Multi-scale methods for fracture: model learning across scales, digital twinning and factors of safety

Pierre KERFRIDEN, Lars BEEX, Olivier GOURY,
Ahmad AKBARI, Hussein RAPPEL, Stéphane P.A. BORDAS

CARDIFF UNIVERSITY
PRIFYSGOL CAERDYDD

THE UNIVERSITY OF WESTERN AUSTRALIA

erc

UNIVERSITÉ DU LUXEMBOURG

Permanent Researchers


Jack S. Hale


Lars Beex

Legato-team
Luxembourg

Post-docs

Comp. Sci.


Alexandre Bilger


Elisa Schenone

Maths


Georgios Bourantas

Physicist


Huu Phuoc BUI


Kostyantyn Malukhin


Satyendra K. Tomar

Maths

PhD Students

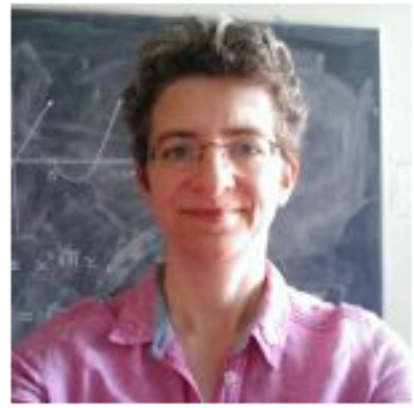

Hussein Rappel

Administrative assistant


Marie Leblanc


Changkye Lee

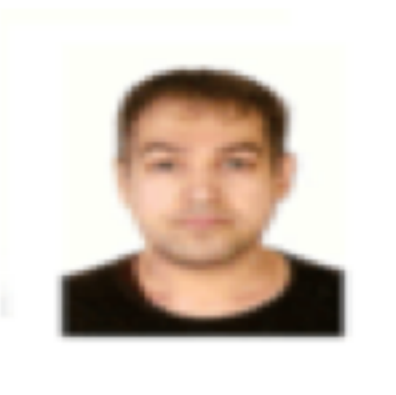

Chi Hoang


Claire Heaney


Danas Sutula


Daniel Alves Paladim


Hung Nguyen Xuan


Kevin Bronik

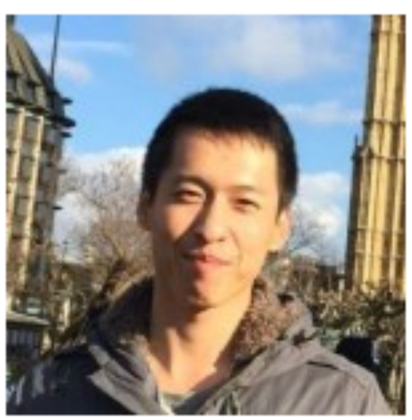

Mohammed Al-Saad


Pedro Bonilla


Peng Yu


Waled Alnaas


Xiaohan Du


Xiude Lin


Xuan Peng

Legato-team
Cardiff

Computational mechanics & computational materials sciences Multiscale/field interface problems

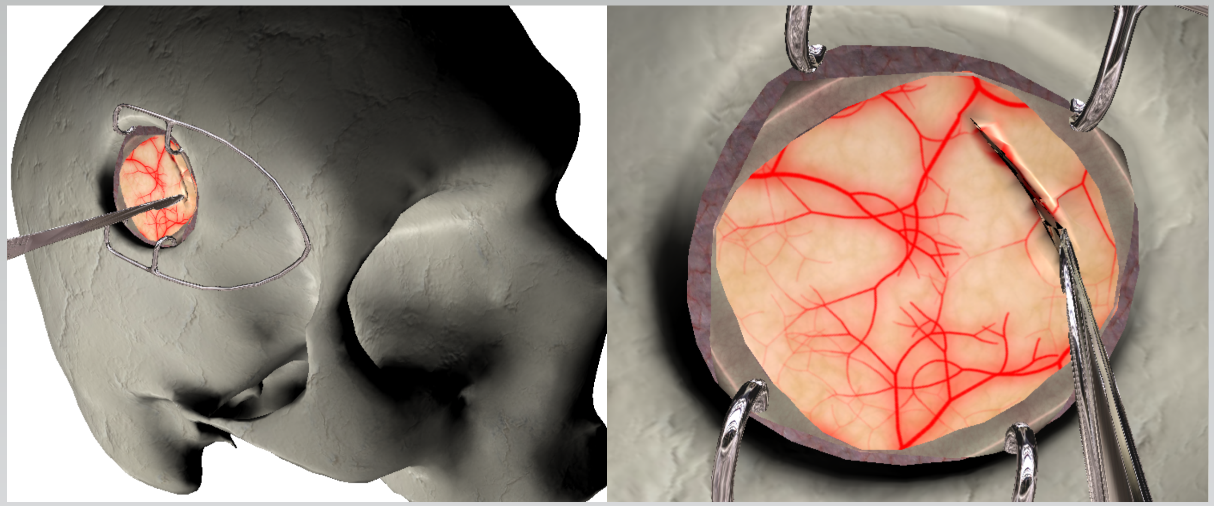
COMPETENCES

DISCRETISATION
discrete and continuum approaches

MULTI-SCALE FRACTURE
aerospace composites, polycrystalline materials

COUPLED PROBLEMS
biofilms, liquid crystals, fluid-structure, batteries

QUALITY & ERROR CONTROL
optimise computational time given an accuracy level



Real-time simulation of cutting during brain surgery, Courtecuisse et al. 2014, Medical Image Analysis

