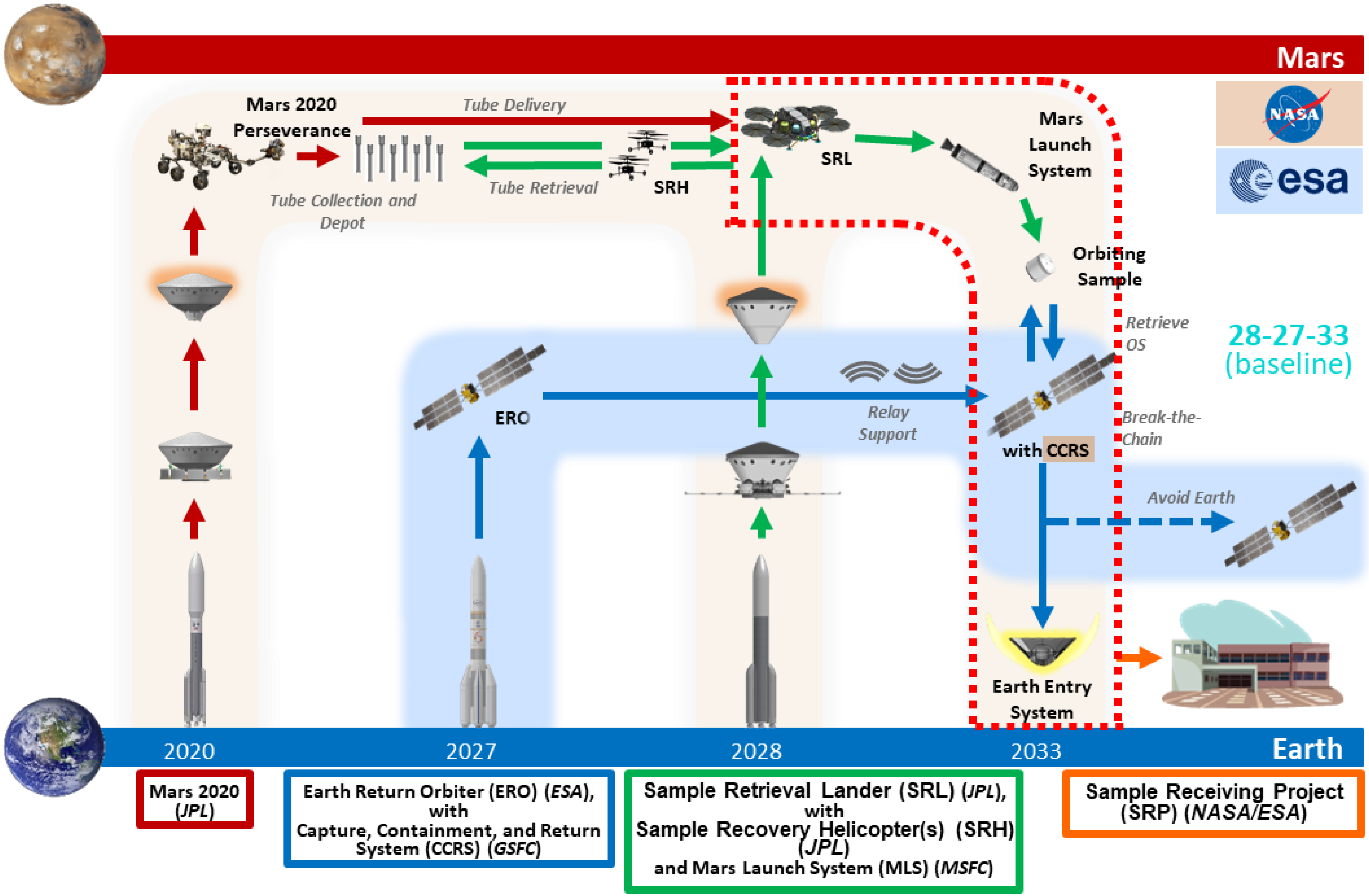
The Containment Assurance Risk Framework of the Mars Sample Return Program

Giuseppe Cataldo[1], Amy Braverman[2], Kyle Grello[1], Aaron Siddens[2], Kevin Carpenter[2]

[1]NASA Goddard Space Flight Center, 8800 Greenbelt Road, Greenbelt, MD 20771
[2]Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109

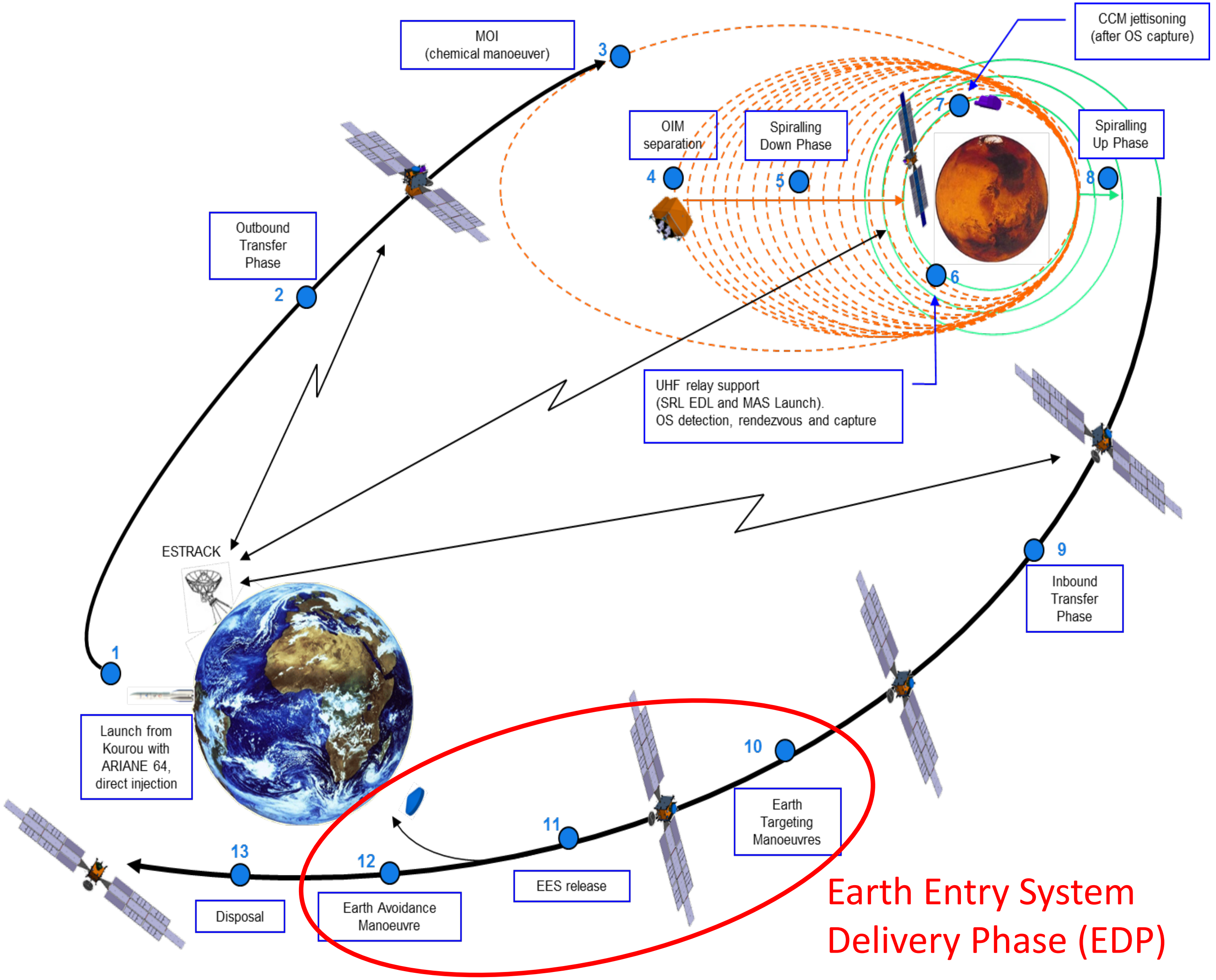
DATAWorks
Defense and Aerospace Test and Analysis Workshop
Alexandria, VA
April 26, 2023

Planned MSR Campaign Architecture Overview

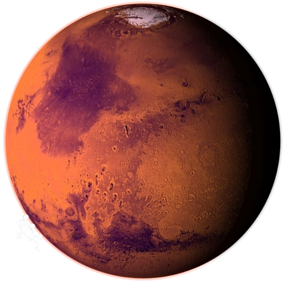


ERO-CCRS mission overview

- Mission objectives:
  - Capture the OS and bring it back to Earth
  - Relay support for Mars assets
- Nominal mission ("28/27/33"):
  - Launch and near-Earth commissioning [30 days]
  - Outbound transfer with heliocentric parking orbit [3 years]
  - Mars orbit insertion [2 weeks]
  - Spiral down [<1 year]
  - Low Mars orbit (relay support, OS rendezvous, OS containment) [1-1.5 years]
  - Spiral up [<1 year]
  - Inbound transfer [1 year]
  - EES delivery phase [few days]
  - Retirement [few days]



What is planetary protection?



FORWARD PLANETARY PROTECTION (FPP)

From NASA Procedural Requirement NPR 8715.24:

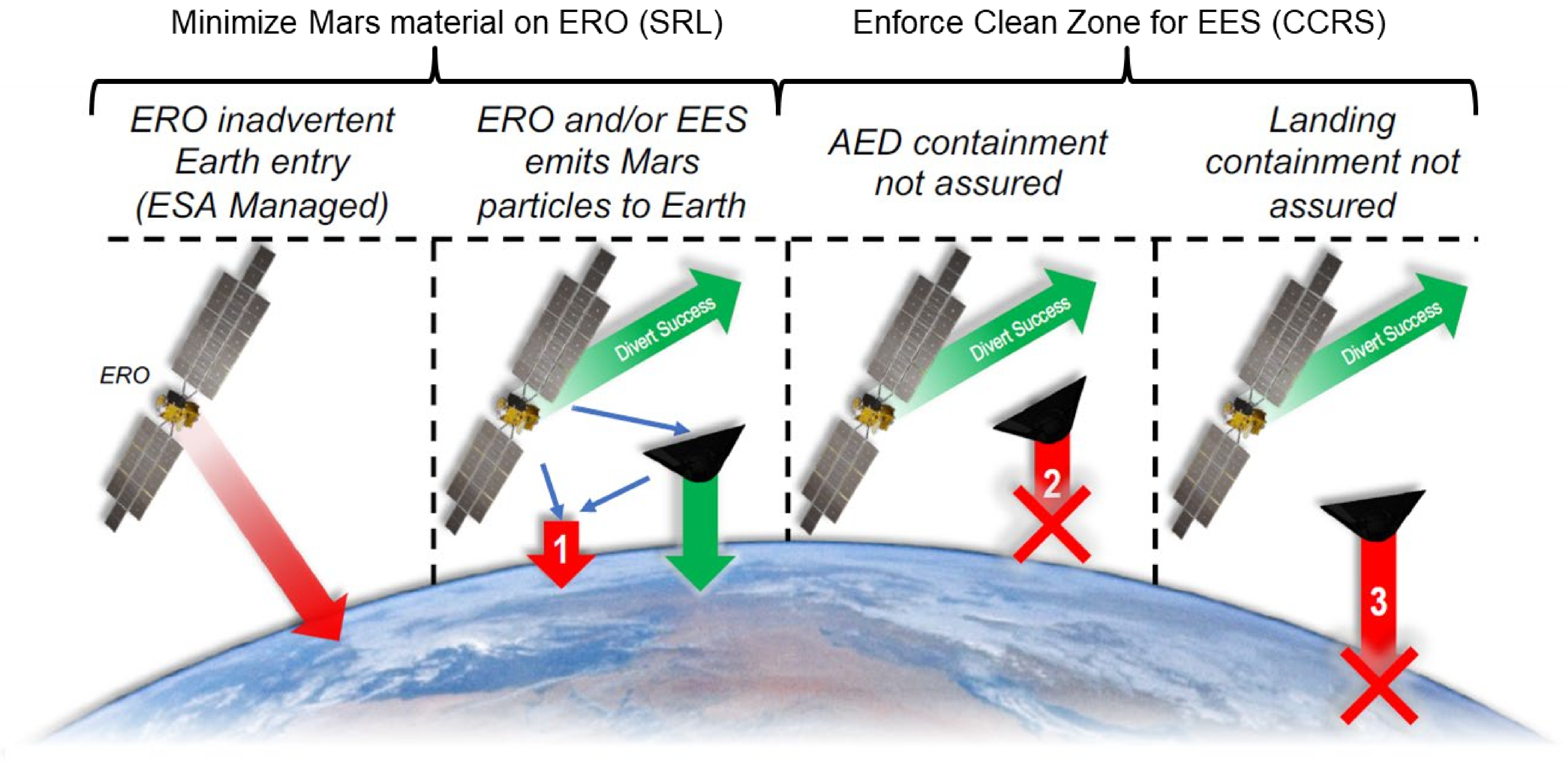
"Planetary protection is the practice of **protecting solar system bodies** from harmful contamination by terrestrial materials to enable scientific exploration and **protecting the Earth-Moon system** from possible harmful extraterrestrial contamination that may be returned from other solar system bodies."

