RAVEN Introduction

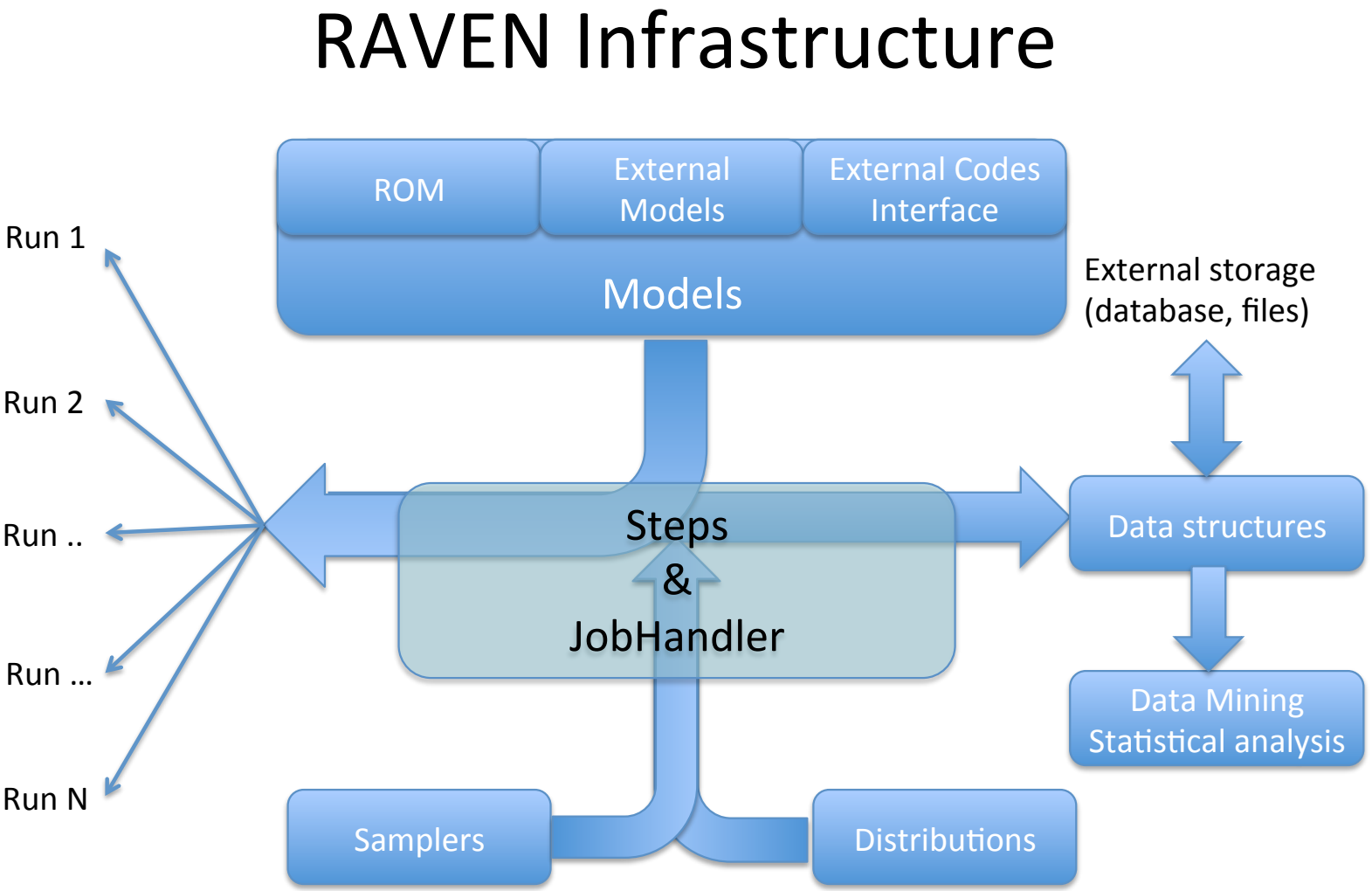
RAVEN Workshop

Presenter: Cristian Rabiti

Capabilities vs. Needs

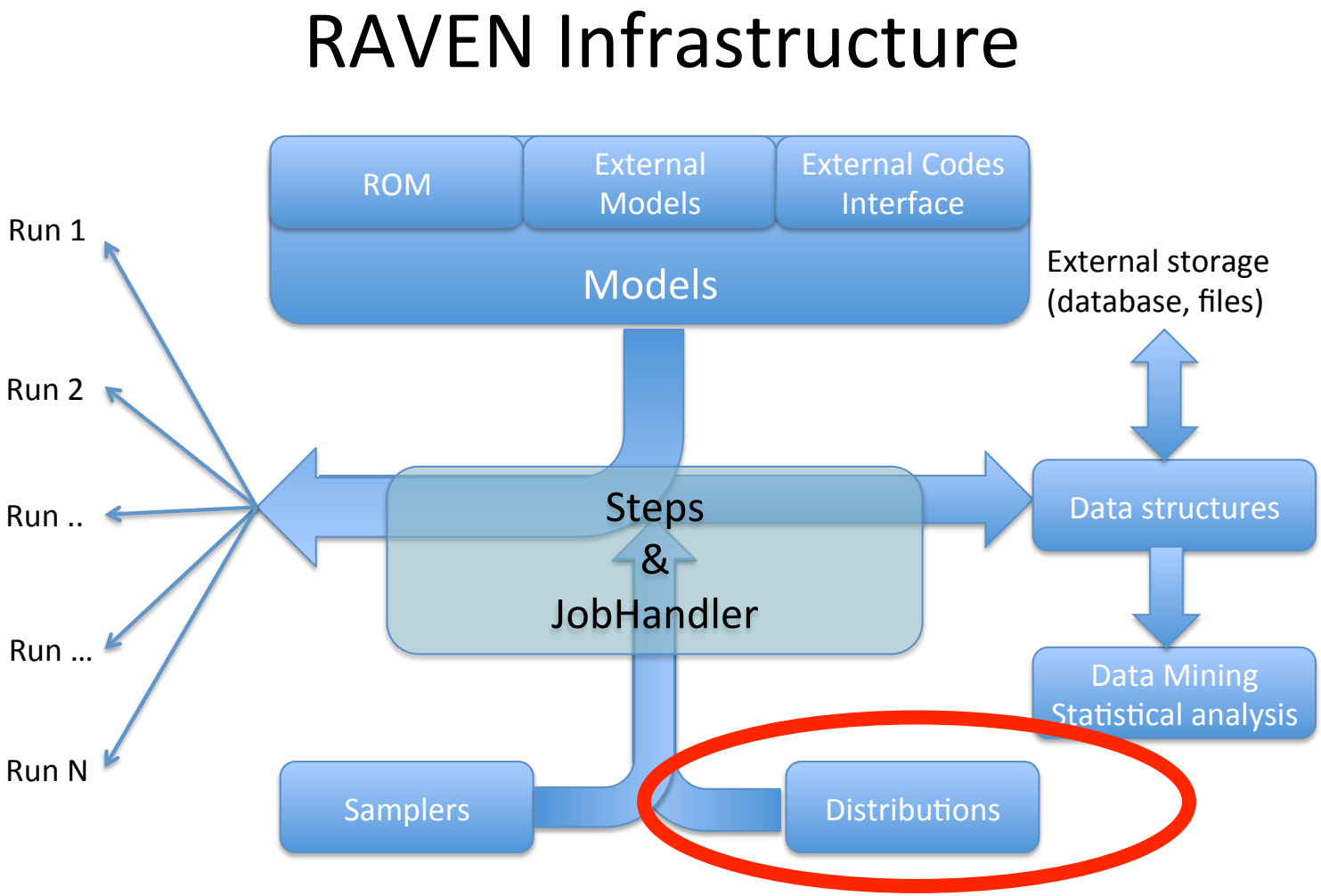
Algorithms	User Cases					
	Uncertainty Quantification	**Risk Analysis**	**Risk Management**	**Validation**	**Experiment design**	**Optimization**
Distribution	Yes	Yes	Yes	Yes	Yes	Yes
Sampling	Yes	Yes	Yes	Yes	Yes	Yes
Surrogate Models	Yes	Yes	Yes	Yes	Yes	Yes
Statistical post processing	Yes	Yes	Yes	Yes	Yes	Yes
Reliability Surface		Yes	Yes			
Data mining		Yes	Yes			
Analytical distribution comparison (ongoing)				Yes	Yes	
Minimization Algorithms (ongoing)			Yes		Yes	Yes

RAVEN Structure



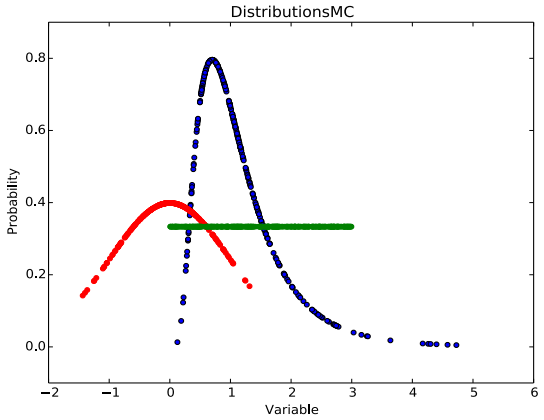
Distributions

- Distribution describe the statistical behavior of input parameters



Standard Distributions

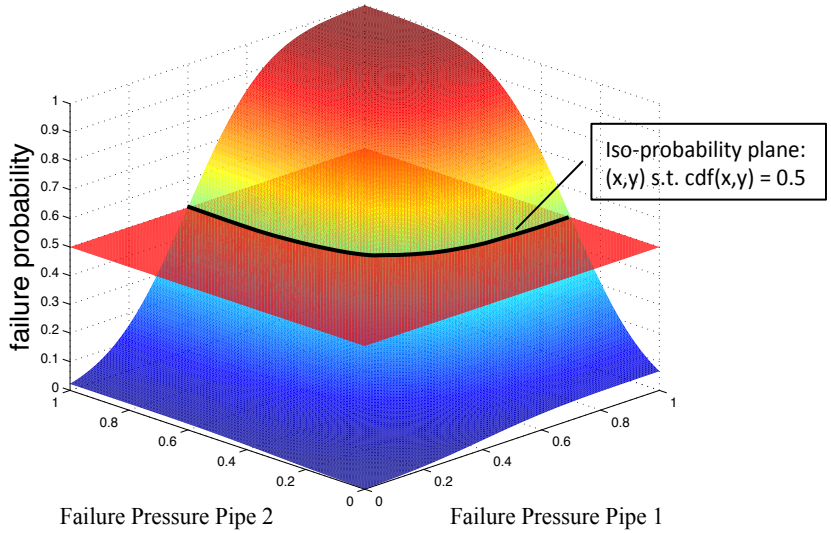
- Most common used 1D distributions



Probability Distribution Function	Truncated Form Available	Probability Distribution Function	Truncated Form Available
Bernoulli	No	Poisson	No
Binomial	No	Triangular	Yes
Exponential	Yes	Uniform	Yes
Logistic	Yes	Weibull	Yes
Lognormal	Yes	Gamma	Yes
Normal	Yes	Beta	Yes

