Robust control design by scenario optimization

Roberto Rocchetta, PhD

DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE

STOCHASTIC OPERATIONS RESEARCH

Acknowledge Dr Luis G. Crespo, Dr Sean P. Kenny for the kind support

B. Wie and D. S. Bernstein, "*Benchmark Problems for Robust Control Design*," in 1991 American Control Conference, 1991

Problem statement

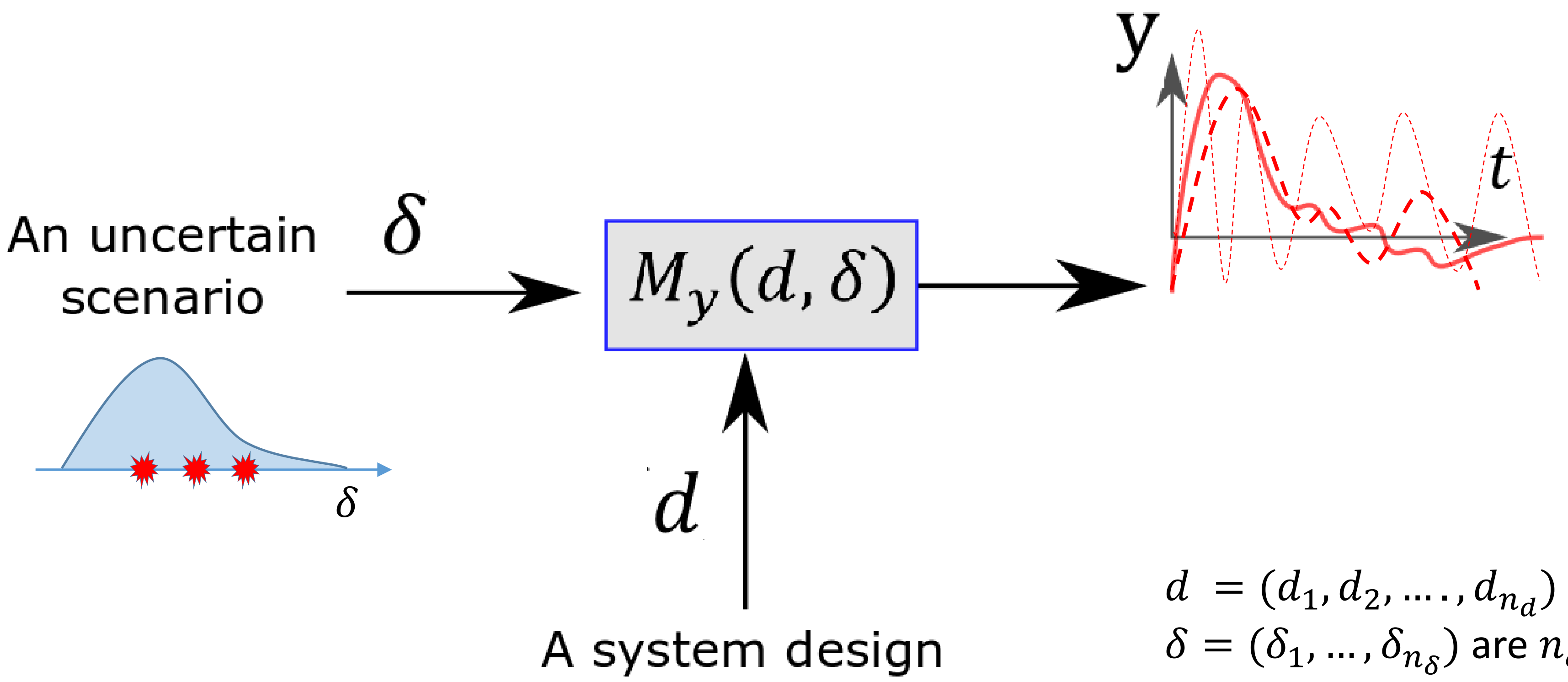
Given a numerical model of a dynamic system $\mathrm{y} = M_y(d, \delta)$, a reliability model $\mathrm{g} = M_g(d, \delta)$ and a set of N of experimental observations

$$\{\delta_i\}_{i=1,..,N}$$

We want to find a reliable design $d^\star$ that maximizes the probability of satisfactory performance (reliability) while avoids catastrophic failures (robustness against worst-case scenarios)

In contrast with d, the random factors δ are inherently variable and non-tunable.

