5. Consistency Based Diagnosis without Dependency Recording Engines

Belarmino Pulido
Grupo Sistemas Inteligentes
Departamento de Informática
Universidad de Valladolid
Outline

1. Problems faced by model-based diagnosis
2. Topological methods
3. Compilation methods
4. A compilation technique: possible conflicts
5. A case study
The consistency based approach to diagnosis

- Compute conflicts at run time, usually with a dependency recording engine
  - Some approaches compile dependencies

- Minimal conflicts and minimal diagnosis usually avoid exponential time and space

- But many different implementations to tackle particular real world applications
1. Problems faced by Consistency Based Diagnosis

**Modelling**
- Where do models come from for complex systems?
  - qualitative versus quantitative
- What kind of ontology:
  - component-oriented vs process oriented

**Dynamic systems**
- Lack of theory to extend the static framework to dynamic systems
- Dynamic vs time-varying systems

**On-line Dependency recording**
- What happens with real (many valued) domains?
- Specially for continuous, dynamic systems
Problems faced by Consistency Based Diagnosis (cont.)

- On line simulation
  - Simulation over time is very demanding for complex systems, specially if coupled to an ATMS

- Fault identification vs Fault detection & localization
  - Including Fault Models
  - Is possible to exonerate components?

- What is the source of complexity?
  - Complex systems or large systems (# components)

- Multiple modelling:
  - At what level of abstraction are we modelling?
  - How we can combine results from different levels of abstraction?
Problems faced by Consistency Based Diagnosis (cont.)

- RT Model-based diagnosis
- Distributed diagnosis
- Diagnosis of Hybrid Systems
- Autonomous Systems

Integration of Model-based diagnosis:
- with other diagnosis techniques
- with other tasks: re-configuration, repair, monitoring, supervision/FTC,...
- model-based diagnosis in the product life-cycle
- (re-usable) model libraries??
Some problems have been (+/-) tackled

- Modelling
- Dynamic systems
- On line simulation
  - inference over time
  - simulation vs prediction
  - open loop, closed-loop, semi-closed loop
- On-line Dependency recording
Early approaches

- DEDALE (Dague et al. 1987) → CATS (Dague et al. 1990) → DOGS (Taillibert & Loiez, 1997)
- MIMIC (Dvorak & Kuipers, 1990)
- SIDIA (Guckenbiehl & Shaffer-Richter, 1990)
- Magellan-MT (Dressler et al., 1994)
Current State

- Simulation-based vs State-based approaches
  - Simulation based diagnosis
    - observations + instantaneous constraints $\rightarrow$ (complete) current state
    - current state + differential constraints $\rightarrow$ (incomplete) next state (integration step)
  - State-based diagnosis (Struss, 97)
    - No "integration step"
    - Check for every possible "next state"

- From now on: Simulation-based
2. Topological methods

- on-line backward search through a causal or functional structure
  - estimation = propagation through causal/functional structure
  - consistency check (of qualitative values)
  - if a discrepancy is found: propagation backward: where is the source of inconsistency?

- off-line dependency-recording → compilation techniques
2.1. On-line backward search

- on-line backward search through a causal or functional structure:
  - CAEN (Bousson & Travé-Massuyès, 92),
    - causal graphs, influences,...
  - DYNAMIS (Chittaro et al., 1996),
    - functional (and teleological) models
  - TRANSCEND (Mosterman & Biswas, 1997)
    - temporal causal graphs from bond-graphs for hybrid systems (continuous & discrete behaviour)
2.2. Off-line dependency recording

- System Description = Structural and Behavioural Information
- Information/Energy paths are restricted (topological information)
- Most of the times topology is fixed
- Set of available observations is fixed
- Is possible to propagate values/energy through every path?
  - No, if no structural faults are present
Compilation Techniques (a survey)

- FDI community:
  - Staroswiecki and Declerk 89, Staroswiecki et al., 1997
  - Lunze and Schiller, 92
  - Nyberg, 01

- AI community:
  - DOGS (Loiez & Taillibert, 1997),
  - DRUM-II (Frölich & Nejdl, 1997),
  - Washio et al., 1997
  - Pulido and Alonso, 1999, 2002
  - Ligeza and Gorny, 2000

- BRIDGE:
  - Cordier et al. 00
  - BRIDGE task group within MONET2
3. Compilation techniques

