The imagination driving Australia's ICT future.

NICTA

Exhaustive Diagnosis of Discrete Event Systems through Exploration of the Hypothesis Space

Alban Grastien, P@trik Haslum, and Sylvie Thiébaux

Australian Government
Department of Communications, Information Technology and the Arts
Australian Research Council

NICTA Members
ANU THE AUSTRALIAN NATIONAL UNIVERSITY
UNSW
NSW Department of State and Regional Development
BusinessACT ACT GOVERNMENT
Victoria The Place To Be
The University of Melbourne
The University of Sydney
Queensland Government
Griffith University
QUT Queensland University of Technology
The University of Queensland Australia
NICTA Partners

Outline

1. Definition
2. Motivation of this Work
3. Diagnosis Tests
4. Exploration Strategies
5. Evaluation

A Definition of the Diagnosis Problem

Input

- Model D
- Observation o
- Hypothesis space $\mathcal{H}$

Diagnosis

$$\Delta \stackrel{def}{=} \{\delta \in \mathcal{H} \mid \mathrm{D}, o, \delta \not\models \perp\}$$

Preferred Diagnosis

Preference relation over hypotheses: $h \preceq h'$
Preferred diagnosis:

$$\Delta_{\preceq} \stackrel{def}{=} \min_{\preceq}(\Delta)$$

(Model-Based) Diagnosis Approaches

"Brute force" Approaches

Two steps:

- Compute all possible system behaviours
- Extract the diagnosis information from these behaviours

Darwiche et al, Lamperti & Zanella, Cordier et al.

Test Approaches

Two steps:

- Test consistency of some propositions $(\mathrm{D}, o, \mathrm{Prop}_i \overset{?}{\not\models} \bot\}$
- Deduce the diagnosis (or test more)

(Model-Based) Diagnosis Approaches

"Brute force" Approaches

Two steps:

- Compute all possible system behaviours
- Extract the diagnosis information from these behaviours

Darwiche et al, Lamperti & Zanella, Cordier et al.

Test Approaches

Two steps:

- Test consistency of some propositions $(\mathrm{D}, o, \mathrm{Prop}_i \overset{?}{\not\models} \perp\}$
- Deduce the diagnosis (or test more)

This Paper

$\rightarrow$ exhaustive diagnosis of DES with test approaches

Outline

1. Definition
2. Motivation of this Work
3. Diagnosis Tests
4. Exploration Strategies
5. Evaluation

"Exotic" Hypothesis Spaces

Set Hypothesis Space (classical definition)

A hypothesis is defined a set of faulty components.

Multi set hypothesis space

A hypothesis records the number of occurrences of each fault.

Sequence hypothesis space

A hypothesis records the order in which the faults occur.

"Exotic" Hypothesis Spaces

Set Hypothesis Space (classical definition)

A hypothesis is defined a set of faulty components.

Multi set hypothesis space

A hypothesis records the number of occurrences of each fault.

Sequence hypothesis space

A hypothesis records the order in which the faults occur.

Question

How does the test approach cop with such hypothesis spaces?

Remarks

Assumption

$\langle \mathcal{H}, \preceq \rangle$ is a partially ordered set

Vocabulary

$h \preceq h'$:

- h' is a descendant of h
- h is an ancestor of h'

$h \prec h' \wedge \forall h'' \in \mathcal{H}, h \preceq h'' \prec h' \Rightarrow h'' = h$

- h' is a child of h
- h is a parent of h'

Outline

1. Definition
2. Motivation of this Work
3. Diagnosis Tests
4. Exploration Strategies
5. Evaluation

Test

Unformally

Is there a diagnosis candidate in the specified set S?

Formally

Given a set $S \subseteq \mathcal{H}$ of hypotheses, does S intersect Δ?

$$\mathrm{D}, o, S \overset{?}{\not\models} \bot$$

If so, return one such candidate $\delta \in S \cap \Delta$

Examples of Tests (1/2)

Candidacy of Hypothesis *h*

‘‘Is *h* a candidate?”

$$S_{cand}(h) = \{h\}$$

Completeness of $D \subseteq \Delta$

“Are there (minimal) candidates besides D?”

$$S_{find}(D) = \mathcal{H} \setminus \{h \in \mathcal{H} \mid \exists h' \in D : h' \preceq h\}$$

Examples of Tests (2/2)

Refinement of δ

"Is there a strictly better candidate than δ?"

$$S_{ref}(\delta) = \{h \in \mathcal{H} \mid h \preceq \delta \wedge h \neq \delta\}$$

Relevance of h with respect to D

"Is there a candidate that is a descendant of h and not a descendant of a hypothesis of D?

$$S_{ess}(h, D) = \{h' \in \mathcal{H} \mid h \preceq h'\} \setminus \{h' \in \mathcal{H} \mid \exists h'' \in D : h'' \preceq h'\}$$

Implementations

Classical Planning

Testing = finding a sequence of actions with certain constraints on the sequence

SAT

Testing = satisfiability testing of a formula

Outline

1. Definition
2. Motivation of this Work
3. Diagnosis Tests
4. Exploration Strategies
5. Evaluation

Preferred-First Strategy

Principle

- Test candidacy of the preferred hypotheses first,
- When the test of h fails, generate successors from children and test candidacy of them
- If a hypothesis h is a descendant of a proven candidate, ignore h

Theorem

PFS terminates and returns the solution if the hypothesis space is finite

Preferred-First Strategy + Essentiality

Principle

- Test the preferred hypotheses first,
- When the test of h fails, generate successors from children and test them
- If a hypothesis h is a descendant of a proven candidate, ignore h
- Ignore irrelevant hypotheses (with respect to the open list and the candidates already found)

Theorem

PFS+e terminates and returns the solution if the diagnosis is finite, the set of children of each hypothesis is finite, and every hypothesis has finite depth

Preferred-Last Strategy

Principle

Iteratively, starting with $D = \emptyset$

- Test completeness of D
- Add the candidate found to D

Remove the non minimal candidates

Theorem

PLS terminates and returns the solution if the hypothesis space is well partially ordered

Preferred-Last Strategy + Refinement

Principle

Iteratively, starting with $D = \emptyset$

- Test completeness of D
- Refine the candidate found until a minimal candidate is found
- Add the candidate found to D

~~Remove the non minimal candidates~~

Theorem

PLS+r terminates and returns the solution if the diagnosis is finite, and the hypothesis space is well-founded

Outline

1. Definition
2. Motivation of this Work
3. Diagnosis Tests
4. Exploration Strategies
5. Evaluation

Theoretical Comparison

Number of Tests

	PFS	PFS+e	PLS	PLS+r
Successes	n	$n \times d$	unb.	unb.
Failures	unb.	$k \times n \times d$	1	n
Types of tests	Q1	Q1 & Q4	Q2	Q2 & Q3

Parameters

- How many tests?
- How "complex" the test set?
- How hard for the test solver?

Experiments

Power Grid Alarms: runtime

Pb	#c	#a	#d	PFS+e	PLS	PLS+r
c-004	2	3	1	0.4	0.3	0.4
c-026	12	11	2	18.5	89.5	23.8
c-076	11	8	2	8.9	7.1	6
c-091	25	16	32	1412	—	785
c-111	26	16	8	358	691	173
w-132	16	7	2	14.4	10.5	8.4
w-331	62	44	16	—	—	545
w-338	42	31	2	820	—	166

Conclusion

Results

- First approach to exhaustive diagnosis of discrete-event systems based on consistency tests
- Definition of two classes of strategies

Next Step

Conflict