Sixth European
Dependable Computing Conference

EDCC-6

Coimbra, Portugal
18-20 October 2006

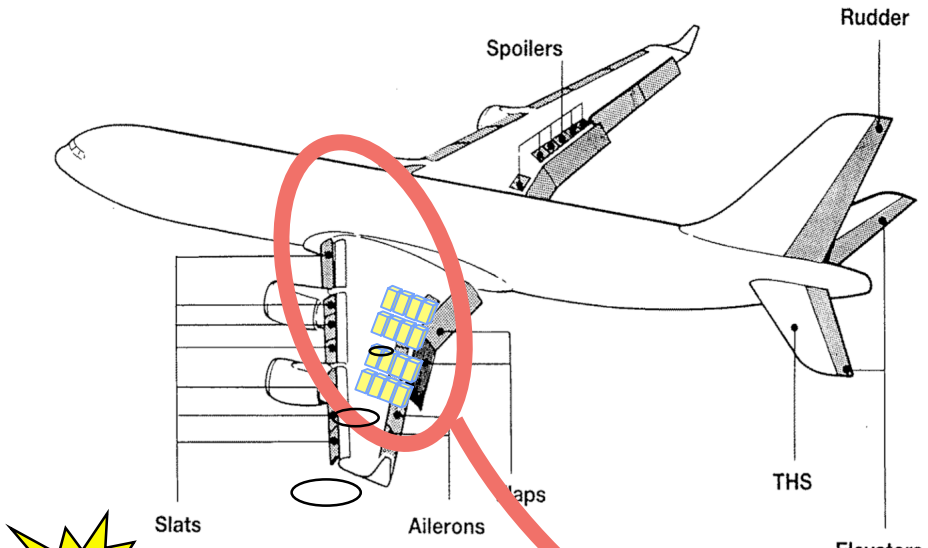
Communication Integrity in Networks for Critical Control Systems

A. Youssef, Y. Crouzet
A. de Bonneval, J. Arlat

J.-J. Aubert, P. Brot

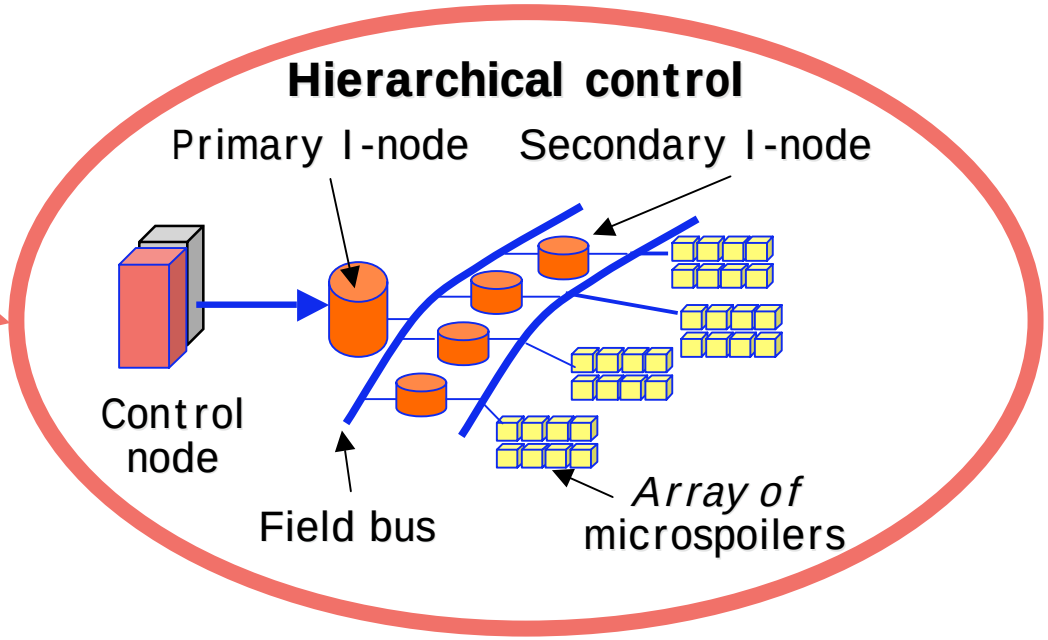
Context and Motivation

- **Usage of fully-digital communication networks into critical embedded systems (commercial aircrafts)**



- **Flexible control**
  - **accommodate distinct commands on different actuators**

    -> all devices cannot be connected to the same bus
  - **Need for intermediate functional nodes "Interstage-nodes" (not simple repeaters)**

