INTRINSIC ID

Secure your digital life™

An Accurate Probabilistic Reliability Model for Silicon PUFs

Roel Maes

Intrinsic-ID, the Netherlands

CHES-2013

Introduction: Silicon PUFs and Reliability

- Basic PUF properties:
  1. **Uniqueness:**
     Equivalent responses from distinct PUF instances are sufficiently different
  2. **(Un)reliability:**
     Equivalent responses from one single PUF instance are sufficiently alike (up to a few errors)
- Both properties have an equally important impact on the PUFs usability and efficiency

PUF **A**

0	1	0	0	1	0	1	1

PUF **B**

1	0	0	1	1	0	0	0

HD(**A**; **B**) = large

PUF **A:enroll**

0	1	0	0	1	0	1	1

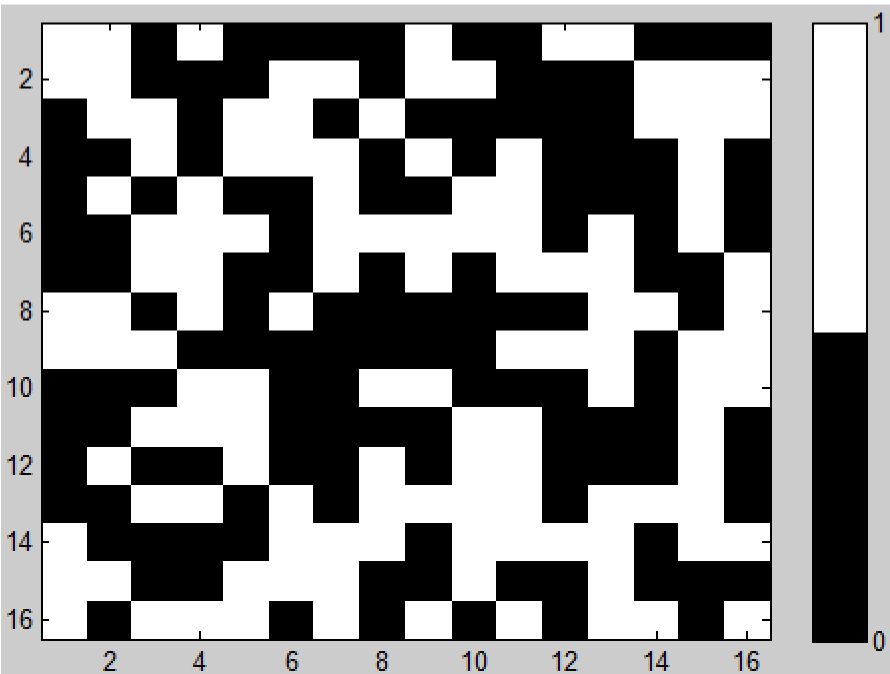
PUF **A:reconstruct**

0	1	1	0	1	0	1	1

HD(**A:enroll**; **A:reconstruct**) = small

Problem Statement: "Old" Error Model

PUF Response



Error model in use until now:

- single fixed error rate p_e
- every cell equally likely to produce an error on every evaluation

(= binary symmetric channel)

Problem:

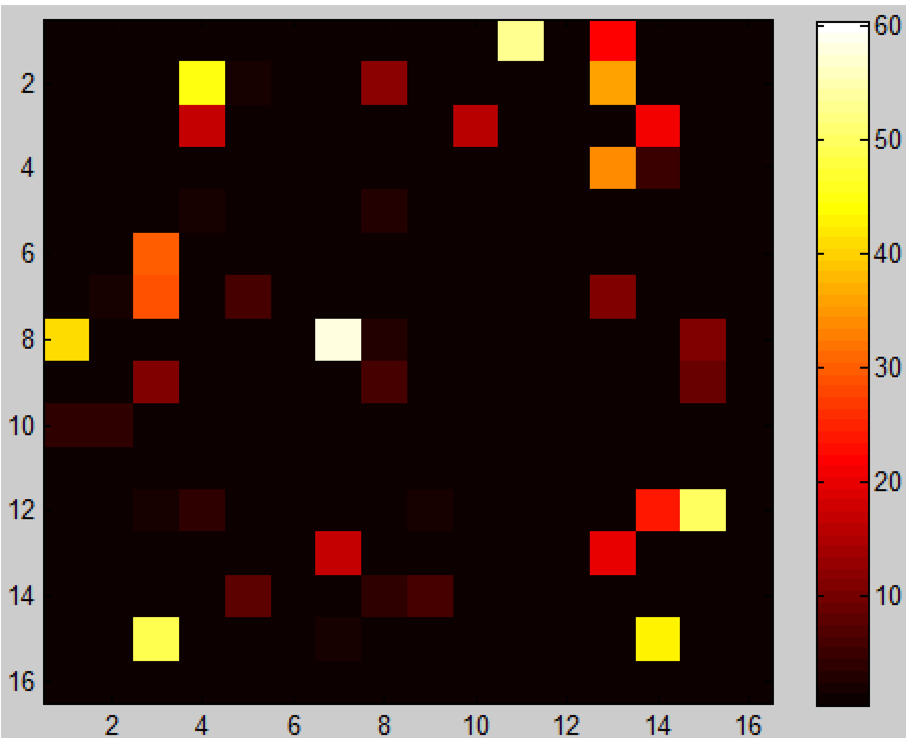
Does not realistically/accurately describe actual PUF reliability behavior...

this becomes apparent when we evaluate the same PUF response many times...

