Information Society Technologies

ARTIST2 – Year 1 Review

Grenoble, October 3rd-4th, 2005

Activity

Diagnosis in Distributed Hard Real-time Systems

Activity leader: H. Kopetz, P. Peti (TUVI)

with an introduction by Albert Benveniste (INRIA)

Introduction by the cluster leader for HRT :

Albert Benveniste (INRIA)

Diagnosis

- **1st meeting at TU Vienna, Vienna 20—21 Dec, 2004**
  - 13 participants
  - Industrials: 1 TTTech
  - 3 affiliates
  - 1 from other clusters (Control)
- **2nd meeting at Verimag, Grenoble 2—3 may, 2005**
  - 11 participants

Diagnosis

Executive summary of Vienna meeting: sample

- **Key objectives and difficulties related to diagnosis in automotive industry**
  - Wanted: accurate diagnosis of critical functions to the point where faulty component for replacement can be traced back (maintenance). For less critical functions, it is still important to achieve a certain degree of on-line diagnosis because data cannot be massively collected for subsequent garage exploitation.
  - Fault isolation, localisation, and root cause analysis, preferably *on-line*, is essential in automotive industry. This is particularly challenging if the fault crosses several subsystems or ECUs.
  - In general, what can still be done if the TT assumption fails? What disappears?
  - Physical redundancy must be limited, for cost reasons.
  - Components and IPs are an unavoidable way to go: what is the impact for diagnosis?

Diagnosis

Executive summary of Vienna meeting: sample

- **Can Discrete Event Diagnosis (timed or not) techniques be useful for computer platform diagnosis?**
  - Can Philipp Peti check whether some assertions related to computer faults may have to involve not just snapshots of values but also states at different instants (i.e., involving dynamical aspects of the system). If the answer is yes, then it is likely that DES diagnosis techniques can be of interest.

- **What (statistical or otherwise) techniques used in control can be useful to deal with transient & intermittent failures?**
  - The statistical methods for on-line fault detection in control were noise oriented. Noise does not occur as such in computer hardware; nevertheless other aspects of randomness occur that may make similar techniques useful. On-line threshold based techniques could be a topic where such a X-fertilization is useful. Andrea Bondavalli & Neeraj Suri will investigate protocols and analytical options and forward the related info to Miroslaw Malek, Albert Benveniste and Qinghua Zhang, for checking possible alternative approaches.

Diagnosis

Executive summary of Grenoble meeting: sample

- **Automata, finite state machines, and related logics**
  - Given a model of the symptom for monitoring, on-line check the possible occurrence of the symptom. The model of the symptom can be, either provided by the designer based on his expertise, or be generated automatically from the system model.
  - Given a model of the plant, plus a model of the symptom, we can check if the symptom is monitorable, i.e., if its detection is unambiguous.
  - Given a model of the plant, plus a collection of faults for diagnosis, we can check diagnosability, i.,e., the ability to separate unambiguously the different cases. This case is a particular case of the former one.
  - To generate so-called *-monitors or diagnosers*, perform the following:

    perform the product of plant model and symptom model

    determinize the result
  - This yields a deterministic automaton triggered by the observations, which returns all possible failures that comply with the past and present observations.
- **Same, but timed**
  - Same, but many steps suffer from undecidability and therefore cannot be automatized in general. See presentation 9 by Stavros Tripakis.
- **Same, but enhanced with non finitary attributes**
  - Subject to the same problems as the timed models.

Diagnosis

Executive summary of Grenoble meeting: sample

❖ Statistics

- When a single fault is monitored, based on single or multiple symptoms, then the so-called Page-Hinkley stopping rule from statistics works well. It is quite general and systematic, and is based on likelihood ratio techniques. The resulting algorithms are simple to implement and robust to imperfect knowledge of the model parameters.
- These algorithms belong to the family of so-called *alpha-count* or *threshold based* procedures found in the literature on dependability (e.g., Bondavalli et al. 2000).
- The base case assume that randomness occurs in an independent manner (noises or disturbances have no spatial/temporal memory). However, it is also possible to handle models with states (Markov chains, semi-Markov chains or HMMs), as long as likelihoods of trajectories can be computed.
- Regarding the monitoring of symptoms for their change in distribution, both *time-triggered* and *event-triggered* versions exist. In the former, the symptom is Boolean and is processed every time it is received. For the second case, symptoms are just events, and intervals between events are monitored.
- *Question:* what if, in the TT case, the symptom is expected but not received? How should we regard this? Should we ignore and skip the instant? Or should we take it as the occurrence of a symptom? Or is it a symptom for another analysis? To be checked.

