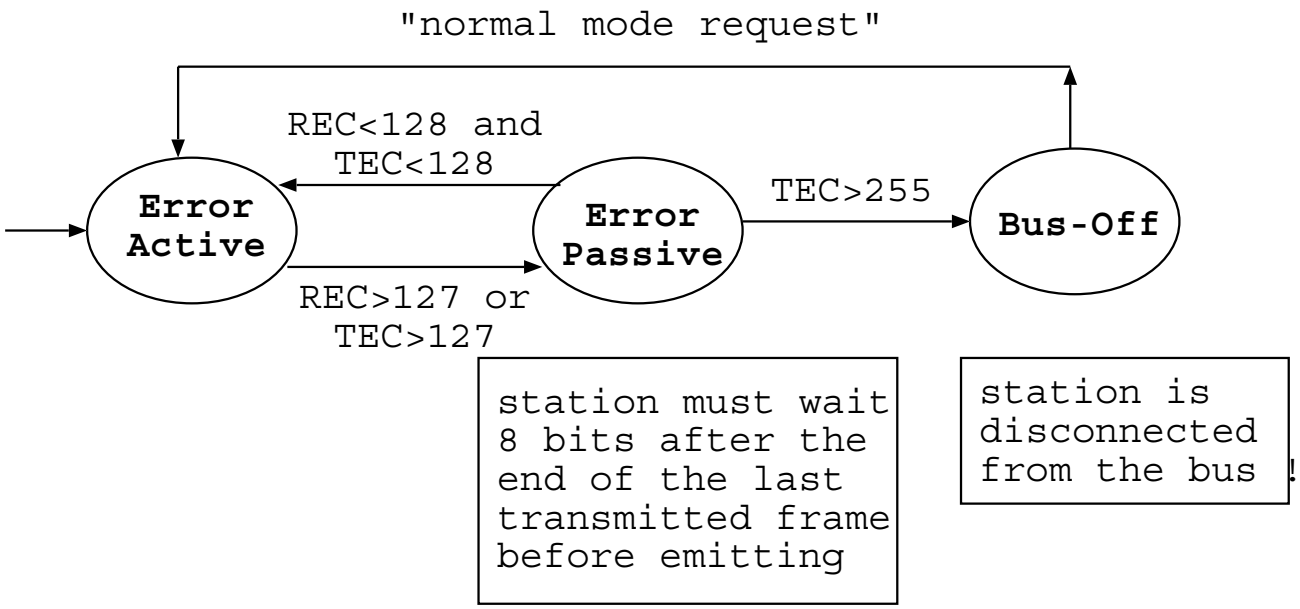
Fault confinement mechanisms on CAN : analysis and improvements

Bruno GAUJAL - Nicolas NAVET
LORIA Lab. - TRIO Team

15 November 2001

CAN fault confinement mechanisms

- **Goal :** prevent defective nodes from perturbing the whole network
- **Functioning scheme :** state of the node decided by two counters



☞ **under high EMI, a correct node can reach bus-off/error-passive because of transmission errors . . .**

Objectives of the study

1. **estimate the probability of being bus-off and error-passive :**
   - bus-off : necessity to design mode changes ?
   - error-passive : worst-case time analysis not respected ?

2. **understand the design choices made for CAN**

3. **propose new mechanisms with clearly-stated hypotheses**

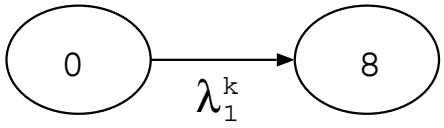
Assumptions :

- all transmission errrors are correctly detected
- state changes are exponentially distributed
- 3 exceptions to the general rules for increasing/decreasing error counters are not considered