BACKWARD PLANETARY PROTECTION (BPP)

Also see:

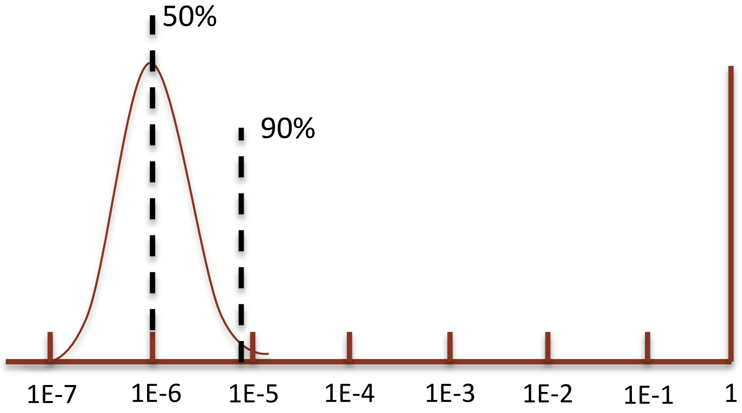
1. *Article IX, UN Space Treaty* (UNOOSA 2017, Report of the Committee on the Peaceful Use of Outer Space, 60th Session, A/72/20, United Nations, New York)
2. *Planetary Protection Policy*, Committee on Space Research (COSPAR), 2021

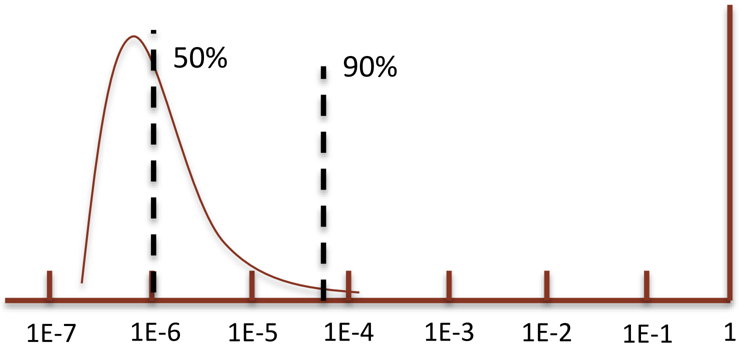
The 5 primary paths of Mars material that can enter Earth's biosphere



Backward Planetary Protection – Key Concepts

- Break-the-chain of contact (BTC)
  - Ensure all Mars material and hardware that contacted Mars material is sterilized or contained inside externally clean vessels.
- **Containment Not Assured** (CNA)
  - The state in which one or more requirements for ensuring containment of Mars material have not been met.
  - CNA does not necessarily equate to Mars material release.
  - If the CNA probability exceeds a target probability, additional analyses and/or mitigation measures *may* be undertaken.
- **Probability Target**
  - Values used in engineering requirements to drive robust containment capabilities.
  - Targets are used to understand if the path to demonstrating PP compliance is quantitative or qualitative.
- Risk elements
  - MSR's BPP requirements are organized in three risk elements, each with a 10^{-6} CNA *probability target.*





CNA probability targets specify both median (50th percentile) and upper bound values (90th percentile) to protect against non-normal distributions.

Outline

TASK: Develop a **framework for verification of containment assurance (CA) requirements** through analysis with input from all relevant phases, including on-orbit assembly, pre-release, approach (A), entry (E), descent (D), and landing (L).

This analytical framework, briefly called Containment Assurance Risk Framework (CARF), is intended to **integrate models from several domains** and requires **extensive coordination among all the teams**.

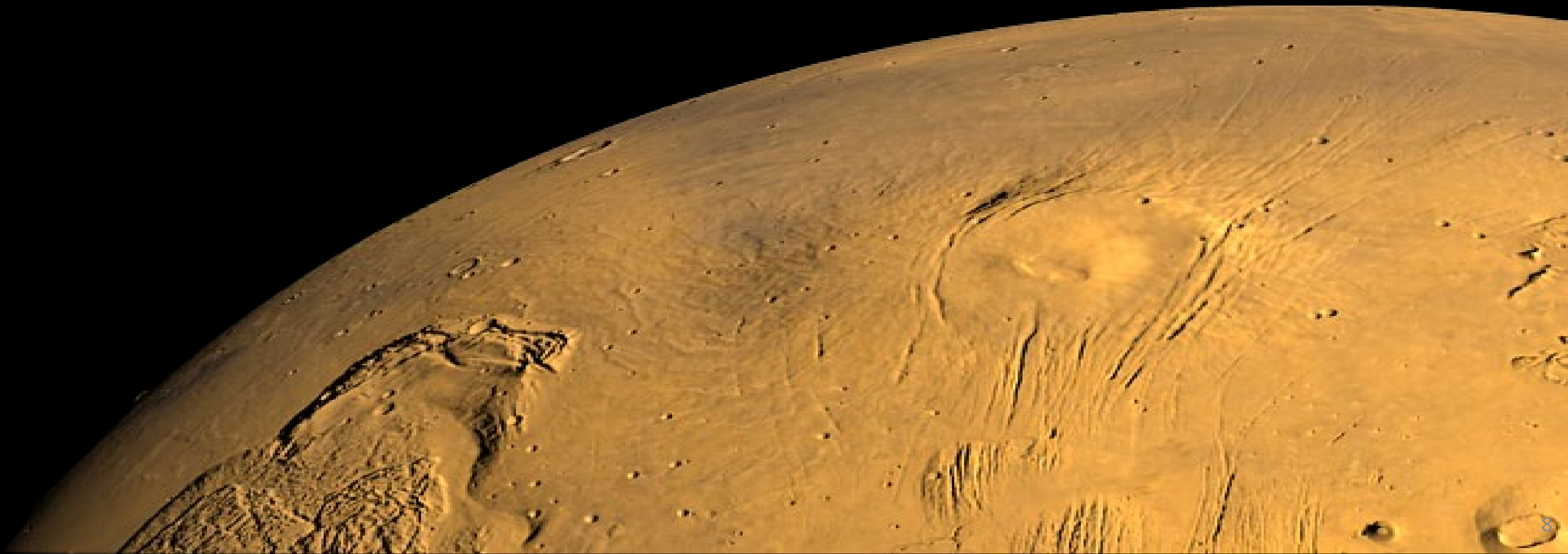
GOAL: This presentation will guide the reader through a conception of this framework and its implementation in terms of **methodologies, resources, and preliminary results**.

Some technical aspects of this framework are described in some detail and are to be seen as proposed tools or methodologies with potential benefits.

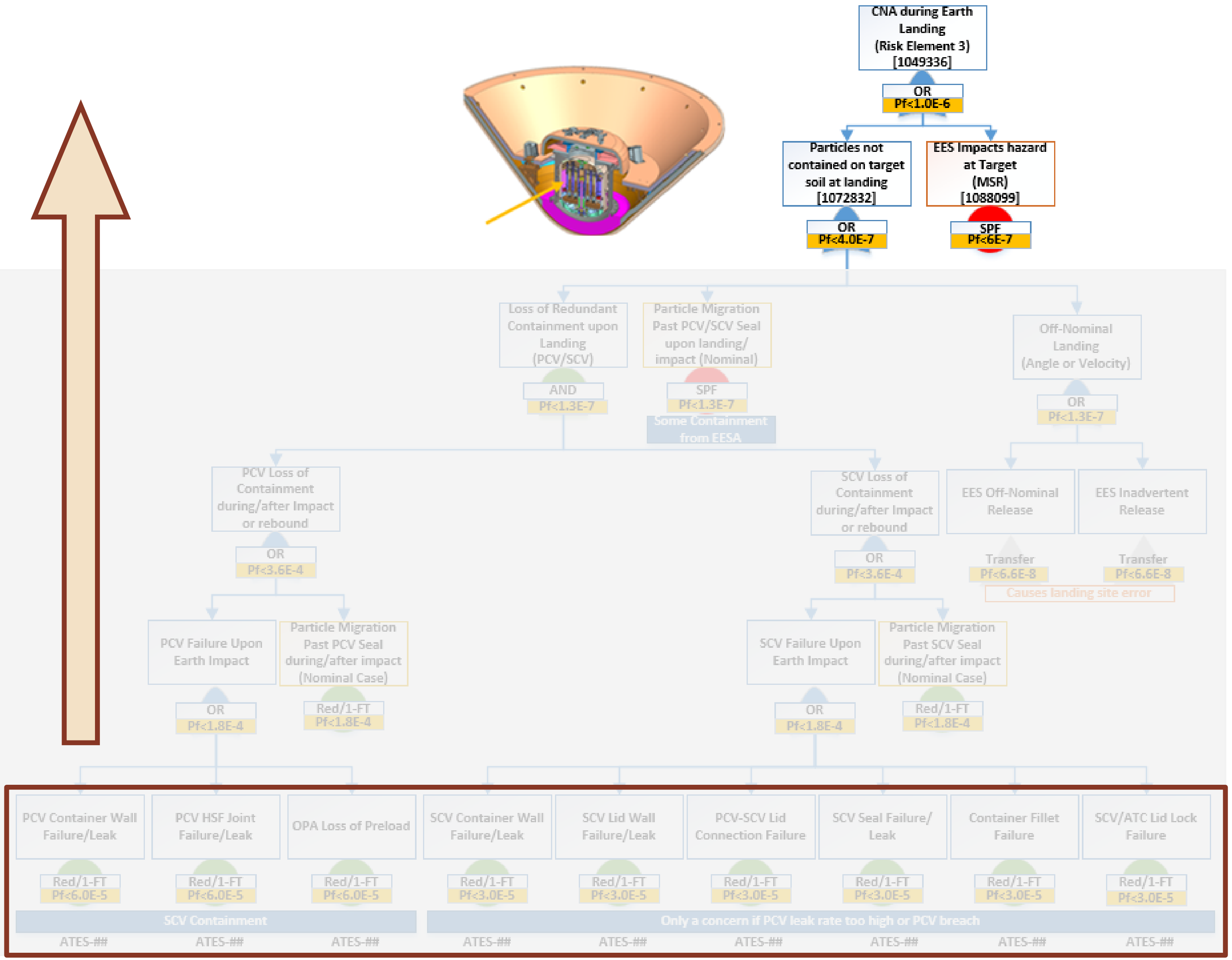
AGENDA

1. CARF Overview
2. CARF Implementation
3. Example of application
4. Conclusions

CARF Overview



Fault tree analysis – What can go wrong and how?



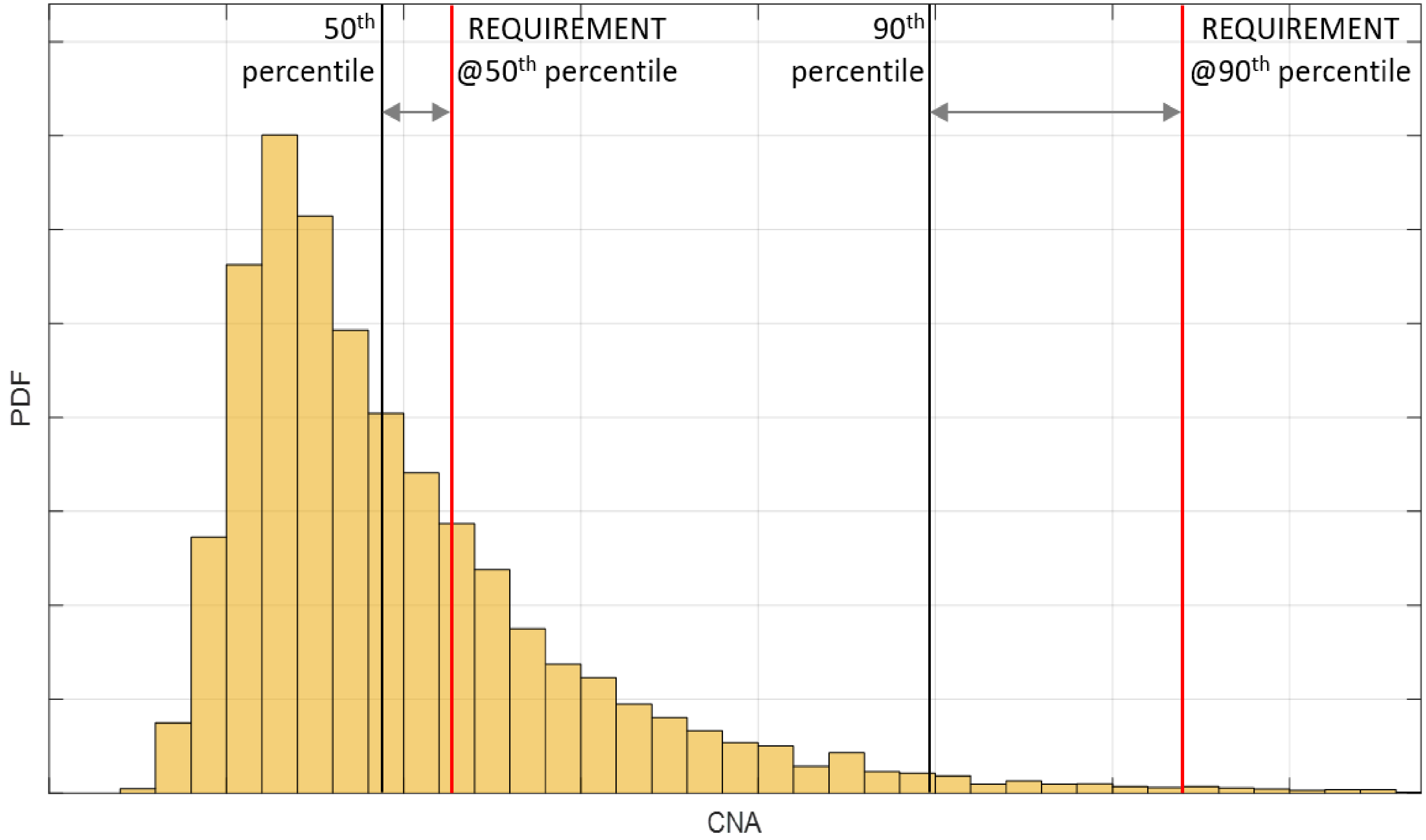
(Credit: Kyle Grello)

1. Identify **failure modes** and **fault conditions** that cause CNA.
2. Combine them in **fault trees** down to the initiating/basic events.
3. Determine **failure rates**, i.e., probabilities of occurrence, of **initiating/basic events**.
   - Note these can be either point estimates or probability density functions (PDFs).
4. Roll up following fault tree logic.