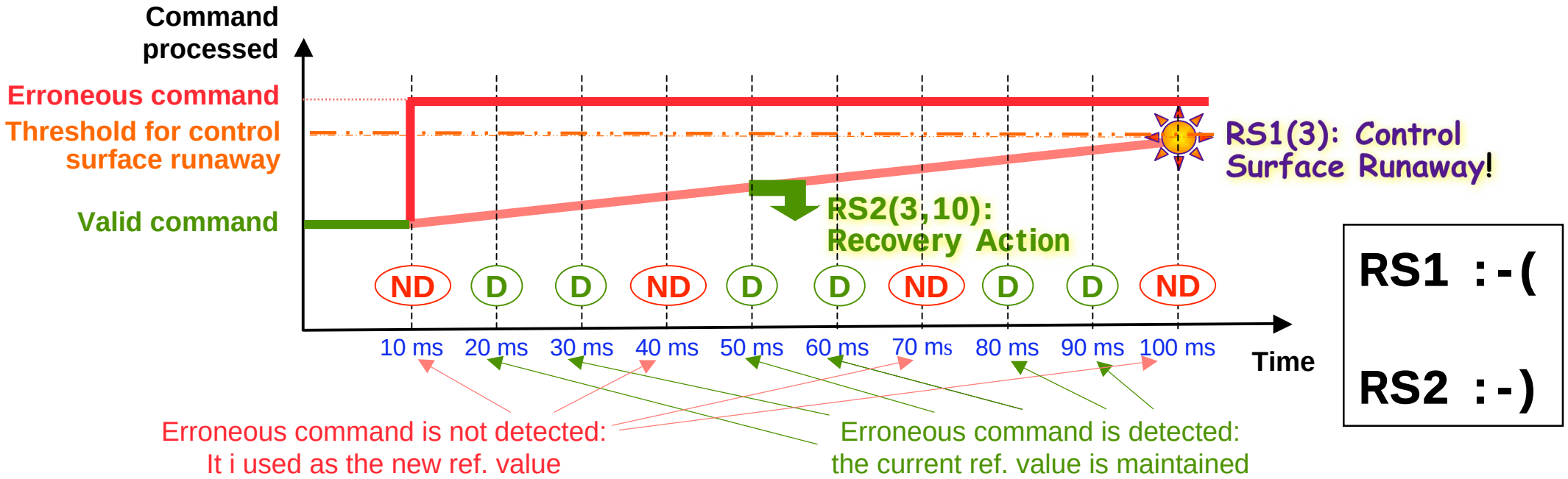


Context and Baseline

- **Slow dynamics of the process** (more than one erroneous command sustained before leading to an undesired event)
  -> Option to (re-)use previous command (even erroneous)
- **Undesired Event (UE) = "Runaway" of the controlled surface**
  -> Discrepancy wrt nominal reference value ≥ 5°
  [servomechanisms with max. speed of 50°/s]
  "Erroneous ref. value applied for 100ms (10 cycl.) => UE"
- **Safety requirement "risk of UE ≤ 10^{-9}/h"**
  -> Constraint on communication system integrity
  "Number of undetected erroneous messages < threshold *t*"
- **Recovery (mitigate issues) -> back-up actions**
  - **Ensure the correct updating of the reference value to the servomechanism**
  - **Do not discard too quickly the communication system**
  - **Do not impair the required safety level**
- **Favor options with limited structural redundancies**

Undesired Event & Recovery Strategies

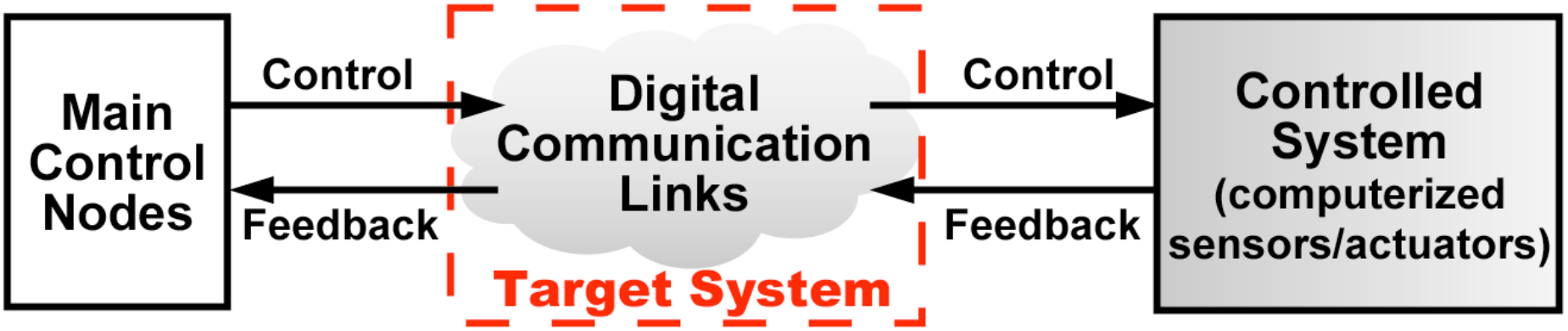
- **Re-use of the previous ("correct") command and "filtering":**
  - **RS1(*r*): launch the recovery after *r* consecutive processing cycles for which an error has been signaled;**
  - **RS2 (r,b) launch the recovery after *r* processing cycles for which an error has been signaled out of a set of *b* successive cycles**
- **Example (*r* = 3 and *b* = 10)**