❖ Possible combination of these

- In a first approach, you design symptoms separately, by using one of the above mentioned approaches; then the symptom is submitted to a statistical analysis.
- In a second approach, probabilities would enter the modeling from scratch. For examples, the diagnosis model could mix nondeterministic branchings with probabilistic ones. This gives raise to Markov Decision Processes (MDP) in use for stochastic control, or, equivalently, to probabilistic automata. But we used them here for diagnosis purposes. Developing systematic algorithms for on-line diagnosis, based, e.g., on a generalization of the likelihood approach, is still open.

Presentation by the activity leader :

Activity leader: H. Kopetz, P. Peti (TUVI)

Outline of the Presentation

Industrial Needs and Experience

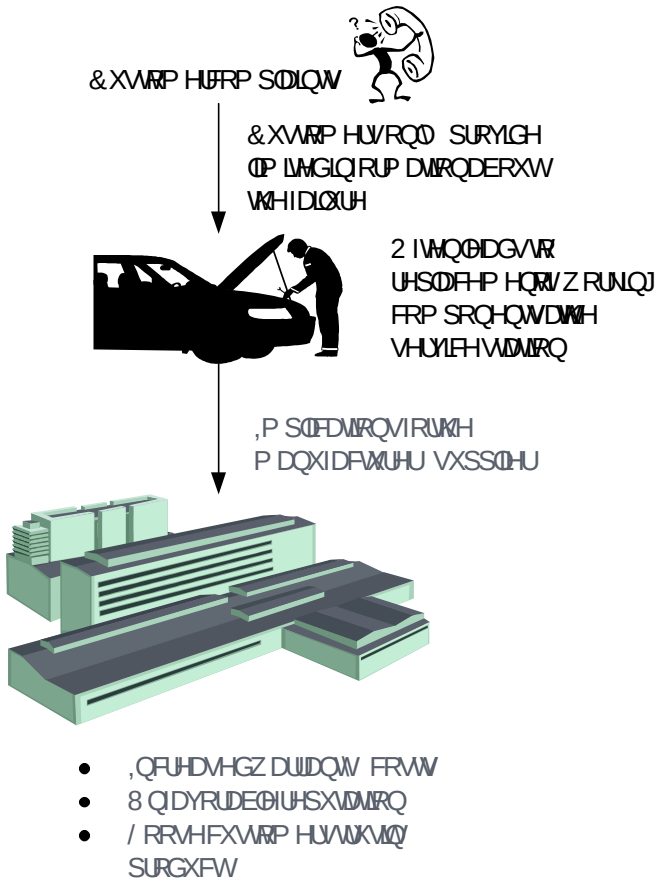
Year 1 Activities

- Achievements & Ongoing Work
- Interaction and Building Excellence Between Partners
- Diagnosis combines methods from
  - Architecture design
  - Contract-based design
  - Formal methods
  - Statistics

"Seems the onboard computer analyzed your driving patterns and determined it was best for all not to start the engine."

Industrial Needs and Experience

- **ARTIST2 Interaction with Industry**
  - Automotive Industry: Audi, Fiat (CRF), TTTech
  - Avionic Industry: Airbus
- **Industrial Needs**
  - Effective diagnostic systems stay behind recent complexity increase of electronic systems
  - Statistics: the number one breakdown cause for cars are electronic problems (negative media coverage)
  - Emerging X-by-wire solutions require new maintenance strategies
  - In automotive embedded controllers, software for diagnosis accounts for around 50% of the entire application and for more than 50% of the validation effort
- **Possible Global Impacts of Research Results**
  - Improvement of Accuracy of Diagnosis
  - Decrease Warranty Costs