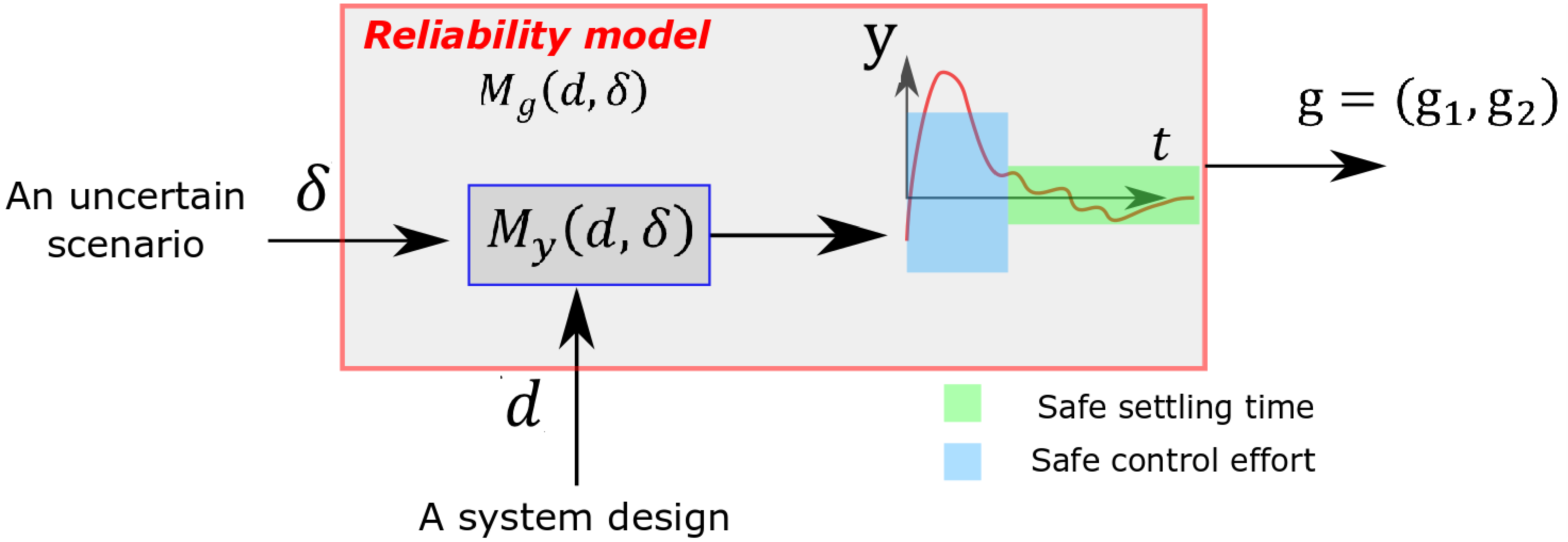
Problem statement



Different uncertain scenarios affect the system's response

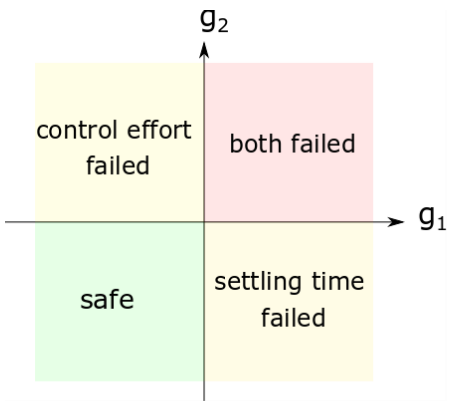
$d = (d_1, d_2, \dots, d_{n_d})$ are n_d design parameters,
$\delta = (\delta_1, \dots, \delta_{n_\delta})$ are n_δ uncertain factors
$y = (y_1, \dots, y_T)$ response at time $t = 1, 2, \dots, T$.

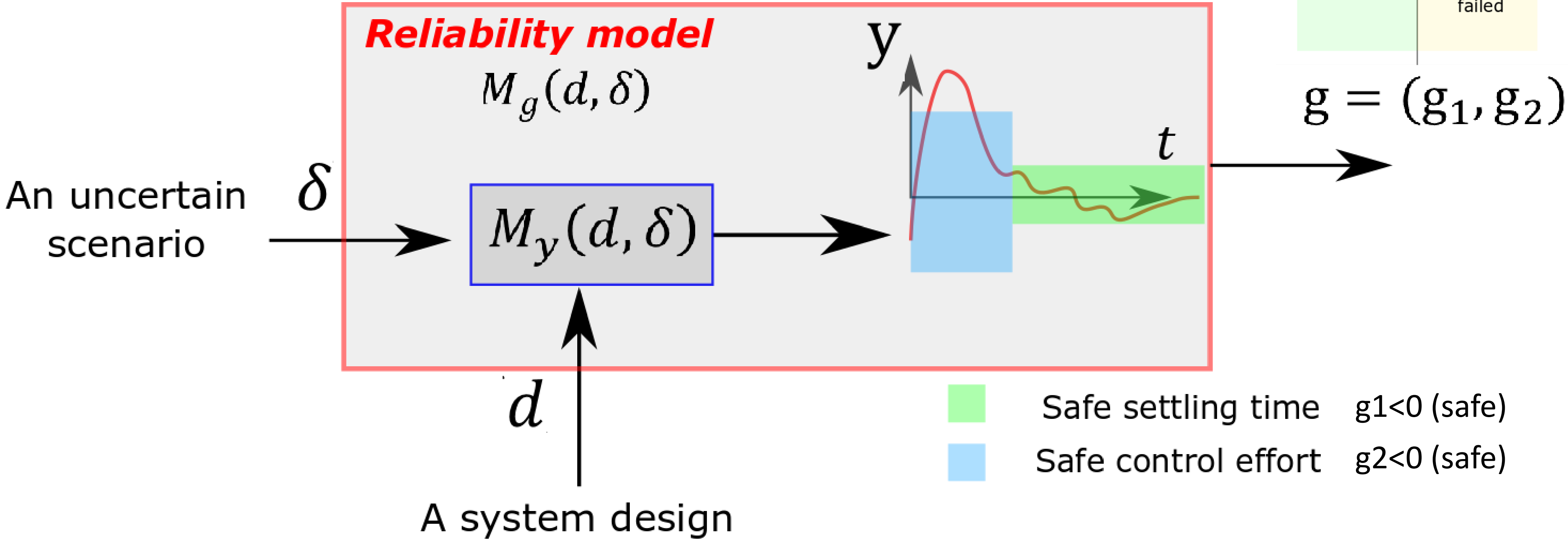
Problem statement



$g = (g_1, \ldots, g_{n_g})$ is a vector of performance scores for n_g reliability requirements.