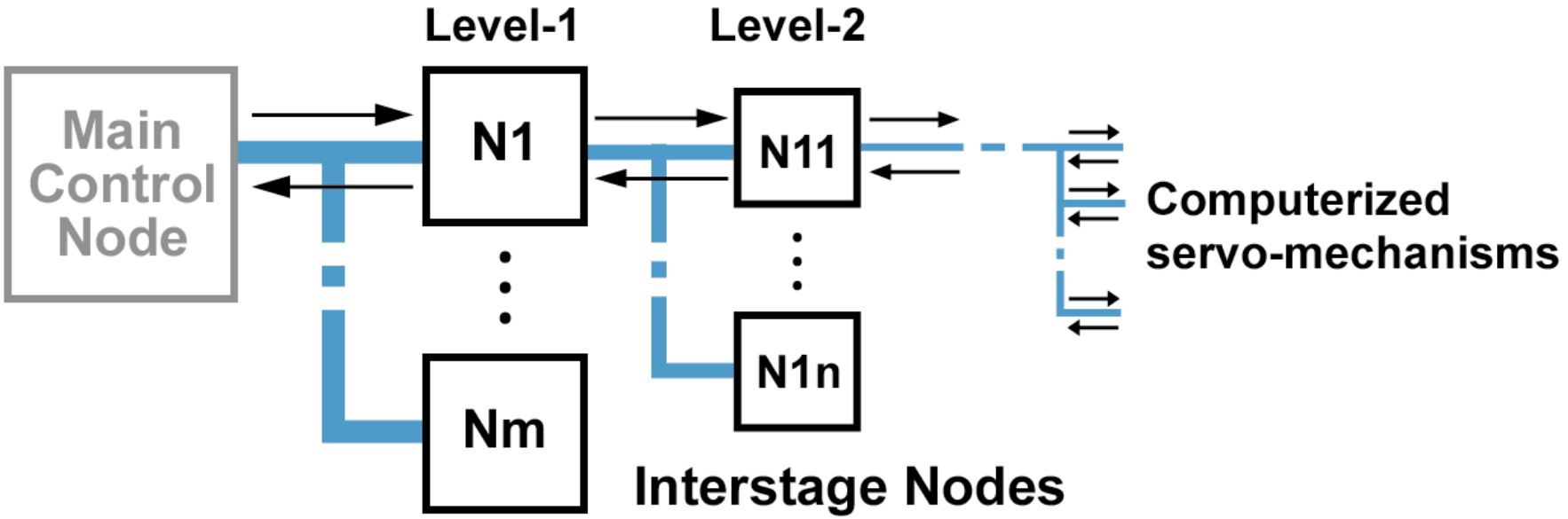
Target UE: Reception of 3 erroneous messages in a set of 10 cycles

