Alarm Processing with Model-Based Diagnosis of Discrete Event Systems

Andreas Bauer – Adi Botea – **Alban Grastien**
P@trik Haslum – Jussi Rintanen

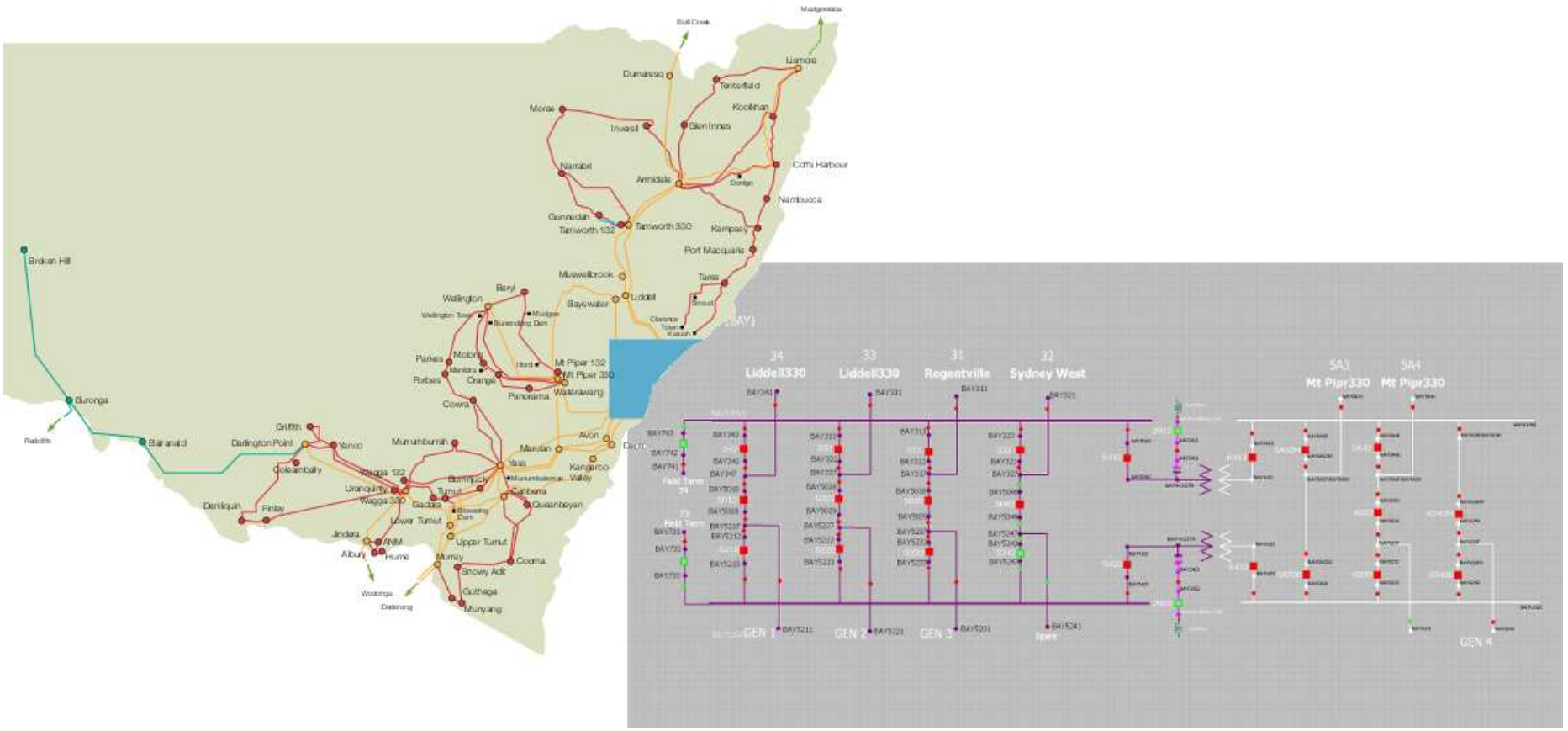
Australian Government
Department of Communications, Information Technology and the Arts
Australian Research Council

NICTA Members
ANU The Australian National University
UNSW
NSW First for Business
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. Alarm Processing of Electricity Networks
2. Related Works
3. Model-Based Alarm Processing
4. Experiments