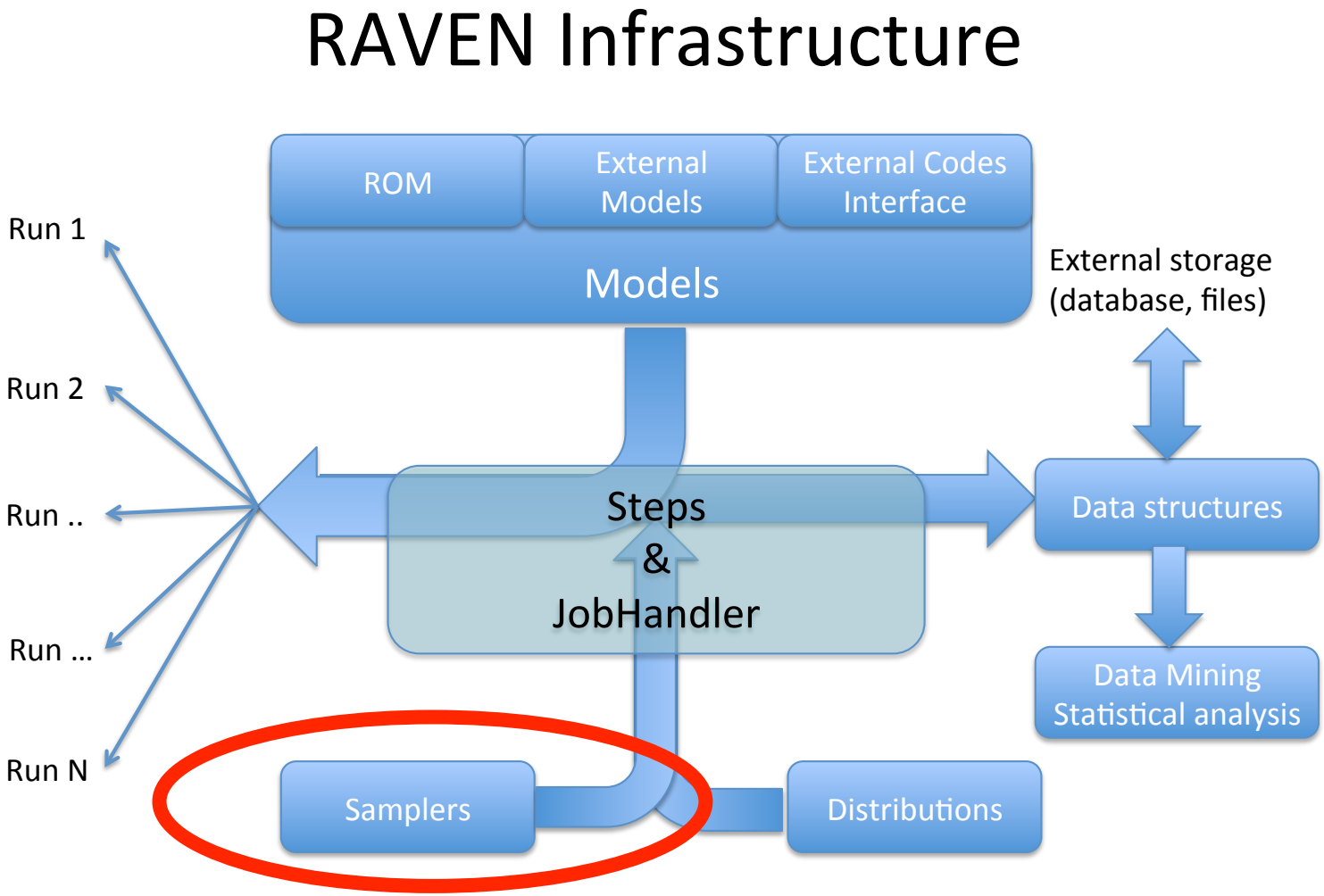
N-Dimensional Distributions (Unique capabilities)

- Import from file for custom N-dimensional distributions:
  - N-dimensional splines on Cartesian grids
  - Inverse weight interpolation
  - Micro sphere interpolation
- Sampling of N-dimensional distributions by not biased random inversion



Sampler

- Samplers determine the strategy for exploring the input space



Samplers (non-Adaptive)

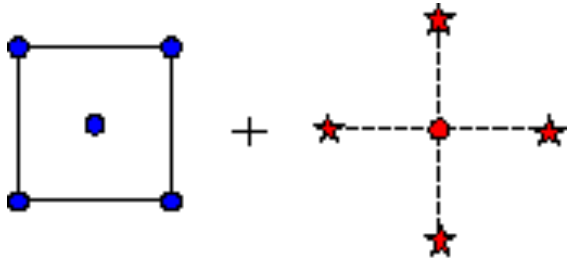
RAVEN supports many non-adaptive samplers

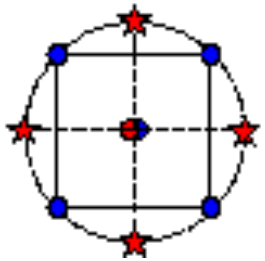
- Monte Carlo
- Grids:
  - equal-spaced in probability
  - equal-spaced in value
  - mixed (probability, custom, value)
  - Custom (used provided values/probability)
- Stratified (LHS type)
  - equal-spaced in probability
  - equal-spaced in value
  - mixed (probability, custom, value)
  - Custom (used provided values/probability)
- Generalized stochastic collocation polynomial chaos

A different sampling strategies can be associated to each variable separately

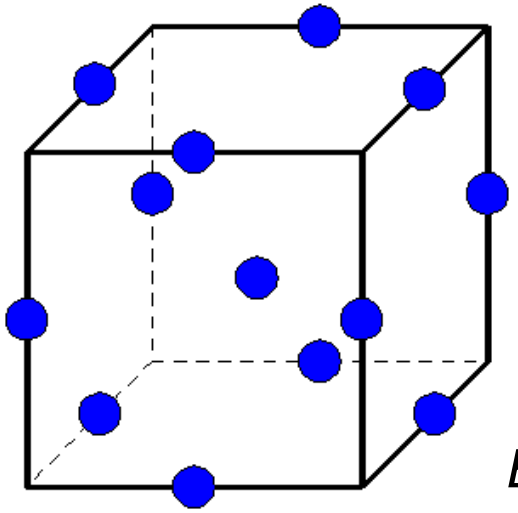
Samplers (non-Adaptive) Cont.

- Factorial Designs:
  - General Full Factorial (grid)
  - 2-Level Fractional-Factorial
  - Plackett-Burman
- Response Surface Designs:
  - Box-Behnken
  - Central Composite





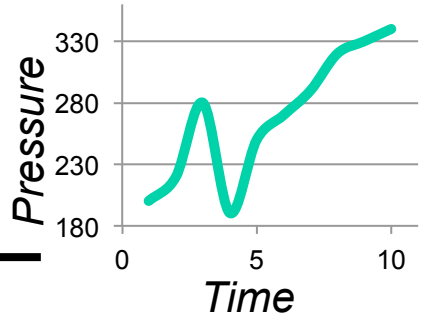
Central Composite

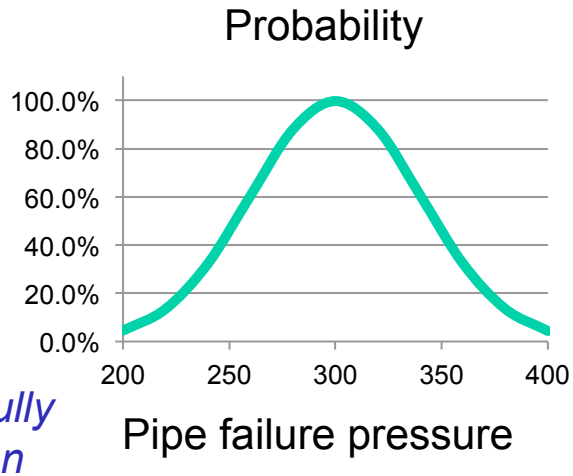


Box-Behnken

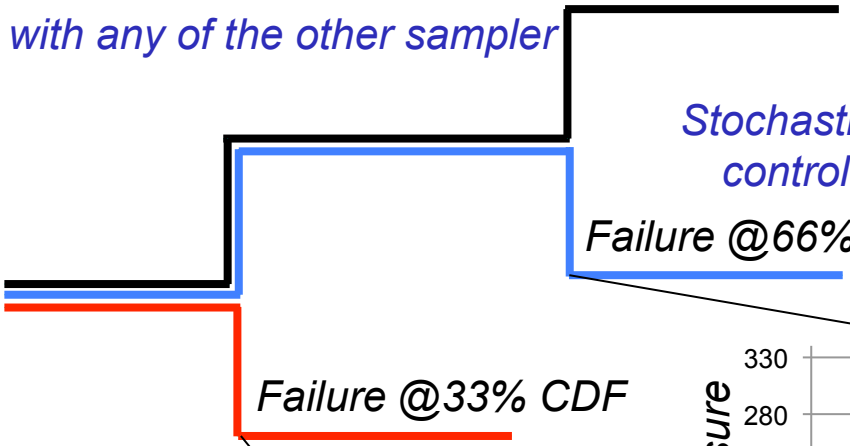
Dynamic Event Trees (DET)