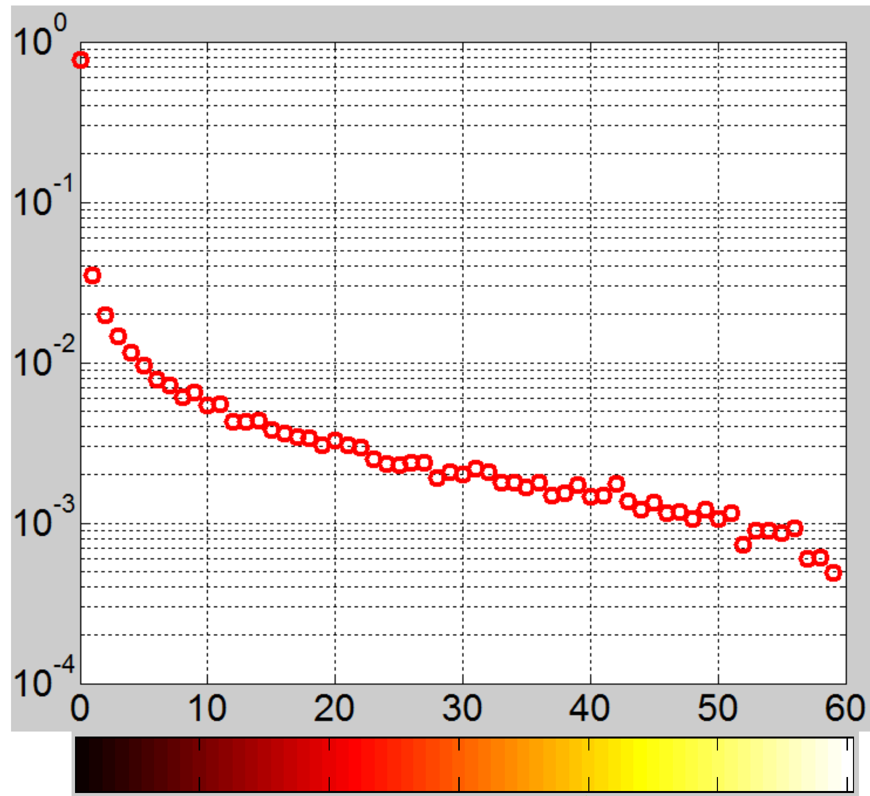
Problem Statement: Experimental Observation

Error behavior over 60 evaluations

Response behavior over 60 evaluations



Histogram of error-counts over cells



 # errors in 60 evaluations of a cell

New Model: Approach = Hidden Variable Model

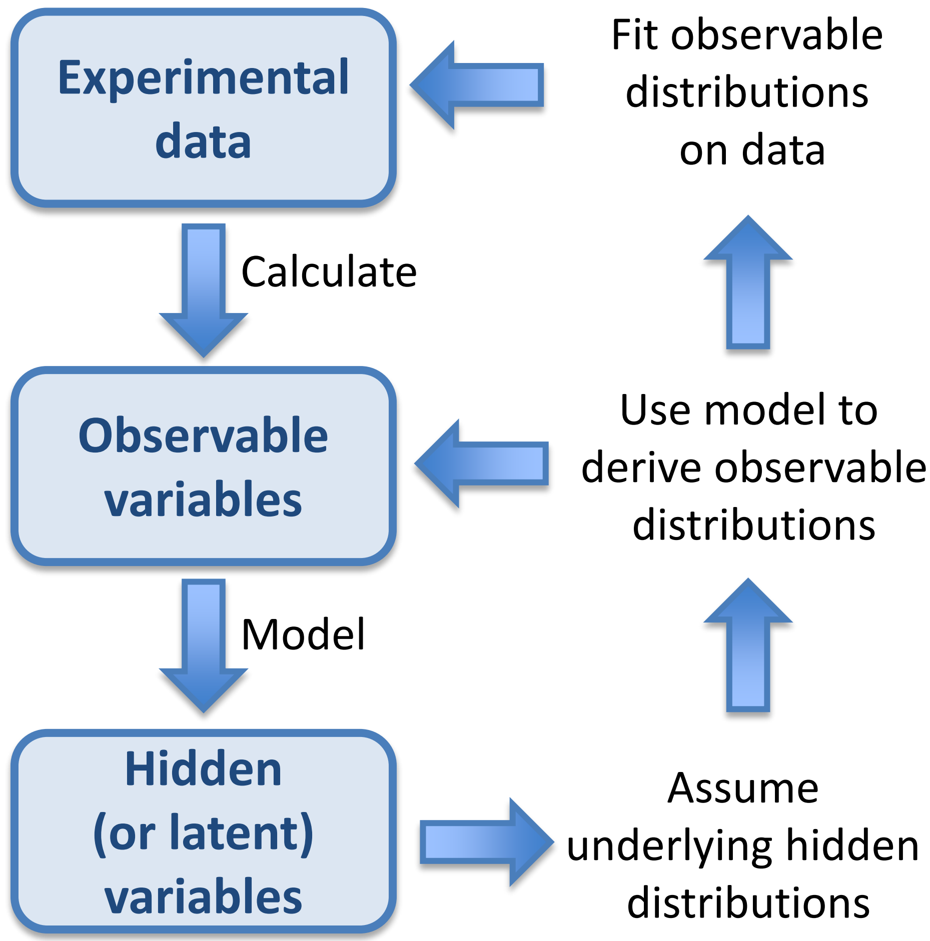
Response bit: $r_i^{(j)}$
(evaluation j of cell i)

