Motivation
- Model-Based Diagnosis (MBD)
- MBD for Unrealizable Specifications
- Experimental Results
- Conclusion
- Questions and Discussion

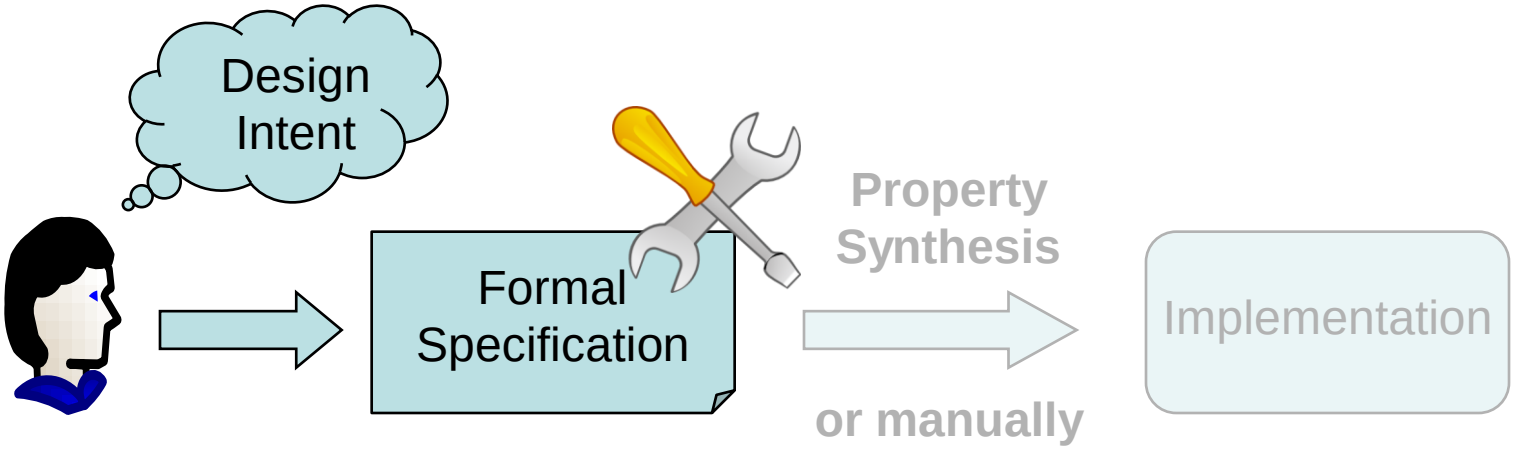
Motivation

- Assumed design flow:



Motivation

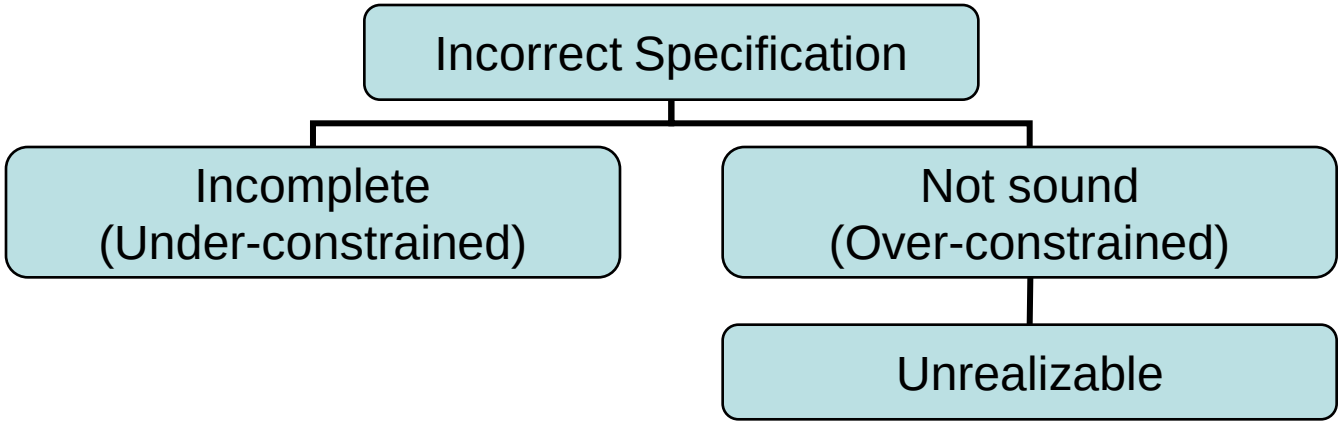
- Assumed design flow:



- **Addressed Problem**:
  Debugging of formal specifications before an implementation is available
- Our main motivation: property synthesis

Motivation

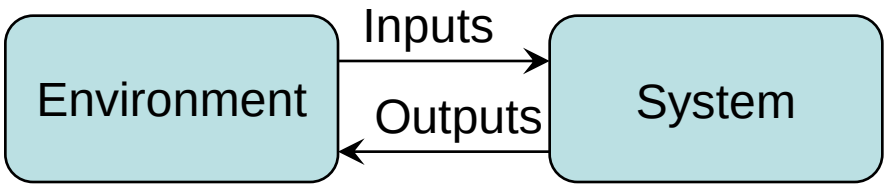
- The specification has to be correct!