NOTE: Given the non-Gaussian nature and interdependence of several of these PDFs, proper statistical methods are needed to combine them as they are rolled up.

Outcome – Quantified CNA

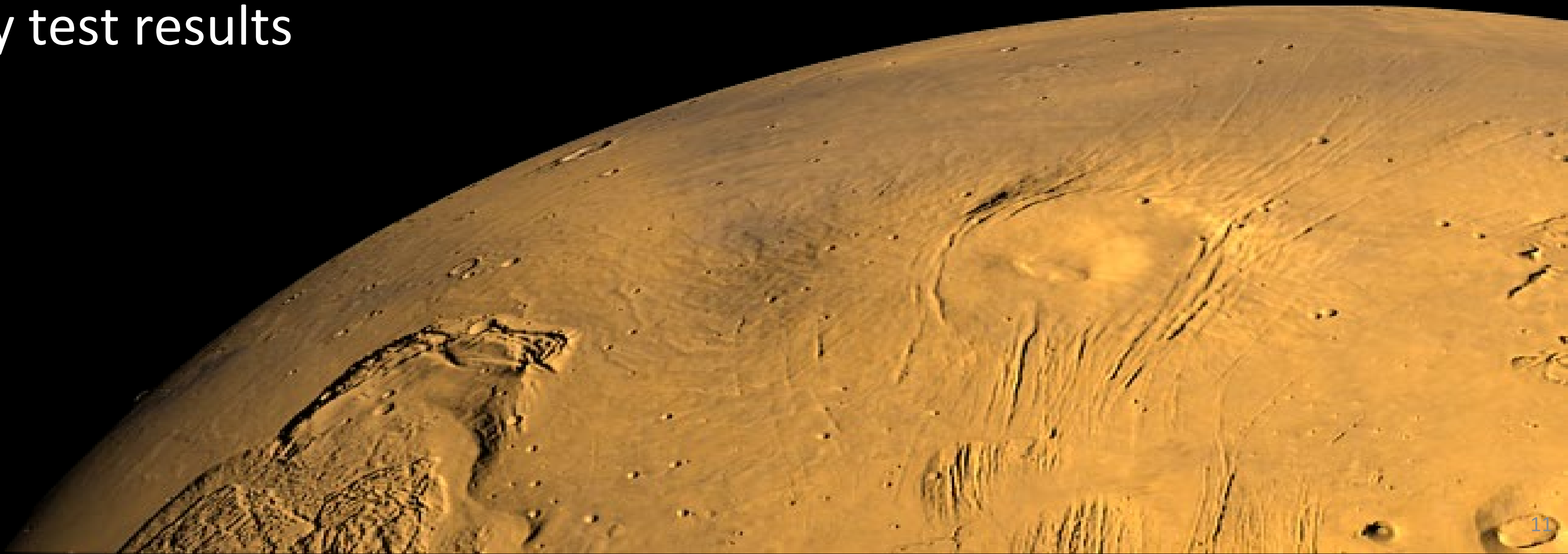
- The ultimate product will be a PDF that allows verifying CNA performance against the requirements.



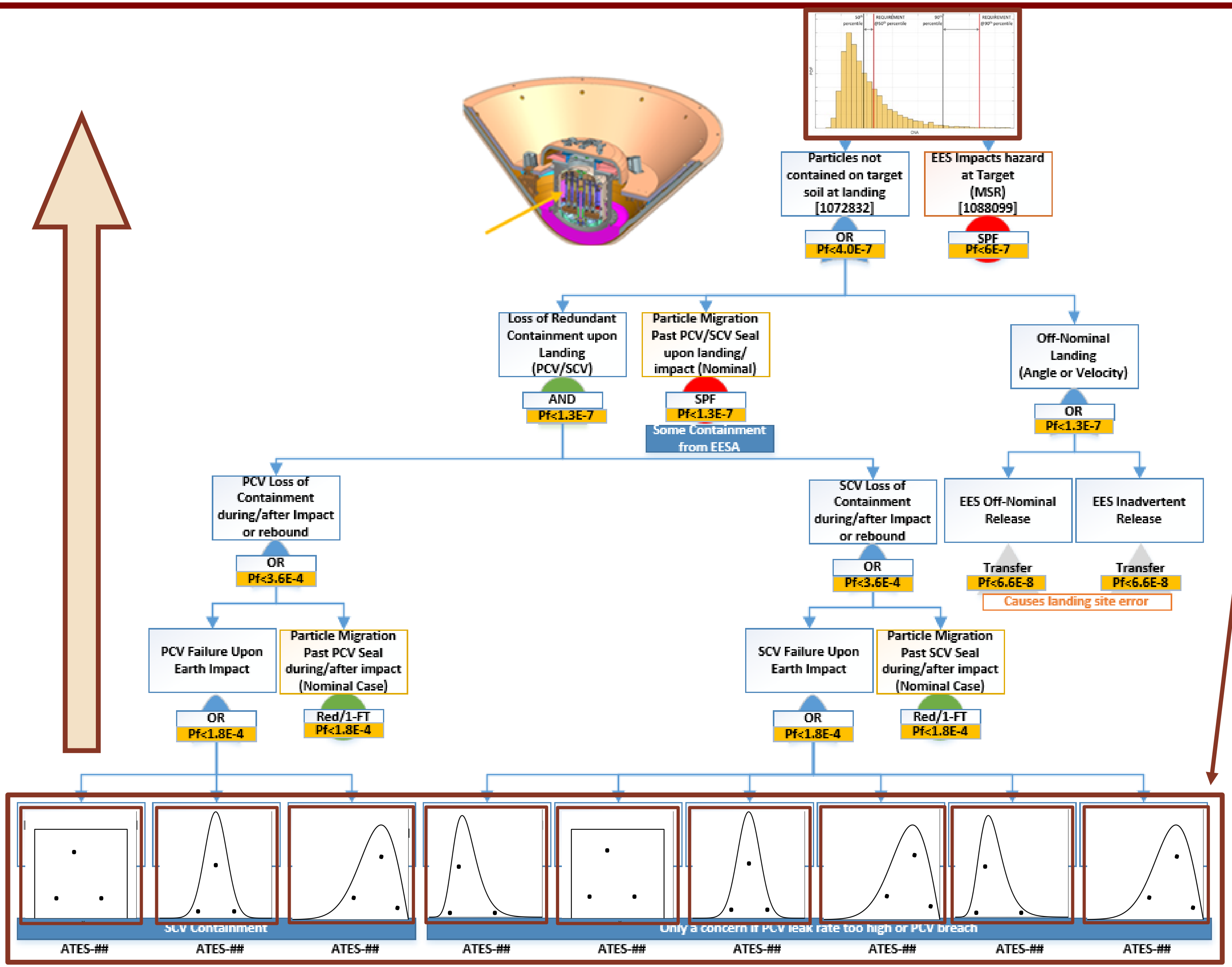
- The following slides will show the **computational methods** proposed to calculate the required (very low) probabilities at the bottom of the fault tree, as well as how to perform the roll-up.
- As an example to guide the reader through the process, a landing model that was used as a pathfinder will be briefly described and results from it will be shown.

CARF Implementation

Modeling and simulation strategies
with preliminary test results



Fault tree analysis – General process



1. Define CNA metrics.
2. Generate distributions at the bottom associated with such metrics.
3. Sample from them.
4. "Add/multiply" as prescribed by fault logic.
   – More mathematical details in following slides.

NOTE: Probability density functions are expected to be different in nature (i.e., non-Gaussian). Their combinations will lead to non-analytical forms that will require advanced sampling techniques, e.g., Markov Chain Monte Carlo.

STEP 1 – Define CNA metrics

IDENTIFY

Quantities of interest associated with specific CNA criteria

–

SPECIFY

Limit states Points or probability distributions beyond which CNA occurs

=

COMPUTE

Containment Not Assured (CNA) metrics

Quantity of interest (QoI)
Peak acceleration, a
Stress, σ
Pitch attitude, β

–

Limit state (LS)
3000 G
Max strength, σ_{max}
90°

=

CNA Metric = QoI – LS
a – 3000 G
$\sigma - \sigma_{max}$
β – 90°

DEFINE

CNA:
CNA metric ≥ 0
(Quantities of interest exceed limit states)

COMPUTE

Probability of CNA
P(CNA)

- Some of these metrics may be defined in terms of:
  - Point estimates (literature data, vendor manuals, etc.)
    - Failure rates
  - Probability density functions (model outputs, design of experiments)

- Statistics & UQ do the rest!

STEP 2a – Generate CNA estimates

- Data readily available:
  1. Literature data
  2. Vendor manuals/datasheets
  3. Previous missions
  4. Tests/Experiments
  5. Etc.

Reliability Report
OMH Series

TT electronics
OPTEK Technology

MTBF Values of Hall-Effect sensor

Summary

The OPTEK Quality Assurance System provides a means to monitor, control and correct product quality on a real-time basis. Each product family that OPTEK produces is tested extensively prior to manufacturing release for quality and reliability. OPTEK's ongoing commitment to quality and product improvement ensures that reliability will be maintained throughout the product's life cycle.

OPTEK requires standard products to complete 1,008 hours of accelerated life test at 70°C or higher with maximum rated operating current and voltage. One hundred sixty-two OMH series Hall-Effect sensors completed 1,008 hours of high temperature operating life test at 127°C with I_{CC} = 3.5 mA without failing. MTBF is generated based on data gathered during operating life test.

Demonstrated performance

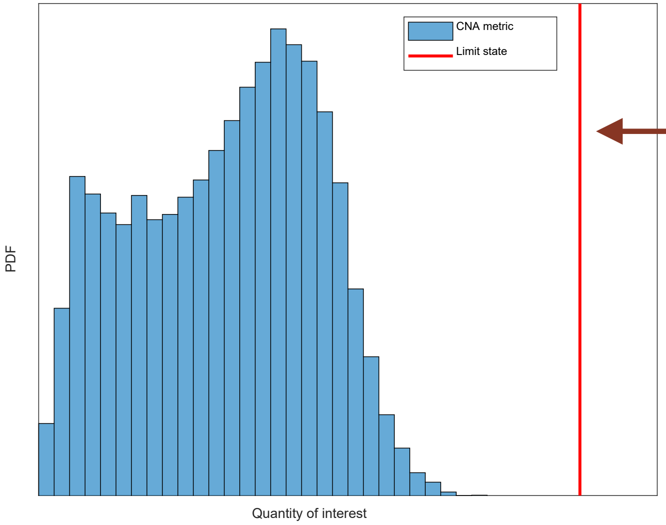
Test	Conditions	Total units tested	Total device hours	Failures[1]
High Temperature Operating life (HTOL)	T_A = 25°C, I_{CC} = 25 mA, V_{CC} @ 24 V	162	163,296	0

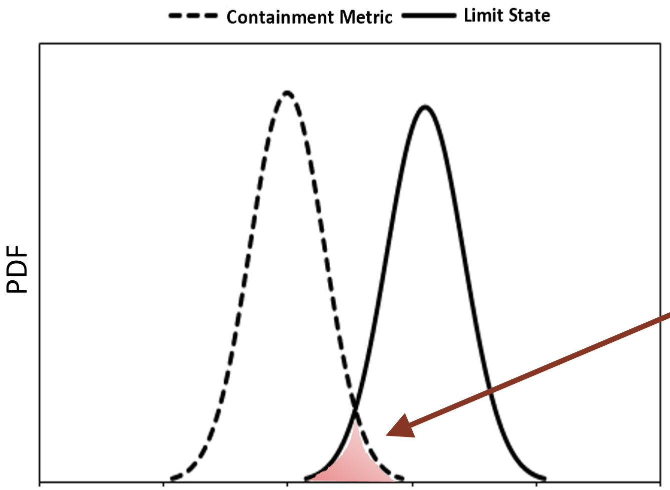
Projected MTBF and FIT of OMH Series operated with I_{CC} = 25 mA DC