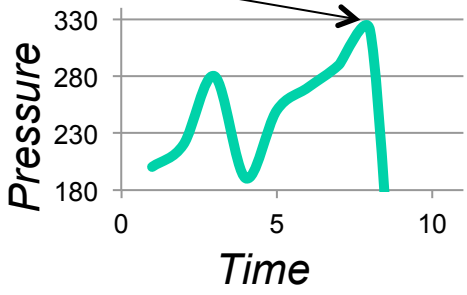
Sampling is decided beforehand for "control variables"
the corresponding simulation is run only if the physics triggers the control variable
DET can be used in a mixed fashion with any of the other sampler

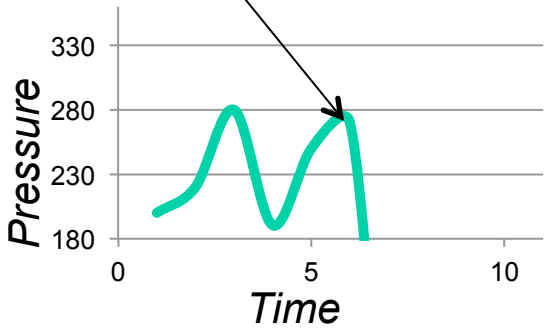


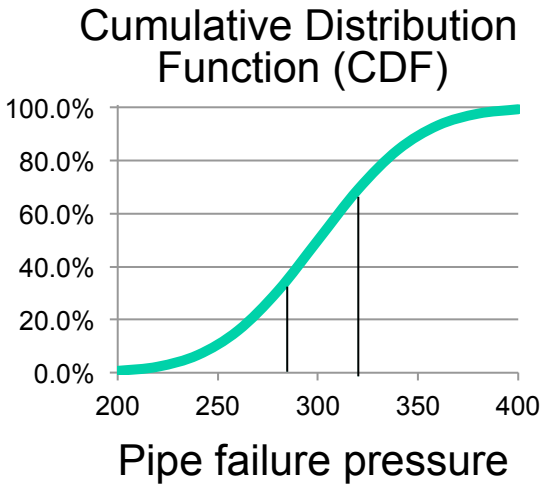


Stochastic system could not be fully controlled by the initial condition



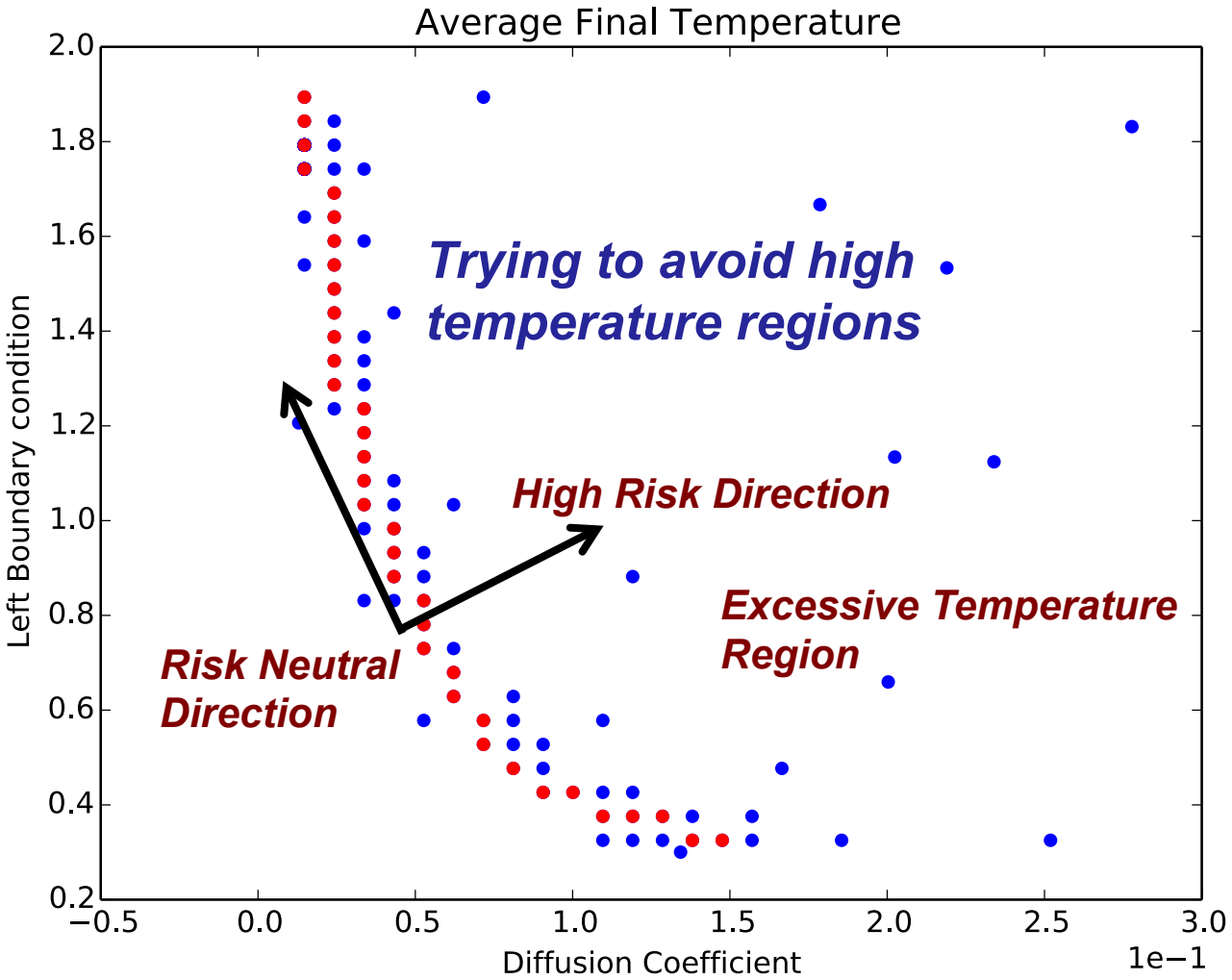






- *DET available with RELAP5-3D, MAAP, RELAP-7*

Adaptive Sampler: Reliability Surfaces Finder



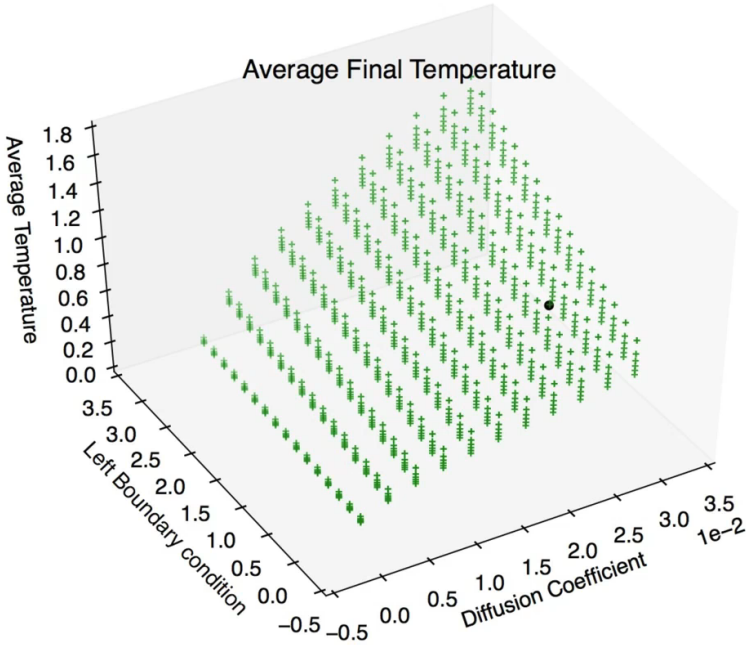
- *The sampler seeks to determine a region in the input parameter space for which a given condition is satisfied*
- *Used to determine operative safe regions*
- *Performance optimization*

Adaptive Limit Surface Demo

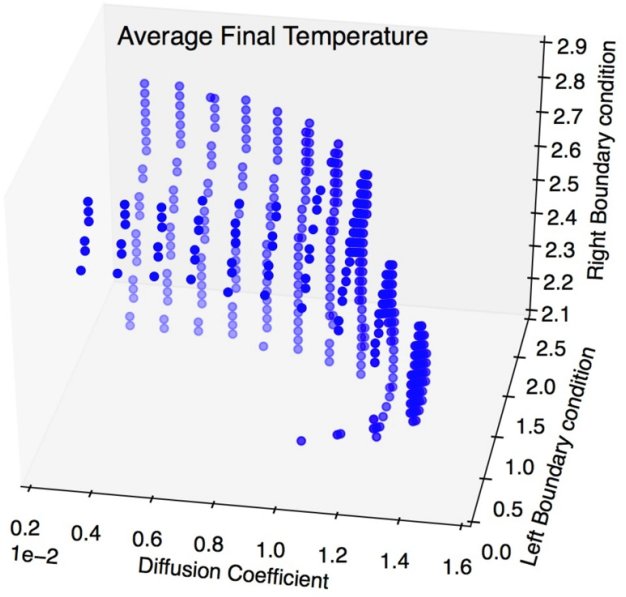
Artificial intelligence (classifiers) is used to speed up the search

3-D input space (surface location)