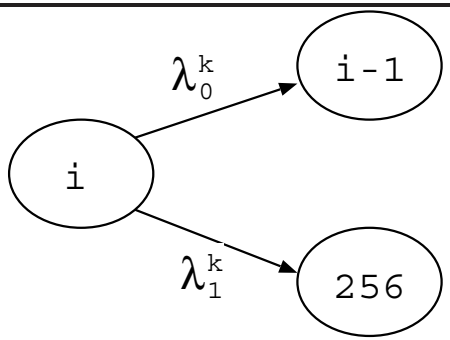
Bus-Off hitting time

- **Modeling :**

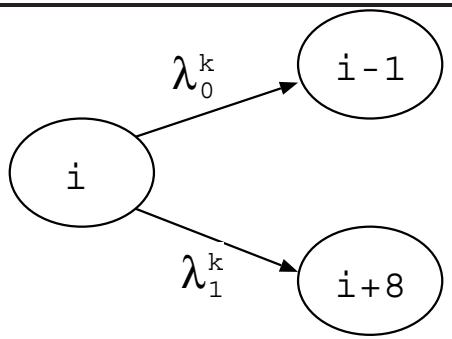
$\mathrm{TEC} = 0$:



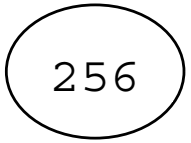
$\mathrm{TEC} \in \{249..255\}$:



$\mathrm{TEC} \in \{1..248\}$:



$\mathrm{TEC} = 256$:



- Markov Process $\overset{\text{uniformization}}{\Longrightarrow}$ Markov Chain :

$$P = \begin{bmatrix} \mathcal{Z} & \mathcal{R} \\ 0 & 1 \end{bmatrix}$$

Bus-Off hitting time

Classical "one-step" analysis to obtain :

- the mean hitting time of the bus-off state (starting from state i) :

$$N_i = \begin{cases} \gamma_i + N_j, & \text{with probability } \sum_{j \in \mathcal{T}} P_{i,j}, \\ \gamma_i, & \text{with probability } P_{i,256} \end{cases}$$

with $\gamma_i = 1$ if $i \neq 256$ and 0 otherwise.

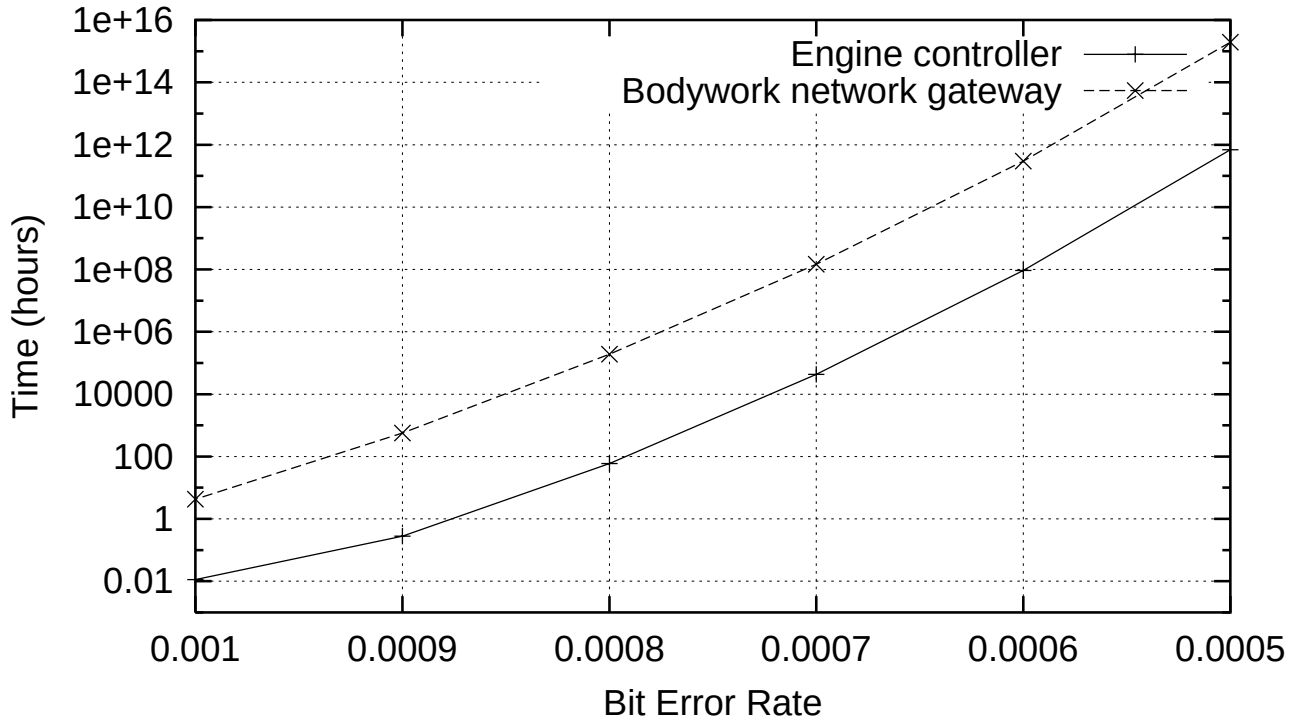
$$E[N_i] = \gamma_i + \sum_{j \in \mathcal{T}} P_{i,j} E[N_j]$$