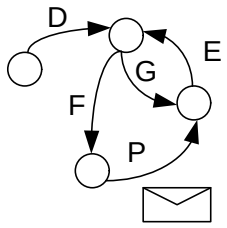
Year 1 activities

Achievements

Logic of Constraints
assume X and Y
assert Z

Random external hardware faults

$ XWRP DWD 7 KHRU

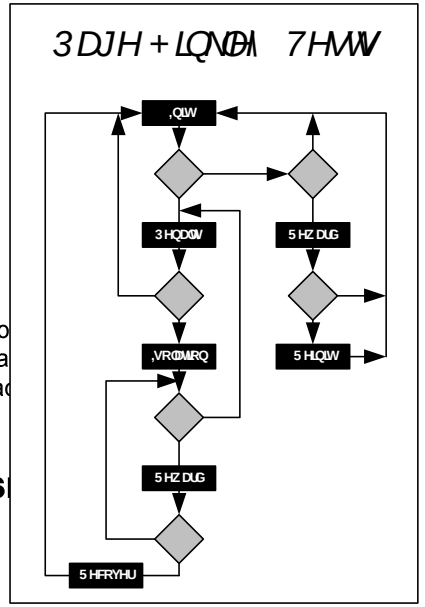


Transport of diagnostic data

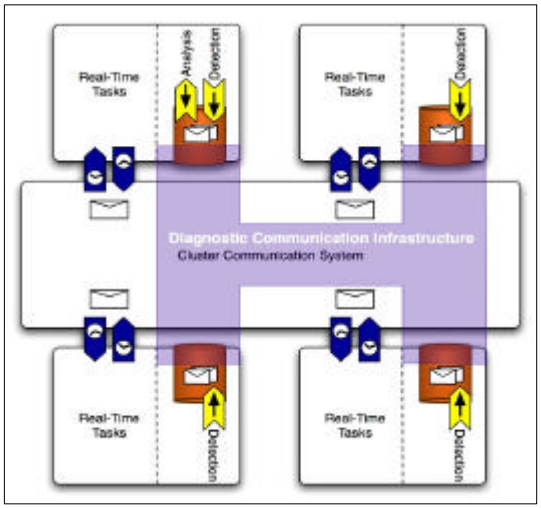
Experienced faults are classfied according to a **fault model suitable for maintenance**

Analysis

- Component conditio
- Advanced maintena
- Engineering feedbac
- Offline analysis



DETECTION **INFORMATION TRANSPORT** **ANALYSI**

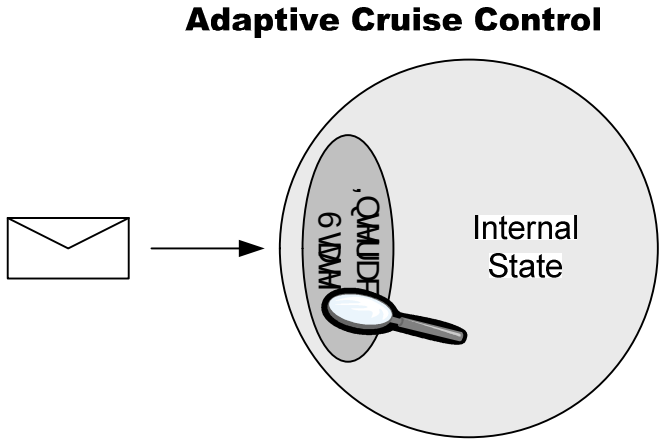


Logic of Constraints (LOC)

- **Detect any illegal behavior, identify the faulty component and provide sufficient information to react (recovery) at run-time.**
- **Illegal behavior might come from plant, platform, and application component faults**
- **Solution:**
  - Decompose the diagnosis problem into a set of assumption/assertion to be validated by each component
  - Capture the assumptions/assertions using the Logic of Constraints (LOC) formal language
  - Synthesize run-time checkers
  - Run-time checking of assumptions/assertions
- **Design-by-Contract**
- **Assumptions/assertions unambiguously specify non-functional and/or functional properties**

Logic of Constraints (2)