Sampling Location



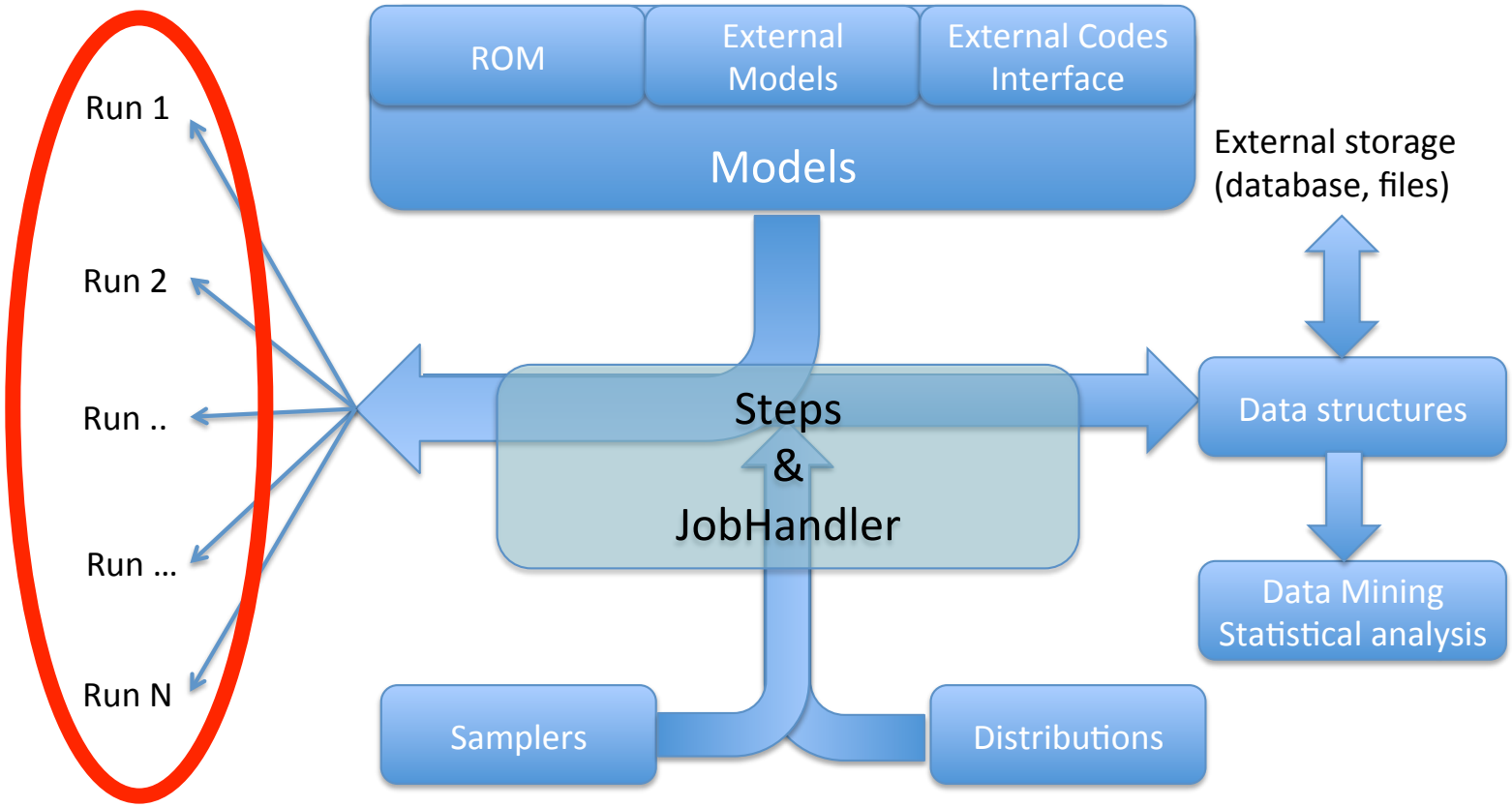
Limit Surface Evolution



Queuing

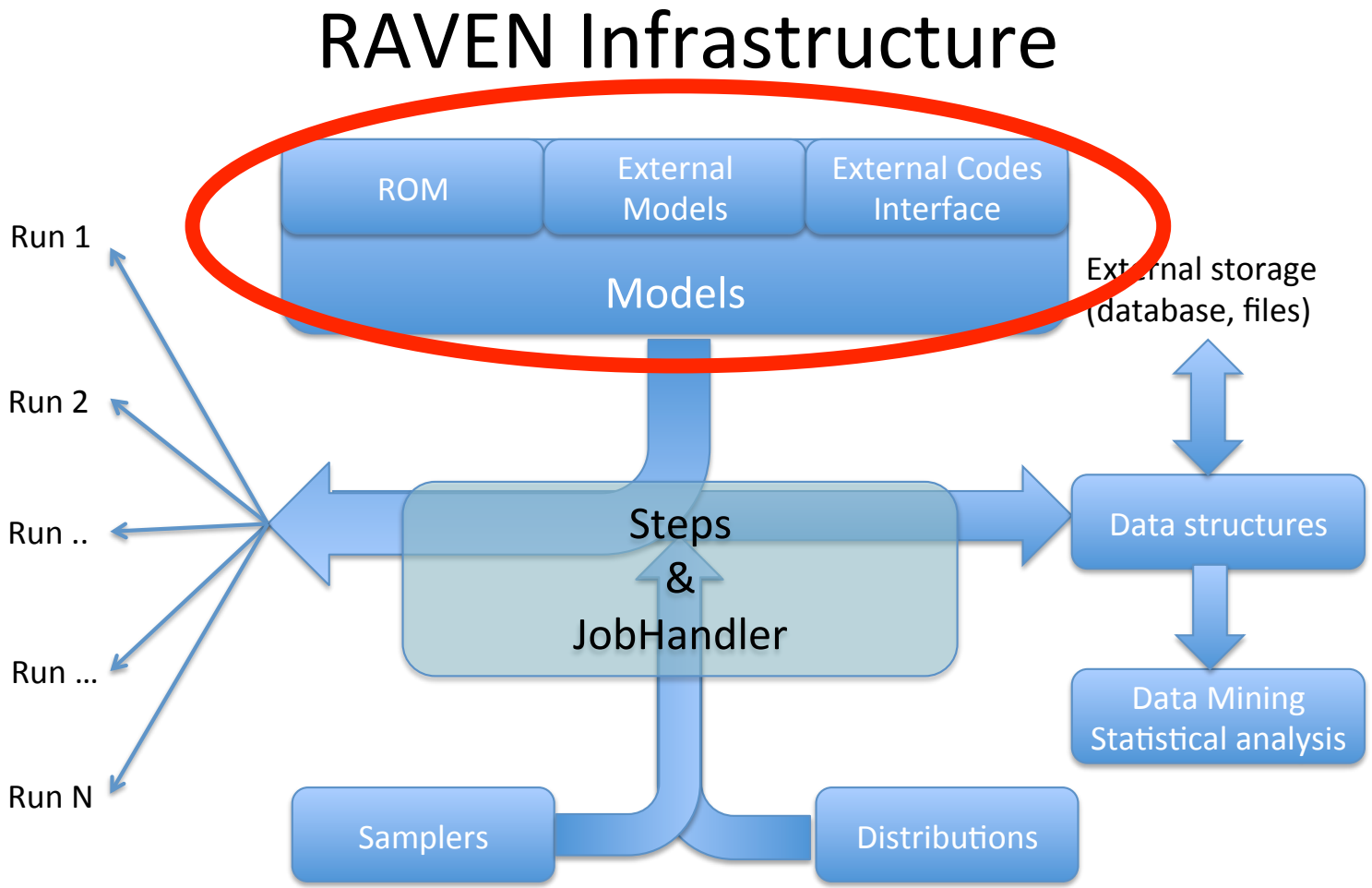
- Queuing strategies is used to take advantage of modern high performance computing

RAVEN Infrastructure



Models API

- Model APIs is the way in which RAVEN access a physical model



Models

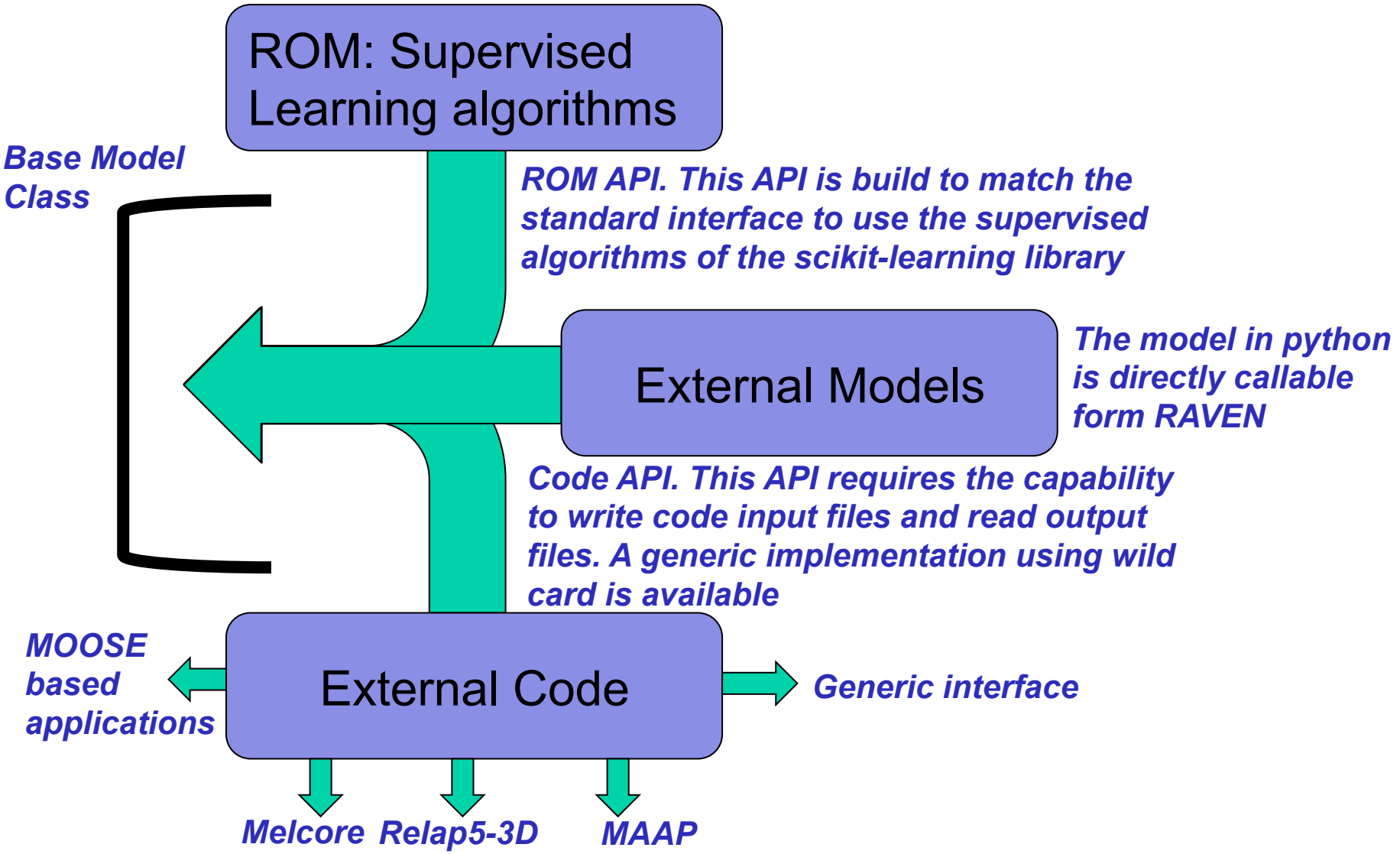
RAVEN support three different models:

- ROM (Reduced Order Models). This type of models are constitute by already trained supervised learning algorithms (RAVEN can use data sets for their training)
- External Models. External models are made of python code that use directly the “model” class
- External codes. These are classical third parties codes. The API requires the implementation of writing/reading to/from the input/output files

Available ROMs

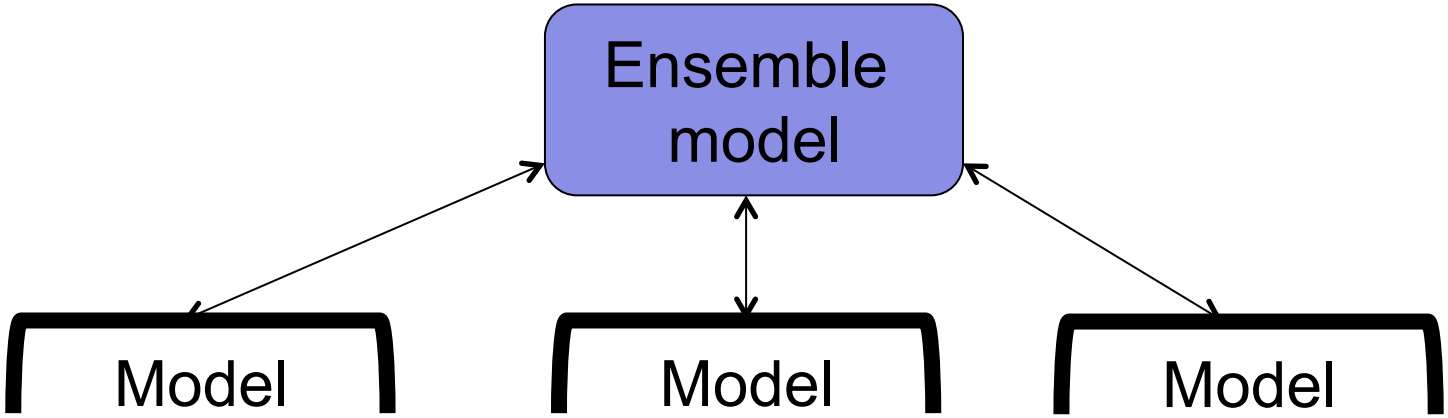
- Nearest neighbor (KD-Tree based)
- Support vector machine:
  - Polynomial kernel
  - Gaussian Kernels
  - Radial basis functions
- Micro Sphere
- Inverse Weight
- N-Dimensional spline
- Gaussian process
- Polynomial (stochastic and not)
- Linear regressors
- *Many more (http://scikit-learn.org/stable/supervised_learning.html)*
- Ensemble models

Model APIs


ROM: Supervised Learning algorithms
Base Model Class
ROM API. This API is build to match the standard interface to use the supervised algorithms of the scikit-learning library
External Models
The model in python is directly callable form RAVEN
Code API. This API requires the capability to write code input files and read output files. A generic implementation using wild card is available
MOOSE based applications
External Code
Generic interface
Melcore
Relap5-3D
MAAP

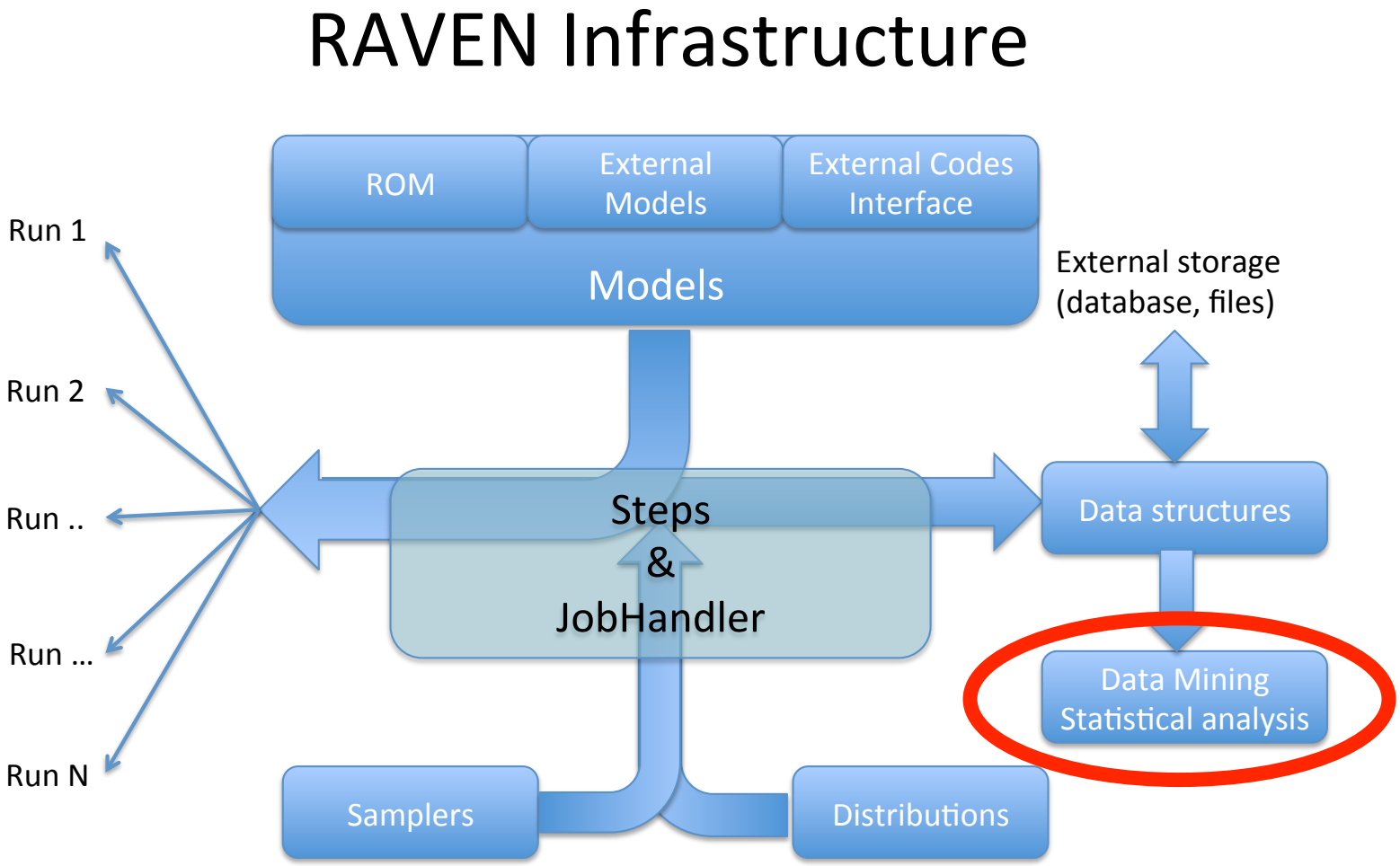
Ensemble model

- Multiple “models” can be assembled together and treated as a single one
- Models can be completely heterogeneous (code, external models, ROM)
- RAVEN acts as a hub for the information exchange
- Information passed between “models” could be:
  - Set of lump values
  - *Set of time series (being currently finalized)*



Models API

- Model APIs is the way in which RAVEN access a physical model



Statistical Post Processing

Statistical characterization of the output (*uncertainty propagation*)

- Mean
- Sigma
- Skewness (asymmetry)
- Kurtosis (more/less peaked than a standard normal)

Input/output relationship (*ranking/sensitivity*)

- Correlation matrix
- Covariance matrix
- Sensitivity matrix (multidimensional linear regression)
- Normalized sensitivity matrix (% change of the response / % change in the answer)

Classical Data Mining

- RAVEN gives access to several data mining algorithms for data post-processing via implementation of APIs between RAVEN and Scikit-learn*
- Cardinality Reduction algorithms aim to recognize commonalities among the data - Clustering, Gaussian Mixture Models, etc.
- Dimensionality Reduction algorithms are used to reduce the number of degrees of freedom in the data under consideration – Manifold Learning, Decomposing Signals in Components, etc.
- **RAVEN extend all the data mining algorithms to perform time dependent data mining**