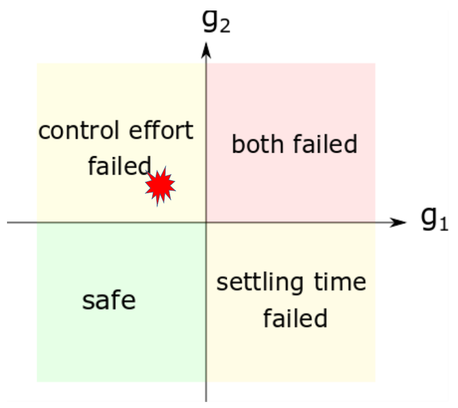
Problem statement

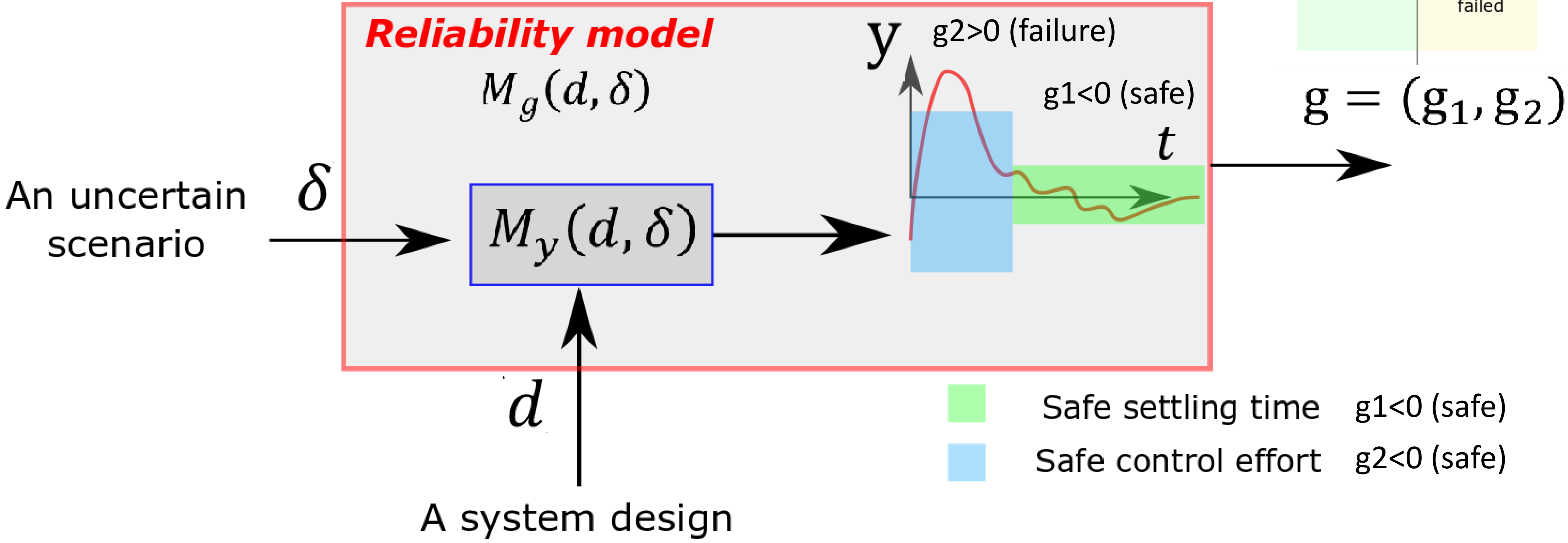




$g = (g_1, \dots, g_{n_g})$ is a vector of performance scores for n_g reliability requirements.