- **An assertion violation of a component provides detection of a failure**
- **Correct assumptions identifies the component:**
  - as faulty if assumptions are exhaustive
  - as possibly faulty otherwise
- **The run-time assumption/assertion checking must be performed in bounded memory**
  - LOC expressiveness must be reduced
- **1st Year activities:**
  - LOC expressiveness: comparison with other modal logics (LTL) and with (Timed) Automata
  - Definition of a methodology based on the proposed solution supporting the AUTOSAR framework




```
assume FSM.State[i].v == ACCon
and abs(ACC.TorqueRequest[i].v) < 20
assert abs(Engine.Torque[i].v) < 20;
```

Diagnosis for Real-Time Systems

- **Given model of real-time system with faults**
  - System model = timed automaton
- **Synthesize automatically a monitor to detect faults**
- **Different types of monitors**
  - Analog-clock: precise but difficult to implement
  - Digital-clock: implementable but conservative (may not detect all faults)
- **Results:**
  - A theory of real-time monitoring with partial observability
  - Plug-ins to the Verimag IF tool-set (model-checking, test generation)
  - Extensions to related research directions (“hot” topics today)

    Real-time testing

    Implementability of “timed objects” (monitors, testers, controllers, …)

Diagnosis for Distributed Systems

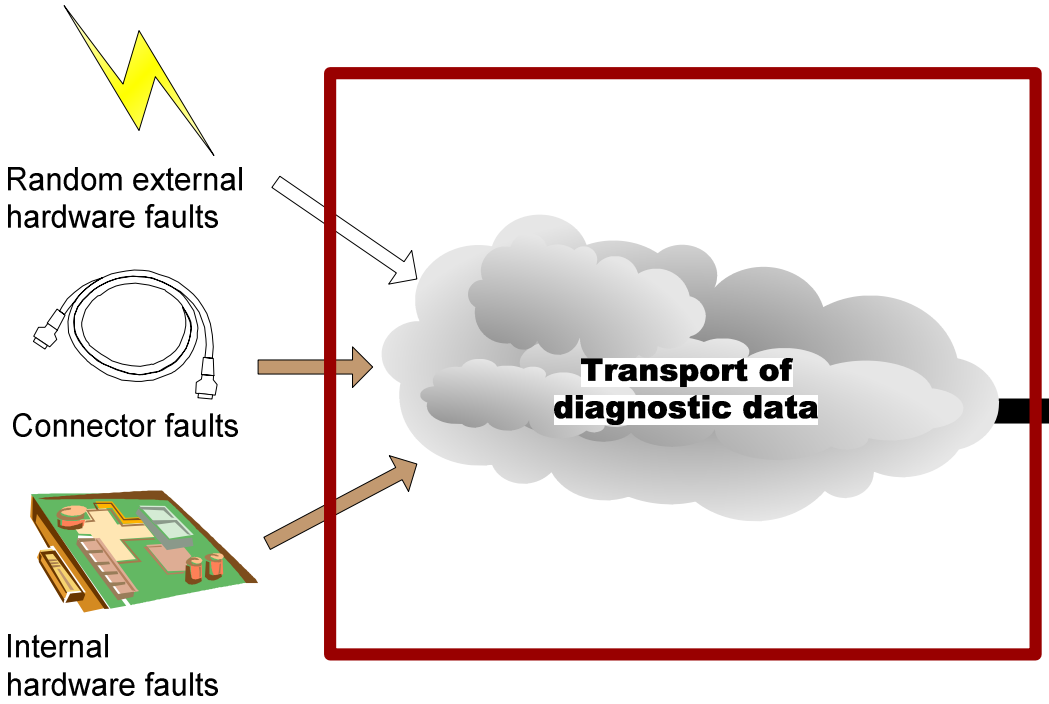
- **Given model of distributed system and property to observe with different types of settings:**
  - With or without communication, communication delays
  - Amount of memory allowed for the monitors, etc
- **Synthesize automatically a set of decentralized monitors**
- **Results:**
  - Basic properties of centralized diagnosis break down:

    Existence of monitors does not imply existence of finite-state monitors

    Checking existence is often undecidable, even in the simplest settings (finite behaviors, regular languages)
  - Recent effort: identify decidable sub-classes
  - Extensions to related research directions