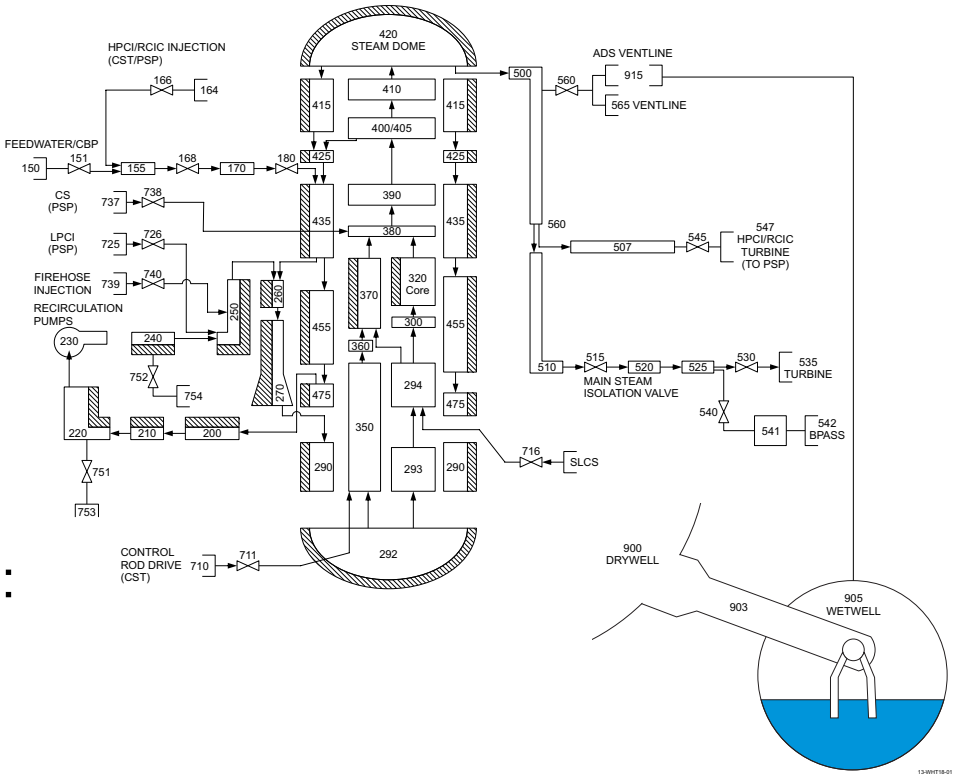
*http://scikit-learn.org

Examples

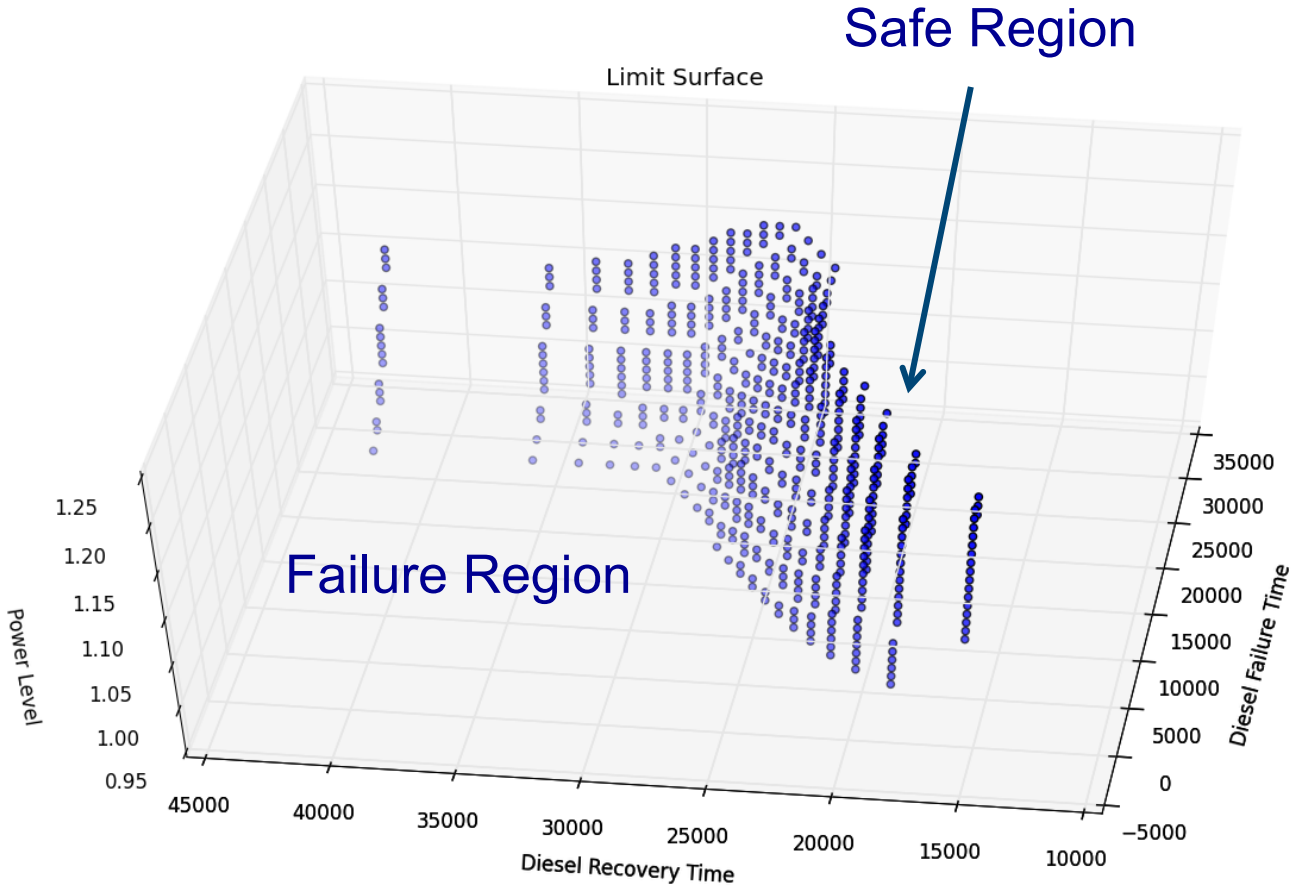
Uncertainty Quantification

- Demo: BWR Station Black Out analysis
- Code: RELAP5-3D (10,000 runs)
- Sampled variables:
  - Elapsed time before Diesel Generation loos
  - Diesel Down time
  - Power Level
- Clad Temperature (system Output):

Entry	Monte Carlo
Mean:	2.32E+03
Sigma:	9.05E+02
Kurtosis:	-1.54E+00
Median:	3.09E+03
Skewness:	-5.25E-01

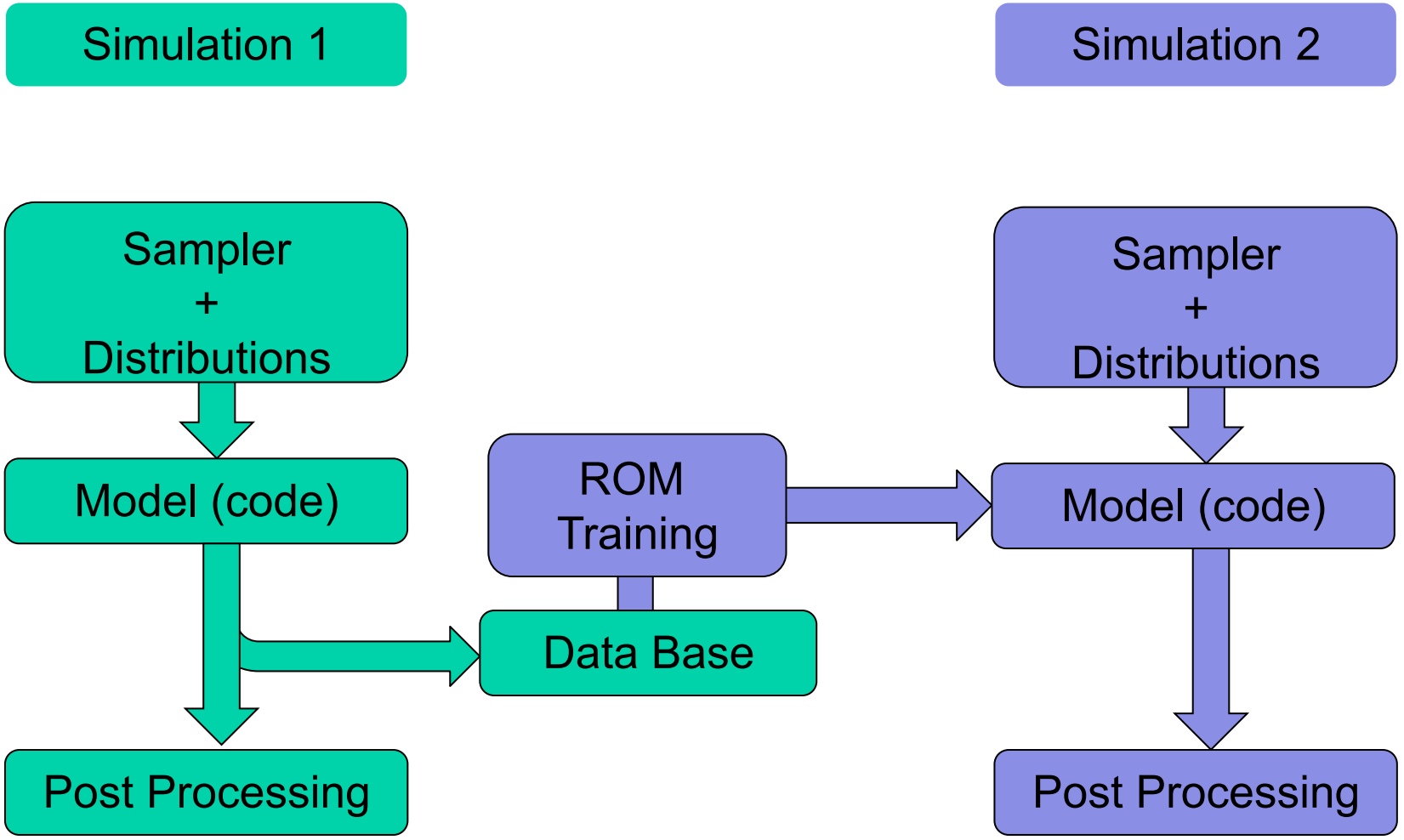


Risk Management: Reliability Surface



Seeking the combination of parameters (axis) that allows not to exceed the maximum clad temperature

Workflows


Simulation 1
Simulation 2
Sampler + Distributions
Model (code)
Post Processing
ROM Training
Data Base
Sampler + Distributions
Model (code)
Post Processing