- We focus on debugging **unrealizability**
- Unsoundness can be reduced to unrealizability [Königh09]

Setting

- Reactive Systems:



- Temporal specifications of the form (A,G)
  - A = set of environment assumptions
  - G = set of system guarantees
  - If the environment fulfills all $a \in A$, the system must fulfill all $g \in G$
  - Special case: A={}

Setting - Realizability

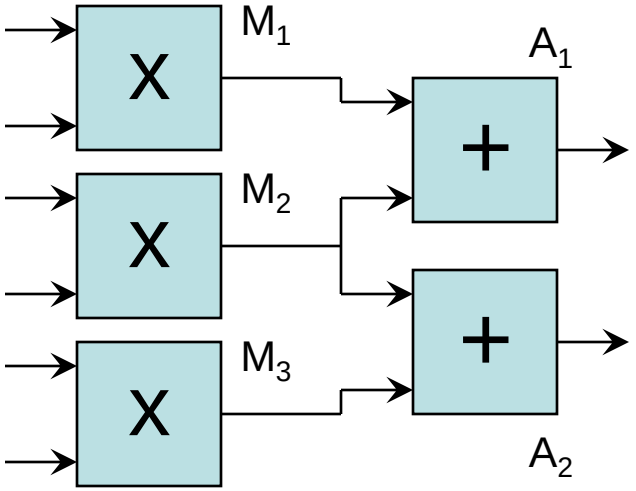
- Realizable:
  - Implementable by a Mealy machine
  - $\forall \overrightarrow{in} : \exists \overrightarrow{out} : (\overrightarrow{in} \,||\, \overrightarrow{out}) \models Spec$
  - + outputs depend on past and present inputs only
- Satisfiable: $\exists \overrightarrow{in} : \exists \overrightarrow{out} : (\overrightarrow{in} \,||\, \overrightarrow{out}) \models Spec$

- ***always***(OUT=1) $\wedge$ ***always***(OUT=0)
  - unsatisfiable, unrealizable
- ***always***(IN=1 $\Rightarrow$ OUT=1) $\wedge$ ***always***(IN=1 $\Rightarrow$ OUT=0)
  - satisfiable, unrealizable

Model-Based Diagnosis [Reiter87]

Given:

- System Description –
- Components
- Observation
- Contradiction



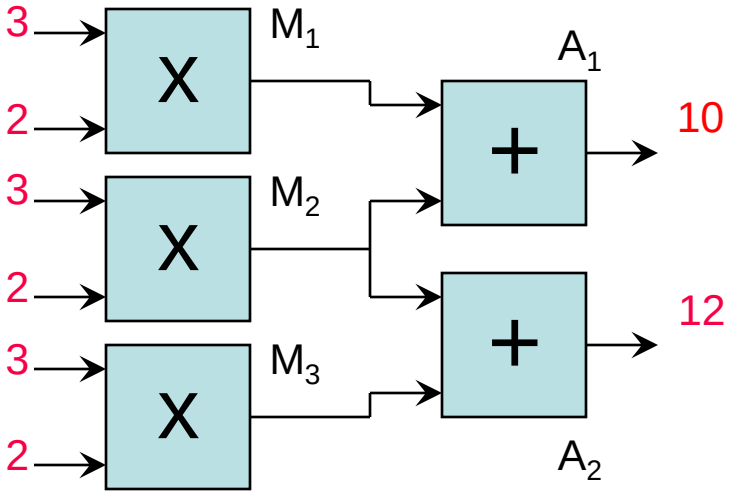
Find:

- Which components may be responsible for the observation?

Model-Based Diagnosis [Reiter87]

Given:

- System Description
- Components
- Observation
- Contradiction



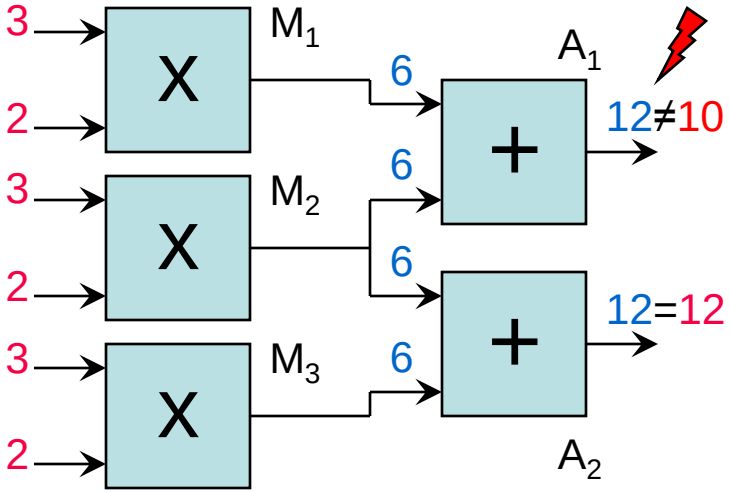
Find:

- Which components may be responsible for the observation?

Model-Based Diagnosis [Reiter87]

Given:

- System Description
- Components
- Observation
- Contradiction



Find:

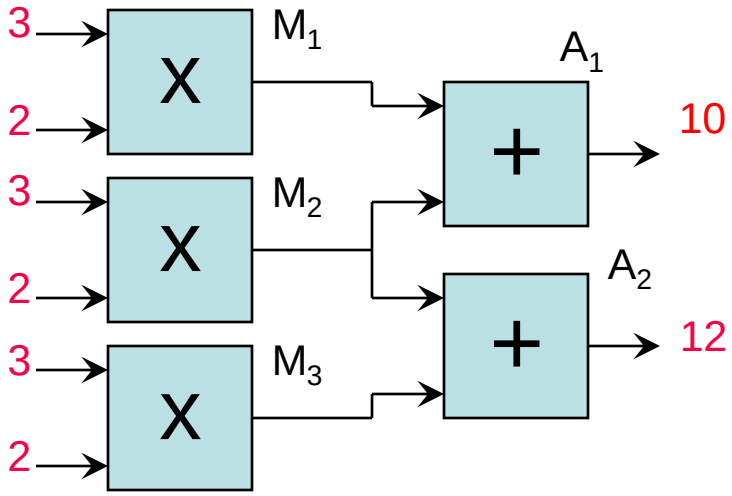
- Which components may be responsible for the observation?