Example: the TransGrid Network



Example (cont.): the Alarm Log

Extract from Incident July 2nd 2009

```
2/07/2009 10:47:27 BAYSWTR PS 023 NO4 GEN UNIT STATUS OFF
2/07/2009 10:47:27 BAYSWTR PS 023 NO4 GEN UNIT STATUS OFF
2/07/2009 10:47:27 BAYSWTR330 330 SYD WEST 322 CB --OPENED--
2/07/2009 10:47:27 BAYSWTR330 330 NO4 BY/CUP 5042 CB --OPENED--
2/07/2009 10:47:27 BAYSWTR330 330 NO4 GEN TX 5242 CB --OPENED--
2/07/2009 10:47:27 BAYSWTR330 CONTROL SYSTEM LAN FAULT ALARM
2/07/2009 10:47:27 BAYSWTR PS 023 NO4 GEN 2242 CB --OPENED--
2/07/2009 10:47:28 LIDDELL330 330 BAYSWTR330 332 CB --OPENED--
2/07/2009 10:47:28 LIDDELL330 330 BAYSWTR330 342 CB --OPENED--
2/07/2009 10:47:28 LIDDELL330 330 NO2 BY/CUP 5022 CB --OPENED--
2/07/2009 10:47:28 LIDDELL330 330 NO3 BY/CUP 5032 CB --OPENED--
2/07/2009 10:47:28 WANG330 FAULT RECORDER OPERATED ALARM
2/07/2009 10:47:28 BAYSWTR330 330 MAIN BUS BAR KV Limit 5 Low
2/07/2009 10:47:28 BAYSWTR330 330 GEN BUS BAR KV Limit 5 Low
2/07/2009 10:47:28 WANG330 BU SUBSTATION MISC EQUIPMENT FAIL ALARM
2/07/2009 10:47:28 SYD WEST 330 BAYSWTR330 322B B CB --OPENED--
2/07/2009 10:47:28 SYD WEST 330 BAYSWTR330 322A A CB --OPENED--
2/07/2009 10:47:28 MT PIPR330 330 FAULT RECORDER OPERATED ALARM
2/07/2009 10:47:28 ERARING500 SUBSTATION MISC EQUIP FAIL ALARM
2/07/2009 10:47:28 MT PIPR330 500 B BUS BAR KV Limit 3 Low
2/07/2009 10:47:28 BAYSWTR330 330 NO3 BY/CUP 5032 CB --OPENED--
2/07/2009 10:47:28 BAYSWTR330 330 NO3 GEN TX 5232 CB --OPENED--
2/07/2009 10:47:28 BAYSWTR330 330 REGENTVILE 312 CB --OPENED--
2/07/2009 10:47:28 BAYSWTR PS 023 NO3 GEN 2232 CB --OPENED--
2/07/2009 10:47:28 BAYSWTR PS 023 NO3 GEN MW1 Entered zero zone
2/07/2009 10:47:28 BAYSWTR PS 023 NO1 GEN RUNBACK URGNT ALARM
...
```

Problem Definition

What is Alarm Filtering

Organise the flow of alarms in order to

- Stress important alarms and hide redundant ones
- Show on-going / finished incidents

What is Alarm Filtering Not

Diagnosis

- The operator wants the alarms, not a diagnosis
- The model is not accurate enough
  (or an accurate model would be too hard to reason on)

Outline

1. Alarm Processing of Electricity Networks
2. Related Works
3. Model-Based Alarm Processing
4. Experiments

Patterns

Definition

Set of alarms, possibly with time constraint, that are symptomatic of a certain fault

Pattern-Based Filtering

With each pattern, is associated the filtering rules:
when the pattern is recognised, the rules are applied

Issues

- Generation
- Completeness
- Intertwined behaviours

Model-Based Approaches

Model

Causality rules of form:

$$\text{alarm}_1 \wedge \cdots \wedge \text{alarm}_k \rightarrow \text{alarm}'$$

Model-Based Filtering

The causality rules are used to determine the *root cause* alarm(s) and ignore the other alarms