Architectural Issues

Basic Architecture

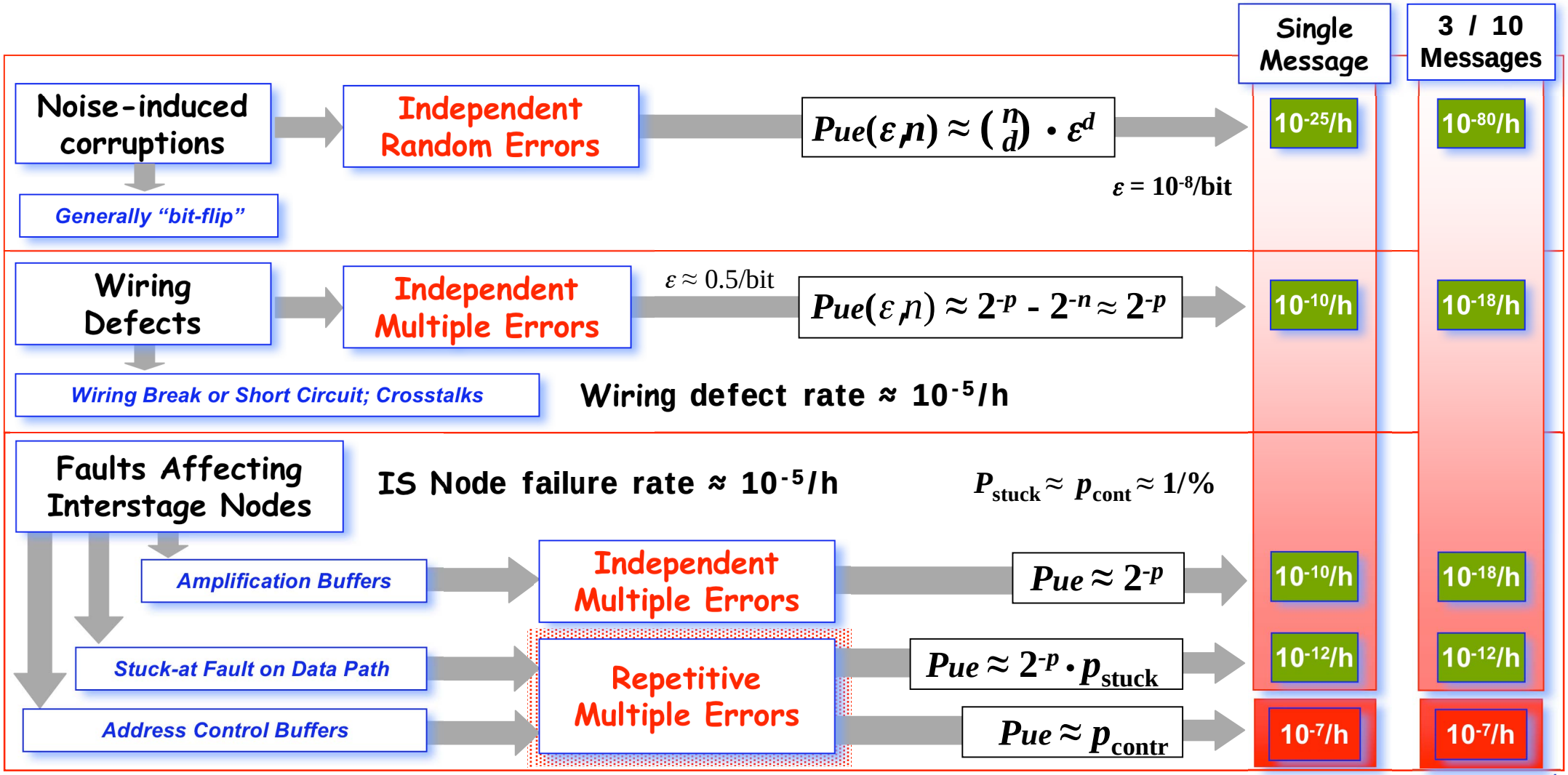


Example: Hierachical Organization of Interstage Nodes



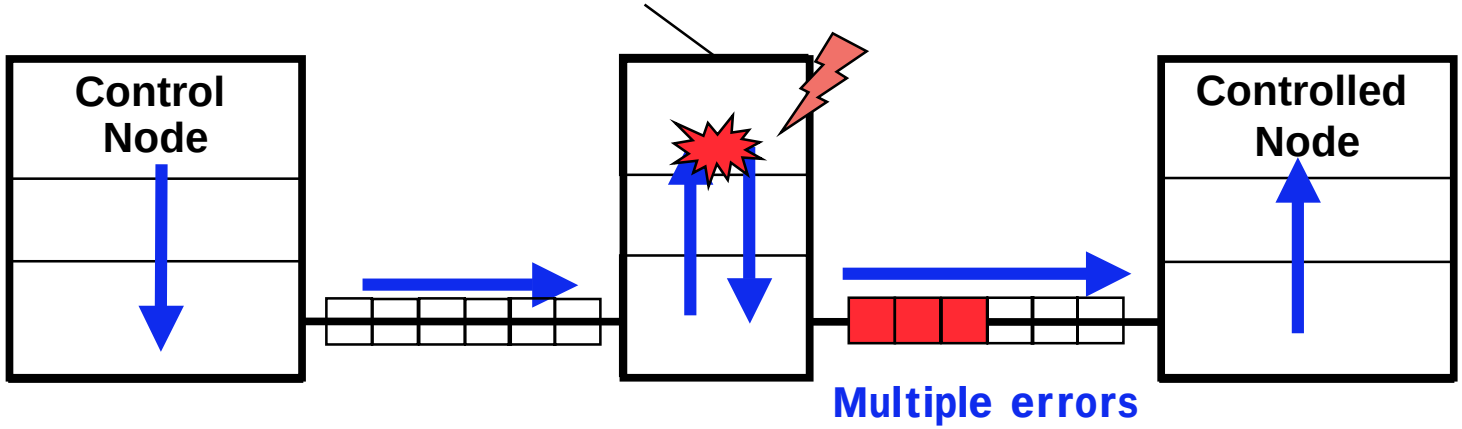
Risk Analysis (Using Classical CRC)

- Message length ≈ 100 bits — Bit Error Rate (ε) ∈ [0,0.5/bit]
- CRC-16: # control bits (p) = 16 — Hamming distance (d) = 5



Impact of Interstage Nodes

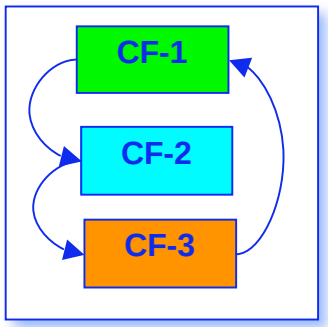
Interstage nodes process data



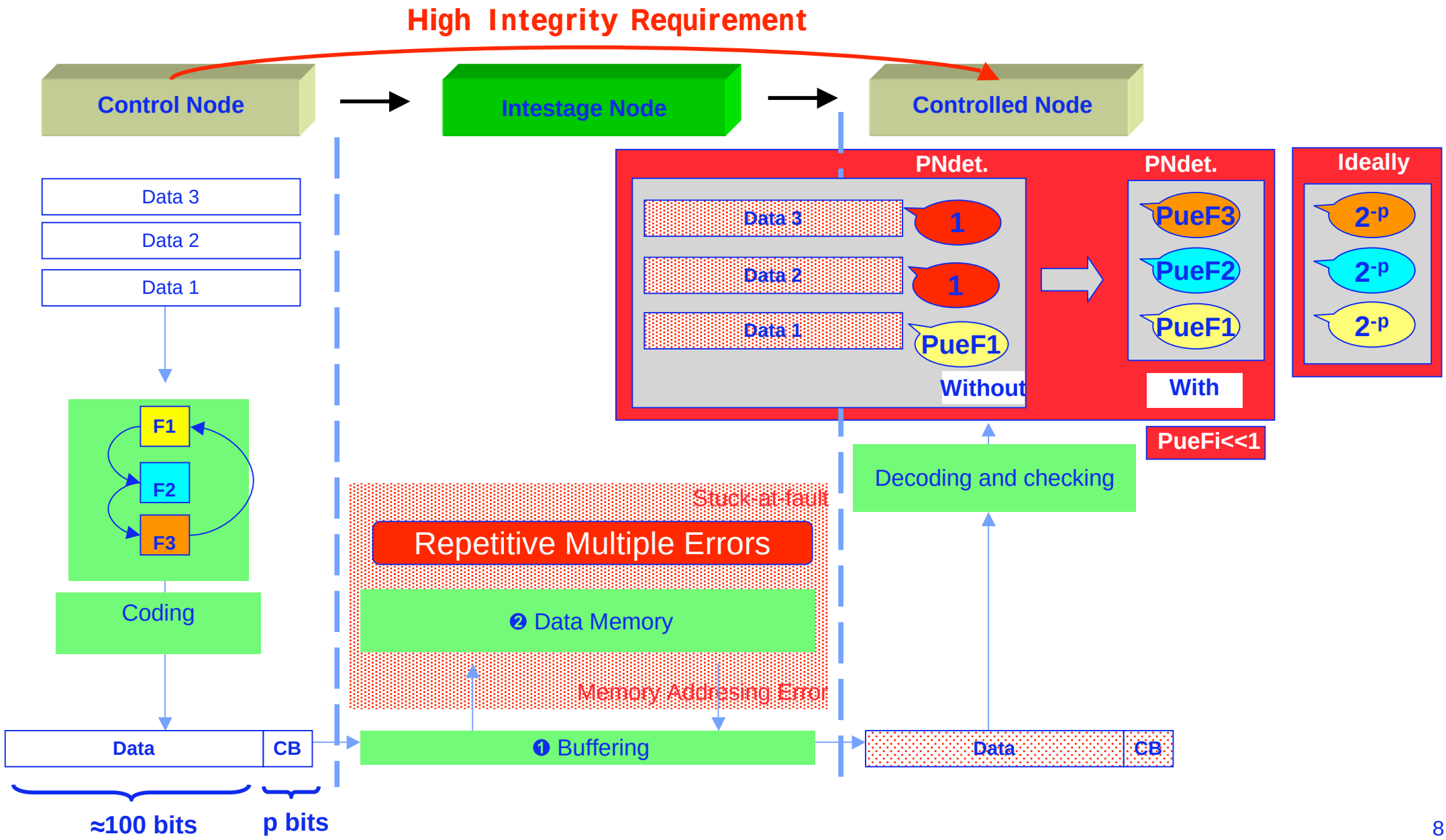
- **Classical approaches: -> Inefficient and/or improper**
  - Basic coding techniques (CRC)
  - End-to-end detection mechanims (HEDC, Keyed CRC, Safety Layer)
    Applicable, but most meant to cope with a single message