Cell error-probability: $p_{e,i}$
(= $\mathbf{Pr}\left(R_i \neq r_i^{\mathrm{enroll}}\right)$)

Cell error-count: $s_{e,i}^{(n)}$
(= #errors in n eval's)

Process variable: m_i
(causes uniqueness)

Noise variable: $n_i^{(j)}$
(causes unreliability)



New Model: Distribution Derivation

Model relation:

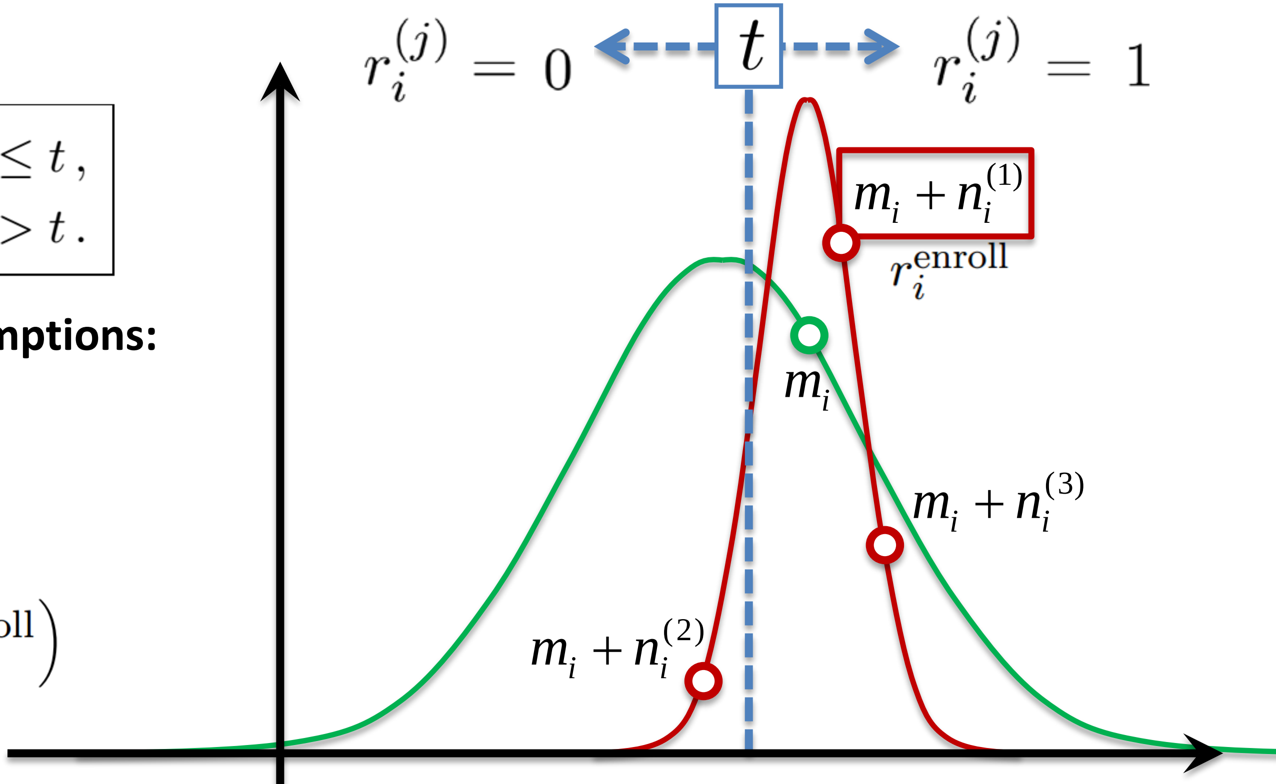
$$\boxed{r_i^{(j)} = \begin{cases} 0\,, \text{ if } m_i + n_i^{(j)} \le t\,, \\ 1\,, \text{ if } m_i + n_i^{(j)} > t\,. \end{cases}}$$

Hidden distribution assumptions:

$$M \sim \mathcal{N}(\mu_M, \sigma_M^2)$$

$$N_i \sim \mathcal{N}(0, \sigma_N^2)$$

$$p_{e,i} \stackrel{\text{def}}{=} \mathbf{Pr}\left(R_i \neq r_i^{\text{enroll}}\right)$$