- the variance of the bus-off hitting times :

$$V[N_i] = E[N_i^2] - E[N_i]^2 \text{ with } E[N_i^2] = \gamma_i + \sum_{j \in \mathcal{T}} P_{i,j} E[N_j^2] + 2 \sum_{j \in \mathcal{T}} P_{i,j} E[N_j] \gamma_i$$

Bus-Off hitting time : application to the PSA benchmark

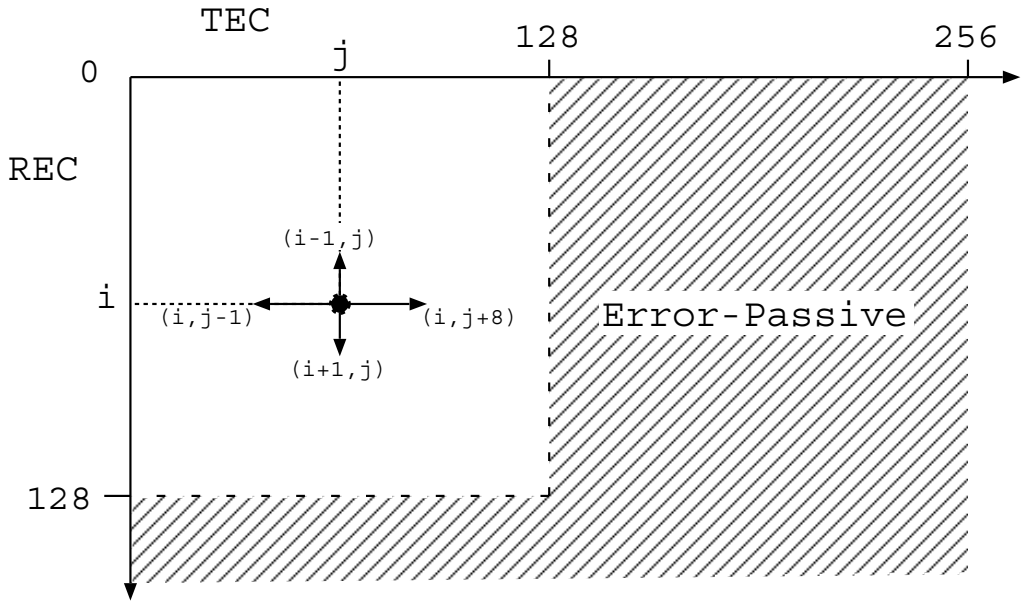
6 devices - 250kbit/s - engine controller load: 7.6% - body. network gateway load : 0.84%



- average hitting times : 40 seconds with BER= 0.001 - 43360 hours with BER= 0.0007
- standard deviation is of the order of the expected value

Error-Passive hitting time : modeling

- state of the process can be identified by the coordinates $(\mathrm{REC}, \mathrm{TEC})$ - error-Passive is reached when $\mathrm{REC} = 128$ or $\mathrm{TEC} > 127$:

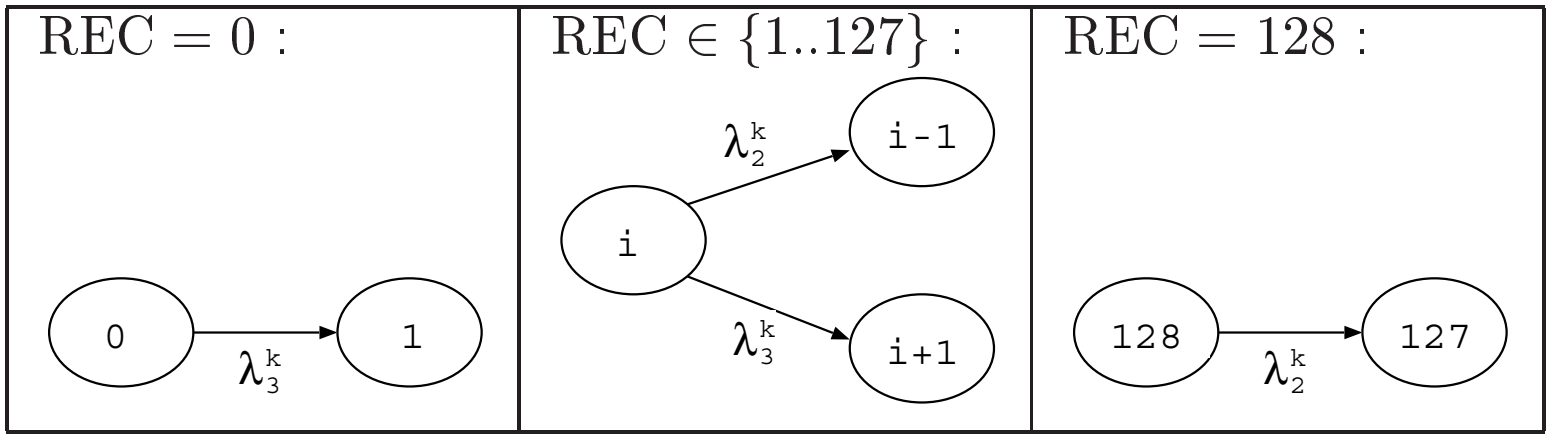


- the probability transition matrix is of size $(257 \cdot 128)^2 \approx 1.09 \cdot 10^9$!

☞ **separate estimate of error-passive due to reception / due to emission**

Error-Passive due to reception

- **Modeling :**



- **Analysis :** computation of the stationary probability vector π

$$\pi = \pi \cdot W \qquad \pi_{128} \text{ is the proportion of time spent in Error-Passive}$$

- **Experiments :** engine controller with BER$= 0.001$,
  - $\pi_{128} = 6.6 \cdot 10^{-131}$
  - expected time between two occurrences is $> 10^{124}$ years !

☞ **The time spent in Error-Passive due to reception is almost nil !**

Error-Passive due to emission

- **Modeling :** same as bus-off
- **Analysis :** Computation of the time spent in a state greater than 127

$$E[M_i] = \gamma_i + \sum P_{i,j} E[M_j],$$

  with $\gamma_i = 1$ if $i \geqslant 128$ and 0 otherwise.

- **Experiments :**
  - engine controller : 26.2% with BER$= 0.001$
  - body. network gateway : 2.7% with BER$= 0.001$

☞ **Time spent in Error-Passive can be very important for high BER !**

Conclusion on existing mechanisms

Performance of the mechanisms

- Bus-off is reached too easily (e.g. 40 seconds with $\mathrm{BER} = 0.001$)
- REC is only useful for stations that do not emit any messages
- Proportion of time spent in error passive can be very important (e.g. 26% with $\mathrm{BER} = 0.001$)

☞ **Application designer has to take account of bus-off and error-passive !**

Design issues :

- errors are assumed to be independent (in practice, they are often bursty!)
- the information given by successive correct transmissions is barely taken into account.

Improved Bus-Off mechanisms

- **Hypotheses :** 2 types of systems
  - "faulty" nodes cannot send correct frames
  - "faulty" nodes may sometimes send correct frames
    q_k is the probability that station k emits a correct frame while being faulty.

- **Transmission errors can be correlated :**
  - p_{k_i} is the probability for a non-faulty nodes k to emit a corrupted frame given that its last $i-1$ were corrupted
  - p_{k_i} can be estimated by

$$p_{k_i} = \frac{P[\text{error burst length on } k \ \geqslant i]}{P[\text{error burst length on } k \geqslant i-1]} \text{ and } p_{k_1} = FER_k/B_k$$

When to decide bus-off ?