INTERACTIVITY
Reduce computational costs by several orders of magnitude

APPLICATIONS

PERSONALISED MEDICINE

Computer-aided surgery

Computer-aided diagnostics

ENGINEERING

Durability & Sustainability

Energy

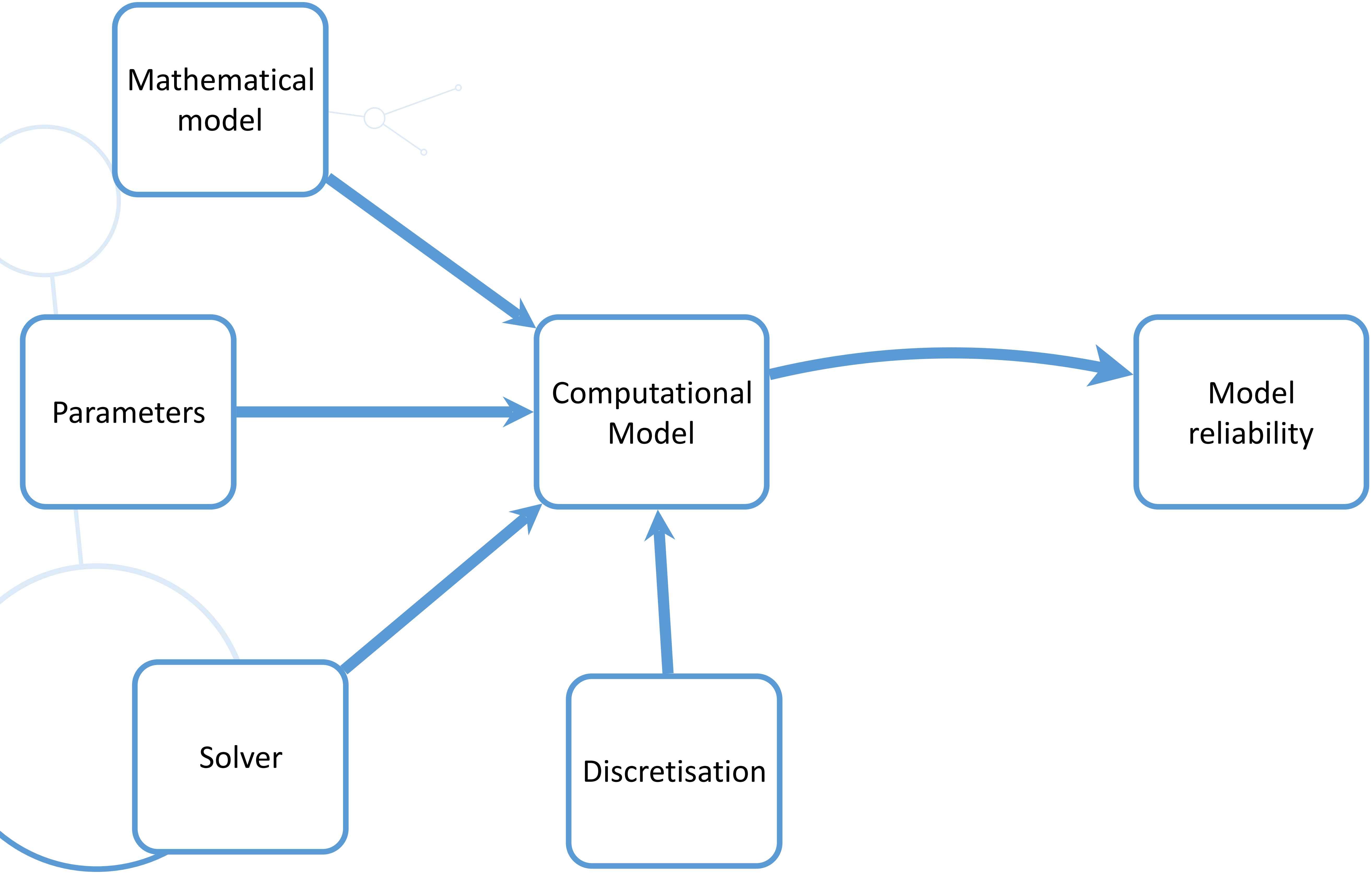
Aerospace

http://legato-team.eu — stephane.bordas@uni.lu S. Bordas, B. Peters, A. Zilian and S. Belouettar (LIST)