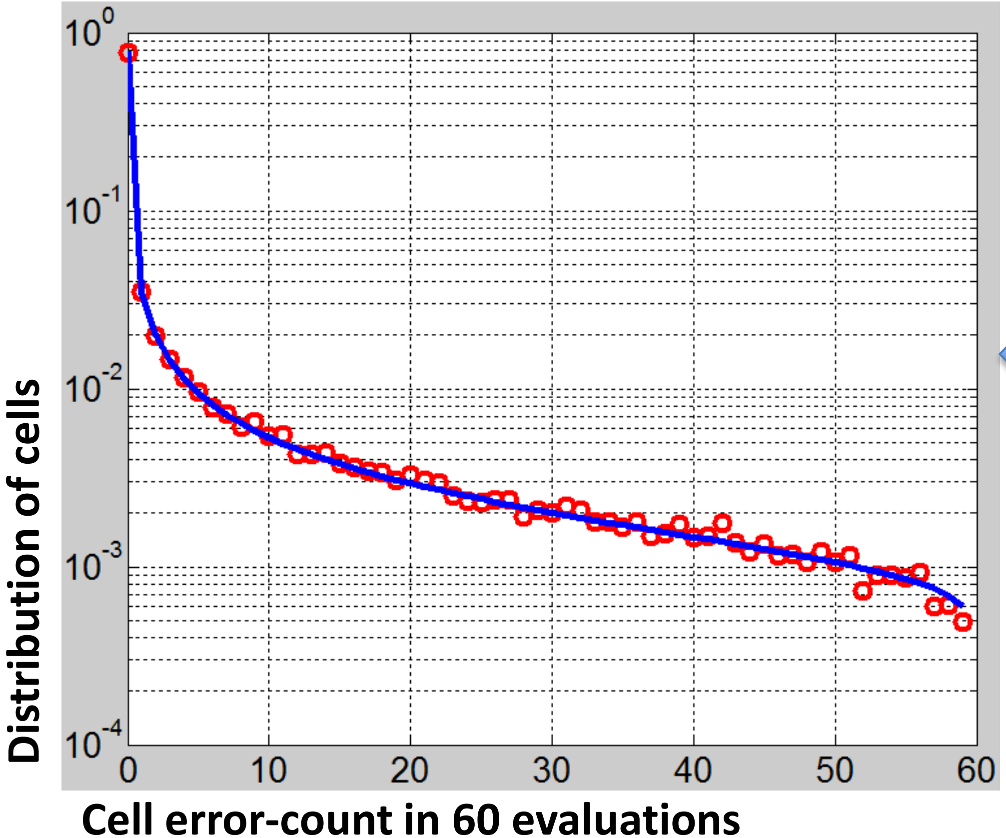


Cell error-probability distribution:

$$\mathbf{cdf}_{P_e}(x) = \lambda_1 \cdot \int_{-\infty}^{\Phi^{-1}(x)} \Phi(-u) \cdot \left(\varphi(\lambda_1 u + \lambda_2) + \varphi(\lambda_1 u - \lambda_2)\right) du$$

New Model: Experimental Fit

Cell error-count distribution: $\mathbf{pmf}_{S_e^{(n)}}(x) = \int_0^1 f_{\text{bino}}(x; n, u) \cdot \mathbf{pdf}_{P_e}(u)\ du$



Experimental data:
from UNIQUE project
[Katzenbeisser et al., CHES-2012]

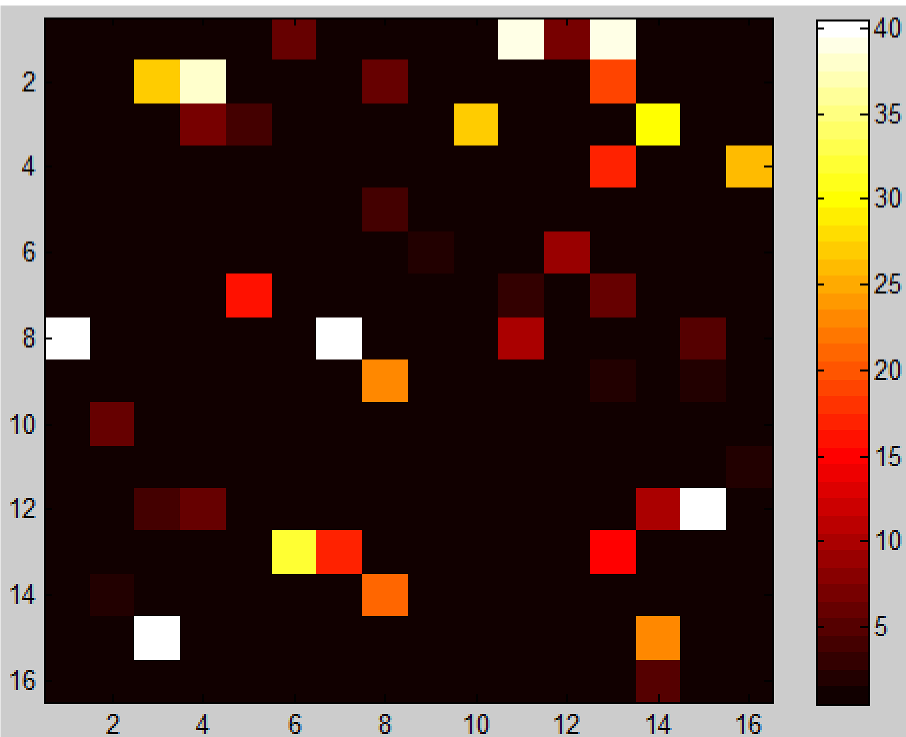
PUF type	MSE of fit	λ_1	λ_2
SRAM	4.5e-9	0.12	0.02
Buskeeper	5.8e-10	0.09	0.03
DFF	1.2e-9	0.08	0.04
Arbiter	1.8e-9	0.07	0.05

New Model + Temperature?