—> Introduce some degree of diversification

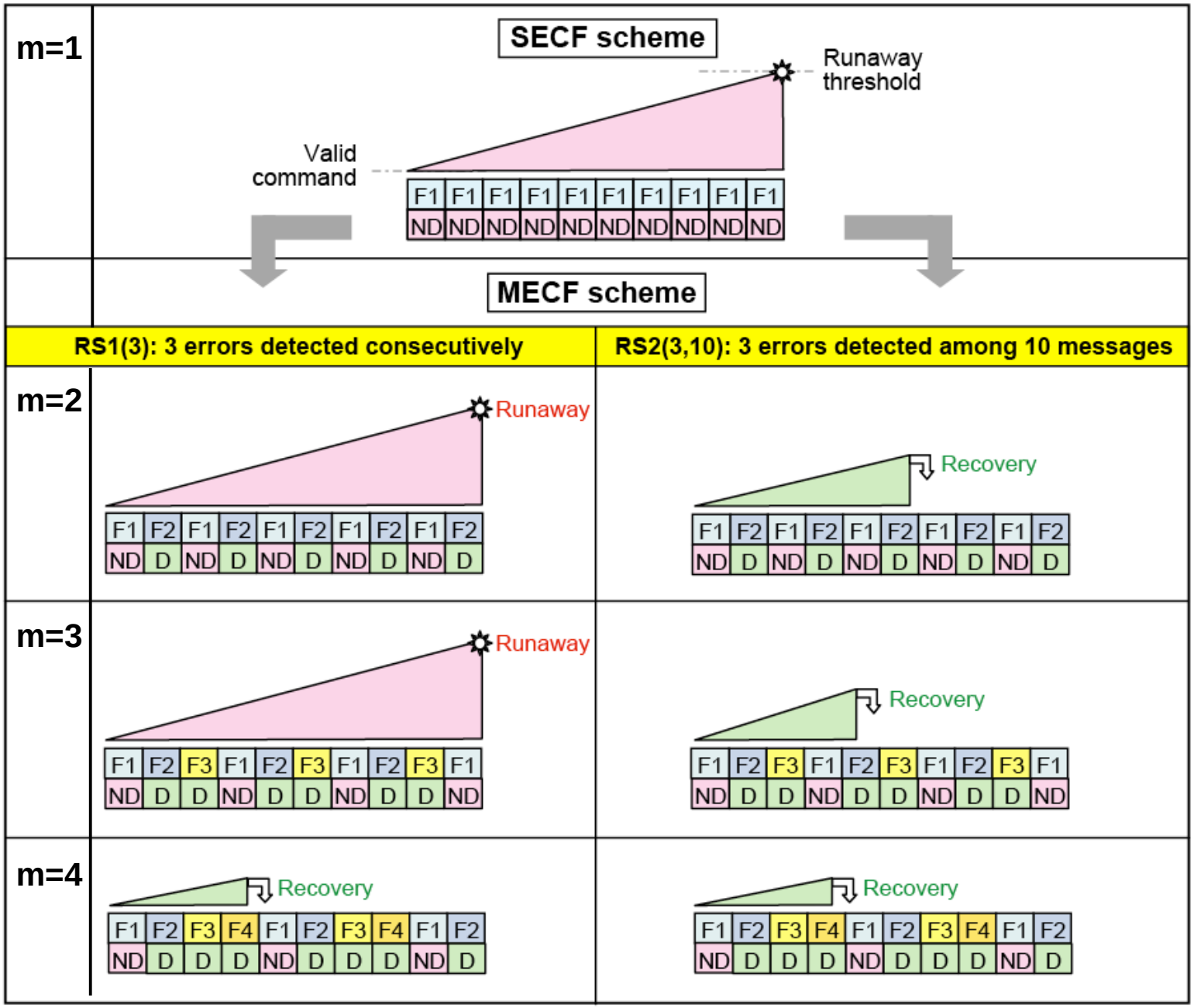
- data and redundancy (e.g., TMR)
- data and coding (Turbo Codes)
- coding function (e.g., rotation of the coding function)
  Multiple Error Coding Function ->
  (m = 3)