Model-Based Diagnosis [Reiter87]:

Conflicts and Diagnoses

Conflicts

- Sets of components that cannot all behave normally
- $\{M_1,M_2,A_1\},\{M_1,M_3,A_1,A_2\}$, …, $\{M_1,M_2,M_3,A_1,A_2\}$



Diagnoses

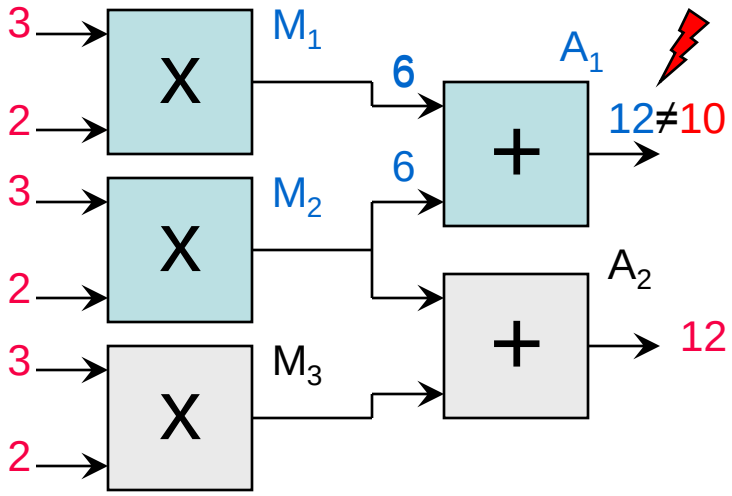
- Minimal sets of components that, if assumed to behave abnormally, explain the observation
- $\{M_1\},\{A_1\},\{M_2,M_3\},\{A_2,M_2\}$, but not e.g. $\{M_3,A_2\}$

Model-Based Diagnosis [Reiter87]:

Conflicts and Diagnoses

Conflicts

- Sets of components that cannot all behave normally
- {M_1,M_2,A_1},{M_1,M_3,A_1,A_2}, …, {M_1,M_2,M_3,A_1,A_2}



Diagnoses

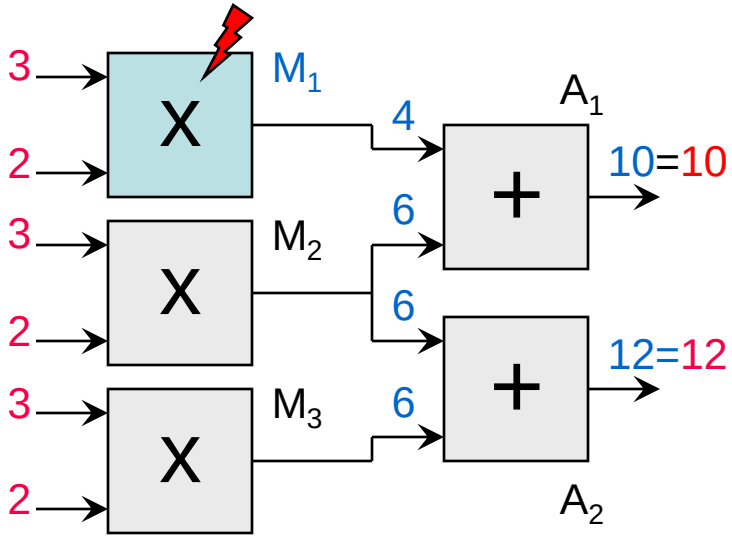
- Minimal sets of components that, if assumed to behave abnormally, explain the observation
- {M_1},{A_1},{M_2,M_3},{A_2,M_2}, but not e.g. {M_3,A_2}

Model-Based Diagnosis [Reiter87]:

Conflicts and Diagnoses

Conflicts

- Sets of components that cannot all behave normally
- $\{M_1,M_2,A_1\},\{M_1,M_3,A_1,A_2\}$, …, $\{M_1,M_2,M_3,A_1,A_2\}$



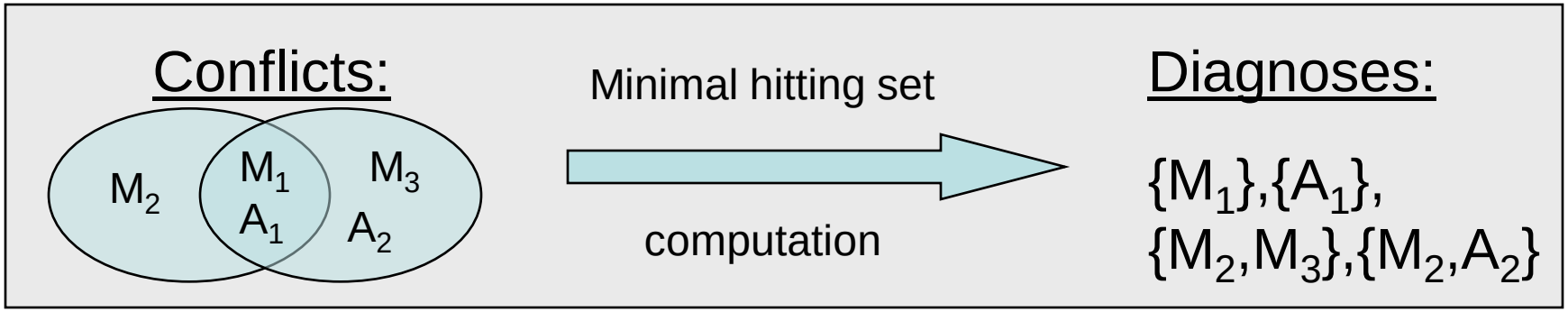
Diagnoses

- Minimal sets of components that, if assumed to behave abnormally, explain the observation
- $\{M_1\},\{A_1\},\{M_2,M_3\},\{A_2,M_2\}$, but not e.g. $\{M_3,A_2\}$