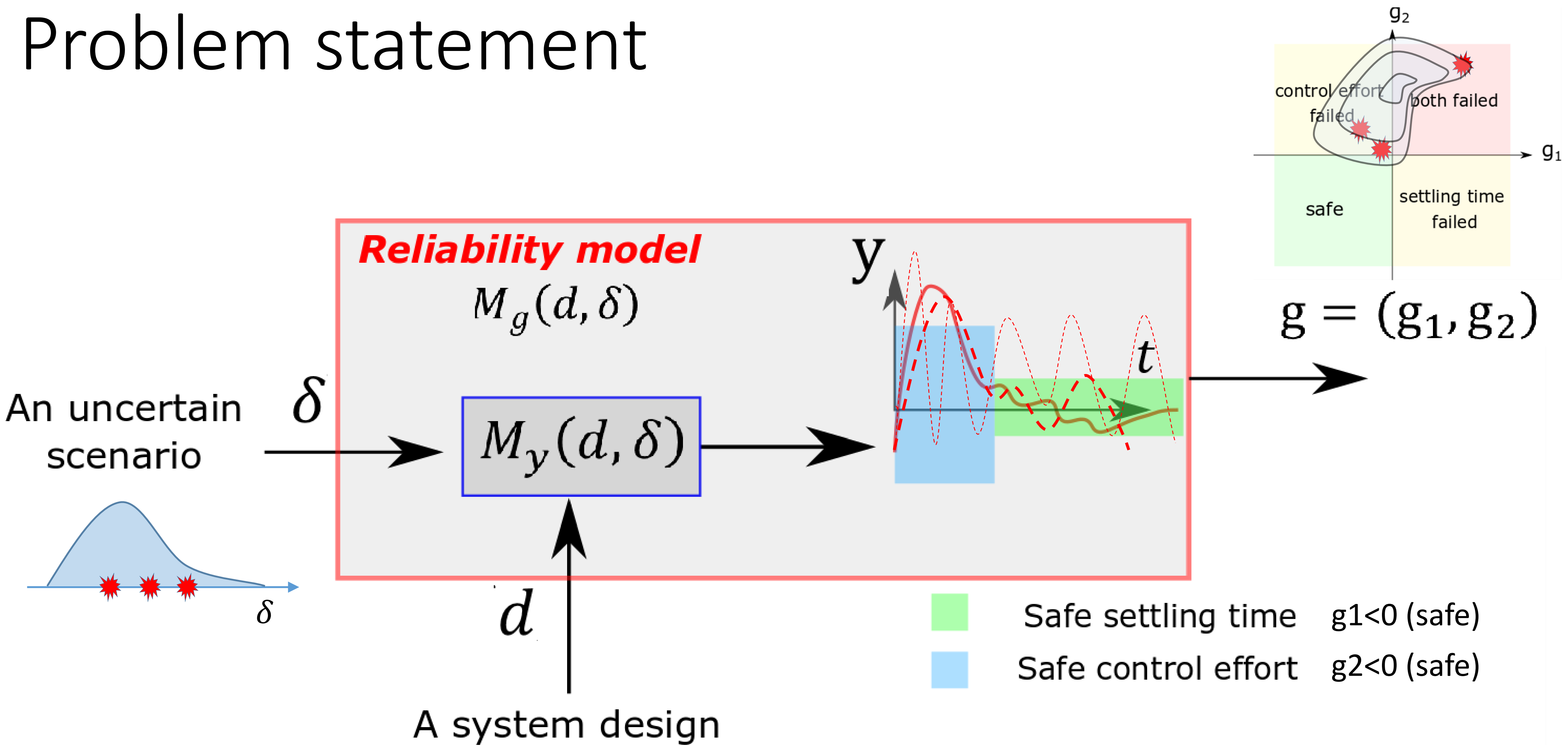
Problem statement





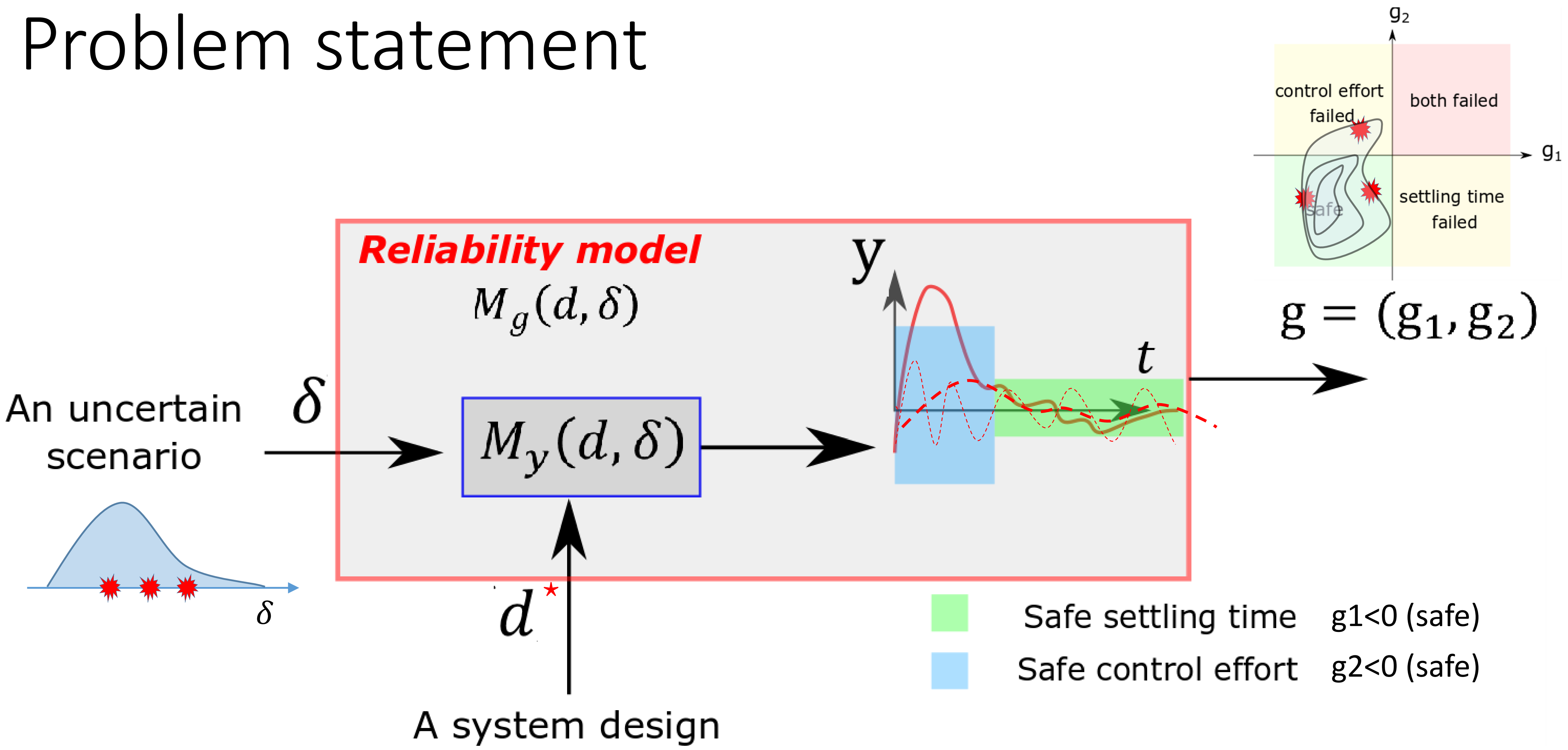
$g = (g_1, \dots, g_{n_g})$ is a vector of performance scores for n_g reliability requirements.

Problem statement



Ideally, an optimal design $d^{\star}$ pushes the realization of g in the safe region, (and far away from severe failure)

Problem statement



Ideally, an optimal design $d^\star$ pushes the realization of g in the safe region, (and far away from severe failure)

Robustness &Reliability Metrics

(to compare different designs)

Worst-case performance score:

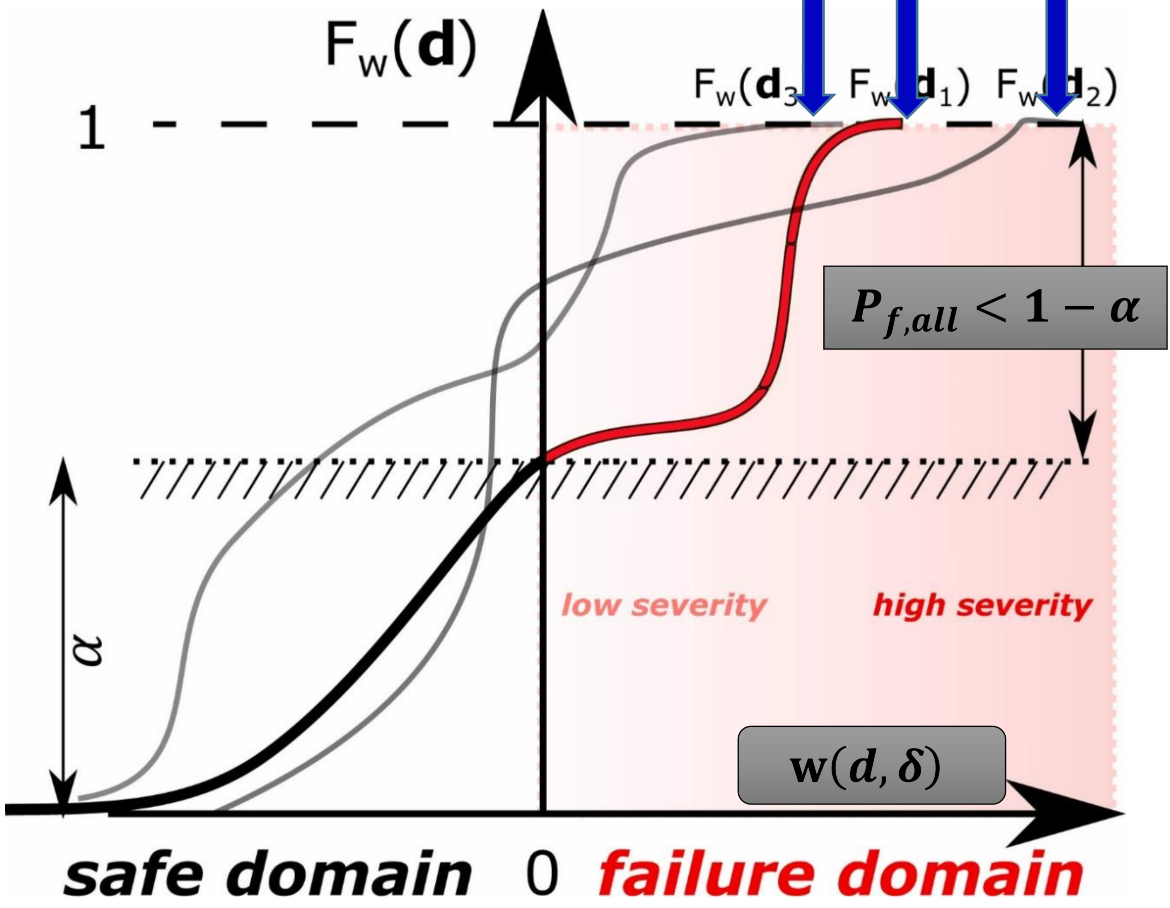
$$\mathrm{w}(d,\delta) = \max_{i=1,..,n_g} g_i\,(d,\delta);$$