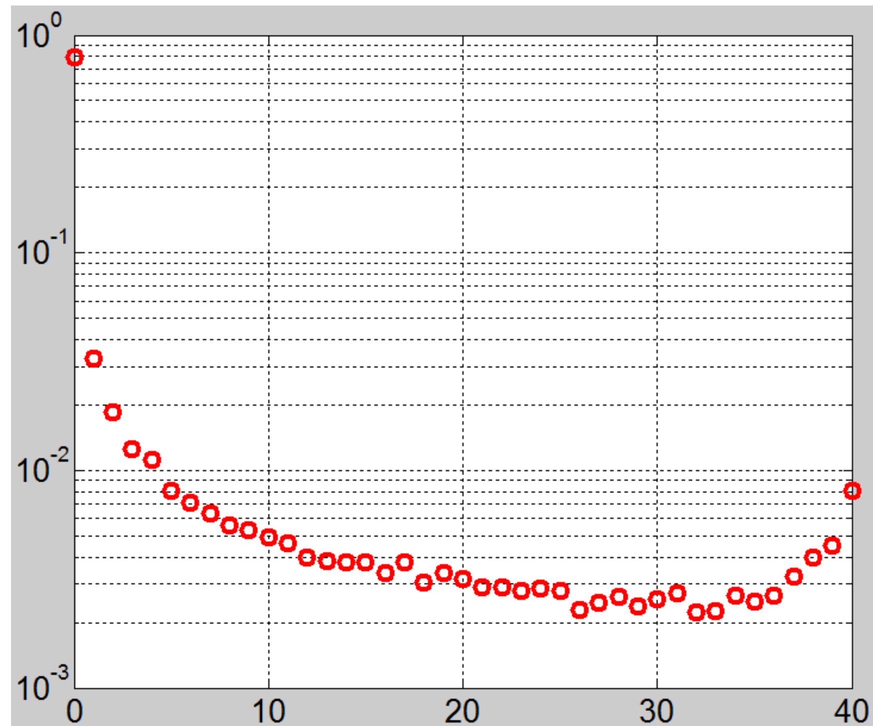
Error behavior at fixed temperature (25°C):

Error behavior under temperature change:

40 evaluations @ 85°C w.r.t. enrollment @ 25°C



Histogram of error-counts over cells



 # errors in 60 evaluations (@25°C) of a cell

 # errors in 40 evaluations (@85°C) of a cell

(w.r.t. enrollment @ 25°C)

New Model + Temperature: Distribution Derivation

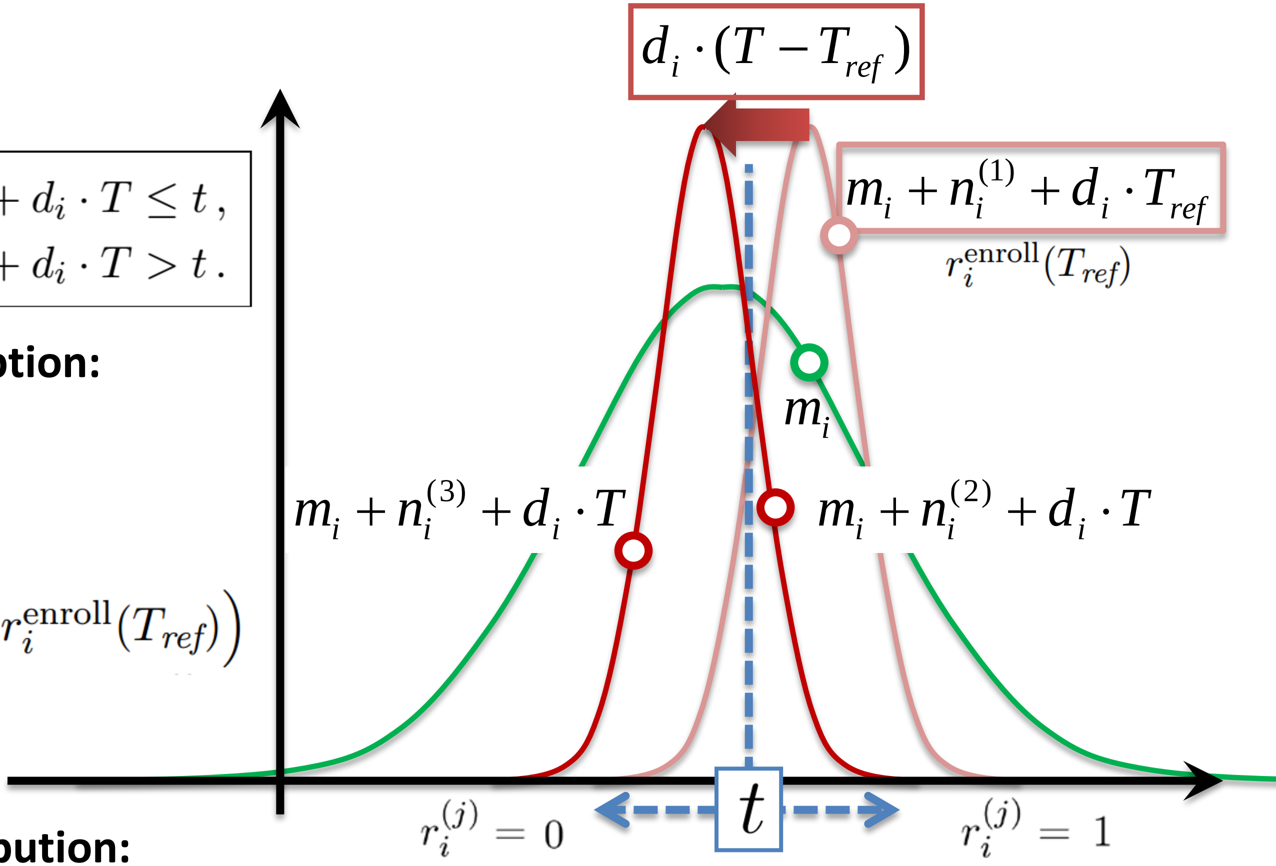
Model relation:

$$r_i^{(j)}(T) = \begin{cases} 0\,, \text{ if } m_i + n_i^{(j)} + d_i \cdot T \leq t\,, \\ 1\,, \text{ if } m_i + n_i^{(j)} + d_i \cdot T > t\,. \end{cases}$$

Hidden distribution assumption:

$$D \sim \mathcal{N}(0, \sigma_D^2)$$

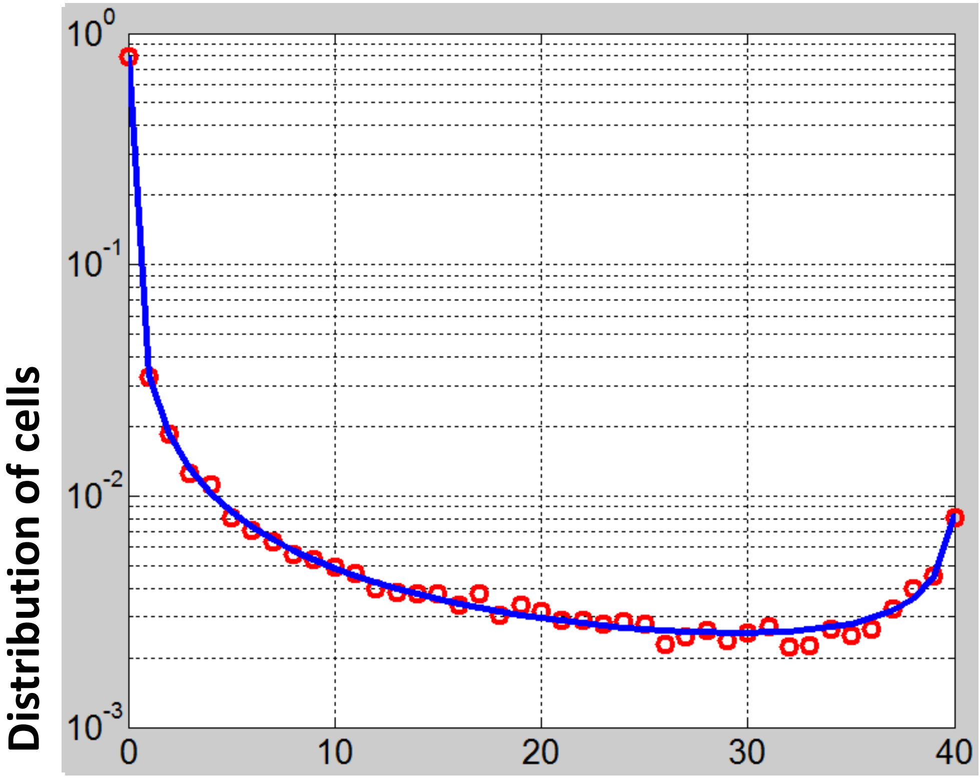
$$p_{e,i}(T; T_{ref}) = \mathbf{Pr}\left(R_i(T) \neq r_i^{\text{enroll}}(T_{ref})\right)$$



Cell error-probability distribution:

$$\mathbf{cdf}_{P_e(T;T_{ref})}(x) = \frac{\lambda_1 \theta}{|\Delta T|} \cdot \int_{-\infty}^{\Phi^{-1}(x)} \int_{-\infty}^{+\infty} \left[\Phi(-u)\,\varphi\left(\theta \tfrac{v-u}{|\Delta T|}\right) + \Phi(u)\,\varphi\left(\theta \tfrac{v+u}{|\Delta T|}\right)\right] \cdot \varphi(\lambda_1 u + \lambda_2)\, du\, dv$$