Ambient temperature	60% confidence		90% confidence	
	MTBF[2]	FIT[3]	MTBF	FIT
(°C)	(hours)	(per 10^9 hours)	(hours)	(per 10^9 hours)
85	128,648	7,773	51,347	19,475
75	192,021	5,208	76,642	13,048
50	582,553	1,717	232,516	4,301
40	954,309	1,048	380,896	2,625
30	1,615,069	619	644,626	1,551
25	2,129,085	470	849,787	1,177
22	2,524,334	396	1,007,543	993
15	3,808,003	263	1,519,897	658

Notes:

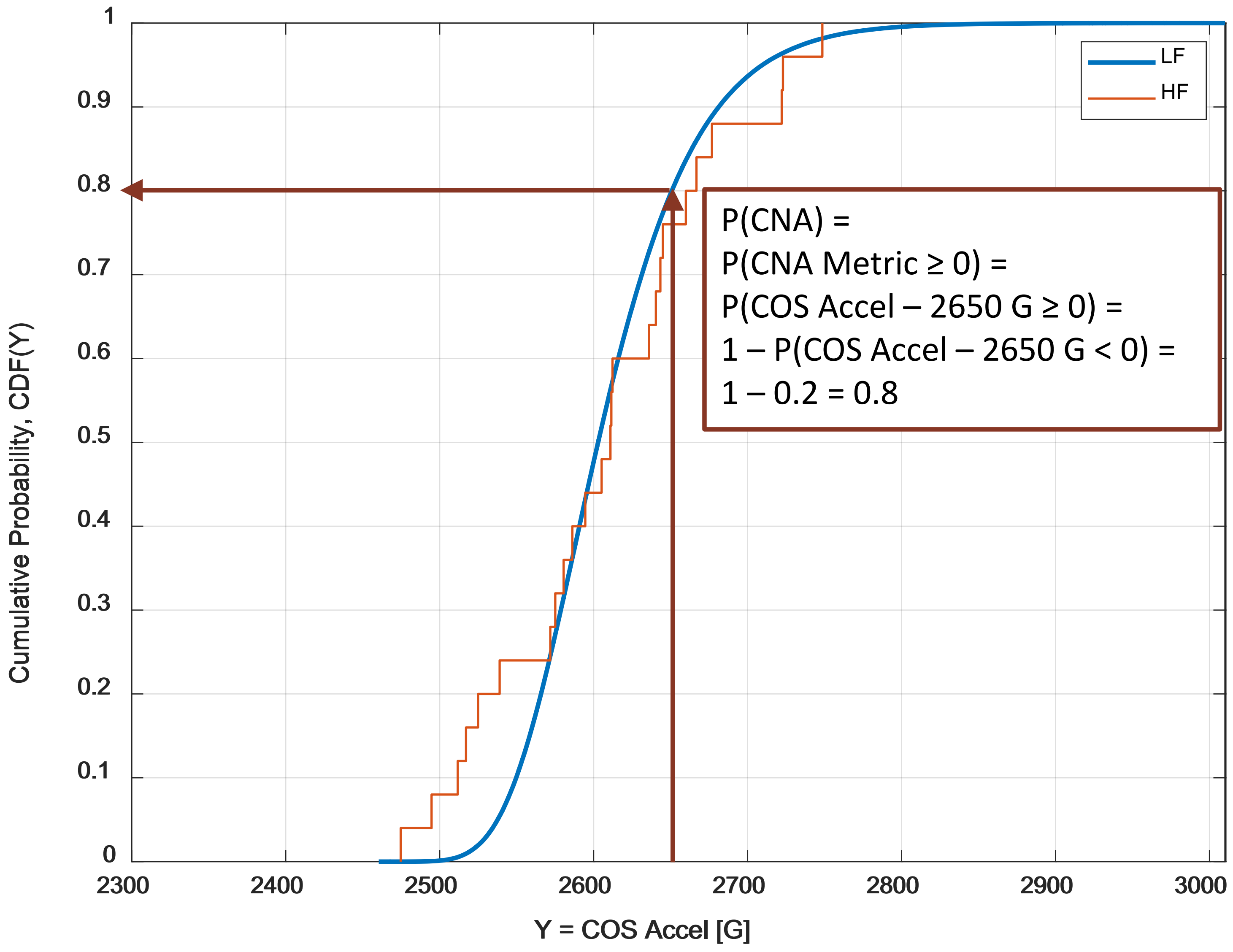
1. End point failure criteria is catastrophic failure or if coupled sensitivity changes by greater than 50%.
2. Mean time between failures (MTBF) is the projected number of operating hours before the first failure and between failures.
3. Failure in time (FIT) is the number of failures expected in one billion device hours. 1 FIT = 1 failure per 1,000,000,000 device-hours.





- Data not readily available:
  - Collect test hardware data
  - Use mathematical models
    - If the limit state is a **point estimate**, use one of the following:
      1. Cumulative distribution functions
      2. Monte Carlo
      3. Importance sampling
      4. Subset simulations
      5. First/Second-order reliability methods (FORM/SORM)
      6. Multifidelity importance sampling
    - If the limit state is a **probability distribution**:
      1. Design of experiments with synthetic data and overlap area

STEP 2a – Point estimates (1): Cumulative Distribution Functions



EXAMPLE

Peak landing acceleration > set critical value

The Cumulative Distribution Functions (CDFs) shown in this plot is come from a high-fidelity (HF) and a low-fidelity (LF) model.

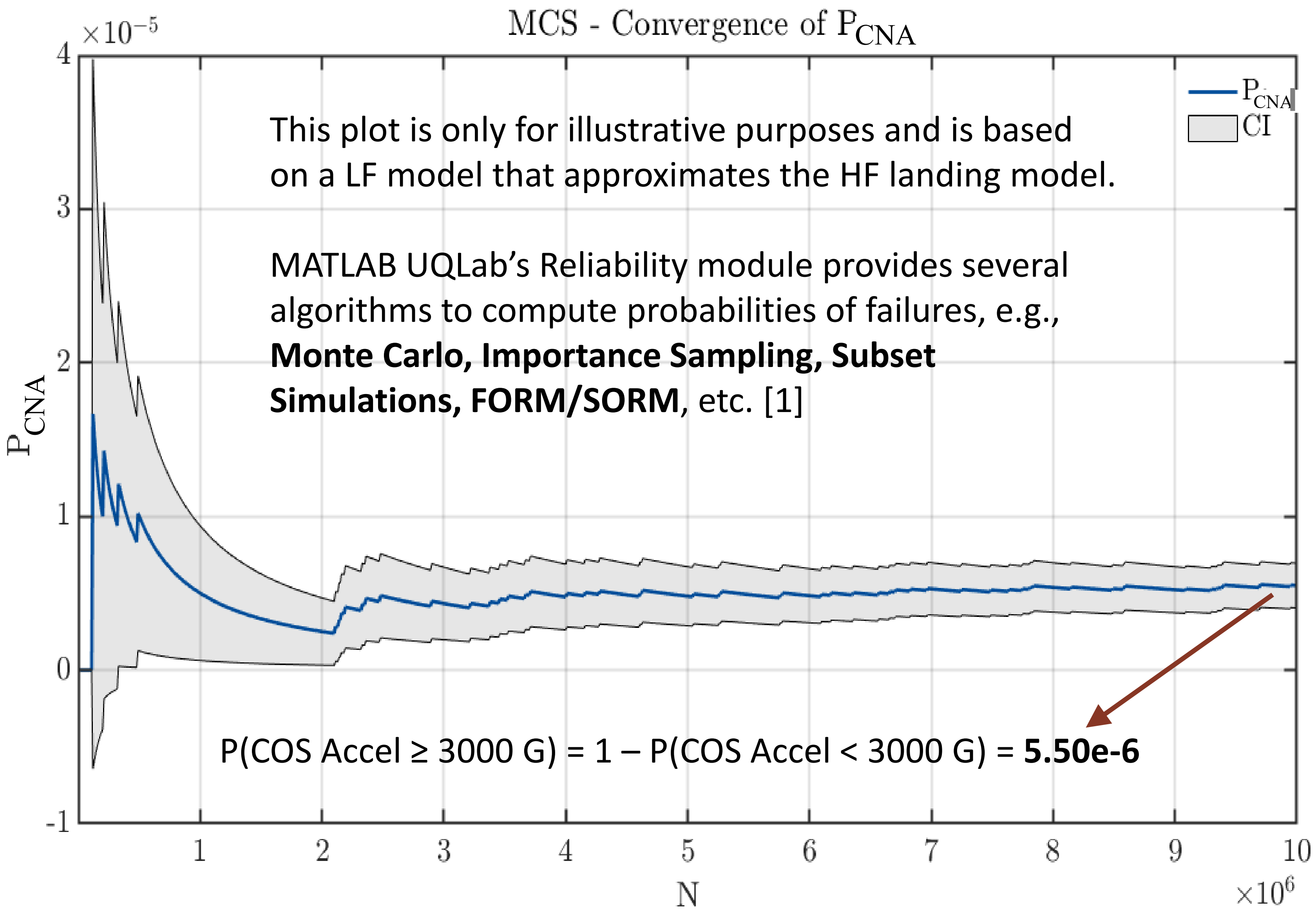
- CDFs from expensive **HF model** with few runs have **very poor resolution**, especially in the tails.
- CDFs from **LF models** may be **smooth** but suffer from **additional uncertainties** due to the models' low fidelity.

TAKEAWAY:

The CDF method is to be used only with **cheap HF models with run times ≲ 10 s***.

*1-s run time = 86,400 runs/day

STEP 2a – Point estimates (2): Monte Carlo and others



This plot is only for illustrative purposes and is based on a LF model that approximates the HF landing model.

MATLAB UQLab's Reliability module provides several algorithms to compute probabilities of failures, e.g., **Monte Carlo, Importance Sampling, Subset Simulations, FORM/SORM**, etc. [1]

P(COS Accel ≥ 3000 G) = 1 – P(COS Accel < 3000 G) = **5.50e-6**

Regular Monte-Carlo approaches may require thousands/millions of runs before reaching convergence, which is impossible with expensive HF models.

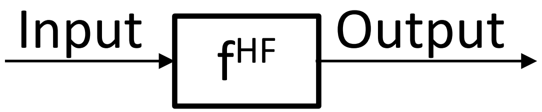
Using LF models implies working with additional uncertainties due to the models' low fidelity.

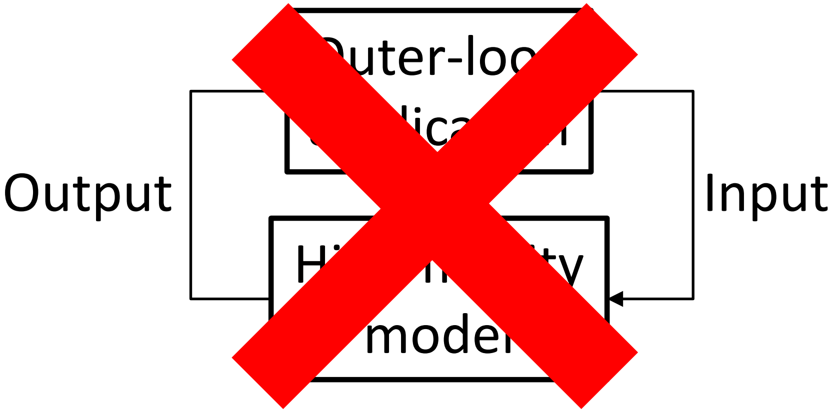
- **Challenge**: it only works with HF models (expensive) or LF models (biased, inaccurate).
- Focus should be on improving knowledge of CDF tails through a **multifidelity approach** (faster and more accurate) as shown on the next slide.