Comparison vs ROM

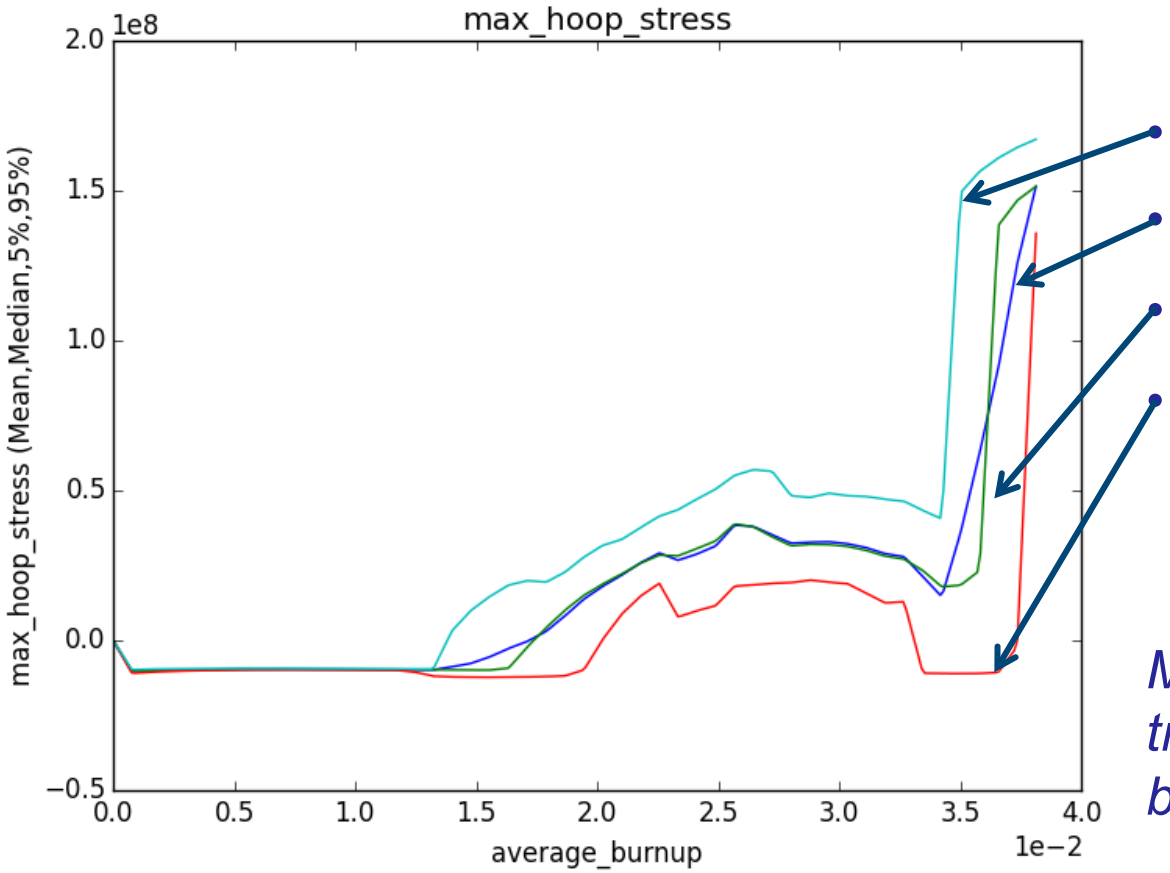
- Statistical figures for max clad temperature:

Entry	Original	Surrogate
Mean:	2.3296E+03	2.3103E+03
Sigma:	9.0559E+02	8.4599E+02
Kurtosis:	-1.547E+00	-1.4497E+00
Median:	3.092E+03	2.683E+03
Skewness:	-5.256E-01	-5.0662E-01
Failure Pb	60.99%	60.82%

- The matching of the results between the Monte Carlo Sampling on the RELAP5-3D code and the build surrogate confirm the good training set

- New input distributions could be tested without re-running the original code but using the ROM

Time Dependent Statistics: Fuel Performance



95% percentile

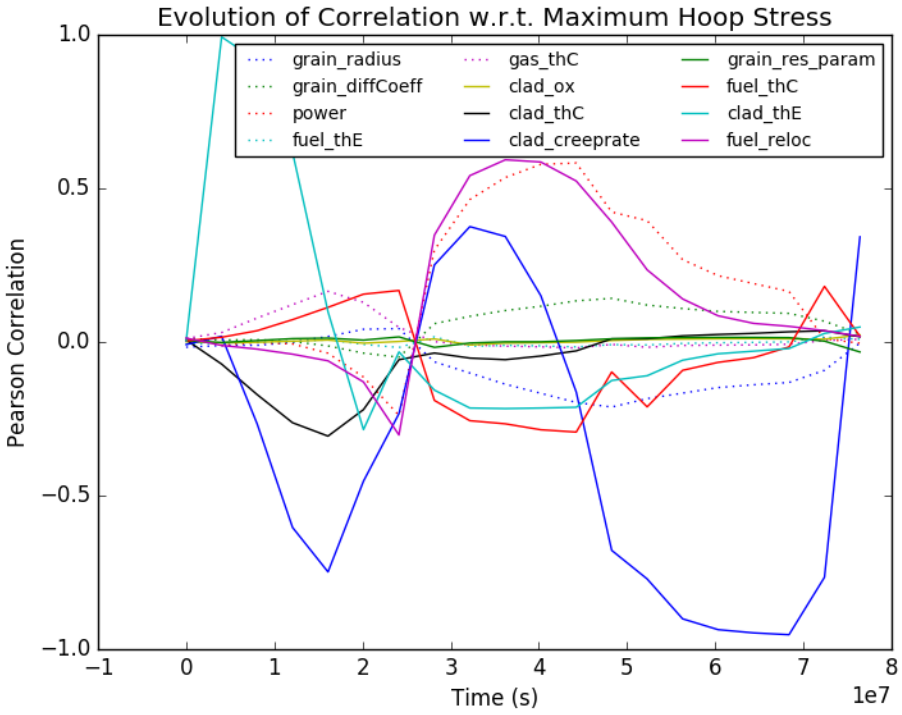
Mean

Median

5% percentile

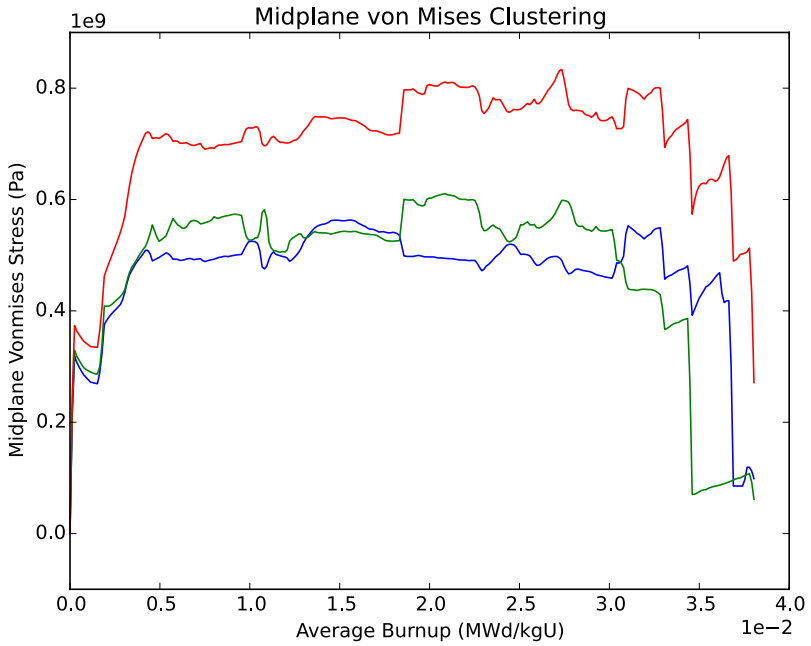
Max hoop stress during transient as a function of burn up

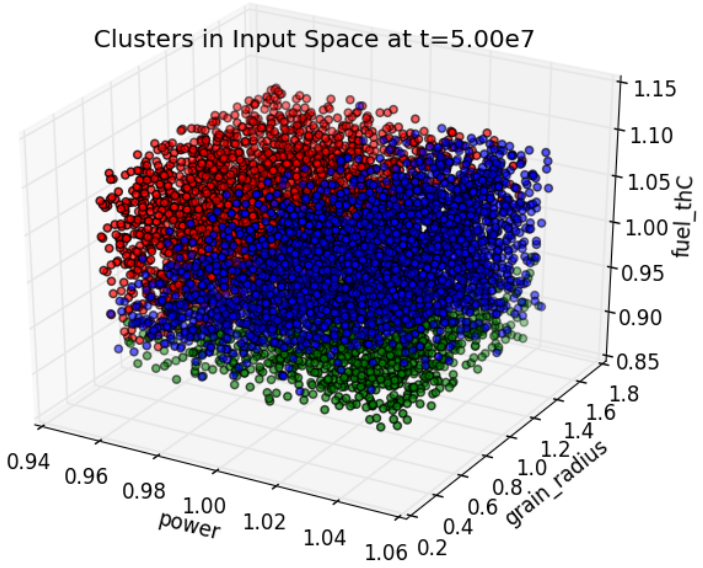
Time Dependent Relational Analysis



- *The correlation analysis reveals how much each input is linearly related to the input space*
- *Changes in the correlation coefficient usually indicate a change in the dominating physics*

Time Dependent Clustering

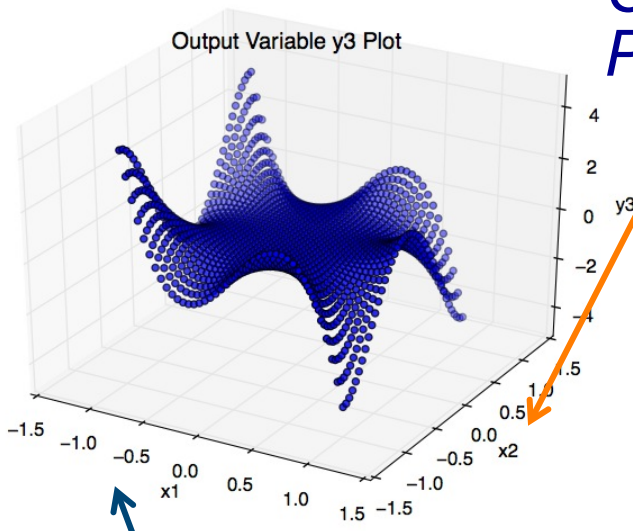




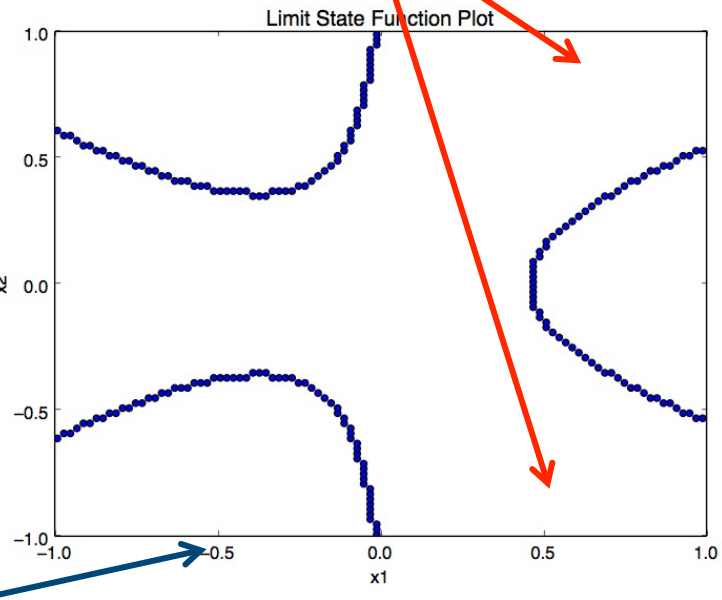
The Safest Point

- Some parameters are within operator/design control
- Some parameters are unknown/ uncontrollable