New Model + Temperature: Experimental Fit

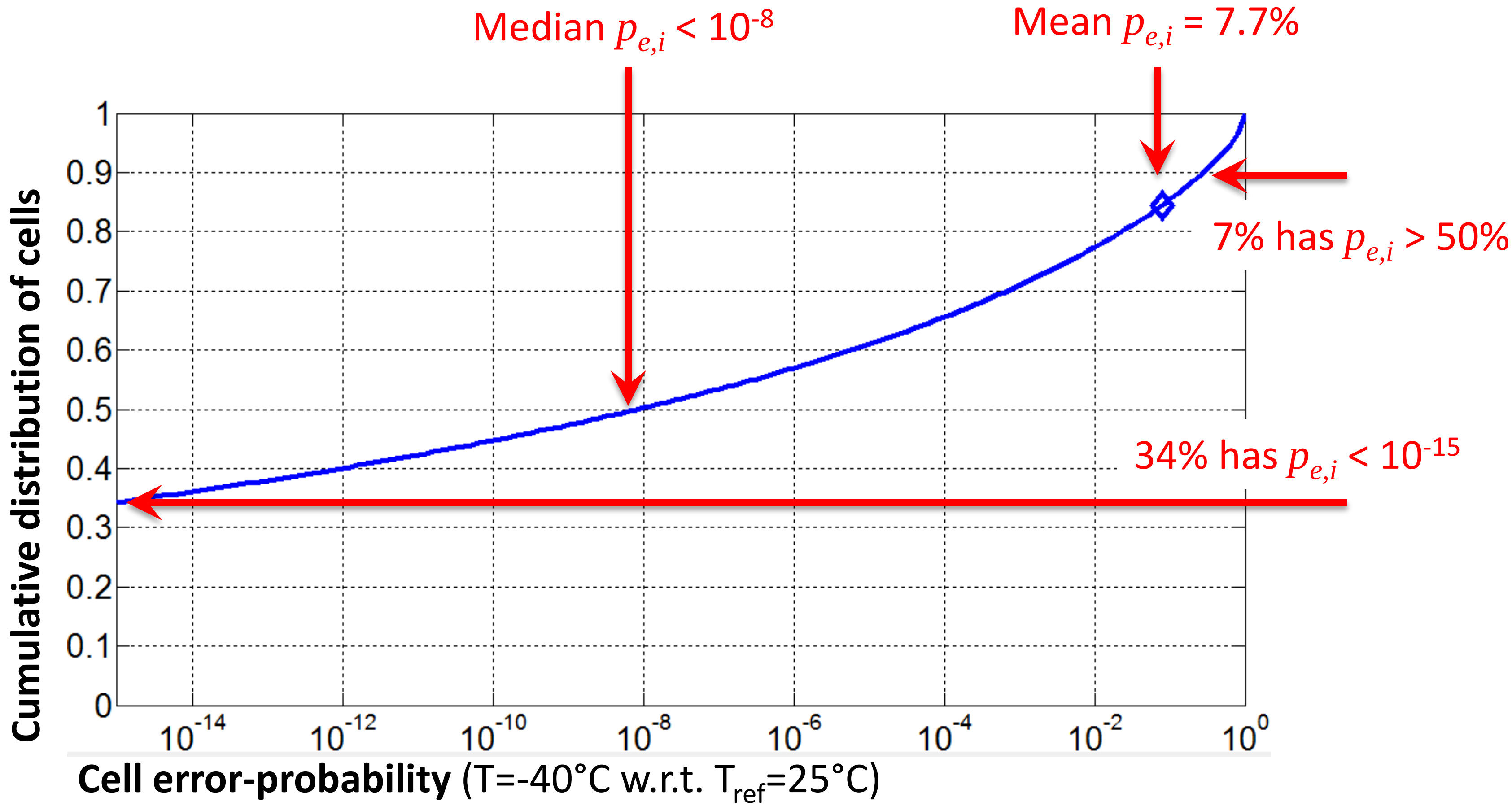


Cell error-count in 40 evaluations (T=-40°C w.r.t. T_{ref}=25°C)

Fit for SRAM PUF (from UNIQUE):

- over range T = [-40°C...+85°C]
- optimal fit for θ = 45.0 *(independent of T)*
- average MSE = 1.6^e-6 *(over full T-range)*

Interpretation of New Model Error Distribution



Majority of errors in a PUF response are caused by a small minority of cells

Implications for PUF-based Key Generation

Fuzzy Extractor System: *(spec.: 128-bit entropy with > 1-10^-9 reliability)*

1060-bit PUF response → **FE-1:** (5, 1, t = 2) repetition code → **FE-2:** (212, 128, t = 11) BCH code → 128-bit entropy for key

Old model: *(for every key generator)*

p_e = 7.7% ⟶ $1 - \mathbf{F}_{\mathbf{bino}}(2, 5, 7.7\%)$ = 0.4% ⟶ $1 - \mathbf{F}_{\mathbf{bino}}(11, 212, 0.4\%)$ = 1.5e-10 = p_{fail}