[1] S. Marelli, R. Schobi, B. Sudret, UQLab user manual – Structural reliability (Rare event estimation), Report #UQLab-V1.4-107, Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland, 2021

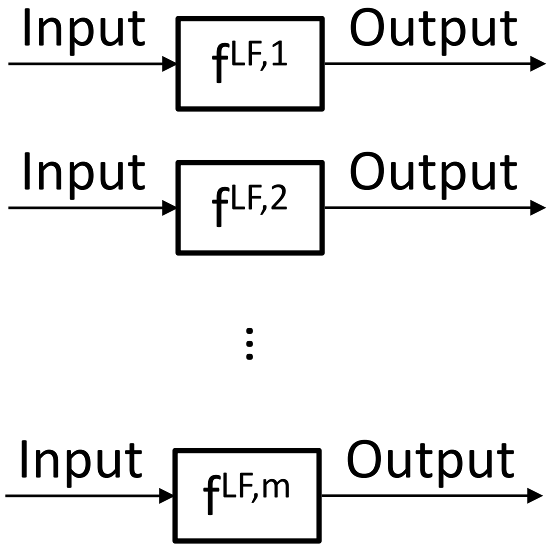
Multifidelity approach to computationally expensive models

High-fidelity (HF) model

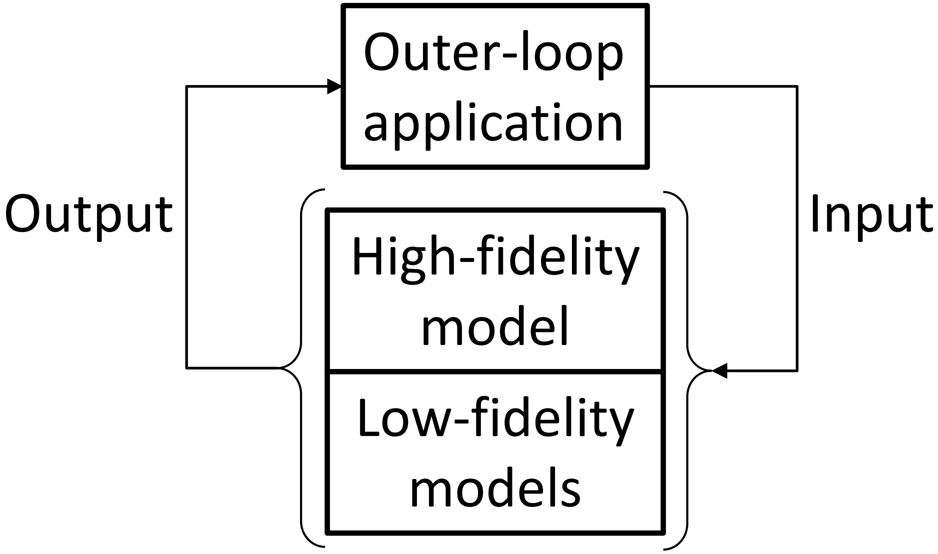




Low-fidelity (LF) models



- Invoke multiple models to reduce computational cost
- Maintain accuracy guarantees on outer-loop results



Multifidelity approach to computationally expensive models

Simplified models
Early-stopping criteria
Natural problem hierarchies (e.g., in CFD: DNS -> LES -> RANS)
Coarse-grid approximations
Linearized models

Projection-based models
Proper Orthogonal Decomposition (POD)
Reduced-basis method
Krylov subspace methods

Data-fit models
Interpolation • Polynomial interpolation • Radial basis functions • Kriging
Regression • Polynomial Chaos Expansion • Support Vector Machines

Information sources beyond models
Expert opinions
Experimental data
Historical data

- Low-fidelity (LF) models should be of **different type**, **approximation quality** and **cost** to inform different aspects of the HF model.
- This is better than using only LF models that are accurate but lack a rich diversity [2,3].
- Successfully applied to JWST [4].

[2] Ng, Willcox, Multifidelity approaches for optimization under uncertainty, *Int. J. Numer. Meth. Engng.* 2014; **100**:746–772
[3] Peherstorfer, Willcox, Gunzburger, Optimal model management for multifidelity Monte Carlo estimation, *SIAM J. Sci. Comput.* 2016; **38**(5):A3163-A3194
[4] Cataldo, Qian, Auclair, Multifidelity uncertainty quantification and model validation of large-scale multidisciplinary systems, *J. Astron. Telesc. Instrum. Syst.*, **8**(3), 038001 (2022)

STEP 2a – Point estimates (3): Multifidelity Importance Sampling

IMPORTANCE SAMPLING

Variance reduction strategy for Monte Carlo estimation
that samples from a problem-dependent biasing distribution

- TWO STEPS:
  1. The biasing distribution is constructed.
  2. Samples are drawn from the biasing distribution and the estimate is derived [6, 38].

Chosen such that **fewer samples** are necessary to obtain an acceptable estimate of the rare event probability than with standard Monte Carlo

The bias introduced by the sampling from the biasing distribution is **corrected** by reweighing the samples in the importance sampling estimator.

CHALLENGE:
Construction of a **suitable biasing distribution** that leads to **variance reduction**

SOLUTION:
MULTIFIDELITY PRECONDITIONING OF THE CROSS-ENTROPY METHOD

by

[5] Peherstorfer, Kramer, Willcox, Multifidelity Preconditioning of the Cross-Entropy Method for Rare Event Simulation and Failure Probability Estimation, *J. Uncertain. Quant.* 2018; **6**(2):737–761

STEP 2a – Point estimates (3): Multifidelity Importance Sampling

Cross-entropy (CE) method

Provides a practical way to approximate the zero-variance density

HOW? By optimizing for a density that minimizes the Kullback-Leibler divergence from the zero-variance density in a set of feasible densities

CHALLENGE:

Even though solving for a biasing density with the CE method typically requires fewer high-fidelity model evaluations than estimating the rare event probability with a standard Monte Carlo approach,
the cost of the optimization problem in the CE method can still be significant if the high-fidelity model is expensive to evaluate

SOLUTION:

Multifidelity method that leverages a hierarchy of low-cost, low-fidelity models to reduce the costs of constructing a CE-optimal biasing density

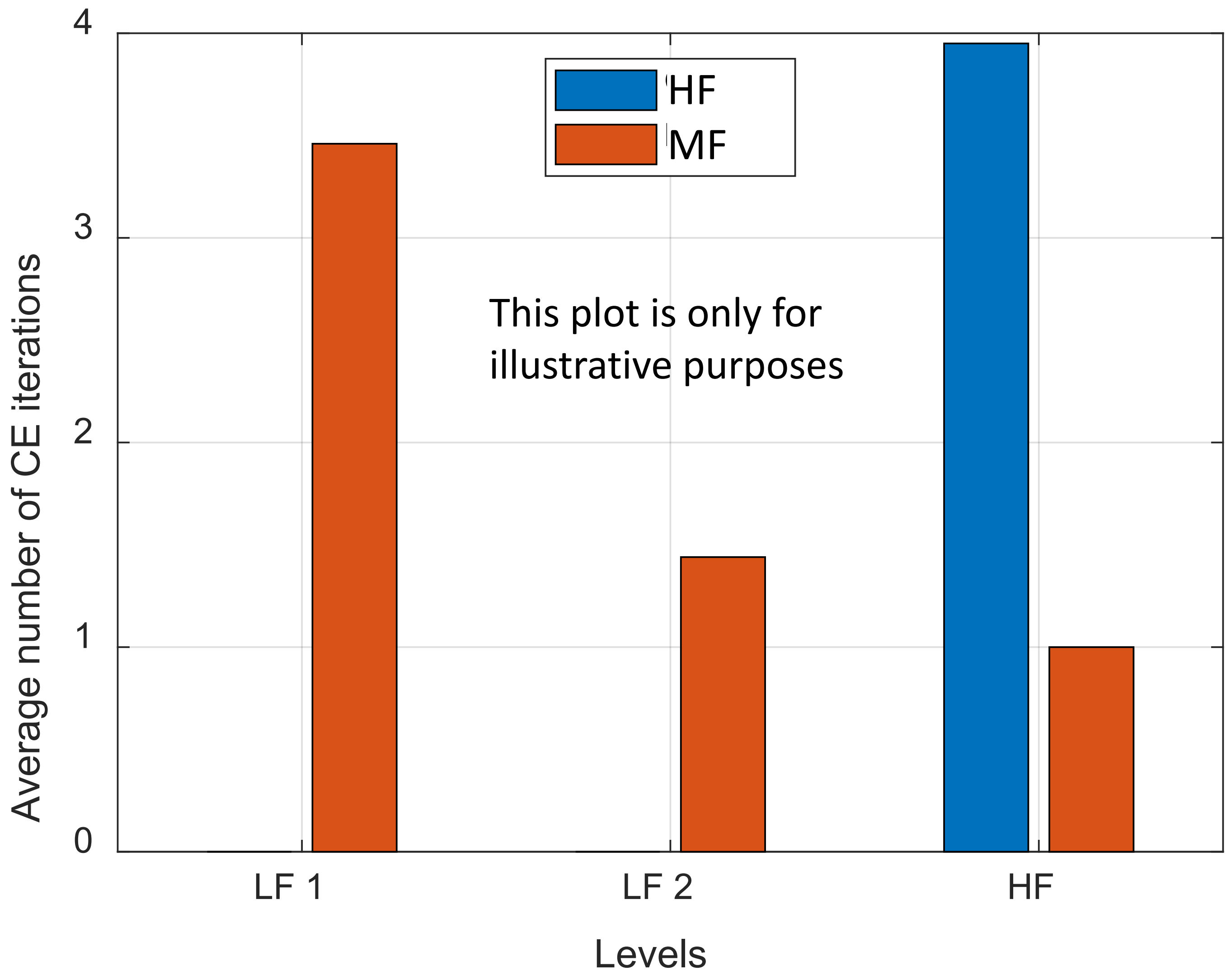
[5] Peherstorfer, Kramer, Willcox, Multifidelity Preconditioning of the Cross-Entropy Method for Rare Event Simulation and Failure Probability Estimation, *J. Uncertain. Quant.* 2018; **6**(2):737–761

STEP 2a – Point estimates (3): Multifidelity Importance Sampling

- The bound on the number of iterations of the CE method depends on the density with which the CE method is initialized.
- The method developed in [5] exploits the fact that the bound on the number of CE iterations can be reduced by a suitable choice of the density with which the CE method is initialized.
- The MFCE method iterates through the different models (1,…,L).
  - L identifies the high-fidelity model.
  - At level 1, the MFCE method constructs a biasing density with the classical CE method, initialized with the nominal density and using low-fidelity model 1.
  - At level 2, the MFCE method uses the CE method to derive a density with model 2.
    - However, the CE method is initialized with the density of the previous level, instead of the nominal density as in the classical CE method.
  - This hierarchical process is continued until level L, where the final estimate uses the high-fidelity model.
- Numerical examples show ability to estimate probability of failures as low as 10^{-9}!

[5] Peherstorfer, Kramer, Willcox, Multifidelity Preconditioning of the Cross-Entropy Method for Rare Event Simulation and Failure Probability Estimation, *J. Uncertain. Quant.* 2018; **6**(2):737–761
And references therein.

STEP 2a – Point estimates (3): Multifidelity Importance Sampling



P(COS Accel ≥ 3000 G) = **8.1062e-6 ± 2.0331e-5**

Convergence achieved with 1 HF model, 2 LF models and **only 6 total iterations**!

Multifidelity importance sampling uses both HF/LF models to recover probability of failures as low as 1e-9! [5]

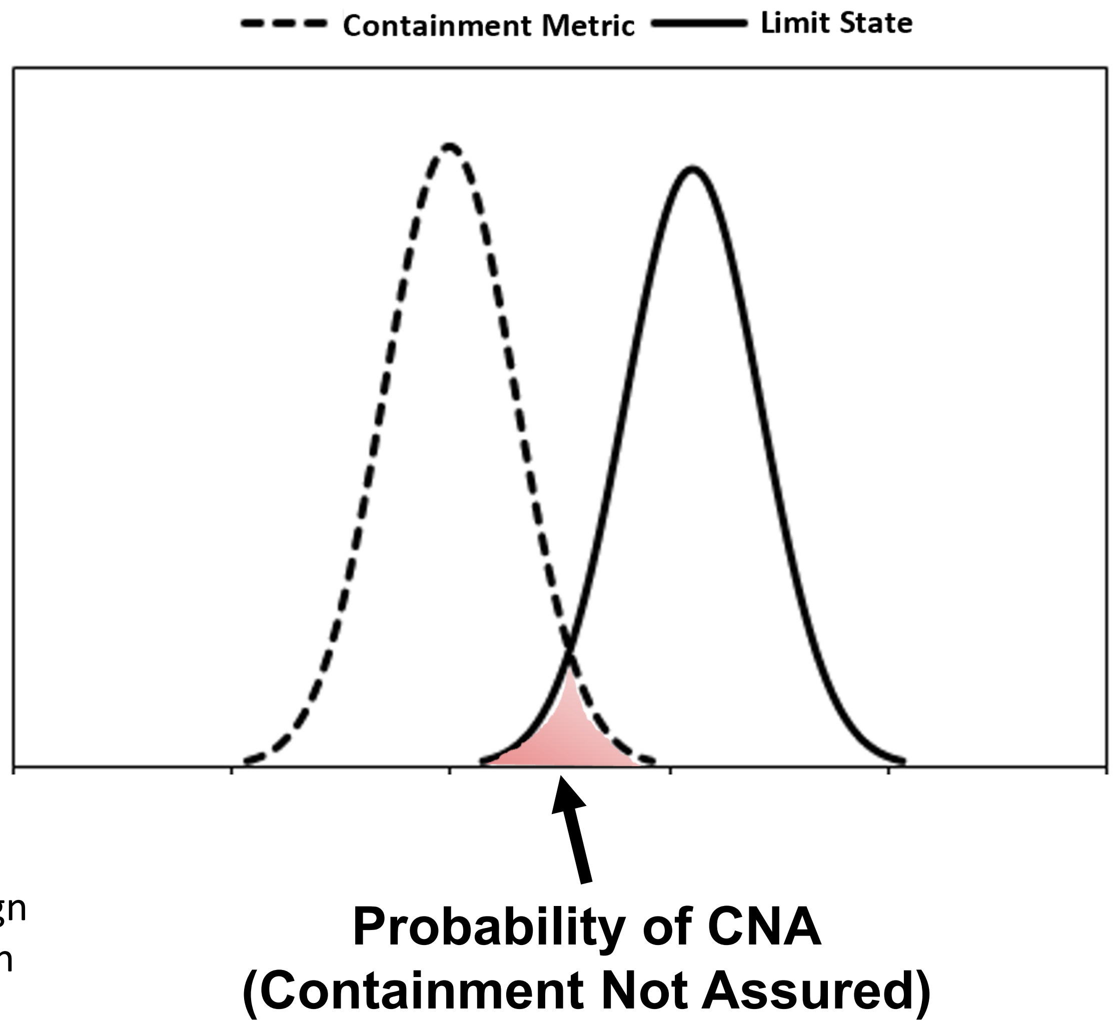
The result accuracy is guaranteed by keeping the HF model in the loop.