The actual problem : detect that a node is faulty only by looking at the correctness of the transmitted frames

☞ **decision can be delayed until the suspected node may jeopardize the real-time behavior of the other stations**

proposed solution :

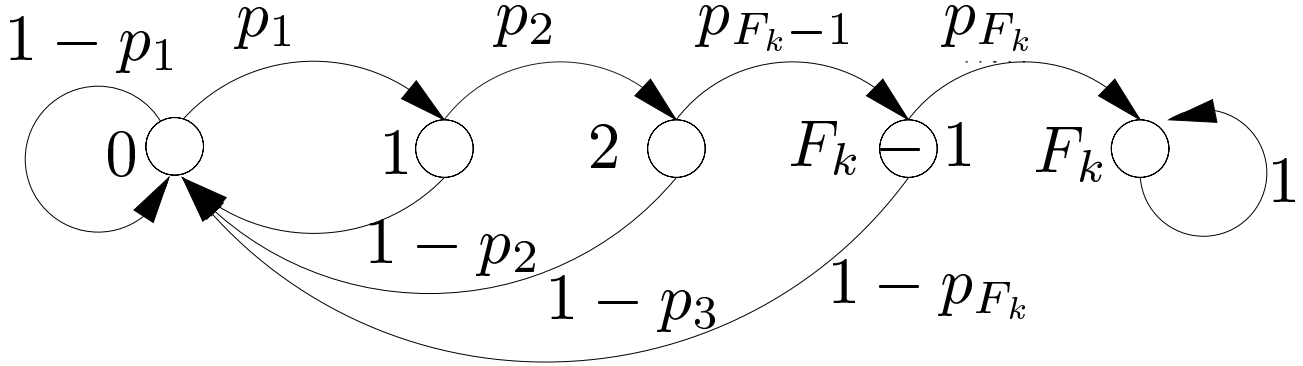
$$F_k = \max\{N_k, \min\{\Phi | \prod_{j=1..\Phi} p_j < \epsilon\}\}$$

where :

- N_k is the maximum number of retransmission of a frame of station k such that the deadlines of all frames of other stations is respected
- $\prod_{j=1..\Phi} p_j$ is the probability to have Φ consecutive corrupted messages while the station is non-faulty.

Faulty nodes cannot send correct frames

Modeling : ($F_k = 4$ here)



Experimentation :

- Much longer bus-off hitting times for a non-faulty node : 221 hours vs 40 seconds for BER$= 0.001$!
- Hitting times less sensitive to BER
- Hitting times very sensitive to F_k ($F_k = 18$ for EC - $F_k = 31$ for BNG)

Faulty nodes can send correct frames (1/2)

Assumption : n consecutive frames while faulty are independent (proba. q^n)

Principle: weight the progression towards bus-off by the quantity of information given by the last transmission

Proposal : state given by two counters (i, j) :

- i is the proximity of bus-off, j is the number of consecutive transmission errors
- on the occurrence of an error $(i, j) \rightarrow (\lceil i/p_{k_j} \rceil, j + 1)$,
- on a successful transmission $(i, j) \rightarrow \lceil i.q_k \rceil, 0)$
- the bus-off state is reached when $i \geqslant 1/\prod_{j=1..F_k} p_{k_j}$.

Faulty nodes can send correct frames (2/2)

Advantages compared to existing mechanisms :

- Errors are not necessarily independent
- Parameters p_k and q_k can be set according to the systems and the environment

CAN current mechanism is a special case :

- Errors are independent ($p_{k_j} = p_k \; \forall j$)
- Taking the log, the chain dynamic is transformed : $/$ replaced by $+$, and $\cdot$ replaced by $-$
- $q_k^8 = p_k$ (for instance $p_k = 10^{-8}$ and $q_k = 10^{-1}$)

☞ Underlying assumption of CAN current mechanisms :

8 correct frames sent by a defective node has the same probability as 1 corrupted frame sent by a correct node

Conclusion and future work

Contribution of the study :

- Analysis of the current CAN fault confinement mechanisms
- Proposition of news mechanisms with clearly-stated hypotheses

future work :

- Validate the analysis of the current mechanisms against simulation with periodic traffic
- New mechanisms for error-passive
- Analysis and comparison of other protocols