- The FDI approach
  - Analytical redundancy
  - Structural Analysis
  - ARRs
3. Compilation techniques

- The AI approach
  - not only used for diagnosis
    - Candidate refinement: Nooteboom & Leemeijer, 93
    - DMBL (Moriarty: Williams & Millar, 1996; 1998)
    - ...
  - not only for consistency-based diagnosis à la GDE
    - Diagnosis based on consequences: Darwiche 95
    - *an alternative to on-line DRE*
4. A compilation technique: the possible conflict approach

Motivation:

- How Consistency-based Diagnosis can be applied to continuous dynamic systems
- GDE-like computational approach?
GDE as paradigm for Consistency-based Diagnosis

- Predict Behaviour
- Identify Conflicts
- Generate Candidates
- Refine Diagnosis
- New Observations
Main difficulties for continuous dynamic systems

- Inclusion of time in the models
  - There is no general extension for dynamic systems and Reiter's theory

- Conflict Generation
  - Consistency-check is not trivial:
    - dynamic systems may exhibit considerable delays
  - We need incremental diagnosis
  - With no additional measurements: how we can discriminate?
Prediction capabilities from local models: drawbacks

- Not obvious how to model continuous systems in a **component-based** approach

On line simulation

- Very demanding for continuous dynamic systems, especially if
  - Local models
  - Local propagation (which may easily stop!)

On line ATMS: difficulties

- Label **registered** with real values + time
- Very demanding on memory terms
4.2. Possible conflicts (Pulido and Alonso, 99)

- **Compilation technique**
  - In industrial environments the set of available measurements is known and fixed beforehand

- **Main ideas**
  - not every sub-system in SD can be a conflict
  - a minimal conflict is a strictly over-determined set of constraints, which can be solved using local propagation
  - the set of (minimal ) over-determined systems can be computed off-line
Possible conflicts can be computed off-line

Computing possible conflicts:
1. SD as a hyper-graph
2. Over-determined systems localization
3. Can these systems be solved using local propagation
Step 1: Representing SD as a hyper-graph
Step 1: Representing SD as a hyper-graph
Step 2: Looking for over-constrained sub-systems

- Minimal evaluable chain, MEC:
  - Connected and over-constrained sub-systems
  - At least, one observation
Step 3: Can the MEC be solved using local propagation?

- Each hyper-arc in a MEC can be solved in different ways
- Each MEC generates an and-or graph
- In the and-or graph zero, one or more Minimal Evaluable Models, MEM, can be found:
  - Predictions are done from observations
  - Only local propagation is used
  - A possible discrepancy is found:
    - if an observed variable is predicted once
    - if a non-observed variable is predicted twice
Since models are not evaluated, what is a possible conflict? Set of relations in a MEC containing, at least, one MEM
CBD using possible conflicts

- Identify Possible Conflicts
- Predict Behaviour
- Possible Conflict Confirmation?
- Generate Candidates
- Refine Diagnosis

New Observations

Off-line | On-line
4.3. Extending possible conflicts to diagnose dynamic systems

- Instantaneous (static) vs Differential (dynamic) constraints
- Temporal information as ODE

- Integration approach, for practical reasons
How loops should be interpreted?

- Loops made up of unknowns and instantaneous constraints
  - Sets of instantaneous equations which can not be solved locally
  - Can be detected off-line; and they could be solved (introducing additional observations (Dague et al. 87) or introducing super-components (Chantler et al. 1998))
• Numerical models + differential arcs do not halt local propagation: spirals not loops (Dressler, 96)
Consistency-based diagnosis and possible conflicts for dynamic systems

1. Identify Minimal Evaluable Chains, MECs
2. Reject MECs without Minimal Evaluable Models, MEMs
3. Select one MEM for each CEM
4. Build SDpc
5. Repeat
   a) Estimate behaviour from observations
   b) Compare estimations against observations
   c) Calculate candidates if discrepancy found

until PC is confirmed

5.a) and 5.b) must deal with lack of precision in model parameters and it must work in noisy environments.
• Diagnosis of dynamic systems
  • noisy environment,
  • sliding windows,
  • semi-closed loop

![Diagram of diagnosis of dynamic systems]

- Real or Simulated Plant
- Filter
- PC$_i$ model
- Filter
- Output data
- Error $< \delta_{pci}$
- Predicted data
- Noise
Measurements and PC predictions. Medium size leakage in TR2 at t=120°.
4.4. Possible conflicts vs conflicts