    Decentralized control and games

Overview

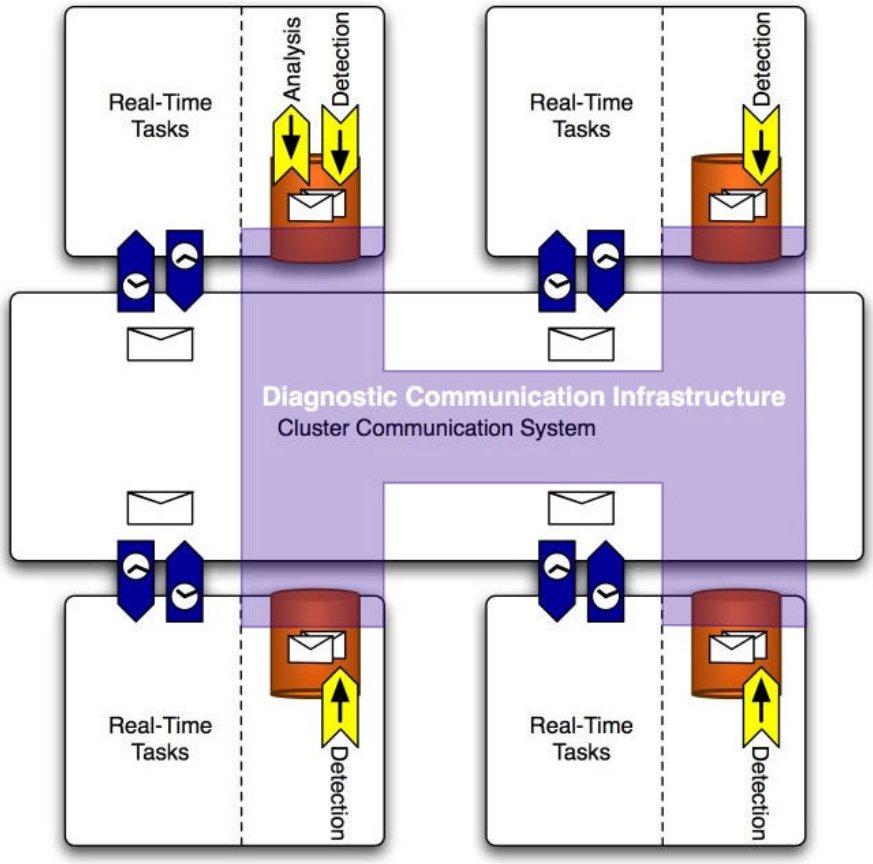


- Component condition assessment
- Advanced maintenance strategies
- Engineering feedback
- Offline analysis