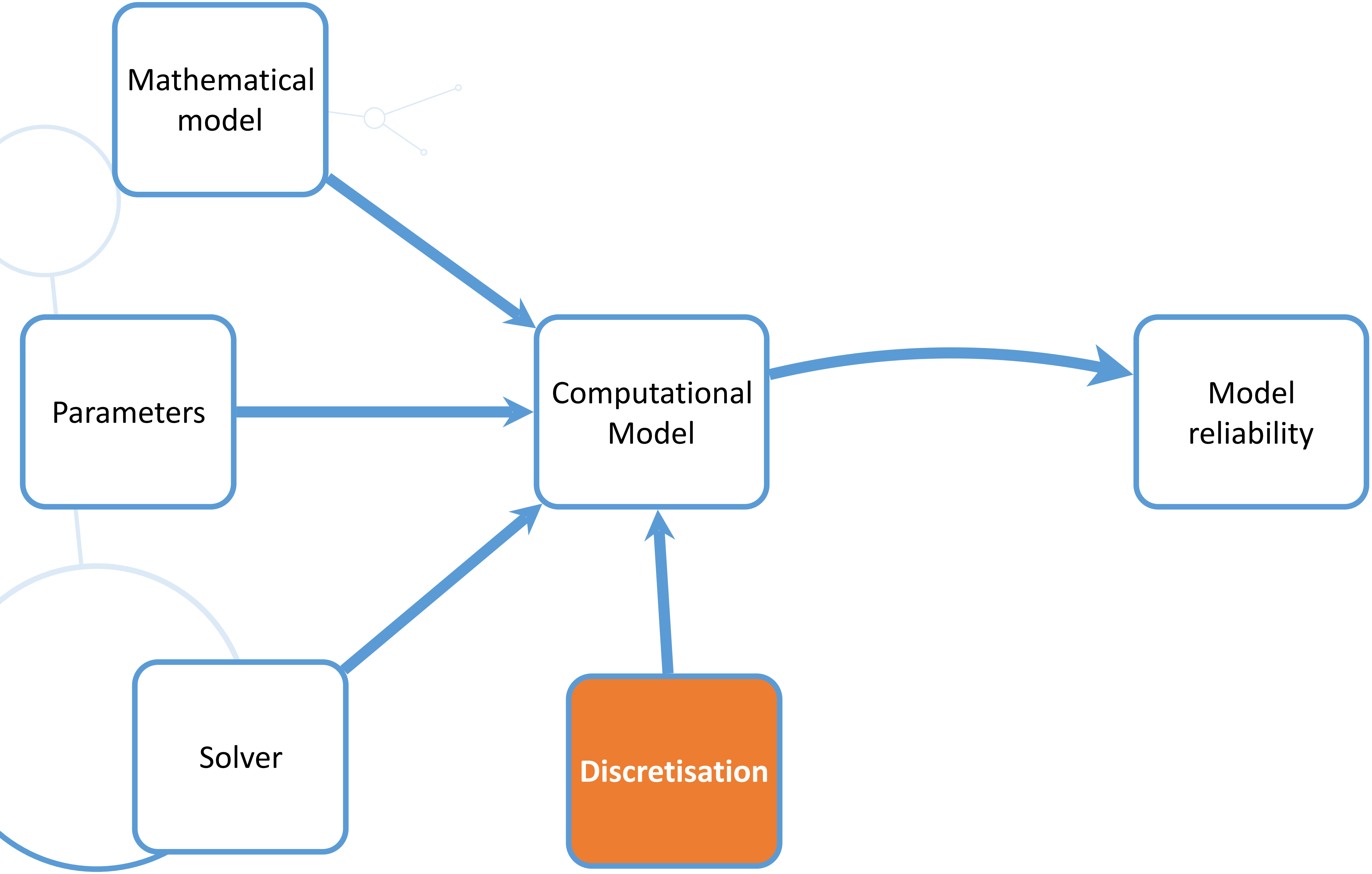
Enabling methodologies

Building blocks

Enabling methodologies



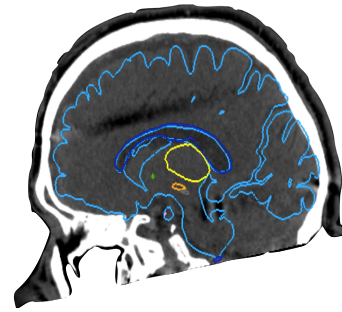
Enabling methodologies



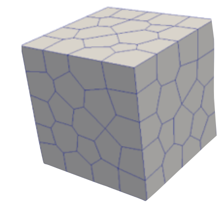
Enabling methodologies

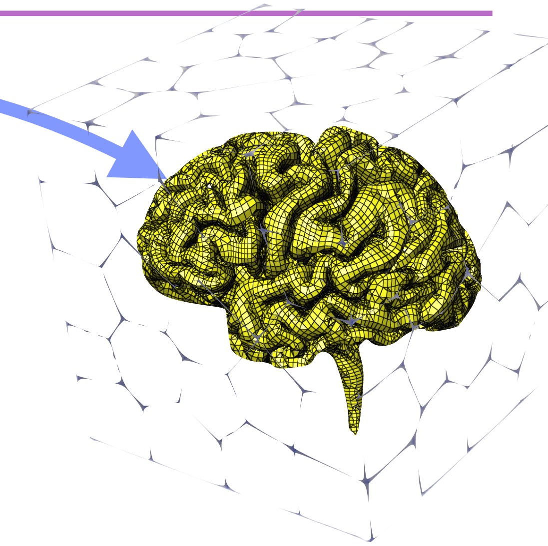