Model-Based Diagnosis [Reiter87]:

Diagnosis Computation

- Every diagnosis must have at least one element in common with every conflict
- Such a set is called “hitting set”
- → Compute all minimal hitting sets for the collection of all (minimal) conflicts
- Algorithm: e.g. hitting set tree algorithm of [Reiter87]



MBD for Unrealizable Specs

- System Description
- Components
- Observation
- Contradiction

g_1: $G(I_1=0 \Rightarrow O_1=1)$

g_2: $G(I_1=1 \Rightarrow O_1=0)$

g_3: $G(I_1 \Leftrightarrow O_1 \wedge I_2 \Leftrightarrow O_2)$

MBD for Unrealizable Specs

- System Description
  - The entire specification
- Components
  - Properties
- Observation

- Contradiction

g_1: $G(I_1=0 \Rightarrow O_1=1)$

g_2: $G(I_1=1 \Rightarrow O_1=0)$

g_3: $G(I_1 \Leftrightarrow O_1 \wedge I_2 \Leftrightarrow O_2)$

MBD for Unrealizable Specs

- System Description
  - The entire specification
- Components
  - Properties
- Observation
  - None
- Contradiction
  - Unrealizability

g_1: $G(I_1=0 \Rightarrow O_1=1)$

g_2: $G(I_1=1 \Rightarrow O_1=0)$

g_3: $G(I_1 \Leftrightarrow O_1 \wedge I_2 \Leftrightarrow O_2)$

MBD for Unrealizable Specs:

Conflicts and Diagnoses

- Specification $\varphi = \{g_1,\ldots,g_n\}$

Conflicts:

- A set $C \subseteq \varphi$ of properties is a conflict iff C forms an unrealizable specification.
- Minimal conflict = minimal unrealizable core
- Example: $\{g_1,g_3\},\{g_2,g_3\}$

g_1: $G(I_1=0 \Rightarrow O_1=1)$

g_2: $G(I_1=1 \Rightarrow O_1=0)$

g_3: $G(I_1 \Leftrightarrow O_1 \wedge I_2 \Leftrightarrow O_2)$

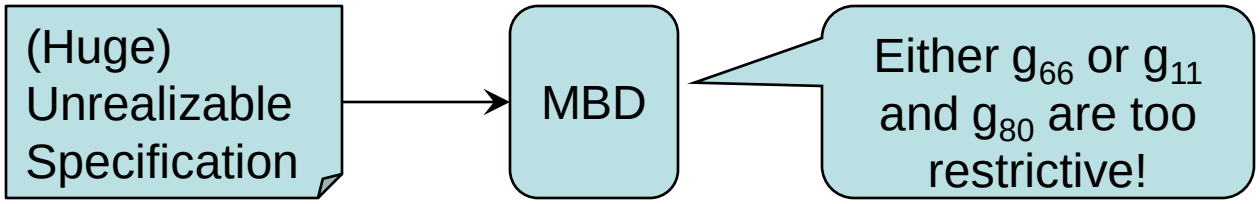
Diagnoses:

- A set $\Delta \subseteq \varphi$ of properties is a diagnosis iff Δ is a minimal set such that $\varphi \setminus \Delta$ is realizable
- Example: $\{g_3\},\{g_1,g_2\}$
- Properties of Δ can be modified in such a way that the spec becomes realizable.

MBD for Unrealizable Specs:

Diagnosis Computation

- Compute minimal unrealizable cores
  - Procedure that checks for realizability
  - Minimization algorithm
- Hitting set tree algorithm of [Reiter87]
- What we have now:



MBD for Unrealizable Specs:

Diagnosing Outputs

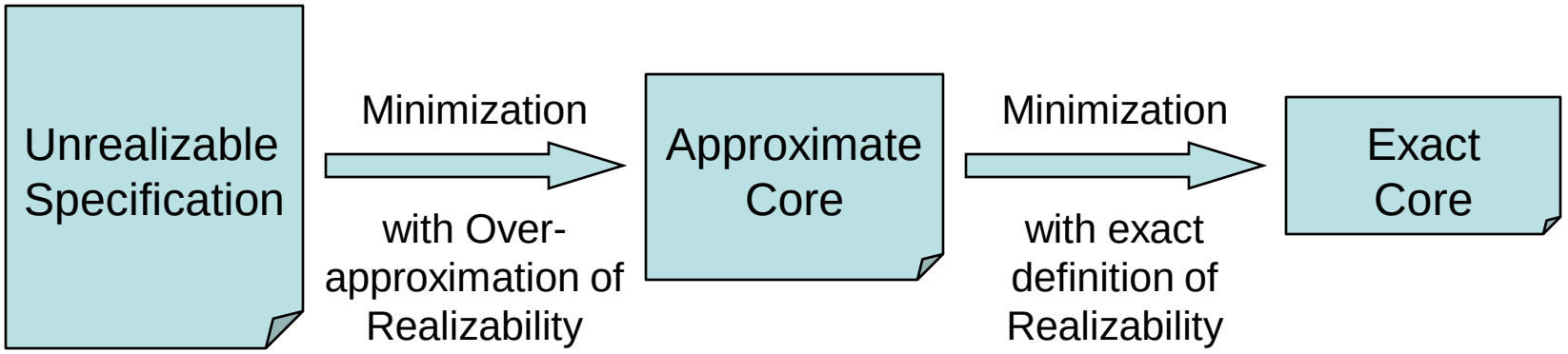
- Outputs as components
- Diagnosis =
  - Set of outputs that can be less restricted to make the spec realizable
  - … that may be over-constrained
- Need to remove restrictions on certain outputs
  - Existential quantification from properties
- Combination: Properties and Outputs
  - 2-Dimensional error localization:

			O_1		O_2	
g_1	**G(**I_1=0	⇒	O_1=1			**)**
g_2	**G(**I_1=1	⇒	O_1=0			**)**
g_3	**G((**I_1=1	⇔	O_1=1	**)** ∧ **(**I_2=1 ⇔	O_2=1	**))**

MBD for Unrealizable Specs:

Performance

- MBD is expensive
  - Many unrealizable cores
  - Many realizability checks
- Performance Optimizations
  - Under- and Over-approximations of realizability
  - 2-Step approach for unrealizable core computation:



Experimental Results

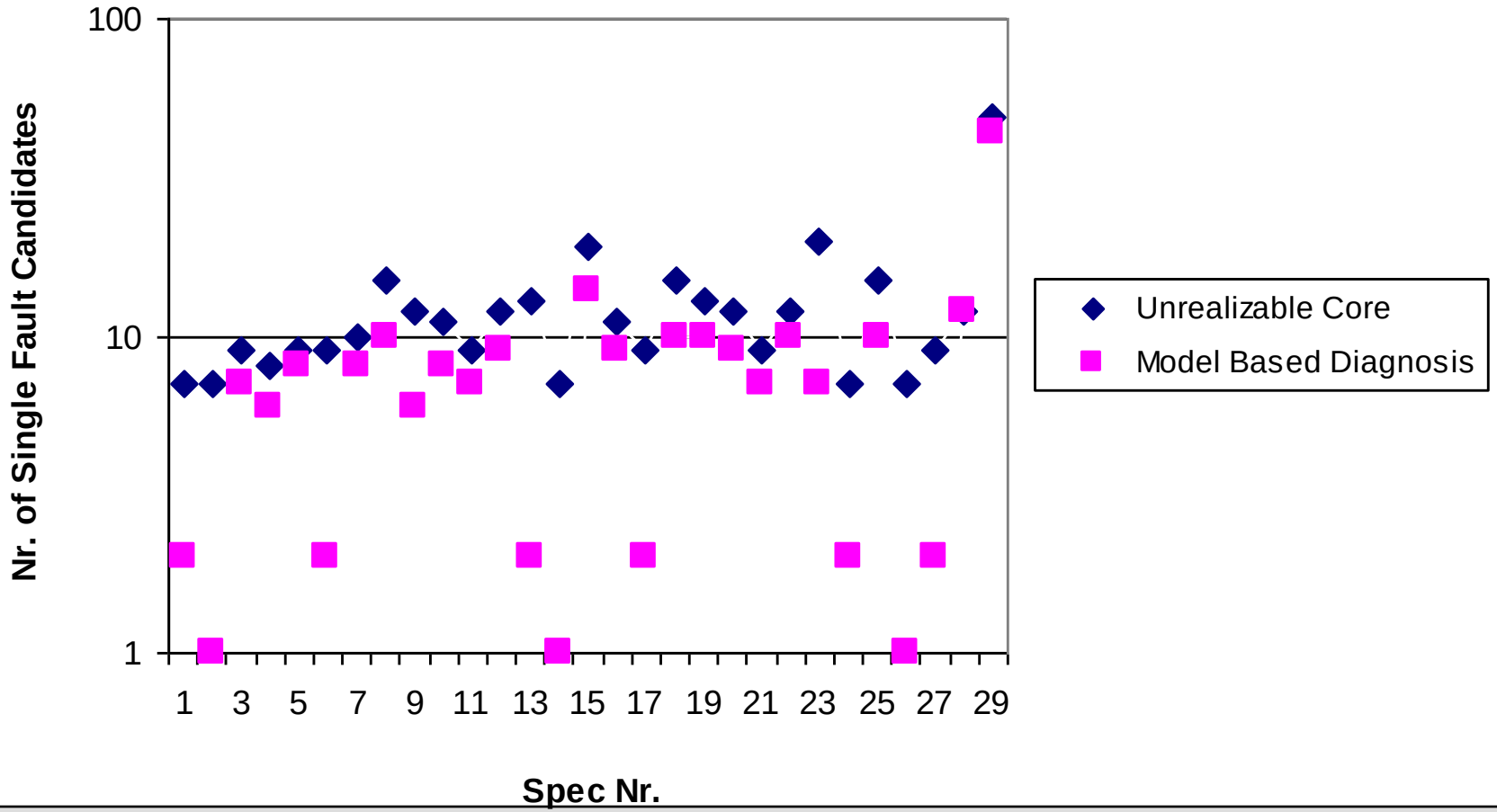
- Implemented in RATSY
- For GR(1) specifications
  - 22 to 218 signals
  - 90 to 6004 properties

- MBD is more precise than a single core
- MBD is more expensive than a single core
- Performance optimizations are effective

Experimental Results:

Improvement in the Precision

- MBD produces 40% less fault candidates



Experimental Results:

Improvement in the Precision: Example

Entire Specification:

67 Guarantees

15 Outputs

Experimental Results:

Improvement in the Precision: Example

Entire Specification:

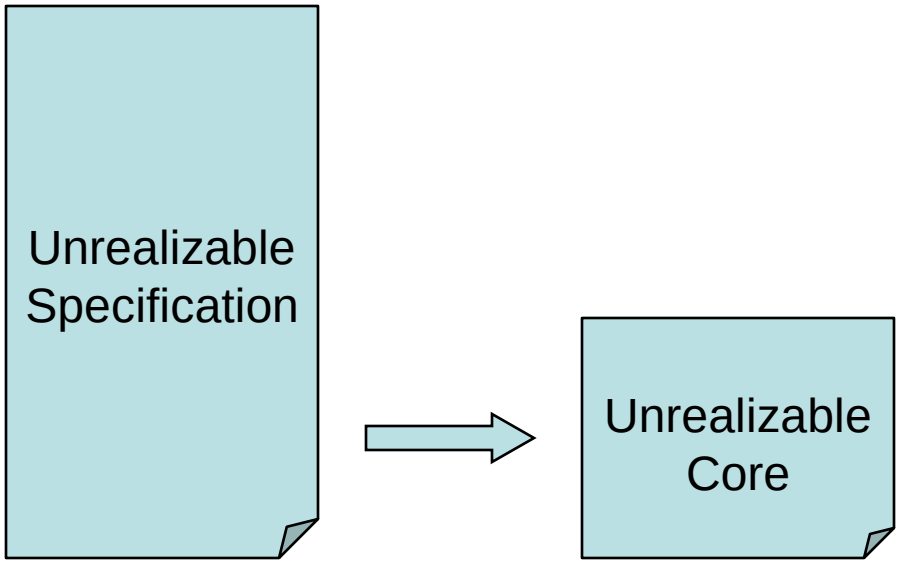
67 Guarantees

15 Outputs

Unrealizable Core:

6 Guarantees

3 Outputs



Experimental Results:

Improvement in the Precision: Example

Entire Specification:

67 Guarantees

15 Outputs

Unrealizable Core:

6 Guarantees

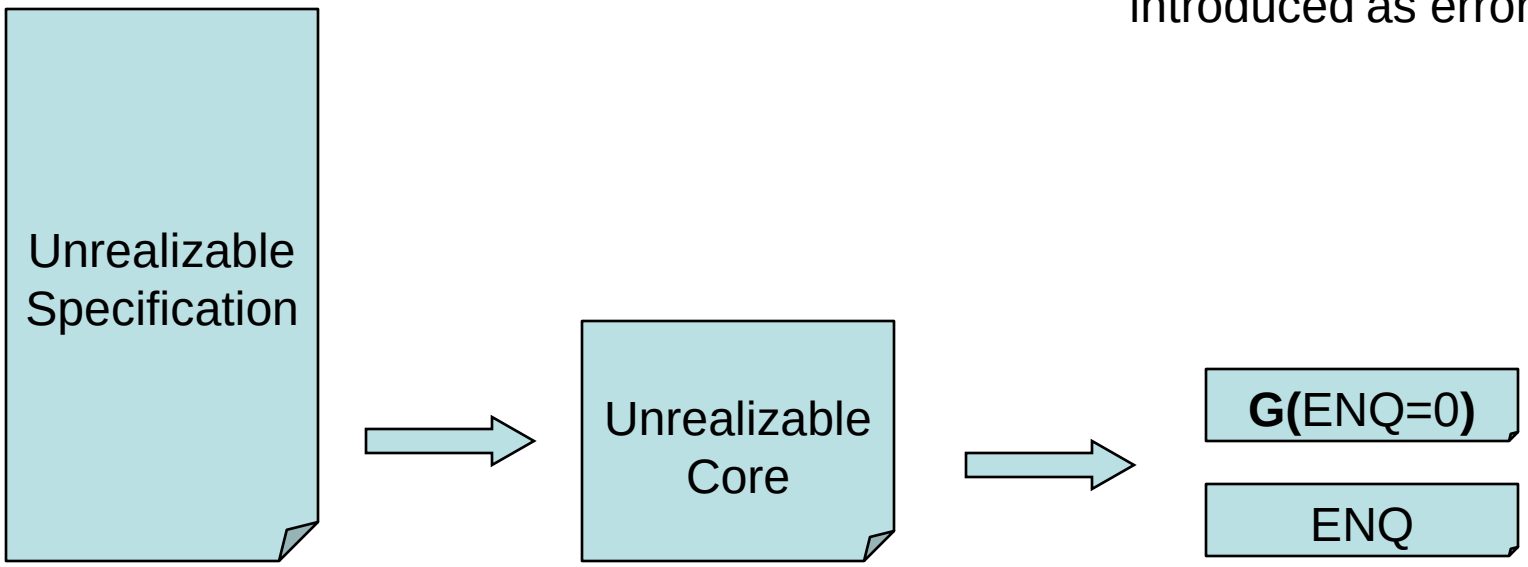
3 Outputs

Single-Fault Diagnoses:

1 Guarantee

1 Output

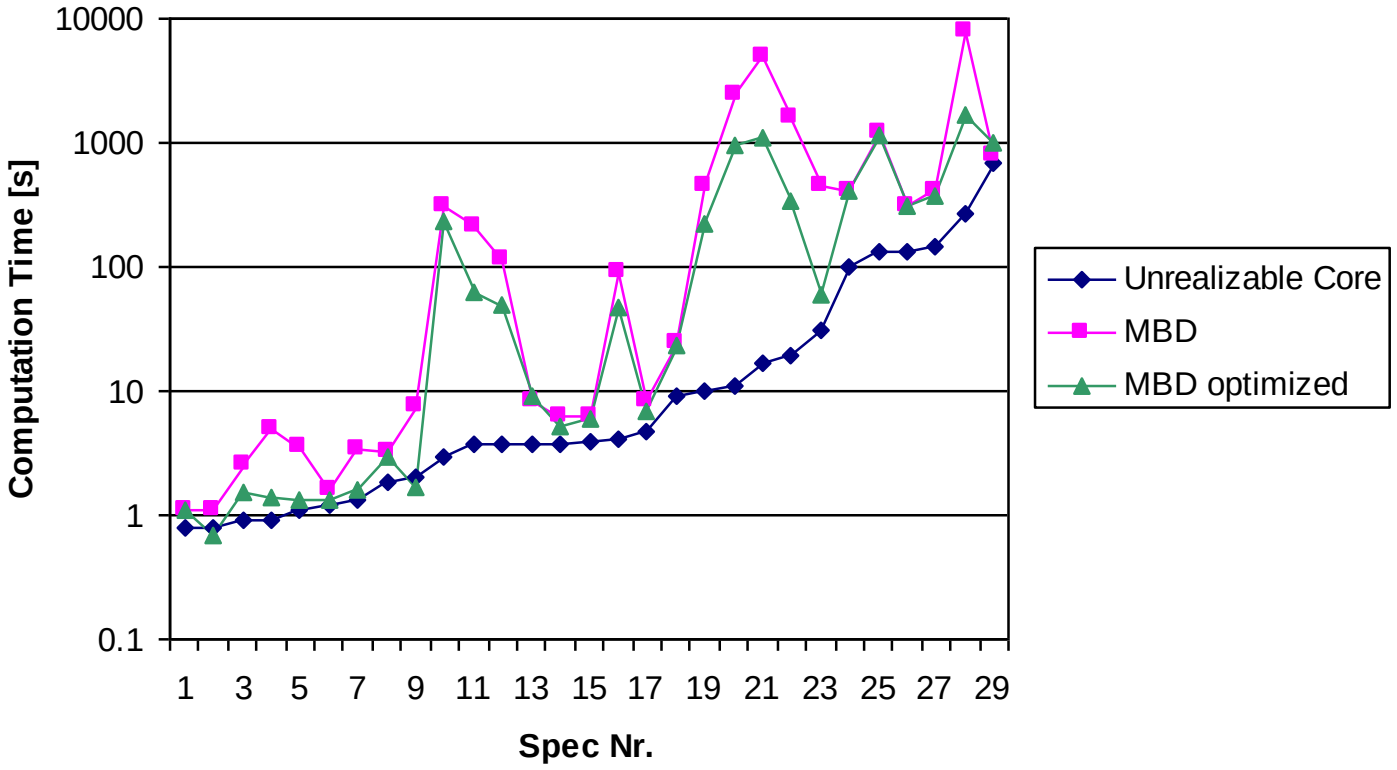
Exactly the guarantee we introduced as error!



Experimental Results:

Computation Time

- Single Fault Diagnosis with MBD is 5 times slower
- Optimizations give a speedup of factor 2.7



Conclusion

- How to apply MBD to debug unrealizability
- Combined existing techniques in a new way
- Performance optimizations with approximations of realizability are effective
- Compared to unrealizable cores:
  - More precise
  - More expensive
- Other applications:
  - Debug unsatisfiable formula in a SAT-Solver

Questions/Discussion

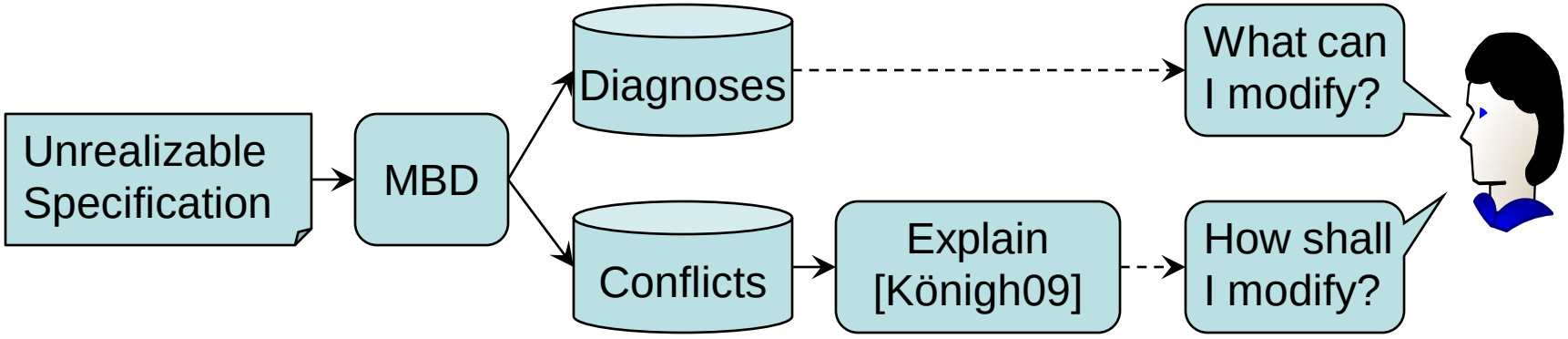
... thank you for your attention!