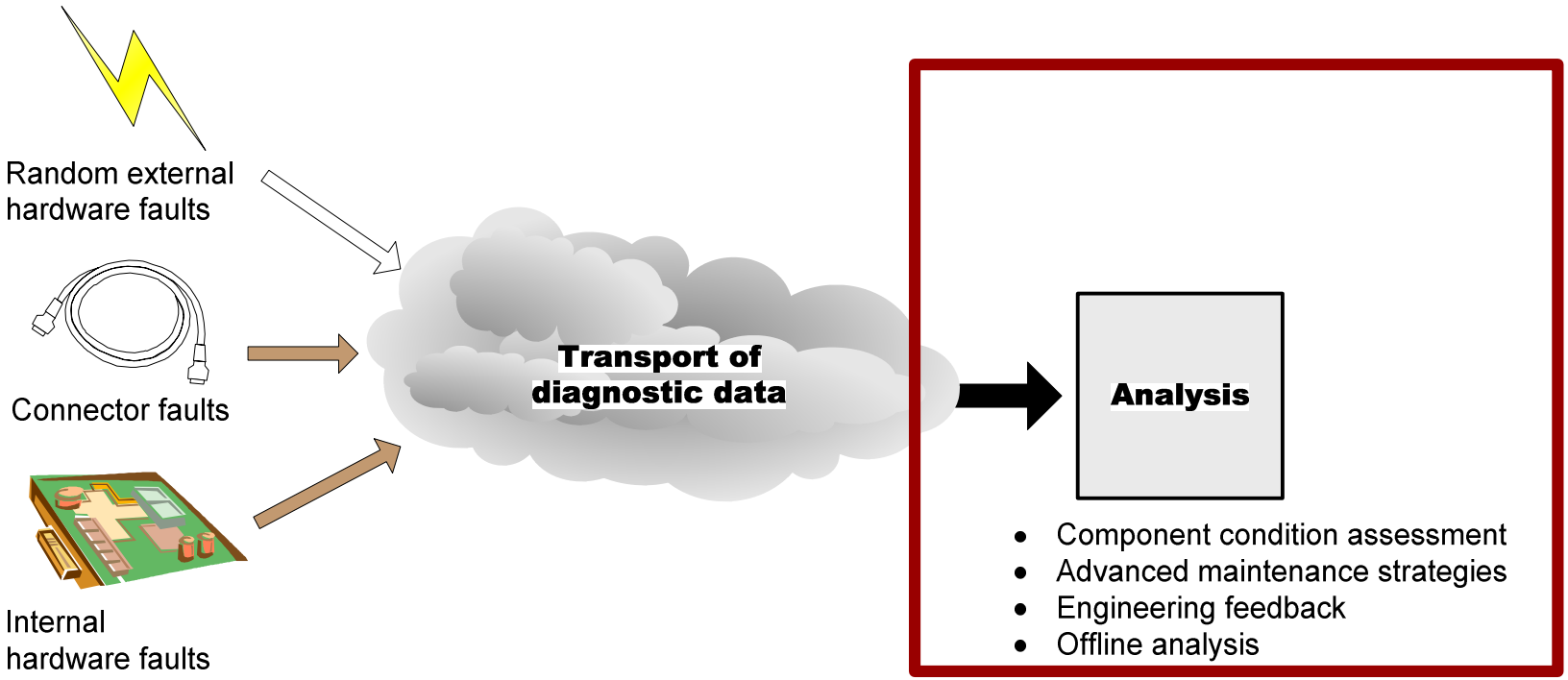
DETECTION **INFORMATION TRANSPORT** **ANALYSIS**

Diagnostic Communication Infrastructure

- A dedicated communication infrastructure for diagnosis
- Encapsulation on network level
- Part of bandwidth statically reserved for diagnosis
- Purely virtual as an overlay network (e.g. TTP, static FlexRay)
- Use of less reliable physical but dynamic communication (e.g. dynamic FlexRay)
- Requires encapsulation at component level (i.e. temporal and spatial partitioning)