PROPOSED METHODOLOGY
for expensive models
with run times ≳ 1 min

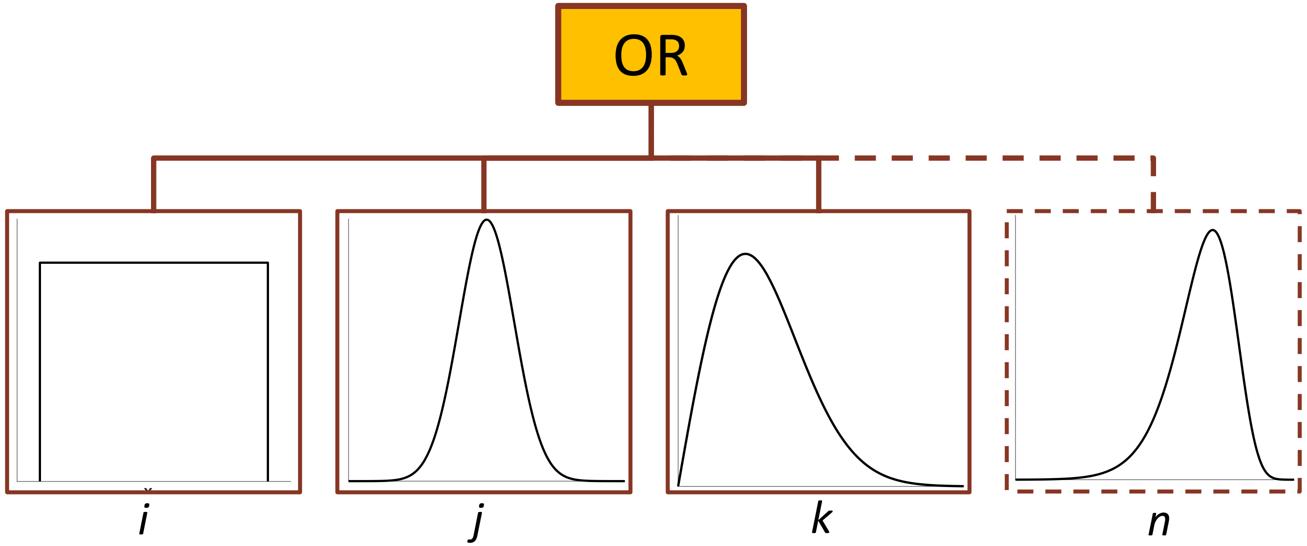
[5] Peherstorfer, Kramer, Willcox, Multifidelity Preconditioning of the Cross-Entropy Method for Rare Event Simulation and Failure Probability Estimation, *J. Uncertain. Quant.* 2018; **6**(2):737–761

STEP 2b – Generate input data for distributions

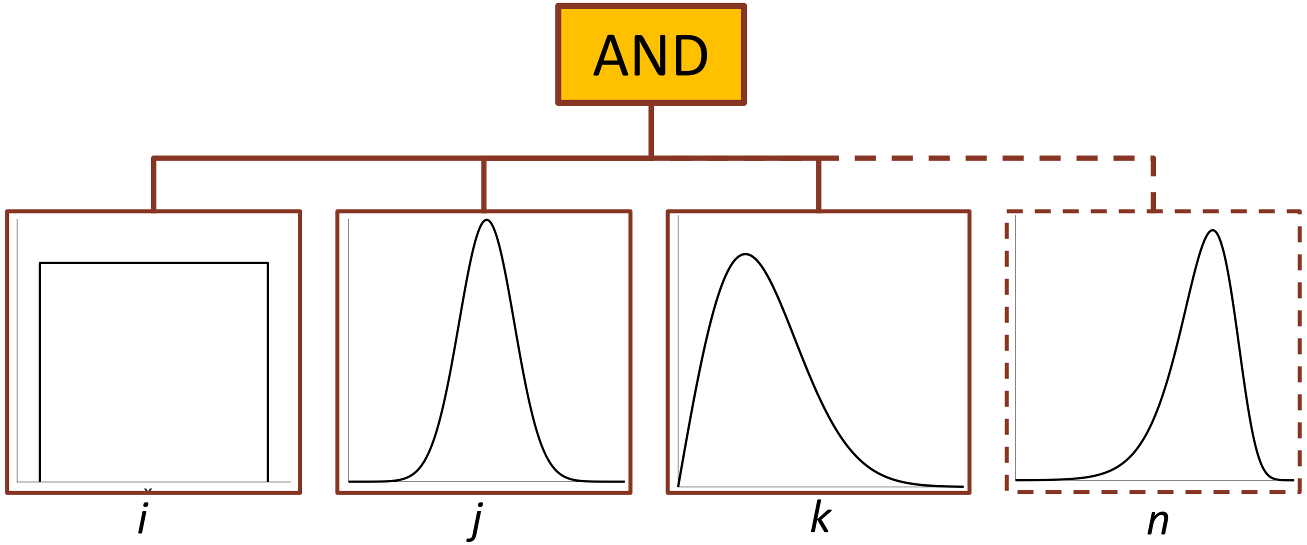
- Design of experiments with numerical models or test hardware
  - Monte Carlo
  - Latin Hypercube
  - Quasi-random
    - Halton
    - Sobol
  - Response surface
    - Box-Behnken
    - Central composite
  - D-Optimal
    - Seeks to minimize the covariance of the parameter estimates for a specified model
    - Good for non-linear models, constrained factors, correlated parameters
  - Maximum projection
    - Maximizes the product of the distances between potential design points with the goal of providing good space-filling properties on projections of factors.
  - Etc.



STEP 3 – Combine all distributions

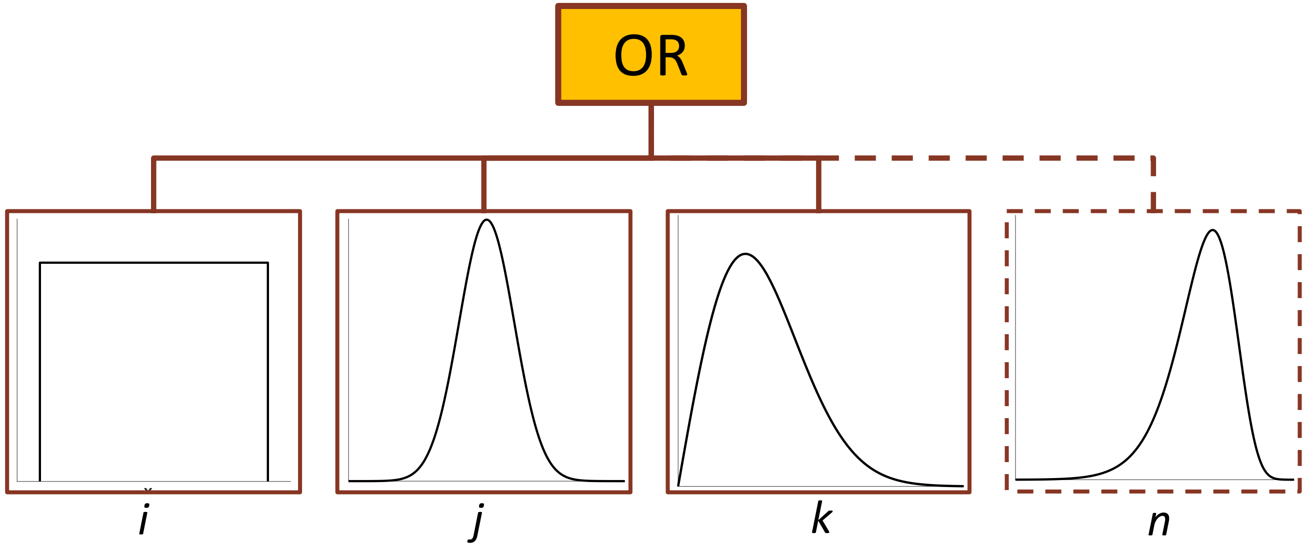


$$P\left(\bigcup_{i=1}^{n} A_i\right) = \sum_{i=1}^{n} P(A_i) - \sum_{i<j} P(A_i \cap A_j) + \sum_{i<j<k} P(A_i \cap A_j \cap A_k) - \ldots + (-1)^{n-1} \sum_{1<\ldots<n} P\left(\bigcap_{i=1}^{n} A_i\right)$$



$$P\left(\bigcap_{i=1}^{n} A_i\right) = P(A_1)P(A_2|A_1)P(A_3|A_2 \cap A_1)\ldots \ldots P(A_n|A_{n-1} \cap \ldots \cap A_1)$$

STEP 3 – Combine all distributions

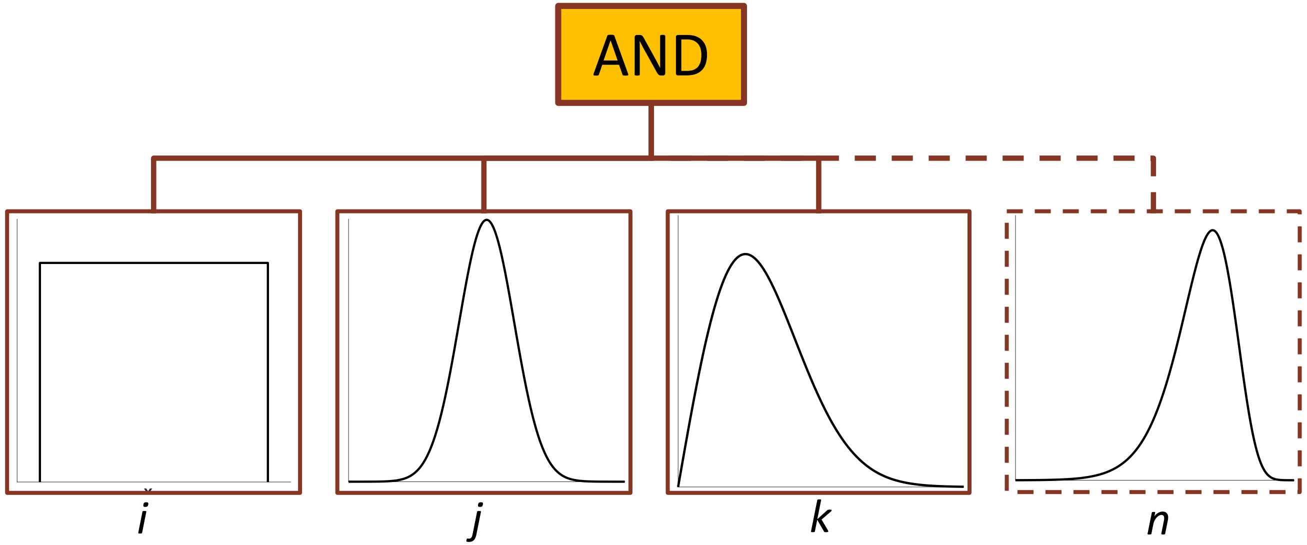


If mutually exclusive

$$P\left(\bigcup_{i=1}^{n} A_i\right) = \sum_{i=1}^{n} P(A_i) - \cancelto{0}{\sum_{i<j} P(A_i \cap A_j)}$$

$$+ \cancelto{0}{\sum_{i<j<k} P(A_i \cap A_j \cap A_k)} - \ldots$$

$$+ \cancelto{0}{(-1)^{n-1} \sum_{1<\ldots<n} P\left(\bigcap_{i=1}^{n} A_i\right)}$$