Literature

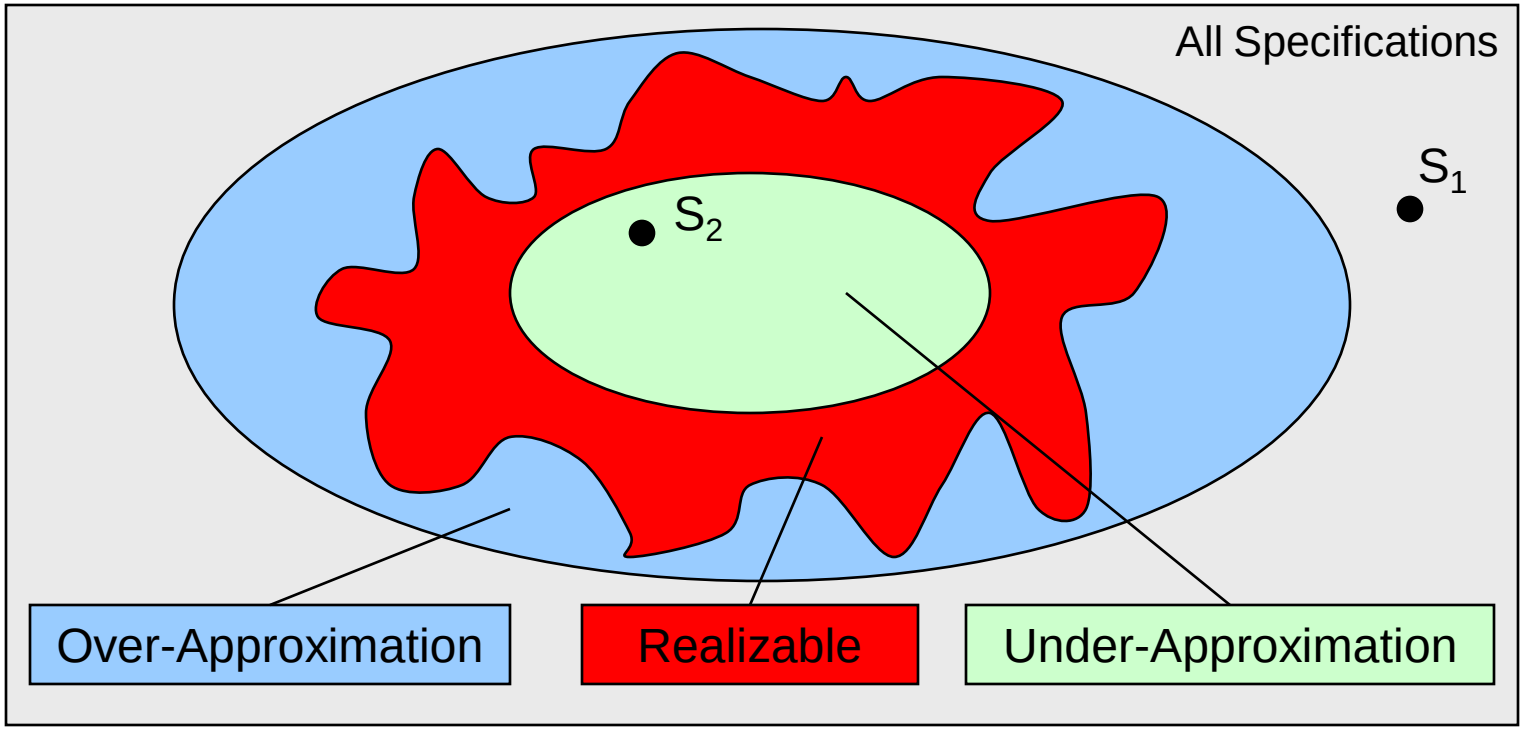
- [Cimatti08] A. Cimatti, M. Roveri, V. Schuppan, and A. Tchaltsev: „Diagnostic information for realizability“, VMCAI, LNCS 4905, 2008.
- [Königh09] R. Könighofer, G. Hofferek, and R. Bloem: „Debugging Formal Specifications Using Simple Counterstrategies“, FMCAD 2009
- [Reiter87] R. Reiter: “A theory of diagnosis from first principles”, Artif. Intell. 32, 1 (Apr. 1987), 57-95.

Related Work on Debugging Unrealizability

- [Cimatti08]: Present an unrealizable core
- [Königh09]: Illustrate unrealizable core with simplified counterstrategy
- This work: Compute fault candidates that can resolve **all** unrealizable cores
- Can be combined:



Approximations of Realizability

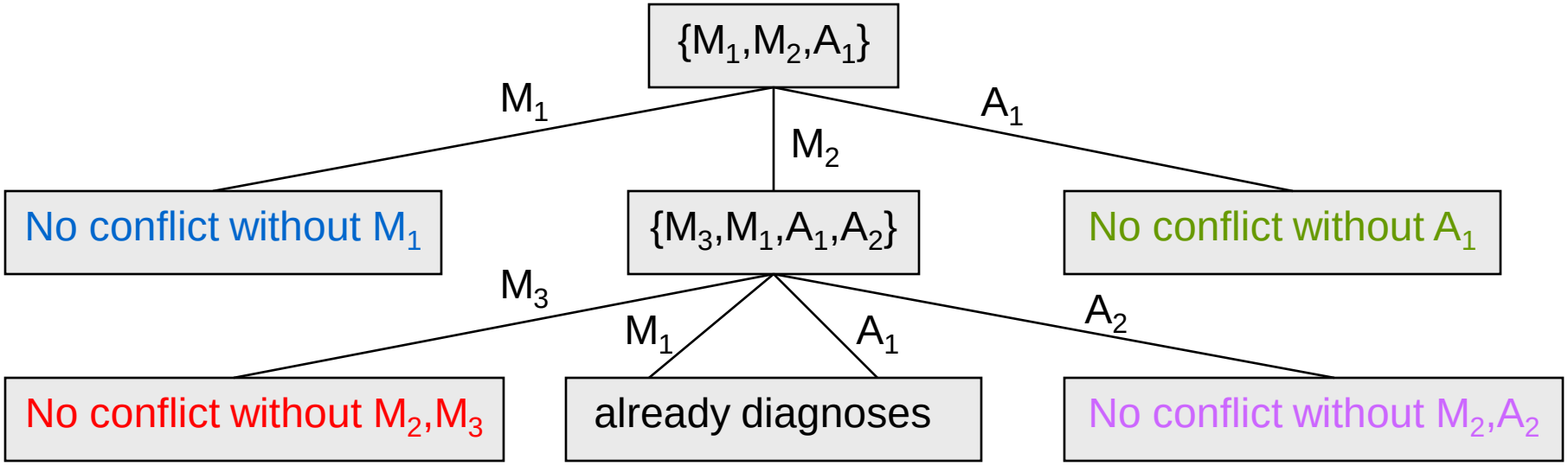


$S_1 \notin$ ⇒ $S_1 \notin$

$S_2 \in$ ⇒ $S_2 \in$

Hitting Set Tree Algorithm [Reiter87]

- Minimal conflicts: $\{M_1,M_2,A_1\},\{M_3,M_1,A_1,A_2\}$
- Hitting set tree:



- Diagnoses: $\{M_1\},\{A_1\},\{M_2,M_3\},\{M_2,A_2\}$