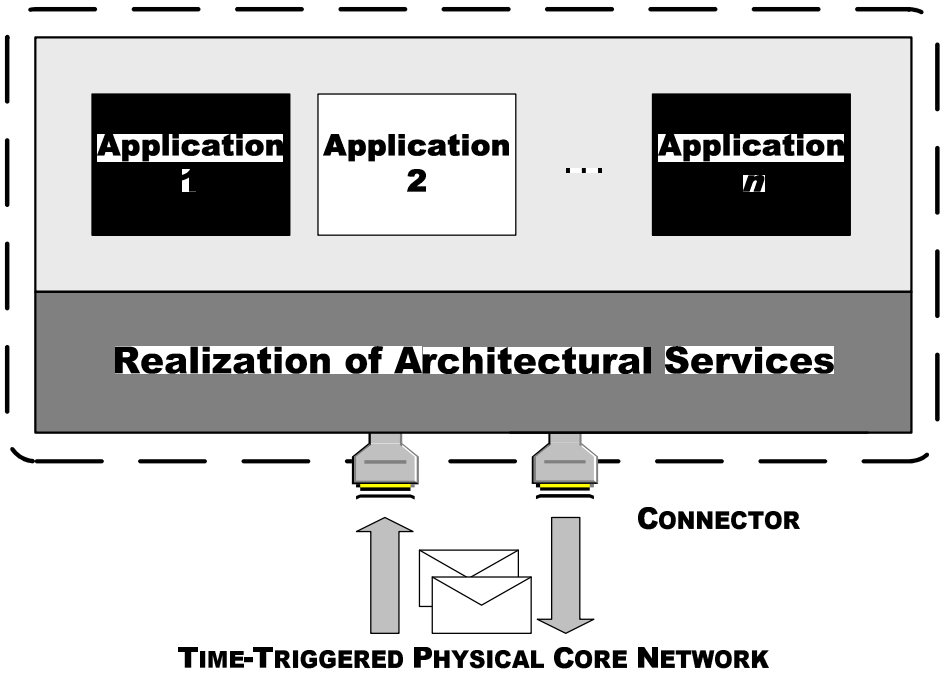


Overview


Random external hardware faults
Connector faults
Internal hardware faults
Transport of diagnostic data
Analysis
Component condition assessment
Advanced maintenance strategies
Engineering feedback
Offline analysis
DETECTION
INFORMATION TRANSPORT
ANALYSIS

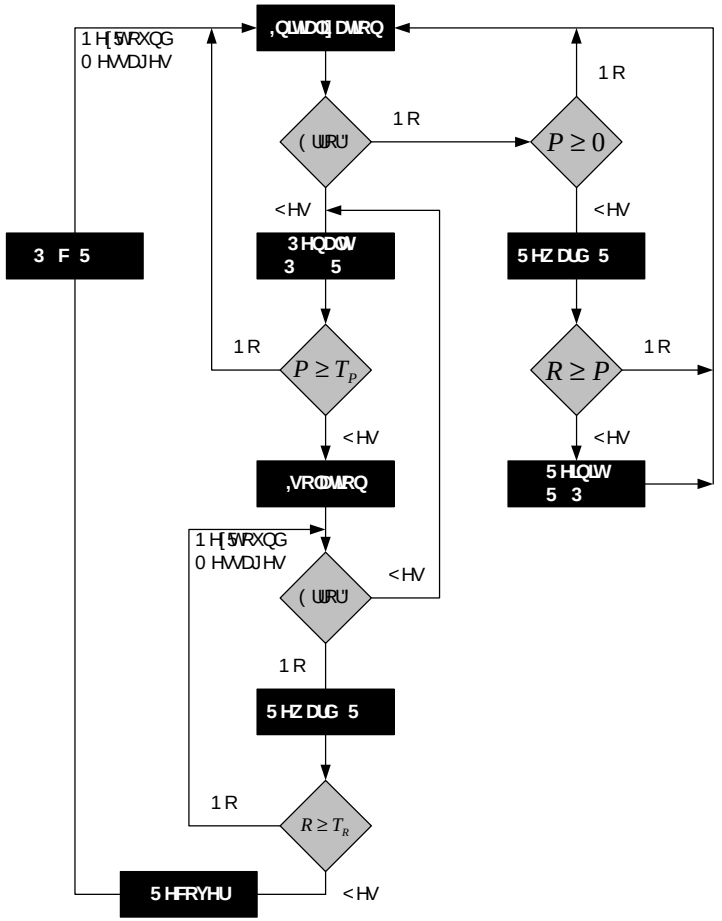
Maintenance-Oriented Fault Model

- **We stop "fault-error-failure" chain at Field Replaceable Unit (FRU) level**
- **Suitable for integrated architectures (e.g. DECOS, IMA, AUTOSAR)**
- **Integrated architectures overcome "1 Function – 1 ECU" limitation**
  - **Hardware faults**: component (ECU) as unit of replacement for hardware faults
  - **Software faults**: software component as unit of update for software faults