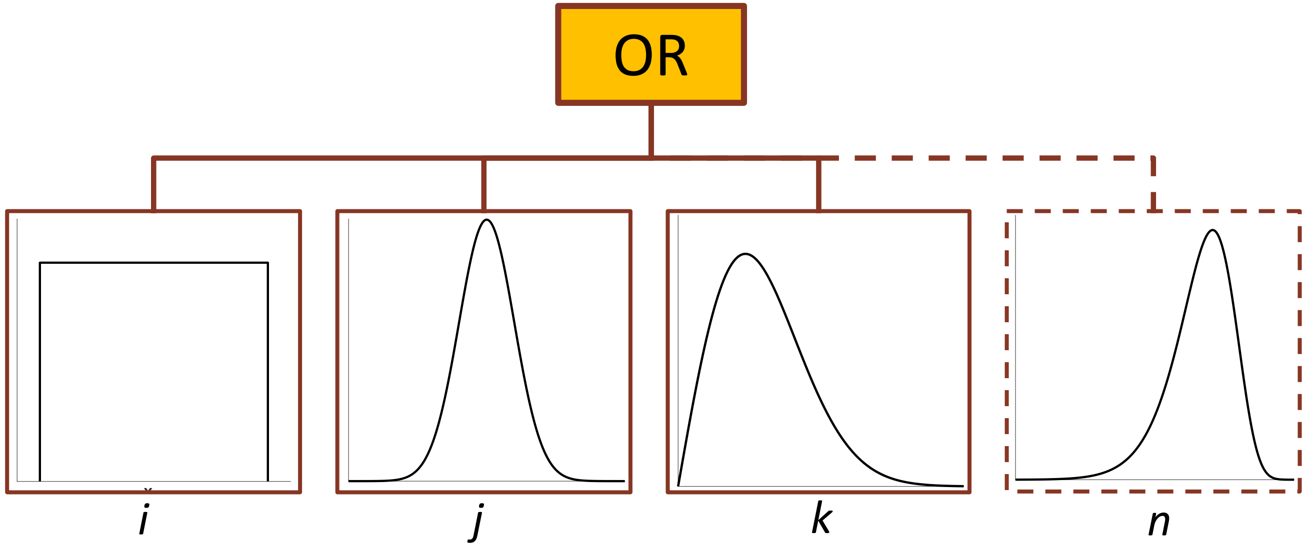


If independent

$$P\left(\bigcap_{i=1}^{n} A_i\right) = P(A_1)P(A_2|\cancel{A_1})P(A_3|\cancel{A_2 \cap A_1})\ldots$$

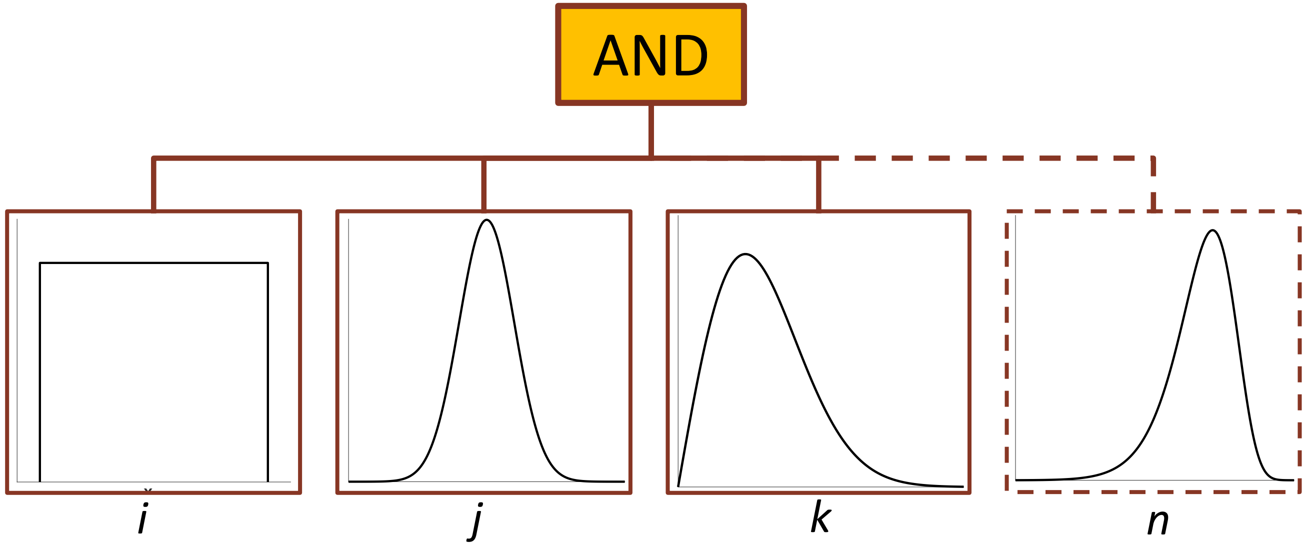
$$\ldots P(A_n|\cancel{A_{n-1} \cap \ldots \cap A_1})$$

STEP 3 – Combine all distributions



If mutually exclusive

$$P\left(\bigcup_{i=1}^{n} A_i\right) = \sum_{i=1}^{n} P(A_i)$$



If independent

$$P\left(\bigcap_{i=1}^{n} A_i\right) = \prod_{i=1}^{n} P(A_i)$$

CARF – First demonstration

- Propagation of CNA probabilities through requirements tree
  - Leaf nodes (green hexagons) represent basic events
  - Distribution of leaf-node success probabilities from stochastic simulations using physical models
  - Compute distributions of success probabilities of interior-nodes' (blue) events sequentially down the tree to the root

- Code written in **Julia** using Graphs.jl and Metagraphs.jl packages.
  - Create ProbabilityGenerator objects to model univariate or joint distributions for leaf nodes based on Monte Carlo results.
  - And/Or logic tests for independence among interior nodes, and manages dependence if needed.
  - Carries whole distributions (as histograms) down the tree, and can report any desired statistics of the distribution in the node label.

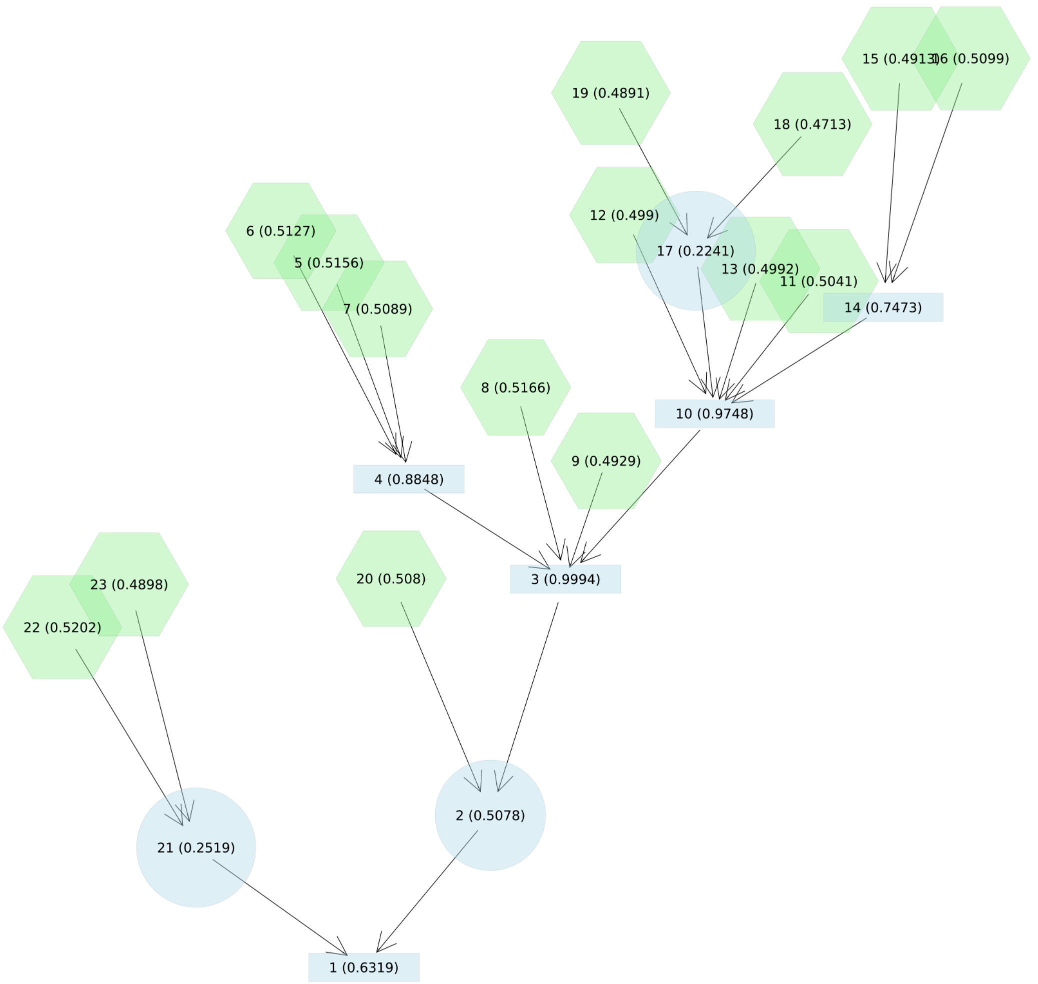
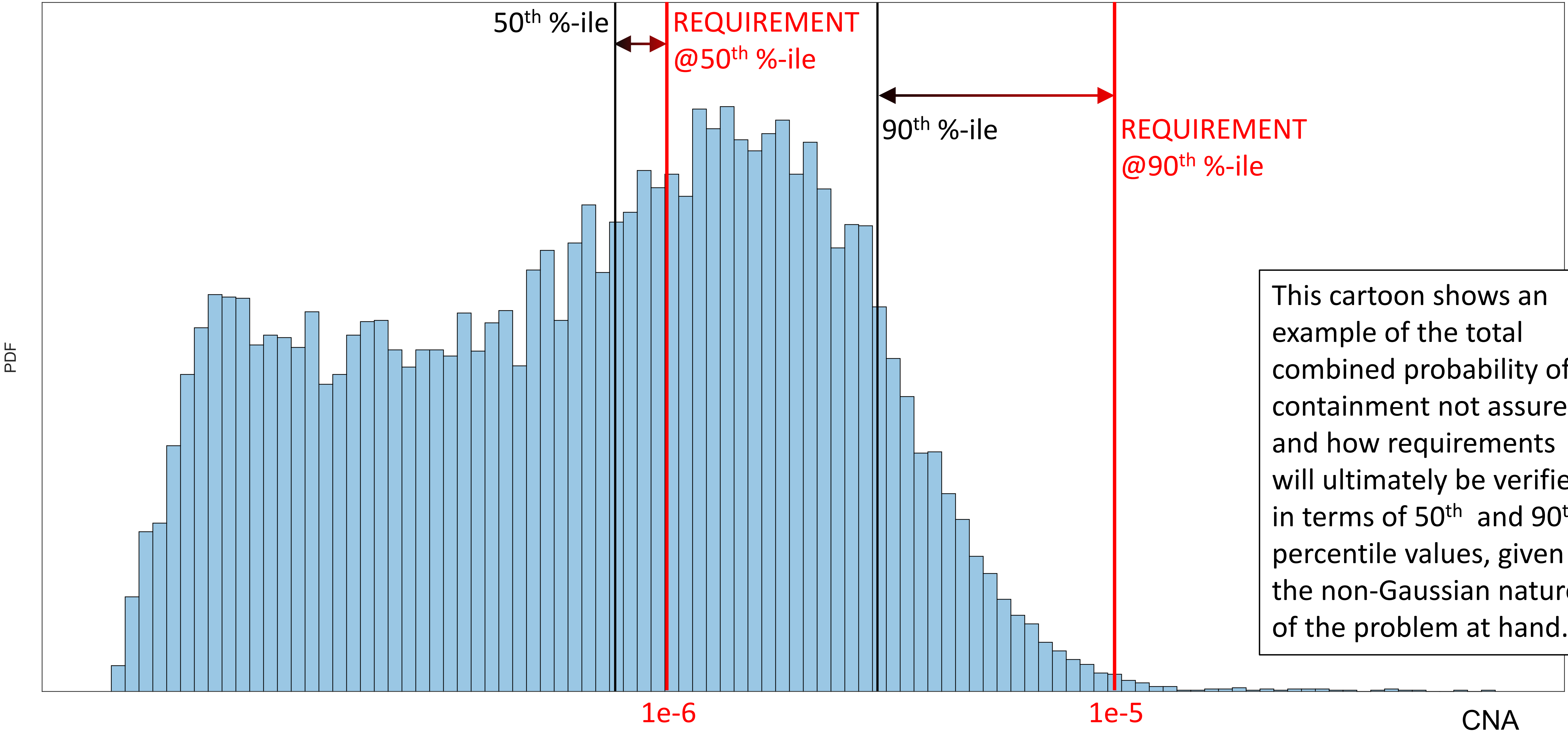


Figure: Small example tree with node labels as node identification number and mean of the distribution (in parentheses) of probabilities. Green hexagons are leaves; blue nodes are interior nodes. AND gates are circles, OR gates are rectangles.

Final product – Requirement verification



This cartoon shows an example of the total combined probability of containment not assured and how requirements will ultimately be verified in terms of 50th and 90th percentile values, given the non-Gaussian nature of the problem at hand.