Discretisation

Image to mesh



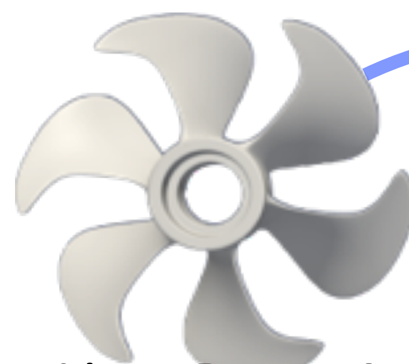
virtual elts



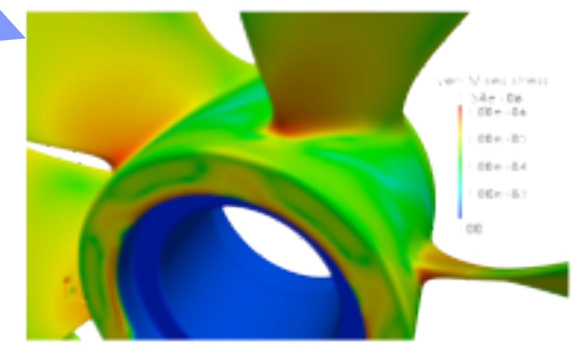


Sundararajan, Tomar, SB, 2015

CAD to mesh

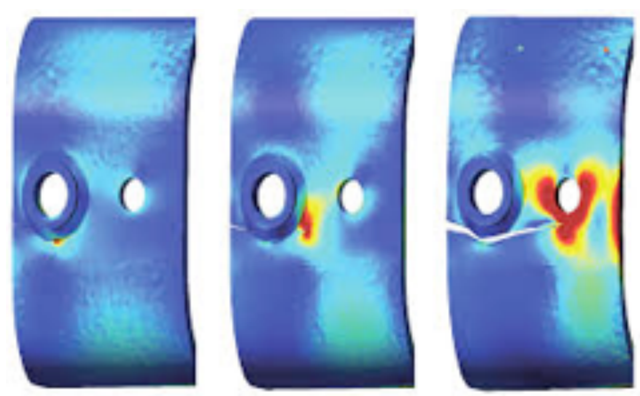


IGABEM



Lian, Peng, Atroshchenko, 2015

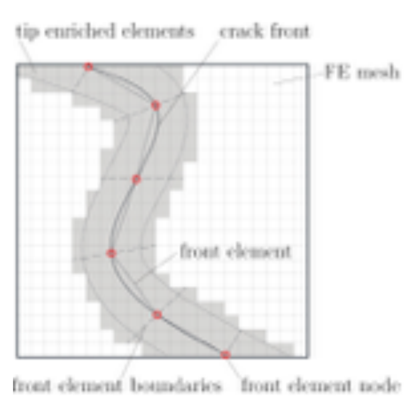