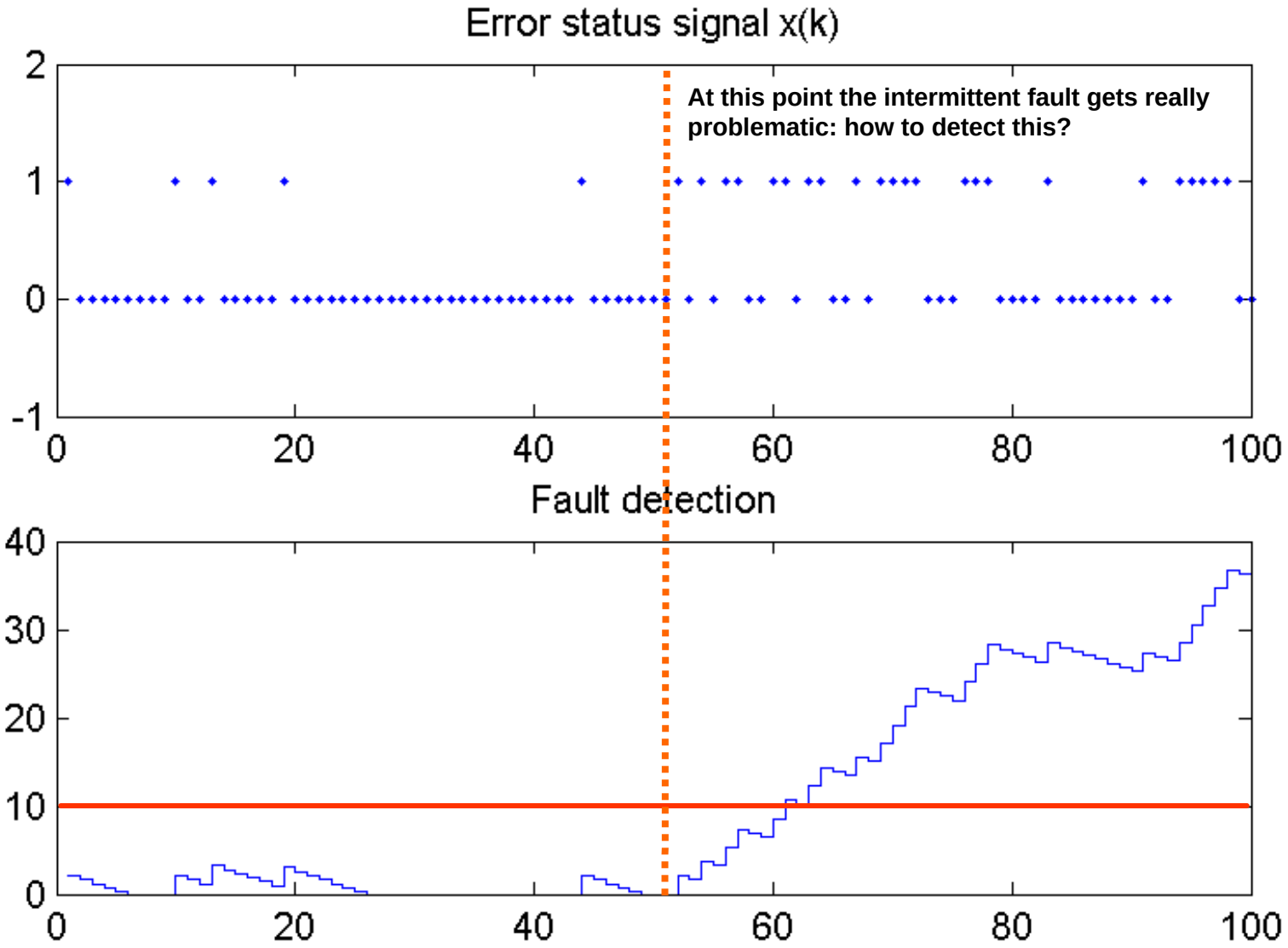


Coping with randomness and uncertainties

- Assertion violations may be intermittent or resulting from “noise effects”; these we like to filter out and keep only violations which occur for a significant amount of time.
- Approach: threshold-based analysis methods
  - Decide on the point in time when a system component should be replaced
  - Keep track of every fault occurrence in each component
  - In case the counter value exceeds a given threshold value, the component is diagnosed as affected by a permanent/intermittent fault
- During and between meeting discussions, it was found that the known threshold-based methods could be cast into the well-known family of statistical *Page-Hinkley Tests* used in quality control for years



Smoothing out intermittent faults



Perspectives

- **Pursue the work on diagnosis for distributed real-time systems based on formal analysis and synthesis methods**
  - This will survive
- **Bring statistical techniques into the former to account for noise effects, and other types of intermittence; link with Hidden Markov Model (HMM) techniques from statistics and pattern analysis**
  - This may not survive, or may be investigated in another life