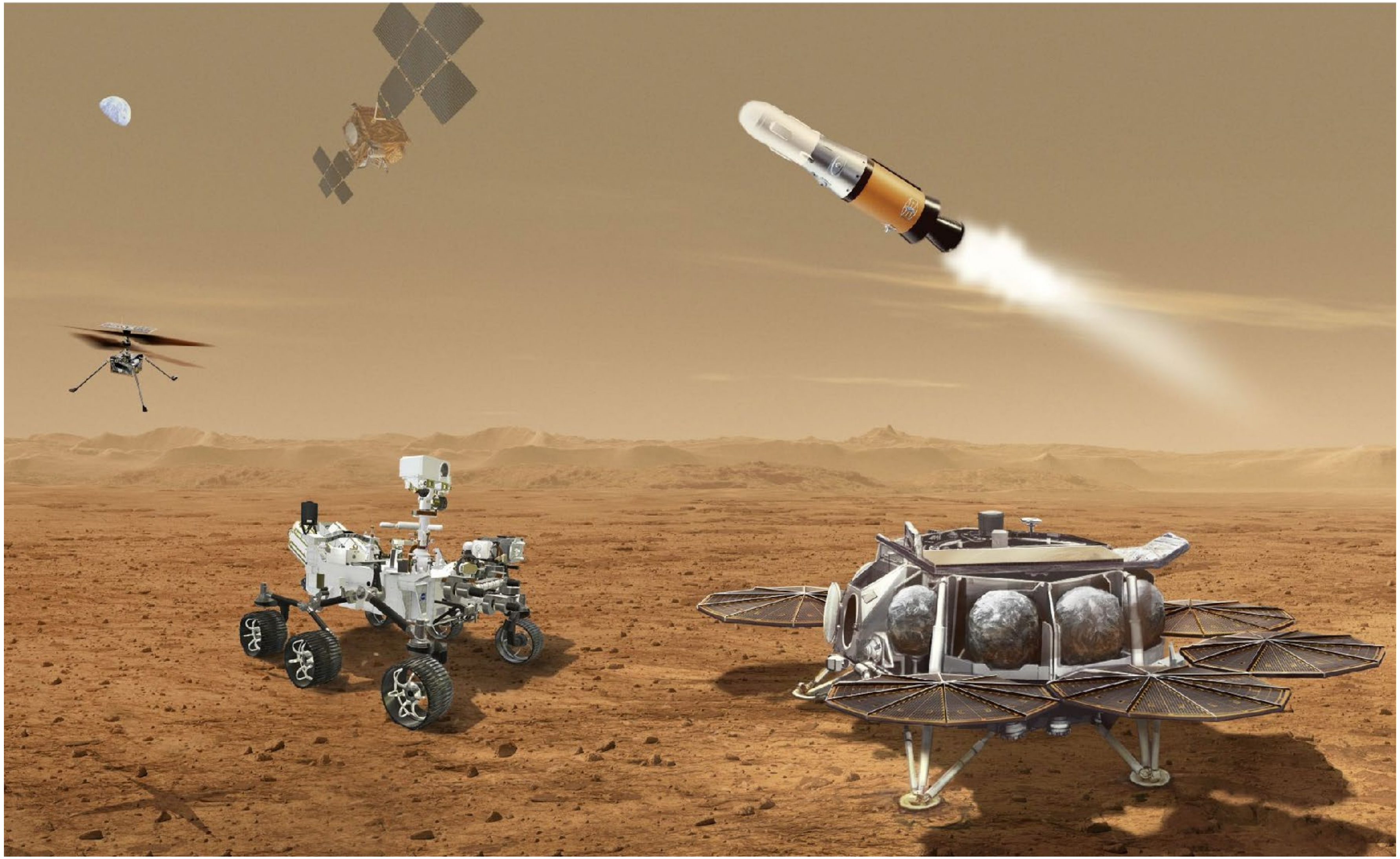
Conclusions

- To demonstrate **compliance** with the strict requirements of MSR, a statistical framework was developed to assess the likelihood of containment loss posed by each sample return phase and make a statement about the total combined mission probability of containment not assured.
- The framework considers failure modes or fault conditions that can initiate propagation sequences ultimately leading to a containment not assured.
- Given the multidisciplinary nature of the problem and the different types of mathematical models used, the statistical tools needed for analysis are required to be computationally efficient.
  - While standard Monte Carlo approaches are used for fast models, a **multi-fidelity approach to rare event probabilities** is proposed for expensive models. In this paradigm, inexpensive low-fidelity models are developed for computational acceleration purposes while the expensive high-fidelity model is kept in the loop to retain accuracy in the results.
  - This work presented an example of end-to-end application of this framework highlighting the computational benefits of a multi-fidelity approach and the overcoming of several complexities inherent to the computation of very low (<1e-6) probabilities.

Future work

- Update process to handle joint and conditional distributions
  - Both in design of experiments and fault tree “roll-up”
- Continue research on sampling techniques for extreme values
- Perform analysis with latest mathematical models and test data as the mission progresses toward more mature designs

Questions?



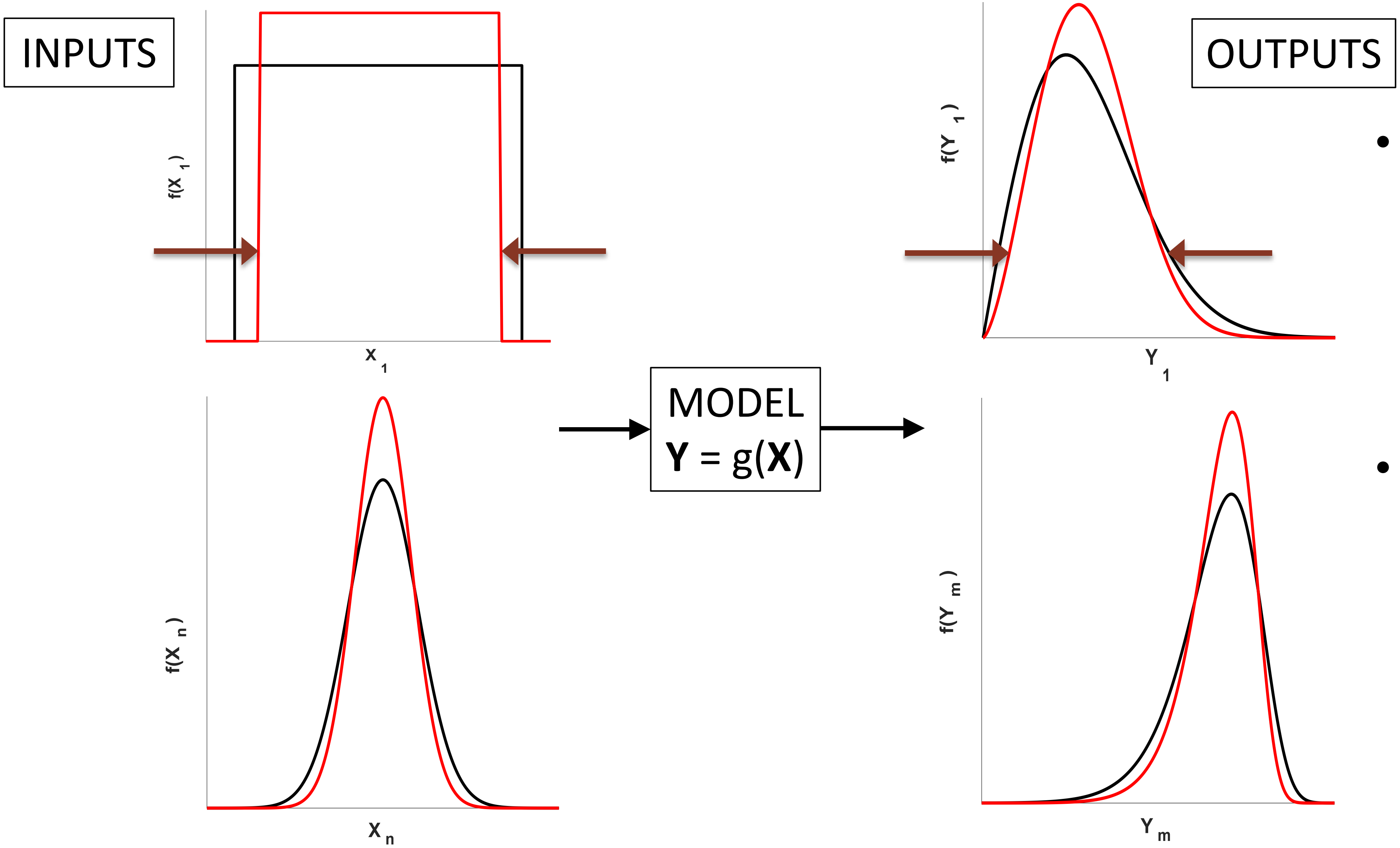
@GCataldoNASA

@g_cataldo

Giuseppe.Cataldo@NASA.gov

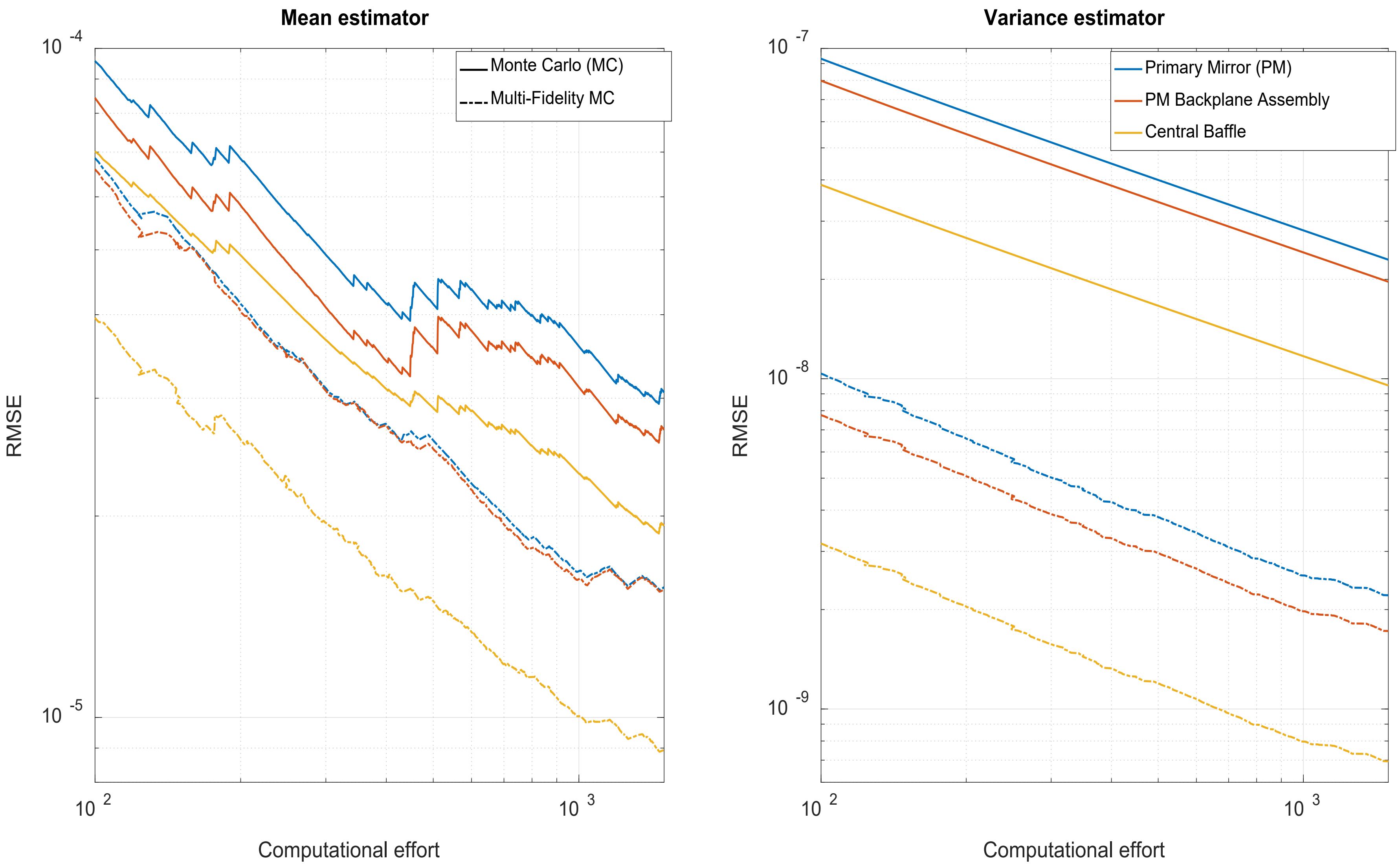
Back-up

Variance-based Sensitivity Analysis



- Determine the amount of variance that would disappear in the outputs if the uncertainty in one or more input parameters were reduced
- **Prioritize parameters** by identifying those which, on average, once reduced their uncertainty, would cause the **greatest reduction in the output variance**

Multifidelity approach to computationally expensive models



- A multifidelity (MF) approach leads to **more accurate results** and **cheaper computational costs**
- The example shown here is from JWST [5]
- What it takes:
  - MATLAB
  - UQLab toolbox (freely available) to develop low-fidelity models
  - mfGSA code (freely available) to run MF analysis
  - A number of samples from HF model

[4] Cataldo, Qian, Auclair, Multifidelity uncertainty quantification and model validation of large-scale multidisciplinary systems, *J. Astron. Telesc. Instrum. Syst.*, **8**(3), 038001 (2022).