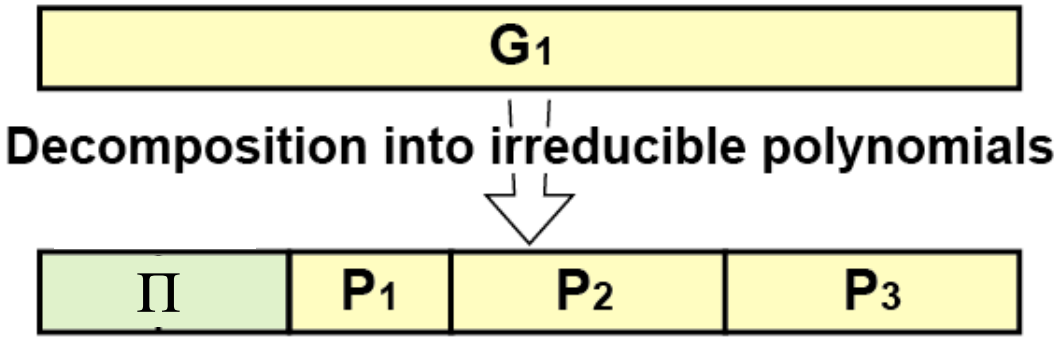
Principle and Objective/Benefit



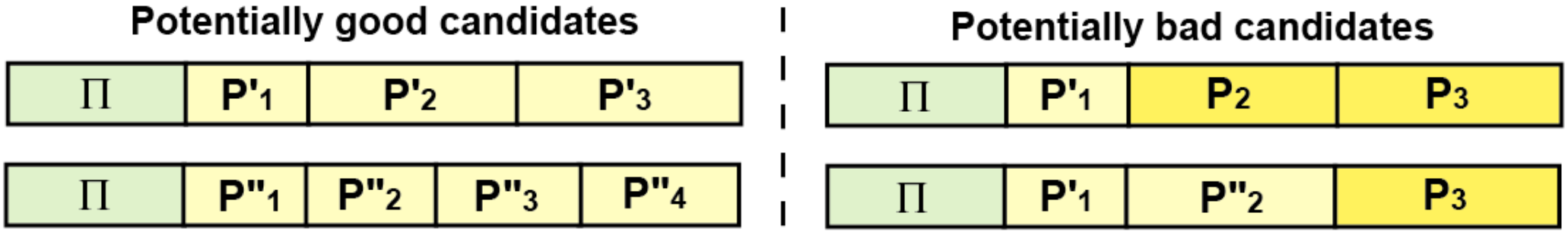
Impact on Detection and Recovery Latency



Implementation Using CRCs:
Selection of the Generator Polynomials



- **Π = small degree polynomial featuring "standard" error detection properties**
  (e.g., [1+x] => detection of odd-weight errors)