Worst-case scenario score:

$$\bar{g}_i(d) = \max_{k=1,..,N} g_i\,(d,\delta_k)$$

$$\bar{\boldsymbol{w}}(\boldsymbol{d}) = \max_{k=1,..,N} \boldsymbol{w}\,(\boldsymbol{d},\boldsymbol{\delta}_k)$$

Empirical cumulative distribution function



Note: *$\mathbb{P}[\cdot]$ and $\mathbb{E}[\cdot]$ require numerical integration, e.g., via sampling-based methods (but we only have a few samples here)*

Robustness &Reliability Metrics

(to compare different designs)

Worst-case performance score:

$$\mathrm{w}(d,\delta) = \max_{i=1,..,n_g} g_i\,(d,\delta);$$

Worst-case scenario score:

$$\bar{g}_i(d) = \max_{k=1,..,N} g_i\,(d,\delta_k)$$

$$\bar{w}(d) = \max_{k=1,..,N} w\,(d,\delta_k)$$

Failure probability for individual and all requirements:

$$P_{f,i}(d) = \mathbb{P}[g_i(d,\delta) > 0]$$

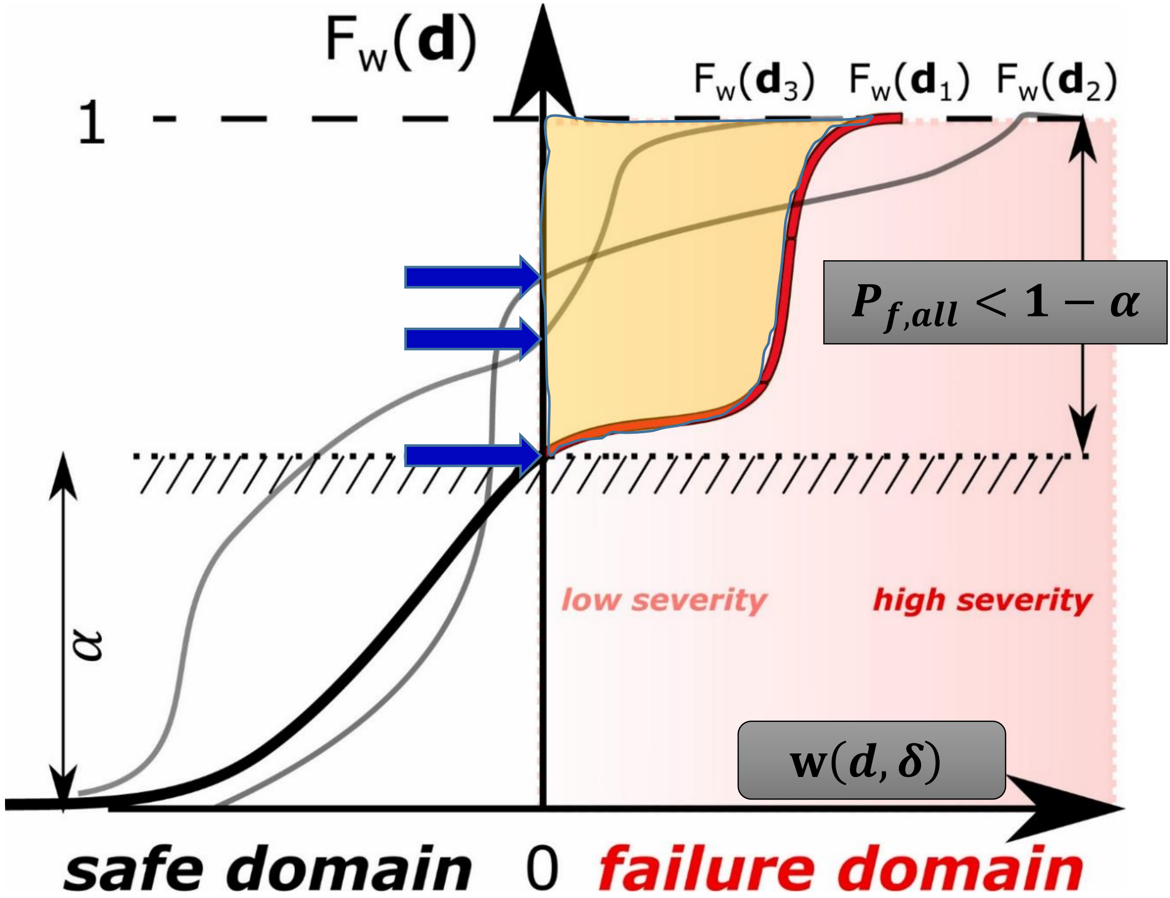
$$\boldsymbol{P_{f,all}(d) = \mathbb{P}[w(d,\delta) > 0]}$$

Severity for individual and all requirements:

$$S_i(d) = \mathbb{E}[g_i(d,\delta) \mid g_i(d,\delta) > 0]$$

$$\boldsymbol{S_{all}(d) = \mathbb{E}[w(d,\delta) \mid w(d,\delta) > 0]}$$

Empirical cumulative distribution function



Note: *$\mathbb{P}[\cdot]$ and $\mathbb{E}[\cdot]$ require numerical integration, e.g., via sampling-based methods (but we only have a few samples here)*

The challenge problem: Reliability-based design and reliability assessment of a controller

Goals:

- Optimize the controller design for a two-mass spring system from given data.
- Verify system reliability and robustness, e.g., reachability of failure domains and "severe" failure regions.