Safest point for a given value of the aleatory variable



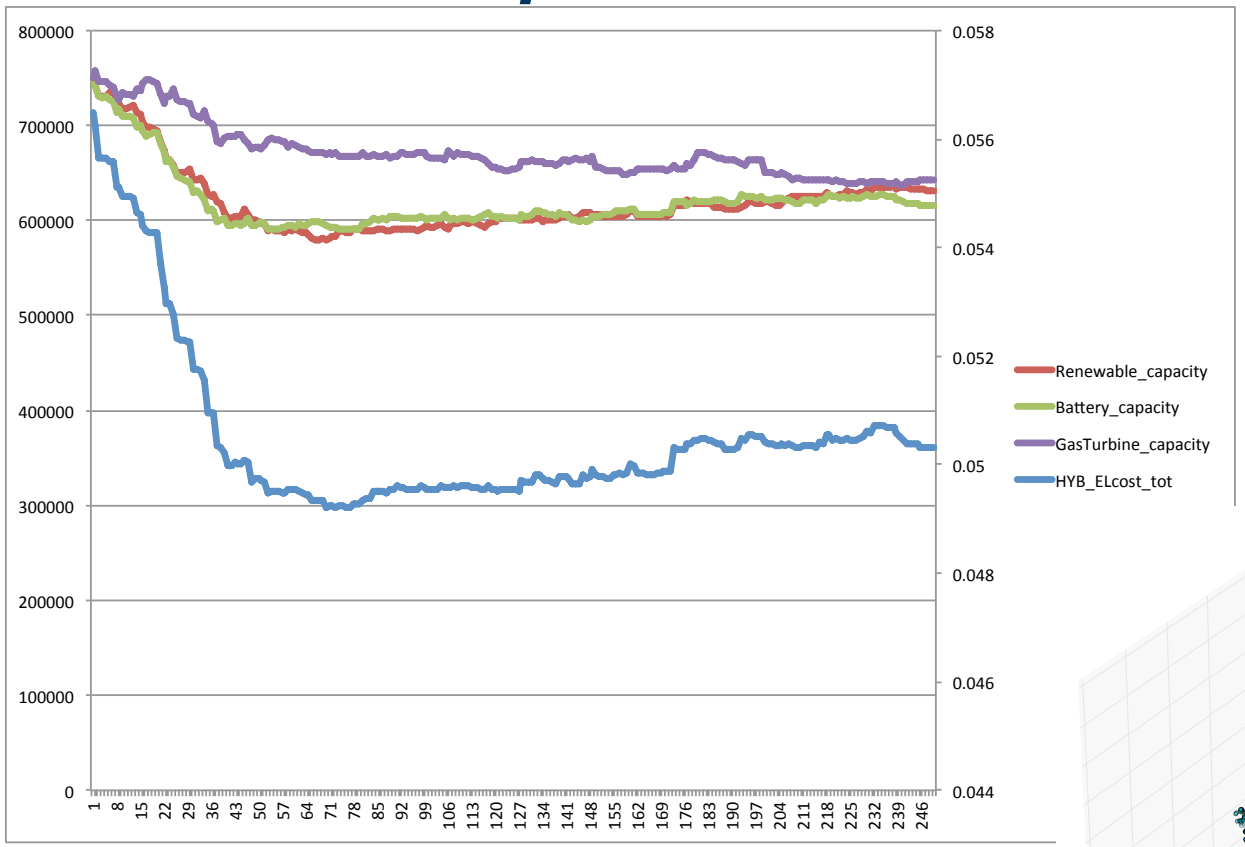
Control Parameter

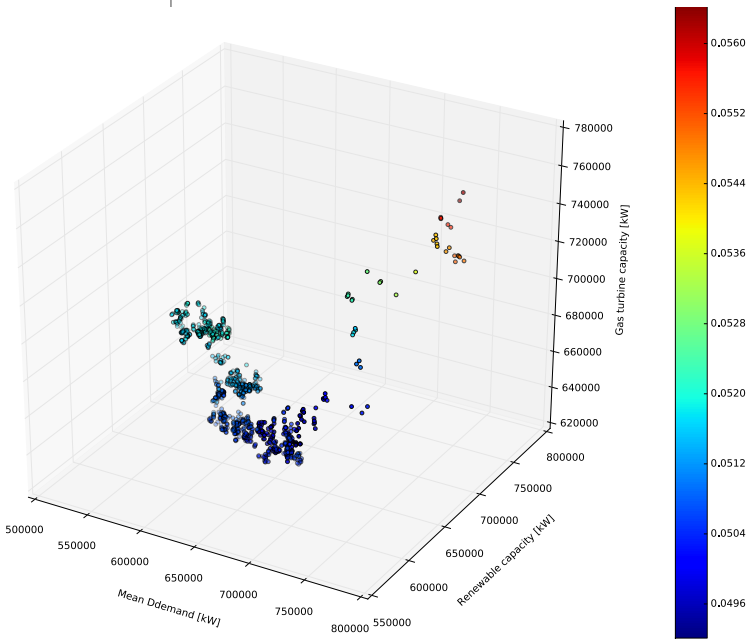


Control Parameter

The safest point needs to be weighted for the probability of the corresponding configuration

Stochastic Optimization

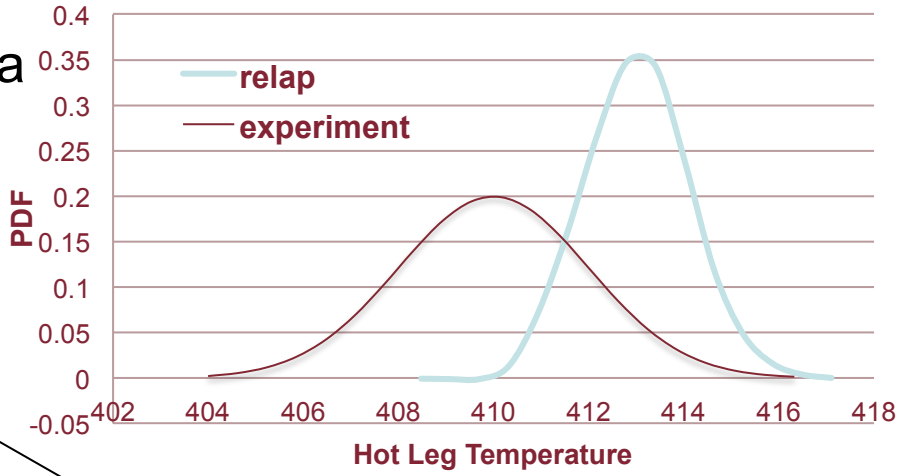




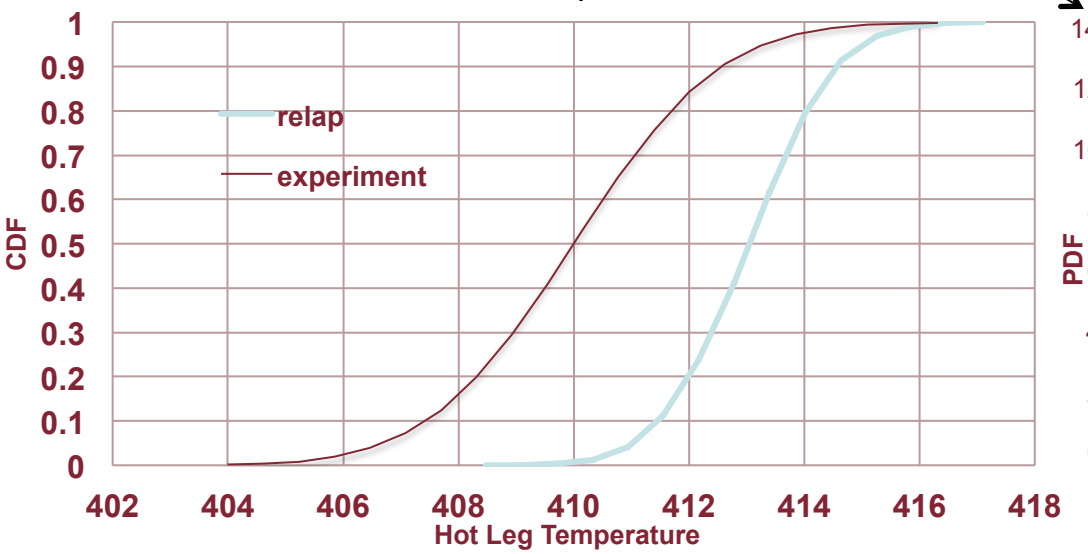
Validation Metrics

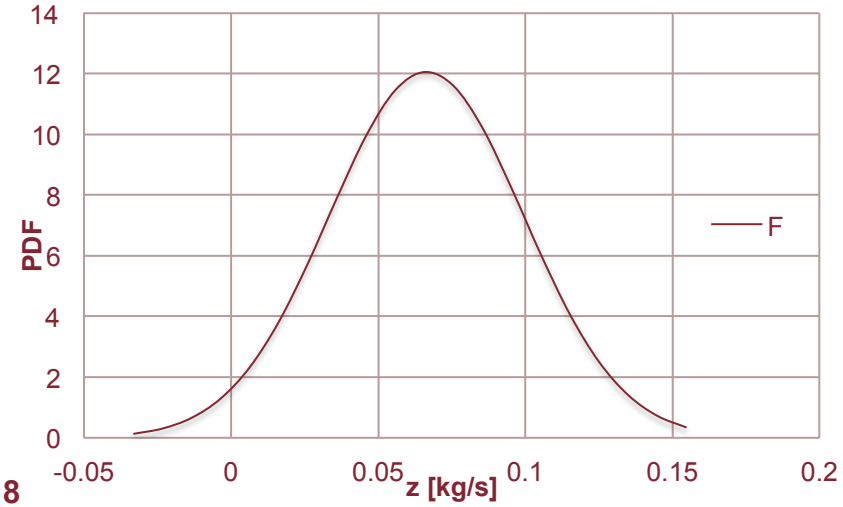
Probability Distribution Function Area Metric

Distance Probability Distribution Function



Minkowski L_1 Metric





Questions