a posteriori error

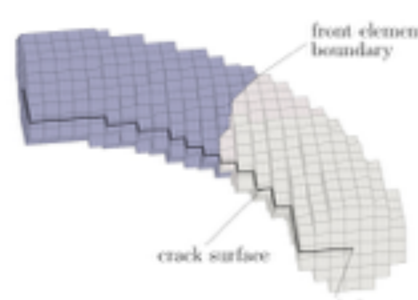


Cenaero

Duflot, SB, Wyart, Pierard, Jin, 2015

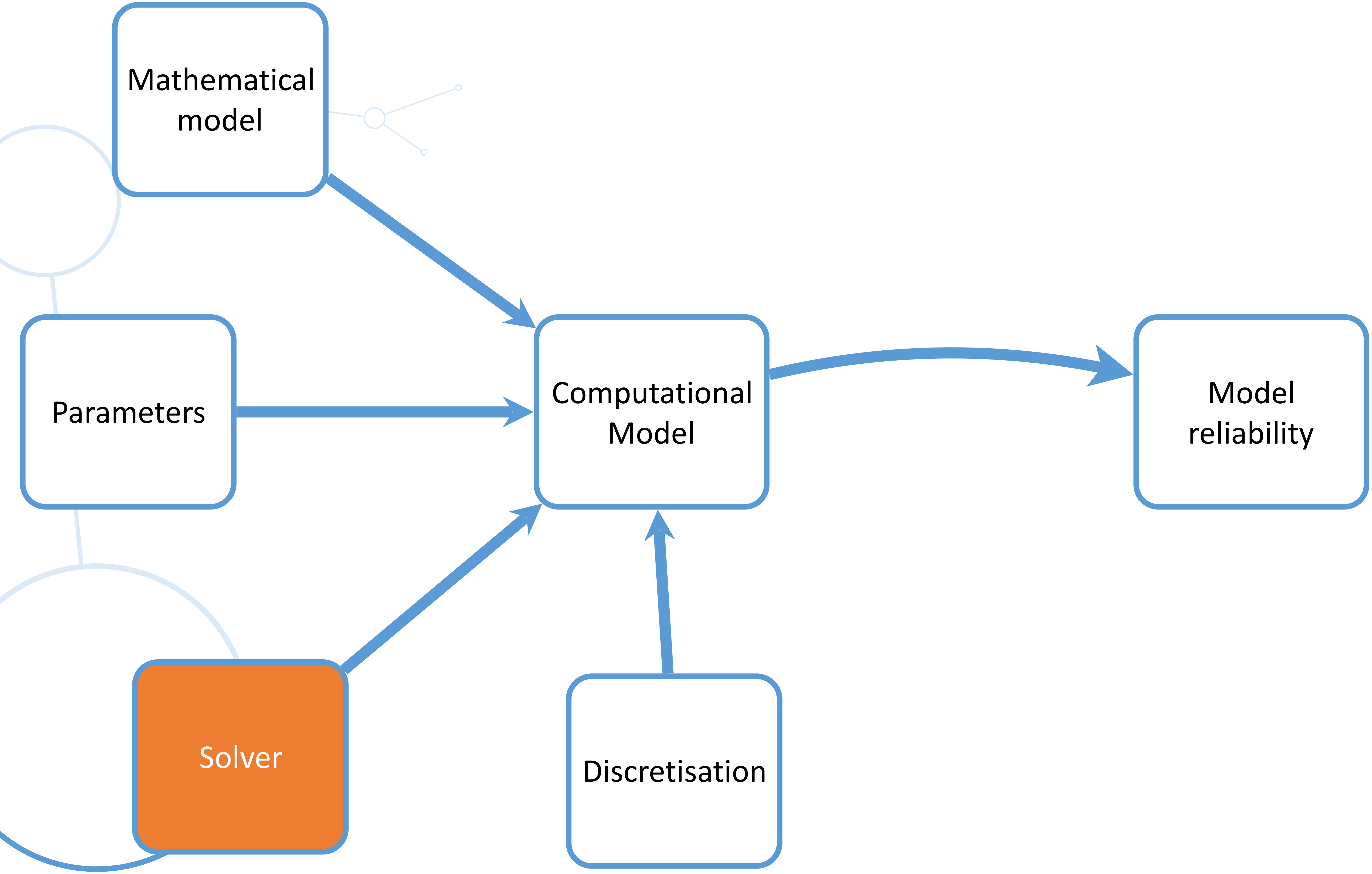
Enrichment Meshless

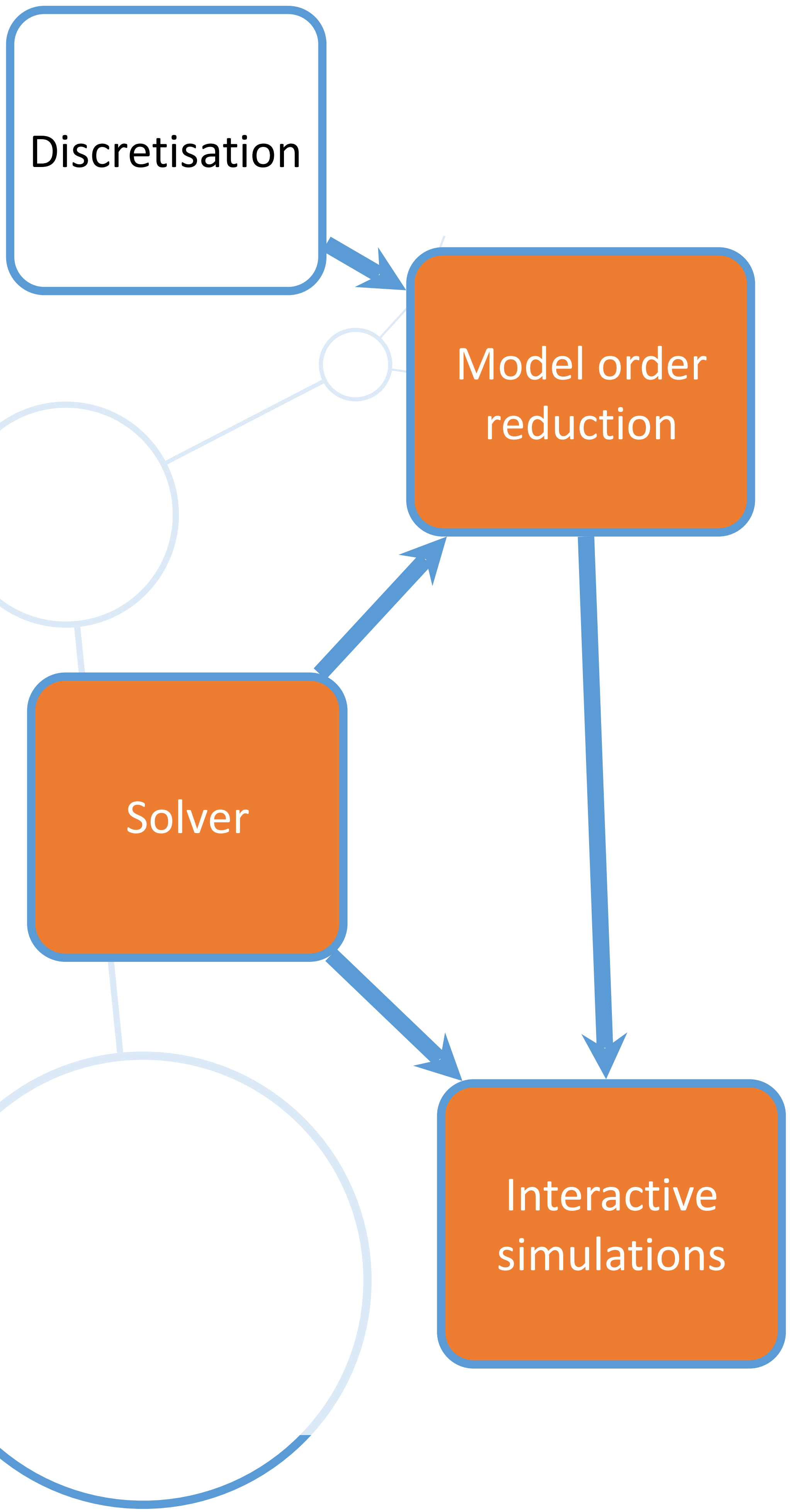


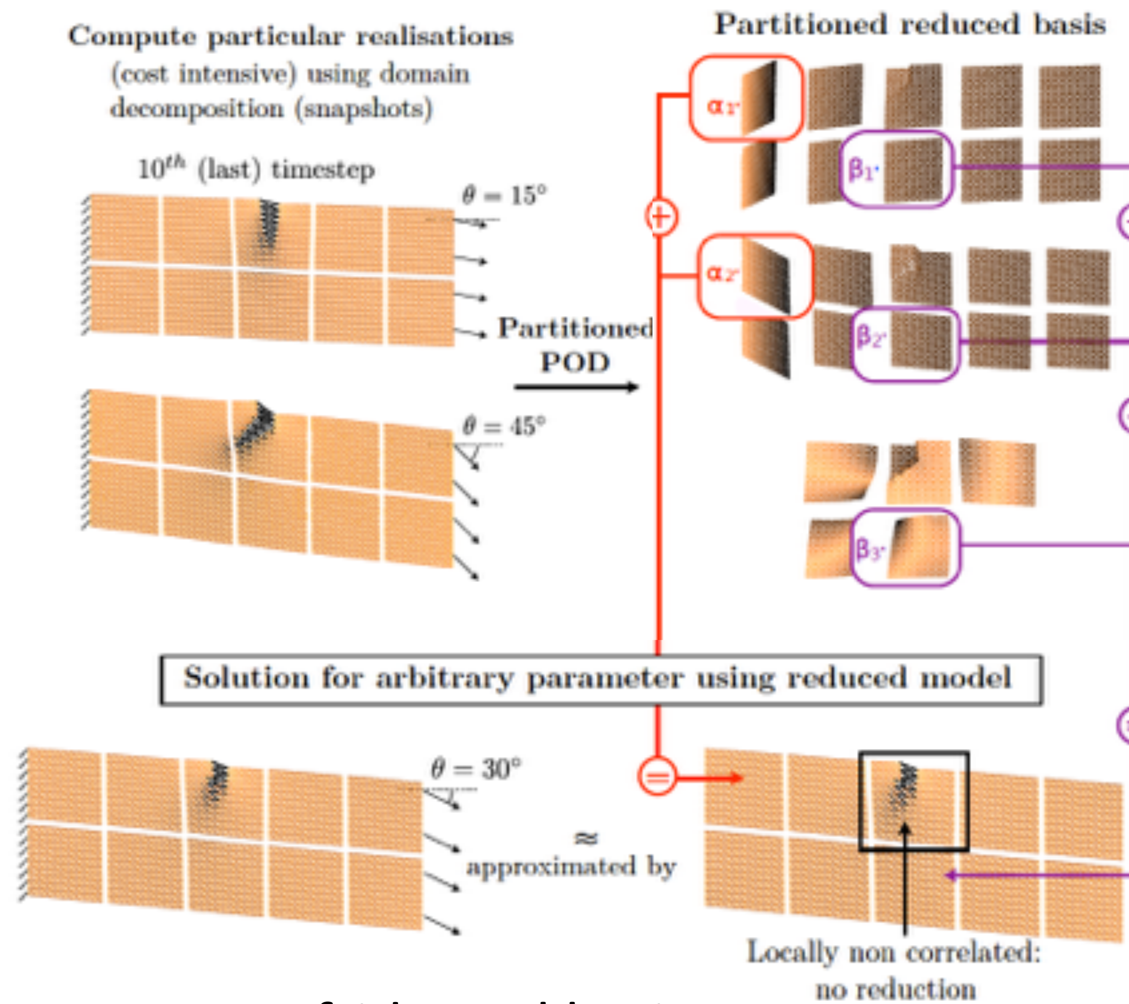


Agathos, Chatzi, SB, 2015