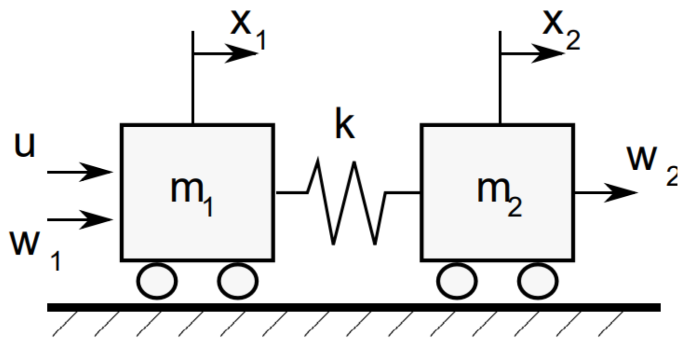


FIGURE 2: THE TWO-MASS SPRING SYSTEM BENCHMARK FOR ROBUST CONTROL DESING.

https://github.com/Roberock/ControllerRobust

The challenge problem:
Reliability and system model

Reliability model $g = M_g(d, \delta)$

g_1: Stability.
g_2: Settling time. position of the mass 2 must fall between ±0.1 from the original state after 15 seconds;
g_3: Control effort. The control signal, u, must fall between ±1.

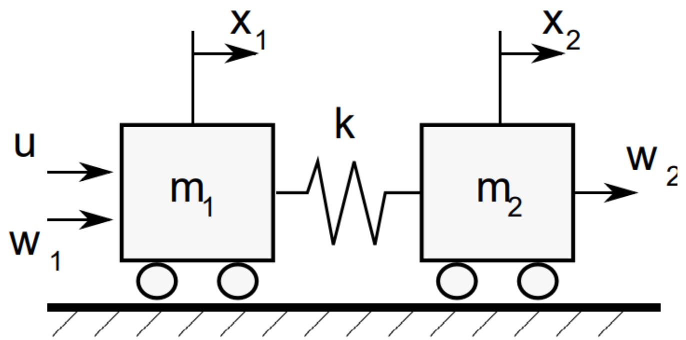


FIGURE 2: THE TWO-MASS SPRING SYSTEM BENCHMARK FOR ROBUST CONTROL DESING.

Note that the time domain requirements require simulating the time response of the system by numerical integration.

Dynamic model $M_y(d, \delta)$: $\begin{aligned} \dot{x}_c &= Ax_c + By \\ u_c &= Cx_c + Dy \end{aligned}$ $\xrightarrow[\text{transfer-function}]{\text{From canonical form to}}$ $H(s) = \dfrac{b_3 s^3 + b_2 s^2 + b_1 s + b_0}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0}$

https://github.com/Roberock/ControllerRobust

The challenge problem:
Design parameters and uncertain factors

Design parameters : $d = [b_0, b_1, b_2, b_3, a_0, a_1, a_2, a_3, a_4] \in \mathbb{R}^9;$

(coefficient of the system transfer function)

Uncertain parameters : $\delta = (m_1, m_2, k_l, k_n, \lambda, \tau) \in \mathbb{R}^6$;

(masses, springs, time delay, control loop lag)

The system-state matrices depend on the uncertain factor:

$$A(\delta), B(\delta), C(\delta), D(\delta).$$

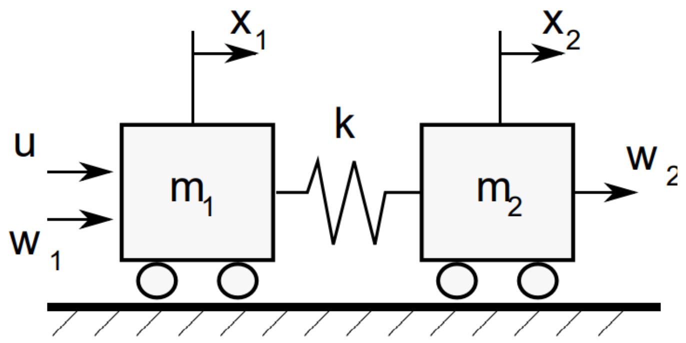
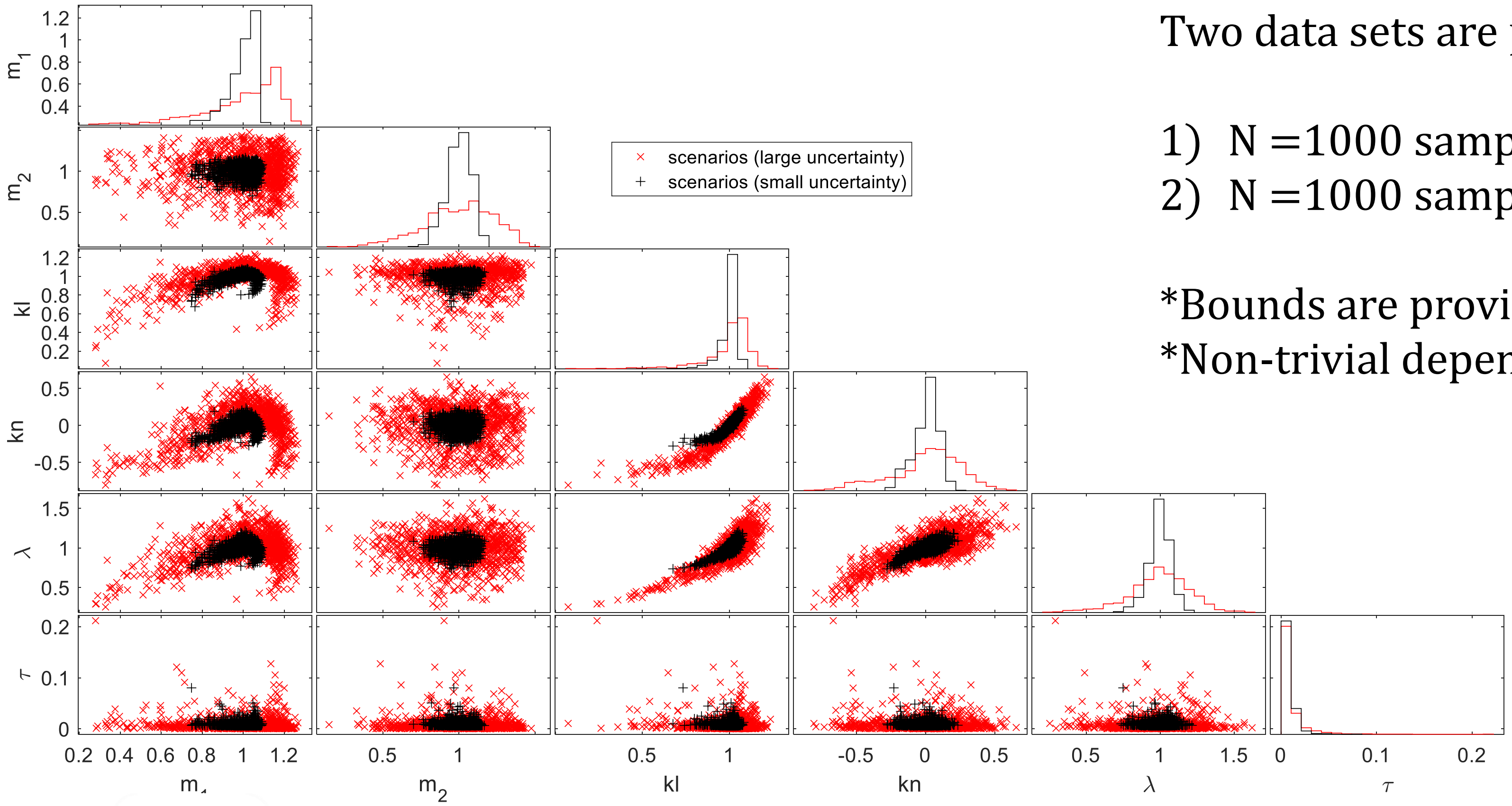