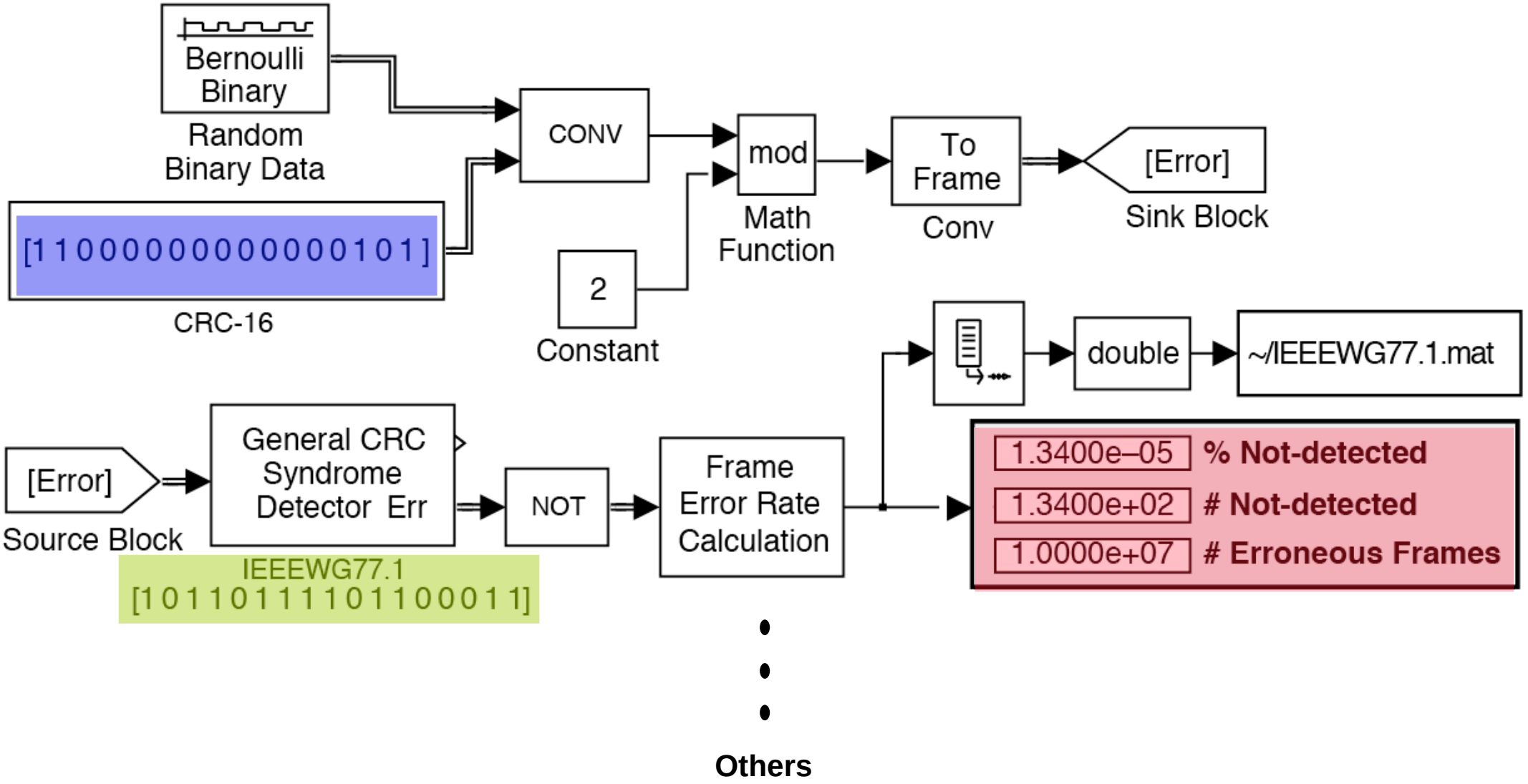
- **P'i and P''i ≠ Pi ∀i**

Generator Polynomial Selection

$G_1(x) = \underline{(1+x)} \cdot \underline{(1+x+x^7)} \cdot (1+x^2+x^3+x^4+x^8) = 1+x^3+x^5+x^6+x^7+x^9+x^{10}+x^{12}+x^{15}+x^{16}$		
Examples of potentially good candidates **$G(x)$ = (1+x) . 7-degree irreducible polynomial . 8-degree irreducible polynomial**		
Identifier	Polynomial representation	Decomposition into irreducible polynomials
$G_2(x)$	$1+x+x^6+x^7+x^8+x^9+x^{10}+x^{13}+x^{15}+x^{16}$	$\underline{(1+x)} \cdot (1+x+x^3+x^5+x^7) \cdot (1+x+x^2+x^4+x^5+x^6+x^8)$
$G_3(x)$	$1+x+x^6+x^{10}+x^{12}+x^{16}$	$\underline{(1+x)} \cdot (1+x+x^2+x^3+x^7) \cdot (1+x+x^4+x^5+x^6+x^7+x^8)$
$G_4(x)$	$1+x^5+x^6+x^7+x^8+x^9+x^{10}+x^{16}$	$\underline{(1+x)} \cdot (1+x^3+x^7) \cdot (1+x+x^2+x^5+x^6+x^7+x^8)$
Examples of potentially bad candidates **$G(x)$ = (1+x) . (1+x+x^7) . 8-degree irreducible polynomial**		
$G_5(x)$	$1+x+x^2+x^3+x^5+x^6+x^9+x^{10}+x^{12}+x^{14}+x^{15}+x^{16}$	$\underline{(1+x)} \cdot \underline{(1+x+x^7)} \cdot (1+x+x^5+x^6+x^8)$
$G_6(x)$	$1+x^3+x^6+x^7+x^{10}+x^{13}+x^{14}+x^{16}$	$\underline{(1+x)} \cdot \underline{(1+x+x^7)} \cdot (1+x^2+x^3+x^4+x^5+x^7+x^8)$

This was analyzed and confirmed via extensive simulation runs

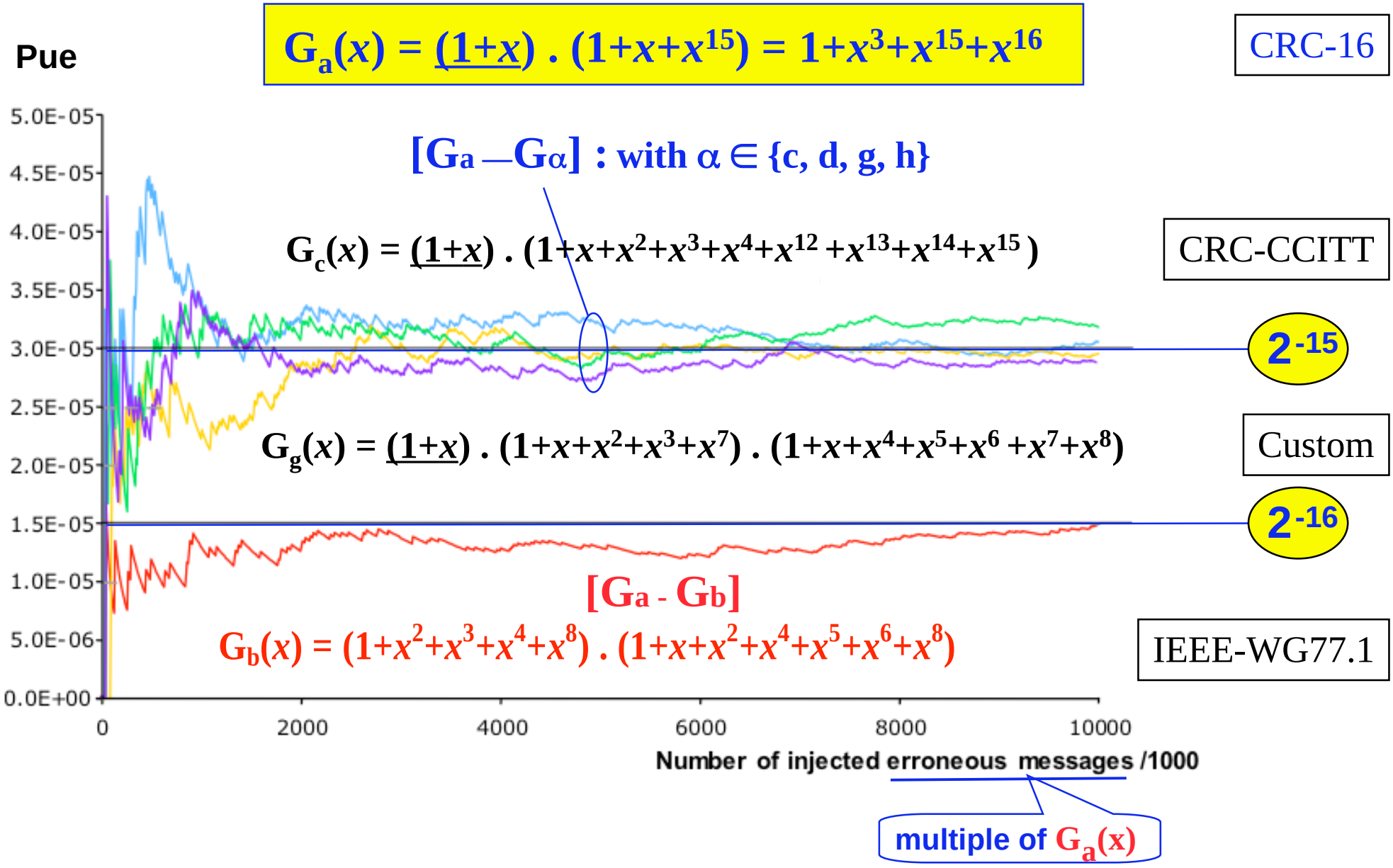
Simulation Framework (Matlab-Simulink)