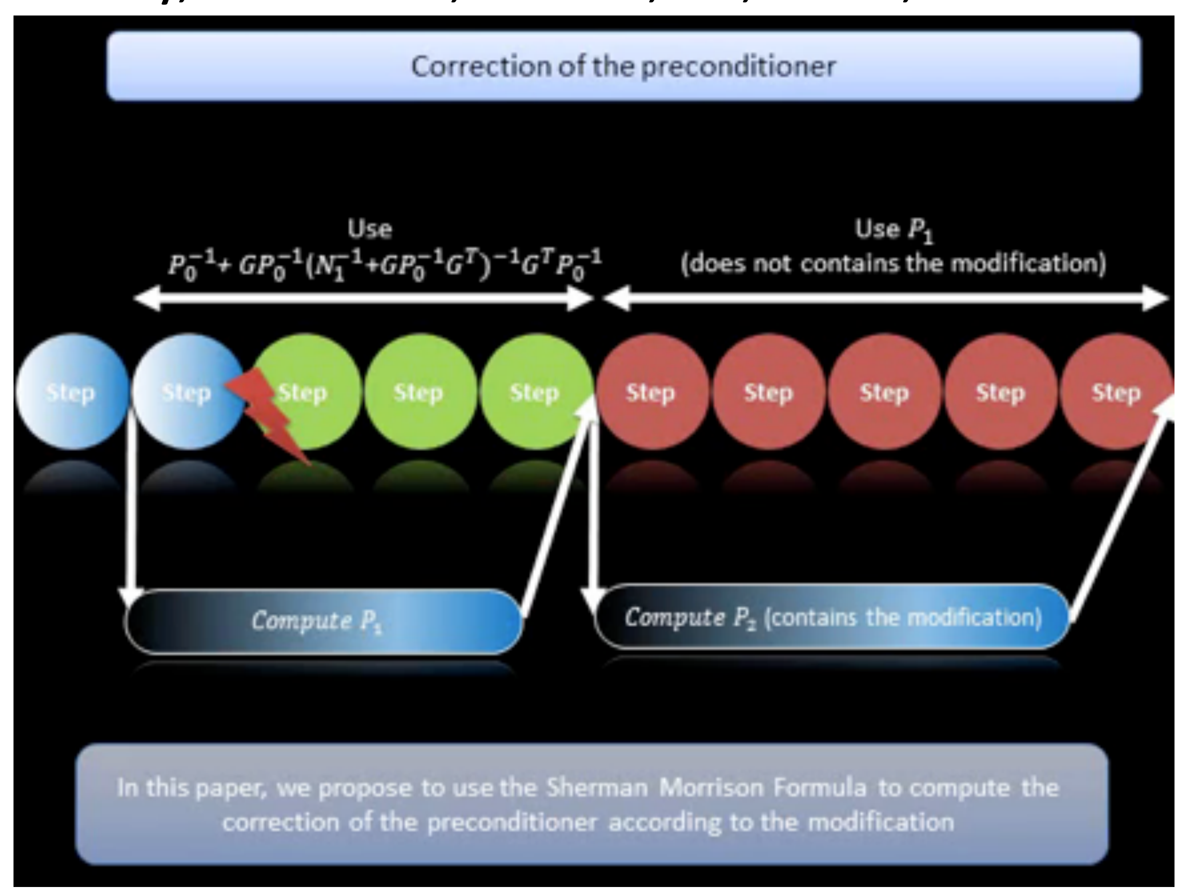
Enabling methodologies





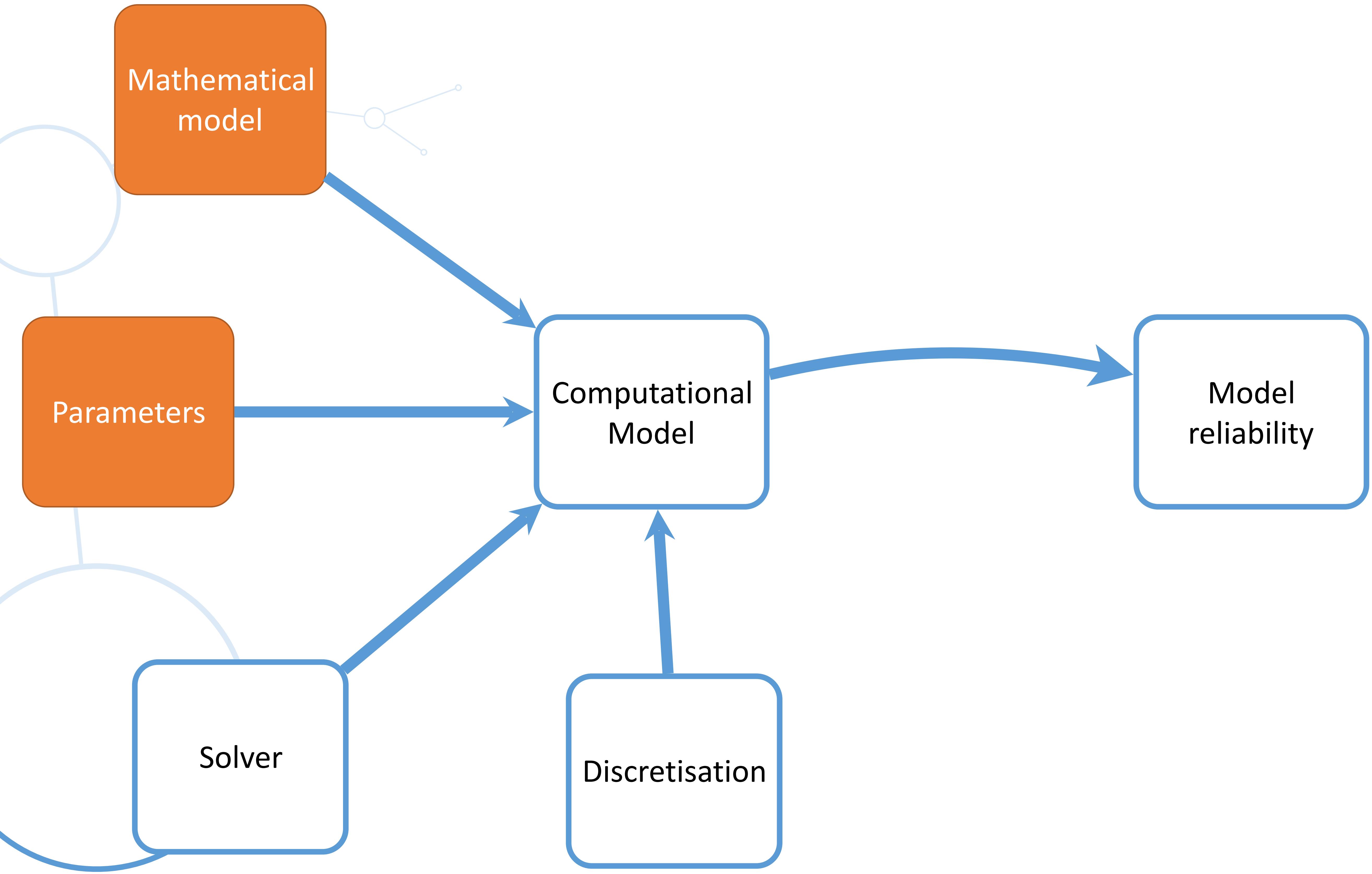


Goury, Kerfriden, Akbari, SB, 2014, 2016

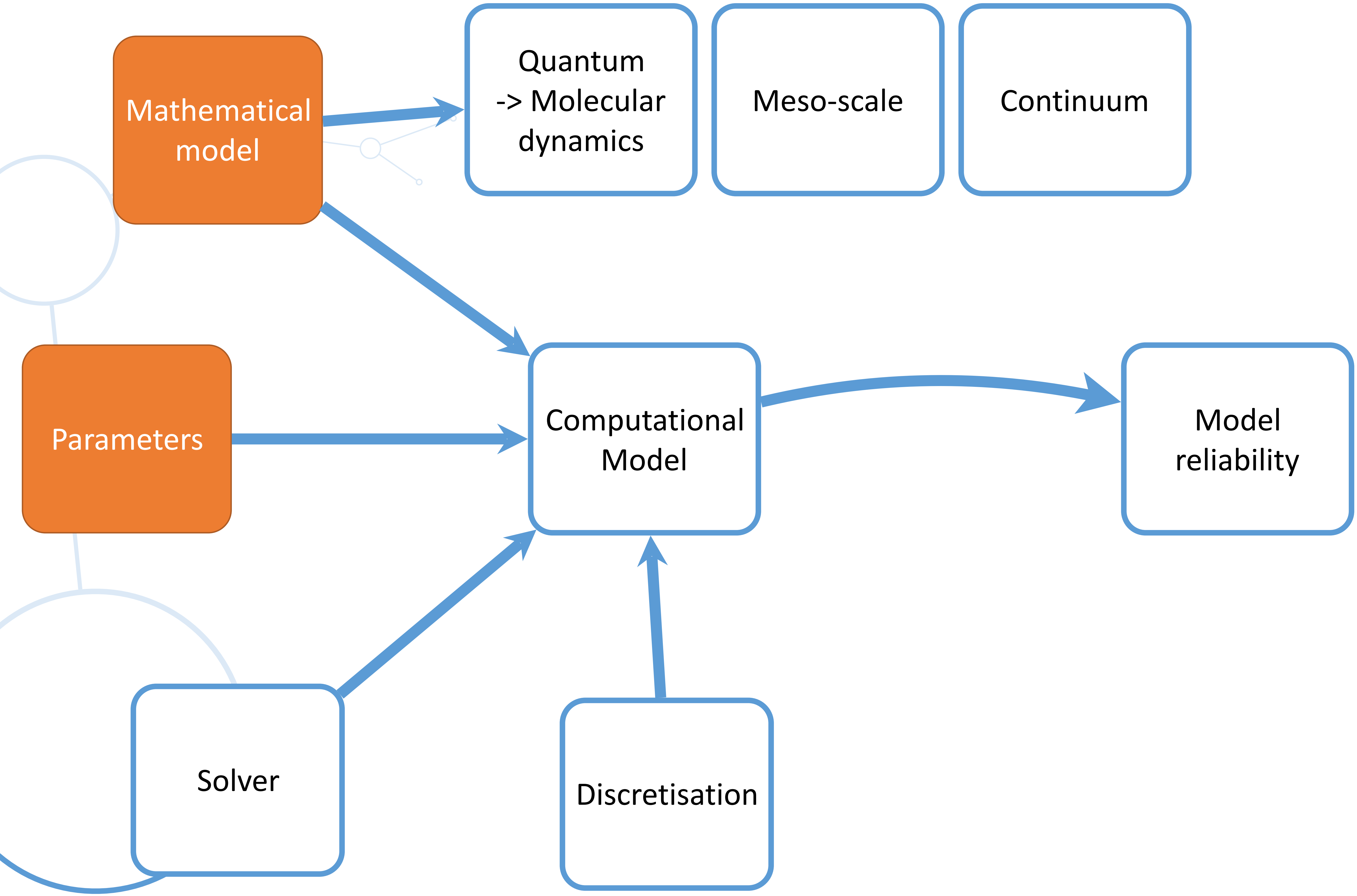


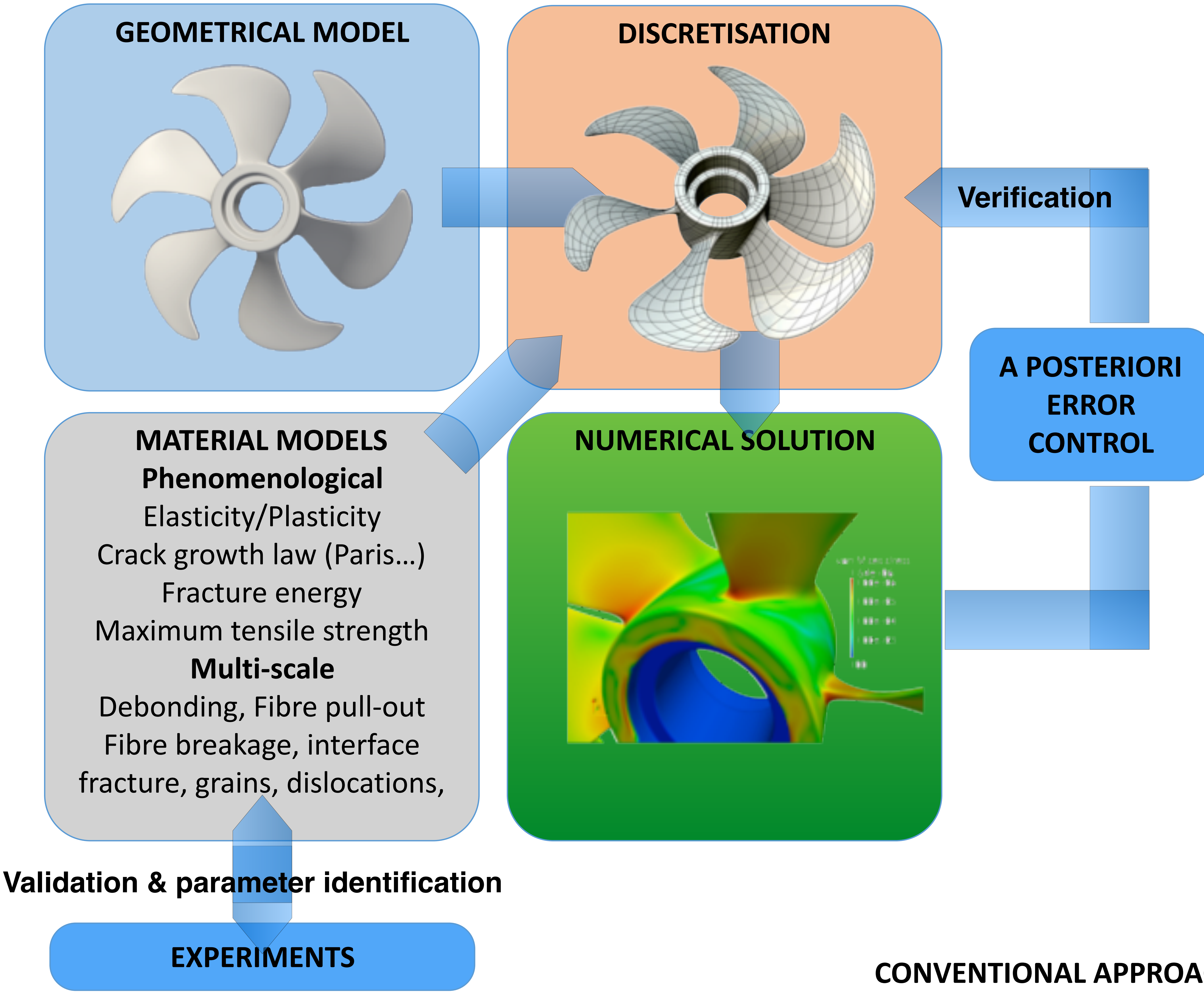
Courtecuisse, Kerfriden, SB, et al. 2014