FIGURE 2: THE TWO-MASS SPRING SYSTEM BENCHMARK FOR ROBUST CONTROL DESING.

https://github.com/Roberock/ControllerRobust

The data set: $D = \{\delta_i\}_{i=1,..,N}$



Two data sets are provided:

1) N =1000 samples, large uncertainty
2) N =1000 samples, small uncertainty

*Bounds are provided
*Non-trivial dependency structures

```
δnames=["mass1" "mass2" "kl" "kn" "λ" "τ" ]
# admissible ranges
δlims=[0.1    2;       0.1    2;
       0.05   1.75;    -1     1;
       0.2    1.8;     0.0001 0.3];
```

https://github.com/Roberock/ControllerRobust

Four designs from [1]:

$\boldsymbol{SP_1(D,0)}$:

Minimizes the worst-case scenario score, $\overline{w}(d)$

$\mathbf{SP_1(D,0.5)}$:

Minimizes 95th quantile of the $w(d,\delta)$ distribution

$\mathbf{SP_2(D)}$:

Minimize $P_{f,all}(d)$

Nominal:

A baseline design

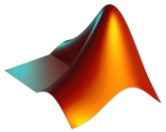
Compromise between severity and reliability optimization! Competing reliability requirements!

Designs Names

	$\mathcal{SP}_1(\mathcal{D},0)$	$\mathcal{SP}_1(\mathcal{D},0.05)$	$\mathcal{SP}_2(\mathcal{D})$	Nominal
$\hat{P}_f$	0.981	0.057	0.175	0.734
$\overline{g}_1$	-0.0272	-0.013	0.069	$7e-4$
$\overline{g}_2$	0.5925	1.576	3.054	1.974
$\overline{g}_3$	0.5925	0.193	1.406	0.202
a_4	0.2238	0.5375	0.7600	0.5503
a_3	0.6811	1.3346	1.9491	1.4175
a_2	3.1275	2.4206	3.0497	2.6531
a_1	2.3615	2.1689	2.7344	2.4802
a_0	1.1833	0.8084	1.0594	1.0000
b_3	-0.0982	2.4802	-0.0831	-0.1324
b_2	0.4702	0.6146	0.6358	0.3533
b_1	0.5886	0.5265	0.7752	0.6005
b_0	0.0777	0.0716	0.0981	0.0728

Scores

The design parameters



https://github.com/Roberock/ControllerRobust

[1] Rocchetta, R., Crespo, L. G., & Kenny, S. P. (2019). Solution of the benchmark control problem by scenario optimization. In Dynamic Systems and Control Conference, USA.

Design resulting from previous works [1]