New model: *(for one particular key generator)*

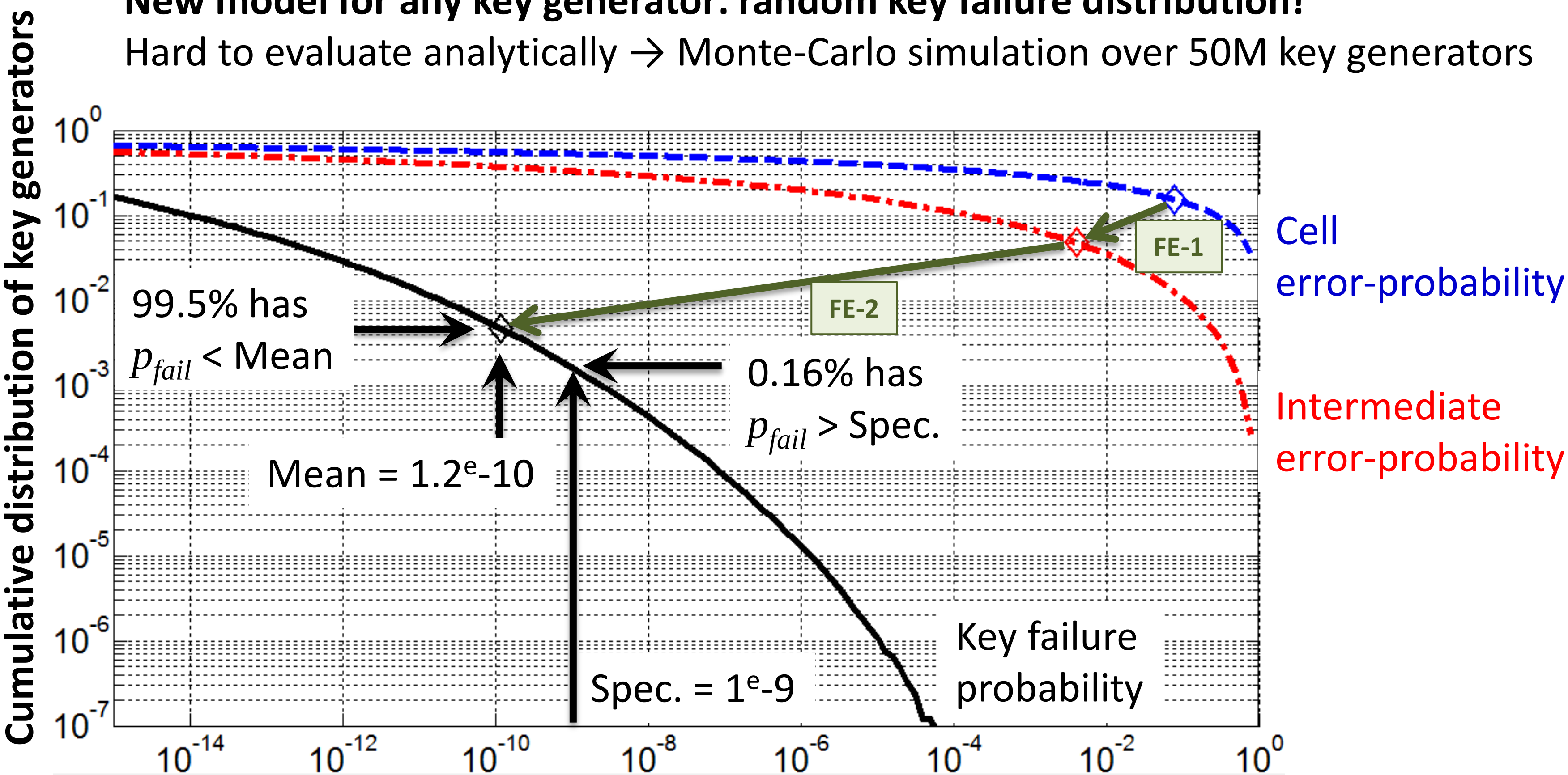
$\mathbf{p_e} = (p_{e,1}, \ldots, p_{e,1060})$,
with $p_{e,i} \sim \mathbf{cdf}_{P_e(T;T_{ref})}$
$(\lambda_1 = 0.12, \lambda_2 = 0.02, \theta = 45.0)$
⟶ $1 - \mathbf{F}_{\mathbf{PB}}(2, 5, \mathbf{p_e})$ = $\mathbf{p_{e,int}}$ ⟶ $1 - \mathbf{F}_{\mathbf{PB}}(11, 212, \mathbf{p_{e,int}})$ = p_{fail}

$\mathbf{F}_{\mathbf{PB}}$ **= Poisson-Binomial distribution:**
when trials are independent but no longer identically distributed

Implications for PUF-based Key Generation

New model for any key generator: random key failure distribution!

Hard to evaluate analytically → Monte-Carlo simulation over 50M key generators



Key failure probability (p_{fail})

Main Conclusions

- New PUF reliability model is realistic, generic and very accurate:
  - Hidden variable model makes **cell-specific behavior** explicit
  - Yields analytic expressions for **error-probability distributions**
  - Can be **fit very accurately** on experimental observations, including **temperature dependent** behavior
  - Applicable to most Silicon PUF types, both memory- and delay-based
- Allows to study full PUF reliability behavior
  - **As opposed to only average-case behavior in old model**
  - Shows **very skewed distributions**:
    "large majority of PUF errors are caused by small minority of PUF cells"
  - Enables **Monte-Carlo simulations** to study effect on PUF-based applications, e.g. key generators

INTRINSIC ID

Secure your digital life™

Thank you!

Any questions?