Issues

- Alarms are *symptoms*, not root causes.
- Context-based causality

Outline

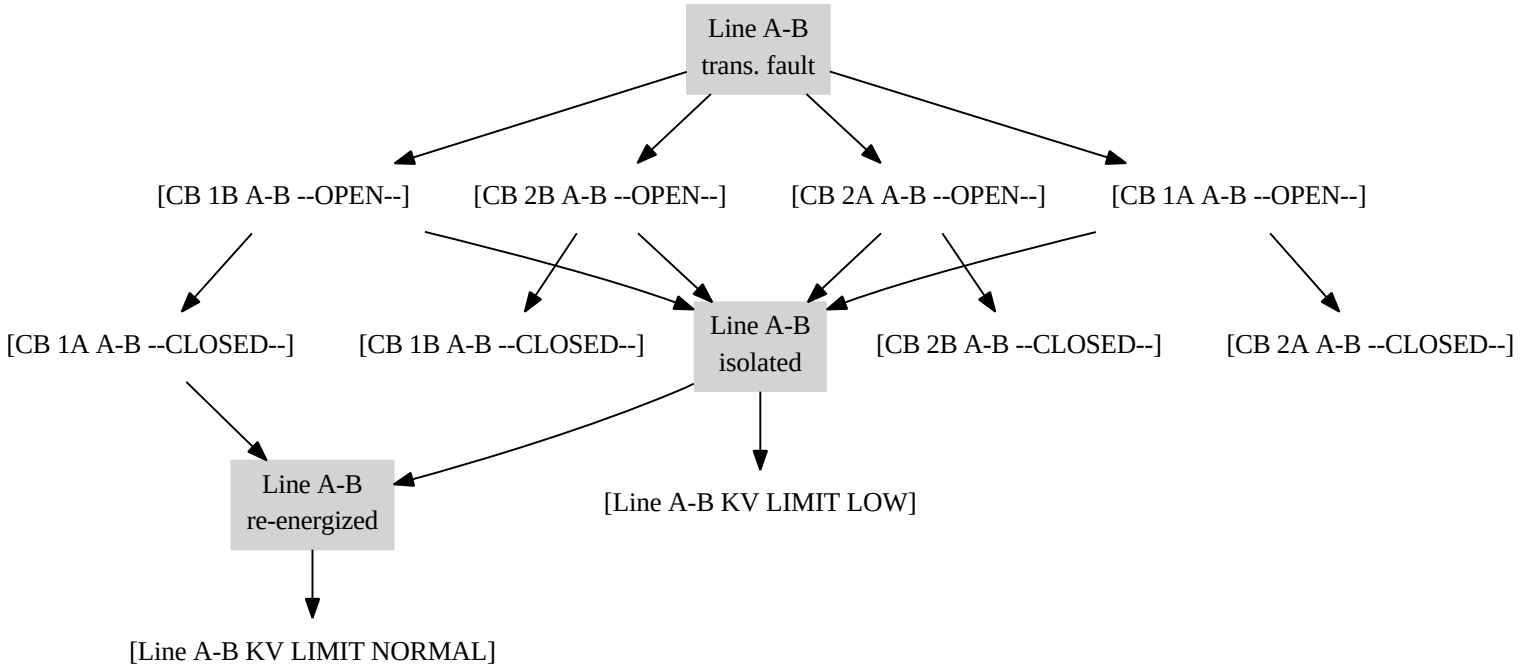
1. Alarm Processing of Electricity Networks
2. Related Works
3. Model-Based Alarm Processing
4. Experiments

Principle

Model-Based Filtering

- Build a causal model of the system, including internal (unobservable) events
- Perform a diagnosis to "explain" the alarms
- Use the diagnosis to determine which and how alarms are related

Example



Representation of an explanatory trajectory

Framed events = unobservable events

Arrows represent causality dependency

How to Compute a Trajectory

Objectives

- Correctness:
  - it is ok not to link related alarms
  - it is *not* ok to link unrelated alarms
- Fast response (rule of thumb: a dozen seconds)

Model

- Timed discrete event systems
- Weak fault model
- "Unexplained event" = a (weakly modeled) fault or an alarm

How to Compute a Trajectory (cont.)

"Diagnoser"

- Searches the trajectory that minimises the number of unexplained events
- Implemented in SAT

Filtering Techniques

Clustering

$\rightarrow$ regroup the events that are logically related

Filtering Techniques

Root cause

$\rightarrow$ select the unexplained event(s) in each cluster

Filtering Techniques

Live Alarms

A set of alarms is live if the situation described by these alarms has not been resolved

$\rightarrow$ test whether the state at the end of the cluster is nominal

Outline

1. Alarm Processing of Electricity Networks
2. Related Works
3. Model-Based Alarm Processing
4. Experiments

Experimental Data (1/2)

Observations

- Incident of July 2nd, 2009
- $2,246$ alarms (731 left)
- Sliced into one-minute diagnosis windows
  and uninterrupted diagnosis windows ($\rightarrow$ 129 problems)

Experimental Data (2/2)

System

- 5,000 components

  but we compute the cone of influence (2 to 104 components)
- Timed automata

Experiments

Diagnoser

- SAT solver using 6 unobservable transitions between two observations
- Permissive (?) implementation of time constraints
- Searches for scenarios with 0, 1, 2, etc., unexplained events

Results

Runtime (for finding the best explanation)

- Only 16 problems not solved on time
- . . . but they are the problems where the filtering is useful

Possible Improvements

- Identify independent subsystems
- Change the model
- Compute any explanation

Conclusion

Summary

- Filtering using model-based diagnosis
- Provides useful information