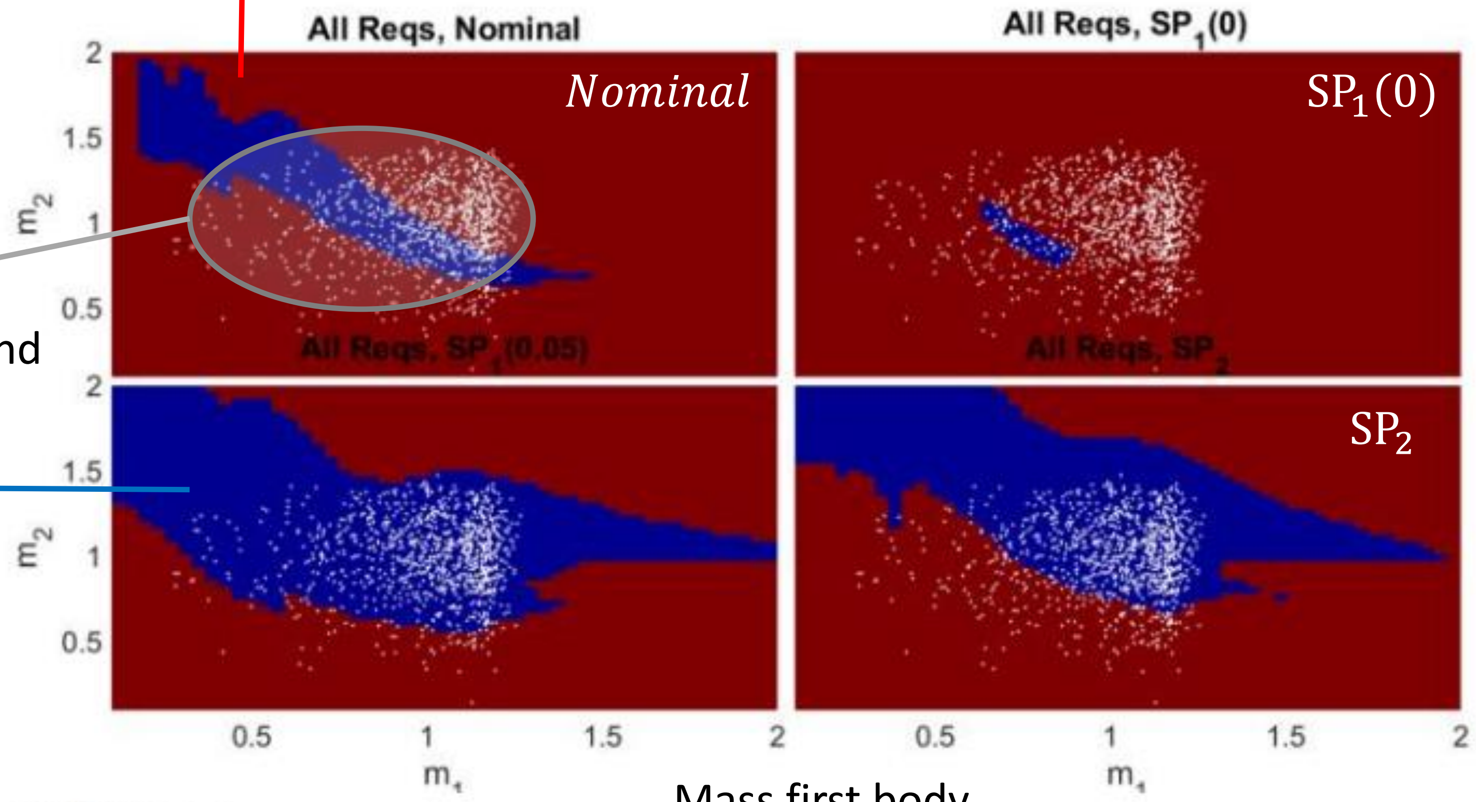
Failure domain,
$w(m_1, m_2, d_{nominal}) > 0$

Also note that w and g are non-convex functions of d and δ

Samples

Mass second body

Safe domain
$w(m_1, m_2, SP(0.05)) < 0$



Mass first body

https://github.com/Roberock/ControllerRobust

[1] Rocchetta, R., Crespo, L. G., & Kenny, S. P. (2019). Solution of the benchmark control problem by scenario optimization. In Dynamic Systems and Control Conference, USA.

Concluding discussion on challenges a few remarks

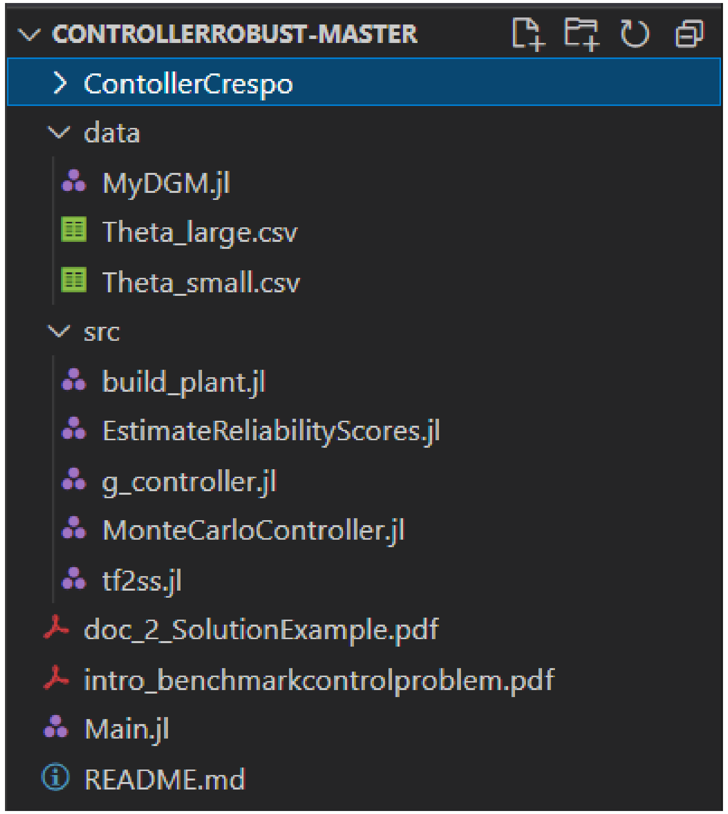
Value of information & uncertainty characterization challenges:
How to use the available data in the best way? Shall we prescribe a model for the uncertainty?
How to keep track of the value of information in the initial data set (only 1k samples) ?
The data present some complex dependency structures. Can we account for this?

Reliability challenges:
- ***Competing reliability requirements.***
- ***Robustness and reliability (competing)***
- ***Estimation of failure probability from set-based characterization (reachability of failure sets)***

Robust vs Reliable design optimization challenges:
- ***Need to define an appropriate optimization program (minimize or constrain severity/reliability/worst-cases?)***
- ***Probabilities and expectations are step-wise discontinuous (gradient based optimizer are inapplicable).***

https://github.com/Roberock/ControllerRobust

Thank you for listening!



[1] Rocchetta, R., Crespo, L. G., & Kenny, S. P. (2019). “*Solution of the benchmark control problem by scenario optimization*”, In Dynamic Systems and Control Conference, USA.

[2] B. Wie and D. S. Bernstein, "*Benchmark Problems for Robust Control Design*," in 1991 American Control Conference, 1991

[3] Roberto Rocchetta, Luis G. Crespo, Sean P. Kenny, “*A scenario optimization approach to reliability-based design*”, Reliability Engineering & System Safety, Volume 196, 2020, https://doi.org/10.1016/j.ress.2019.106755

[4] R. Rocchetta, Luis G. Crespo, *“A scenario optimization approach to reliability-based and risk-based design: soft-constrained modulation of failure probability bounds”*, Reliability Engineering & System Safety, https://doi.org/10.1016/j.ress.2021.107900