- Sets of MEC and MEM are computed off-line

- No model evaluation / simulation off-line

- Possible conflict: 
  *Set of constraints in a MEC giving rise to, at least, one MEM*

- Is the set of possible conflicts equivalent to the set of minimal conflicts?
Possible conflicts are:
\{m_1, a_1, m_2\},
\{m_2, a_2, m_3\},
\{m_1, a_1, a_2, m_3\}
While computing possible conflicts:

- every over-determined system, EC, is detected
- for each MEC every way of solving is detected

**Prop. 2**

for each minimal conflict found by GDE, there will be a MEC

**Minimal conflicts:**

\{M_1, A_1, M_2\},
\{M_2, A_2, M_3\},
\{M_1, A_1, A_2, M_3\}

**MECs:**

\{m_1, a_1, m_2\},
\{m_2, a_2, m_3\},
\{m_1, a_1, a_2, m_3\}
Prop. 3
for each minimal conflict found by GDE, one MEM will provide the same discrepancy

Minimal conflict:  
\{M_1, A_1, M_2\}

Minimal conflict:  
\{M_2, A_2, M_3\}

Minimal conflict:  
\{M_1, A_1, A_2, M_3\}
Prop. 4
a discrepancy found using a MEM will lead to a conflict

In theory
the set of possible conflicts provides same results as of the set of minimal conflicts

In practice
- The number of MEM for each MEC grows exponentially in the number of interpretations
- We just select one MEM for each MEC
- Suboptimal results
Prop. 4 always holds; the converse does not always hold.

Equivalence assumption:
Every MEM in a MEC provides the same set of solutions for any given set of input observations.

Prop. 5
if the equivalence assumption holds, in practice the set of possible conflicts is equivalent to the set of minimal conflicts
4.5. Possible conflicts vs ARRs

- Compilation approach from the FDI community

- Analytical redundancy relations, ARRs

- The structural analysis approach (Staroswiecki and Declerk, 89) to compute ARRs:
  - canonical decomposition
  - over-determined systems = just-determined systems + redundant relations

- Each ARR can be used for diagnosis purposes
Similarities between PCs and ARRs
- exhaustive search for over-determined sub-systems
- interpretations in the and-or graph / causal assignment
- only local propagation is used, loops can be analyzed off-line
Differences
- the equivalence assumption is implicitly used
- pcs approach look for strictly (minimal) over-determined systems
- the way they are used in the diagnosis process

Summary
- pcs are theoretically equivalent to minimal conflicts
- in practice, pcs are equivalent to ARRs, suboptimal results with conflicts
4.6. Conclusions

Within BRIDGE framework

- Cordier et al. have proposed a framework to compare DX and FDI approaches within MBD
  - Potential R-conflicts associated to relations in ARRs are equivalent to minimal R-conflicts

- From a computational point of view:
  - the equivalence assumption must hold,
  - set of ARRs must be computed with minimality criteria,
  - same minimality criterion: minimality w.r.t. constraints when models have more than one constraint
5. A case study
Its scheme

FT-01
Flujo Entrada
TR-1
Línea 10

FT-04

Línea 9
T2
Línea 12
Flujo Salida FT-02

Línea 8

LT-05
P3
Flujo Salida FT-03

TR-2
Línea 14
Línea 7

Escuela de Diagnosis, Barcelona, 2004
PC minimal w.r.t. constraints, minimal w.r.t. components
PC minimal w.r.t. constraints, minimal w.r.t. components
PC minimal w.r.t. constraints, not minimal w.r.t. components
MEC with no MEM
References

- Cordier et al., 2000
- Dague et al. 1987
- Darwiche & Provan, 1997
- Dressler et al., 1994
- Dressler & Struss, 1996
- Dvorak & Kuipers, 1990
- Guckenbiehl & Shaffer-Richter, 1990
- Loiez & Taillibert, 1997
- Lunze & Schiller, 1992
- Mosterman, 1997
- Mosterman & Biswas, 1997
References

- Pulido & Alonso, 1999; 2000
- Pulido et al., 2001
- Staroswiecki & Declerq, 1989
- Staroswiecki et al., 1997
- Travé-Massuyès & Milne, 1997
- Struss, 1997
- Washio et al., 1997
- Williams & Millar, 1998