Example of Analysis: Target Codes

$G_a(x) = \underline{(1+x)} \cdot (1+x+x^{15}) = 1+x^2+x^{15}+x^{16}$ — Standard generator polynomial: CRC-16		
Standard generator polynomials **$G(x) = (1+x)$. 15-degree polynomial**		
Identifier	Polynomial representation	Decomposition into irreducible polynomials
$G_b(x)$: IEEE-WG77.1	$1+x+x^5+x^6+x^8+x^9+x^{10}+x^{11}+x^{13}+x^{14}+x^{16}$	$(1+x^2+x^3+x^4+x^8) \cdot (1+x+x^2+x^4+x^5+x^6+x^8)$
$G_c(x)$: CRC-CCITT	$1+x^5+x^{12}+x^{16}$	$\underline{(1+x)} \cdot (1+x+x^2+x^3+x^4+x^{12}+x^{13}+x^{14}+x^{15})$
$G_d(x)$: IBM-SDLC	$1+x+x^2+x^4+x^7+x^{13}+x^{15}+x^{16}$	$\underline{(1+x)}^2 \cdot (1+x+x^3+x^4+x^5+x^6+x^8+x^{10}+x^{12}+x^{13}+x^{14})$
$G_e(x)$: CRC-16Q*	$1+x+x^3+x^4+x^5+x^6+x^8+x^{11}+x^{15}+x^{16}$	$\underline{(1+x)} \cdot (1+x^3+x^5+x^8+x^9+x^{10}+x^{15})$
$G_f(x)$: IEC-TC57	$1+x+x^4+x^7+x^8+x^9+x^{11}+x^{12}+x^{14}+x^{16}$	$\underline{(1+x)}^2 \cdot (1+x+x^3+x^6+x^7) \cdot (1+x^2+x^3+x^4+x^5+x^6+x^7)$
Custom generator polynomials **$G(x) = (1+x)$. 7-degree irreducible polynomial . 8-degree irreducible polynomial**		
$G_g(x) = G_3(x)$	$1+x+x^6+x^{10}+x^{12}+x^{16}$	$\underline{(1+x)} \cdot (1+x+x^2+x^3+x^7) \cdot (1+x+x^4+x^5+x^6+x^7+x^8)$
$G_h(x) = G_4(x)$	$1+x^5+x^6+x^7+x^8+x^9+x^{10}+x^{16}$	$\underline{(1+x)} \cdot (1+x^3+x^7) \cdot (1+x+x^2+x^5+x^6+x^7+x^8)$

Examples of Results from Simulation Runs



About Improvement Achieved

Threshold: 10^{-9}/h

Protection schemes → ↓ Fault classes: stuck-at on	SECF (m = 1)	MECF m = 2	MECF m = 3
- buffer memory	1.5×10^{-12} / h	2.3×10^{-17} / h	6.9×10^{-22} / h
- address control	10^{-7} / h	1.5×10^{-12} / h	4.5×10^{-17} / h

Concluding Remarks

- **Pragmatic and Novel Approach for Mitigating High Integrity Requirements in Critical Communications Systems**
- **CRC-based Implementation:**
  - Theoretical issues associated to properties of generator polynomials provide a sound basis for identifying criteria for selecting suitable coding functions
  - Criteria validated via extensive simulation runs
- **Generalization: investigation of alternative policies for mixing distinct coding functions (CF)**
- **Formalization: derivation of closed-form expressions**
  - Probability of undetected errors (Pue)
  - (Min) Latency for system recovery action after an error is undetected (LRA) [# of message cycles]

Example: $m>1$ # of distinct CF; r # of reported error detections, b size of frame of messages (only for RS2)

$\mathbf{LRA(RS1) = r+1}$ for $r < m$; $\mathbf{LRA(RS2)} = \lceil m \times r/(m-1) \rceil$ for $LRA < b$