Enabling methodologies

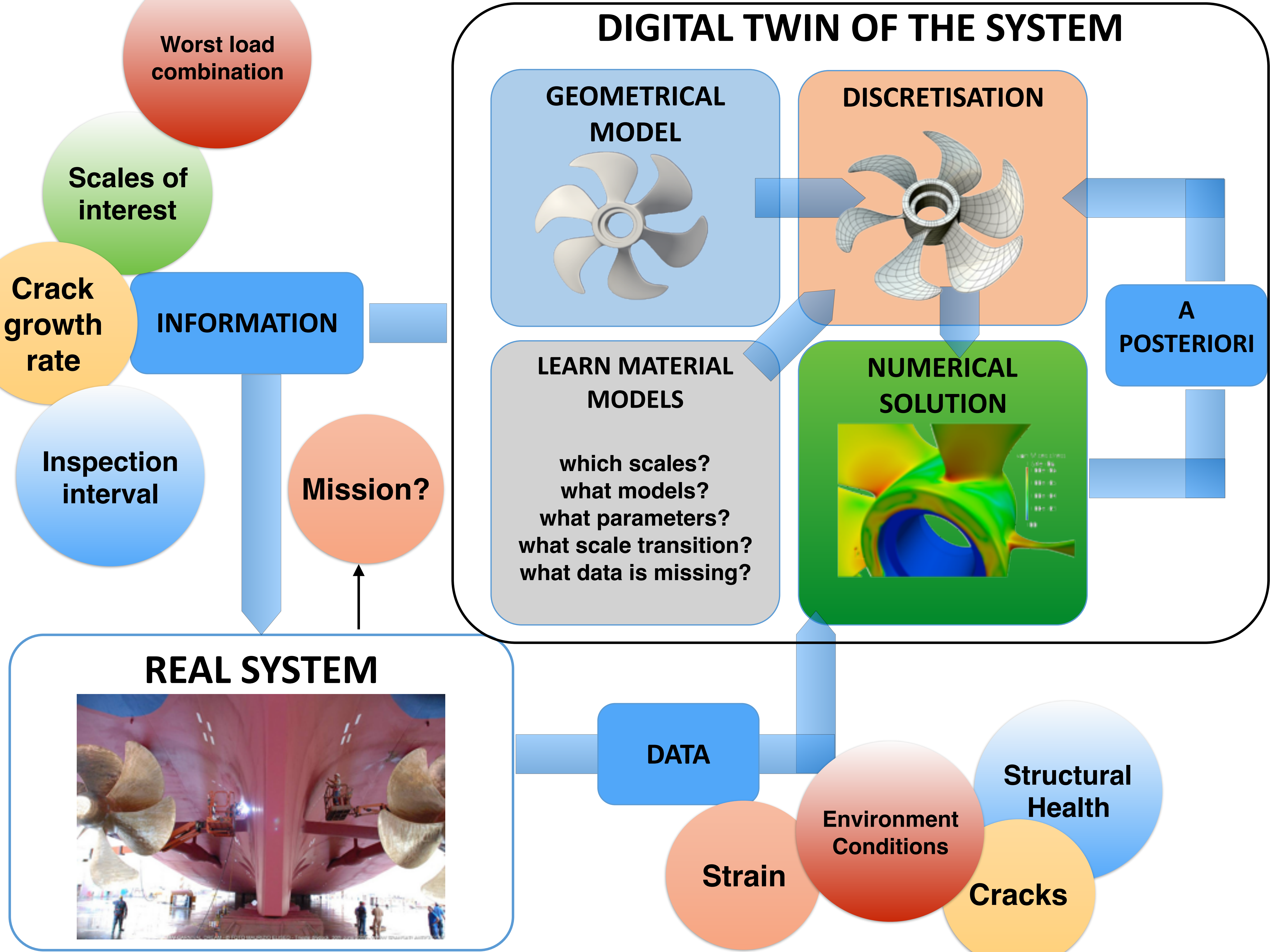


Enabling methodologies





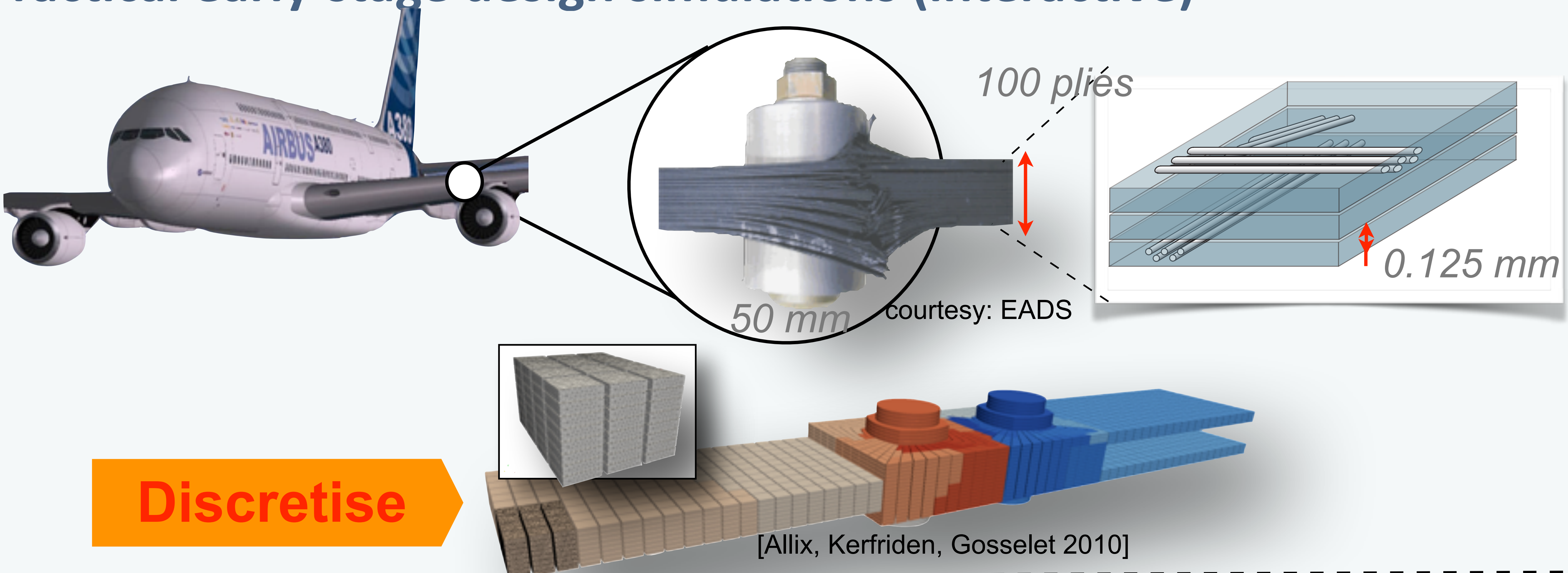
CONVENTIONAL APPROACH


DIGITAL TWIN OF THE SYSTEM
GEOMETRICAL MODEL
DISCRETISATION
LEARN MATERIAL MODELS
which scales?
what models?
what parameters?
what scale transition?
what data is missing?
NUMERICAL SOLUTION
A POSTERIORI
Worst load combination
Scales of interest
Crack growth rate
INFORMATION
Inspection interval
Mission?
REAL SYSTEM
DATA
Strain
Environment Conditions
Structural Health
Cracks

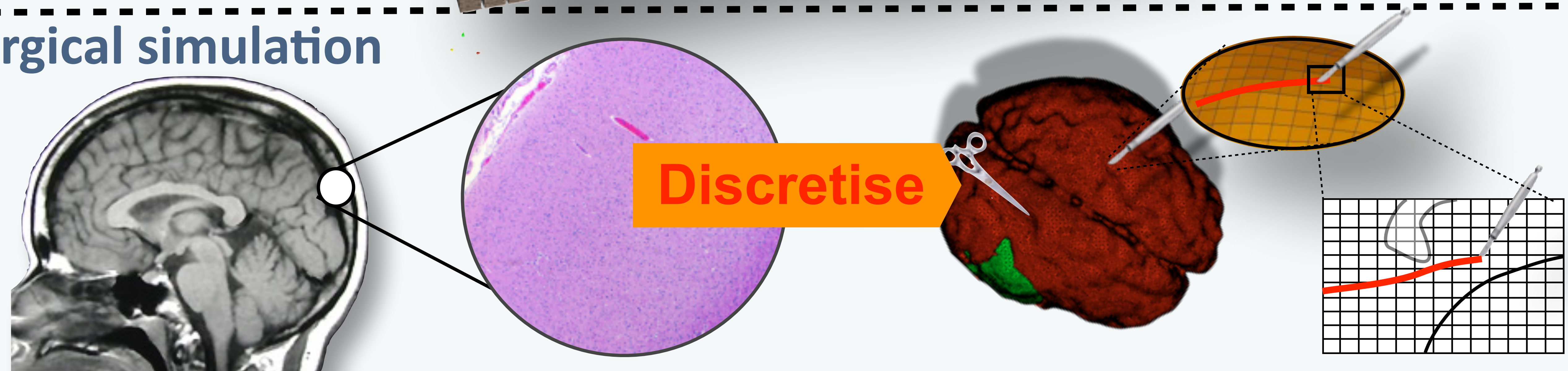

www.imagespk.com www.imagespk.com www.imagespk.com www.imagespk.com

Motivation: multiscale fracture of engineering structures and materials

Practical early-stage design simulations (interactive)



Surgical simulation

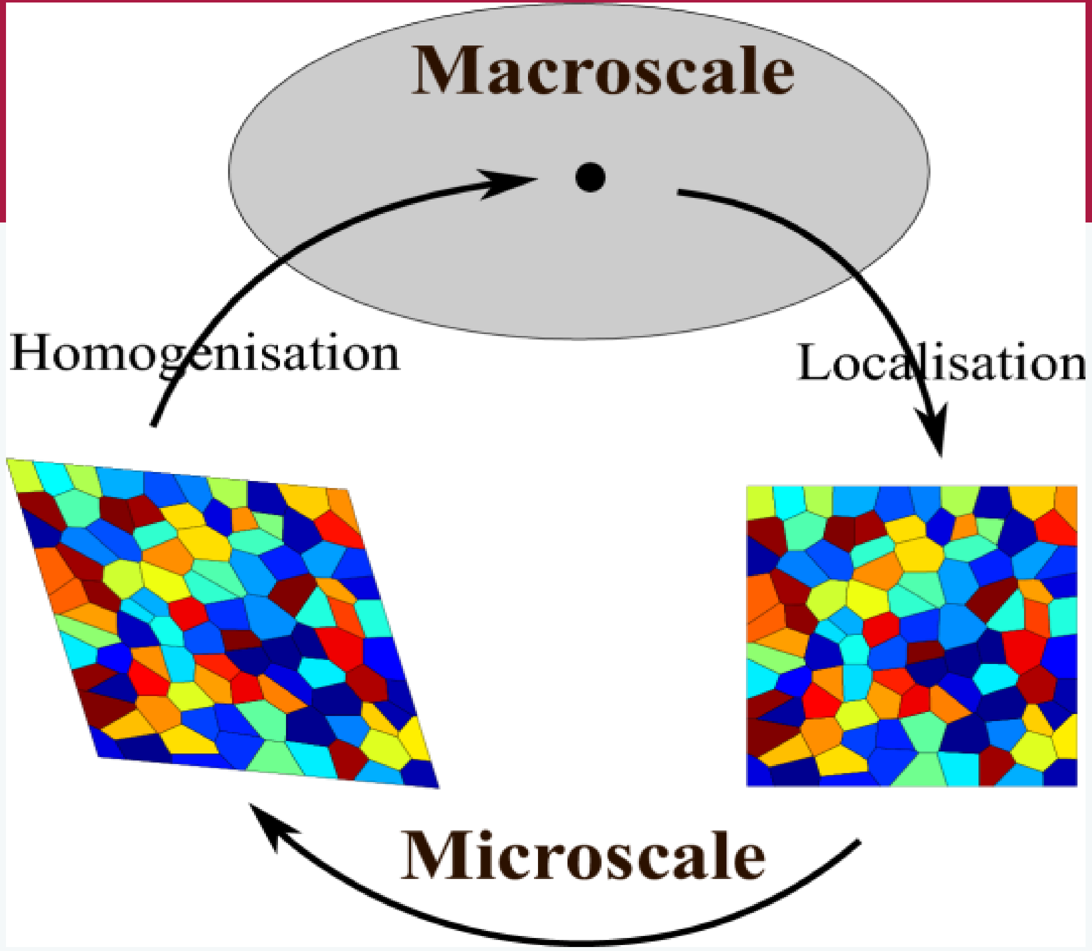


- Reduce the problem size while controlling the error (in QoI) when solving very large (multiscale) mechanics problems

Ways to reduce the models

- Homogenisation (FE^2, etc.) - Hierarchical
- Concurrent and hybrid (bridging domain, ARLEQUIN, etc.)
- Enrichment (PUFEM, XFEM, GFEM)
- Model reduction (algebraic)

Reduction methods based on homogenisation



Definition of an RVE

$$l^c \gg l^f \gg l^g$$

Coupling of macroscopic and microscopic levels

The volume averaging theorem is postulated for:

1) Strain tensor:

$$\boldsymbol{\epsilon}^c = \frac{1}{|\Omega(\mathbf{x}^c)|} \int_{\partial\Omega(\mathbf{x}^c)} \mathbf{u}^f \otimes_s \mathbf{n} \, d\Gamma$$

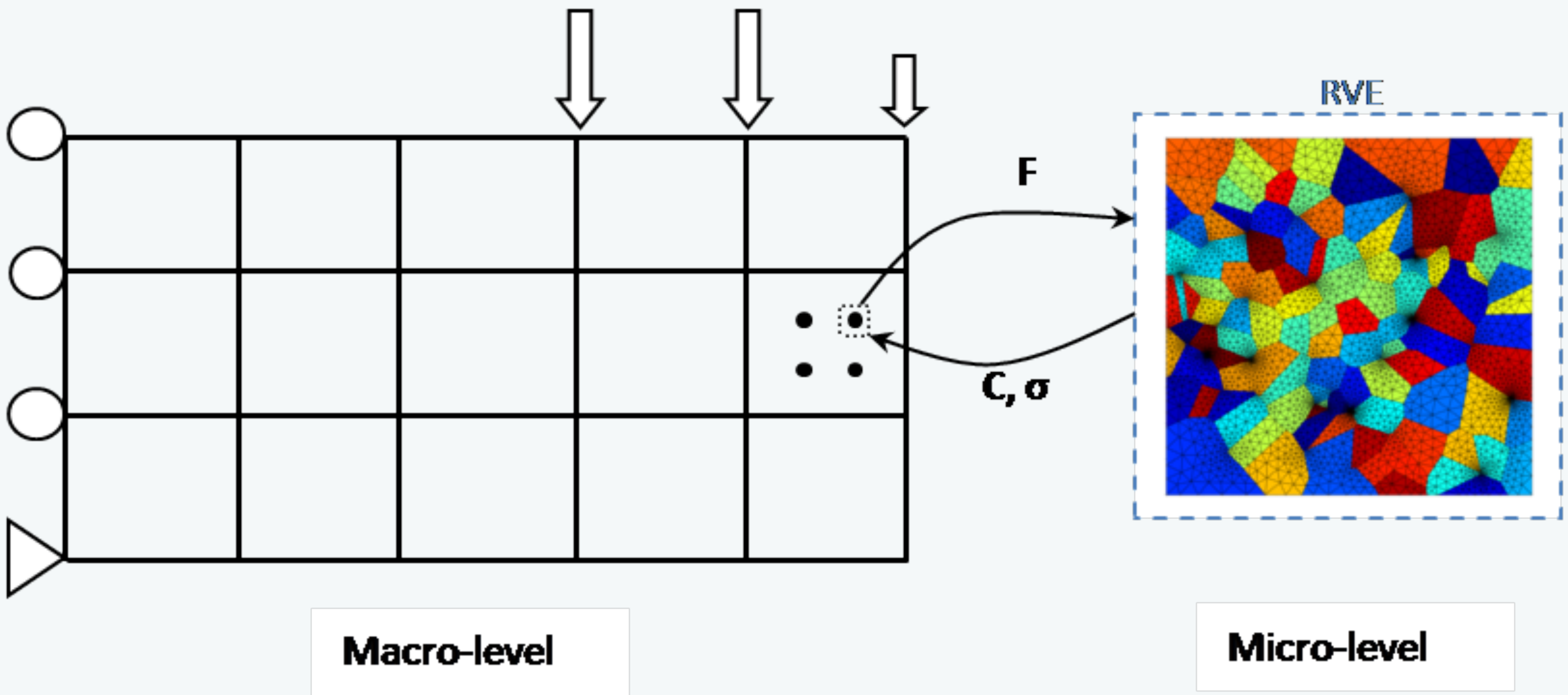
2) Virtual work (Hill-Mandel condition):

$$\boldsymbol{\sigma}^c : \delta\boldsymbol{\epsilon}^c = \frac{1}{|\Omega(\mathbf{x}^c)|} \int_{\partial\Omega(\mathbf{x}^c)} \mathbf{t}^f \cdot \delta\mathbf{u}^f \, d\Gamma$$

3) Stress tensor:

$$\boldsymbol{\sigma}^c = \frac{1}{|\Omega(\mathbf{x}^c)|} \int_{\partial\Omega(\mathbf{x}^c)} \mathbf{t}^f \otimes \mathbf{x}^f \, d\Gamma$$

Hierarchical multi-scale approaches (FE^2)



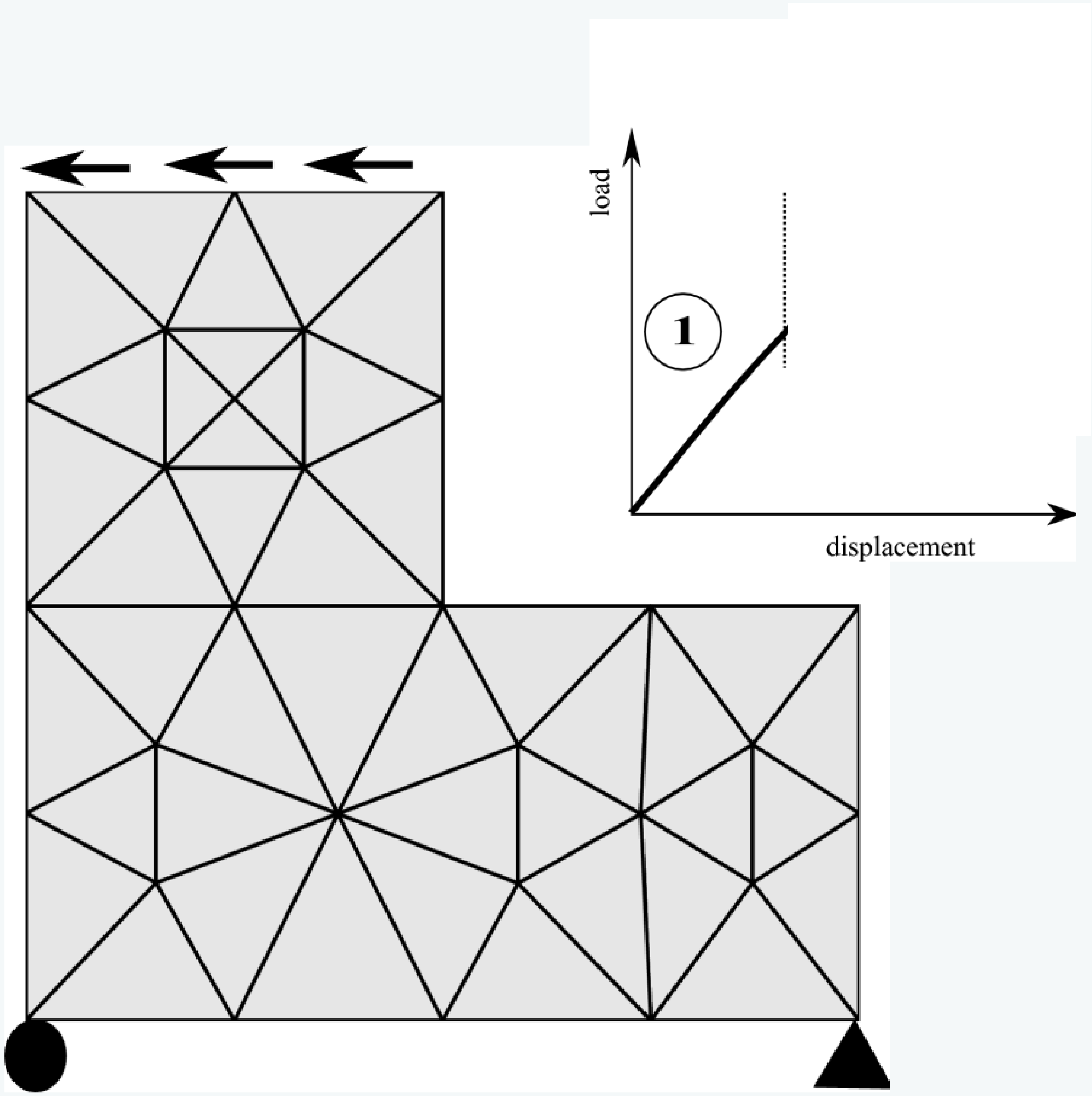
Advantages and abilities:

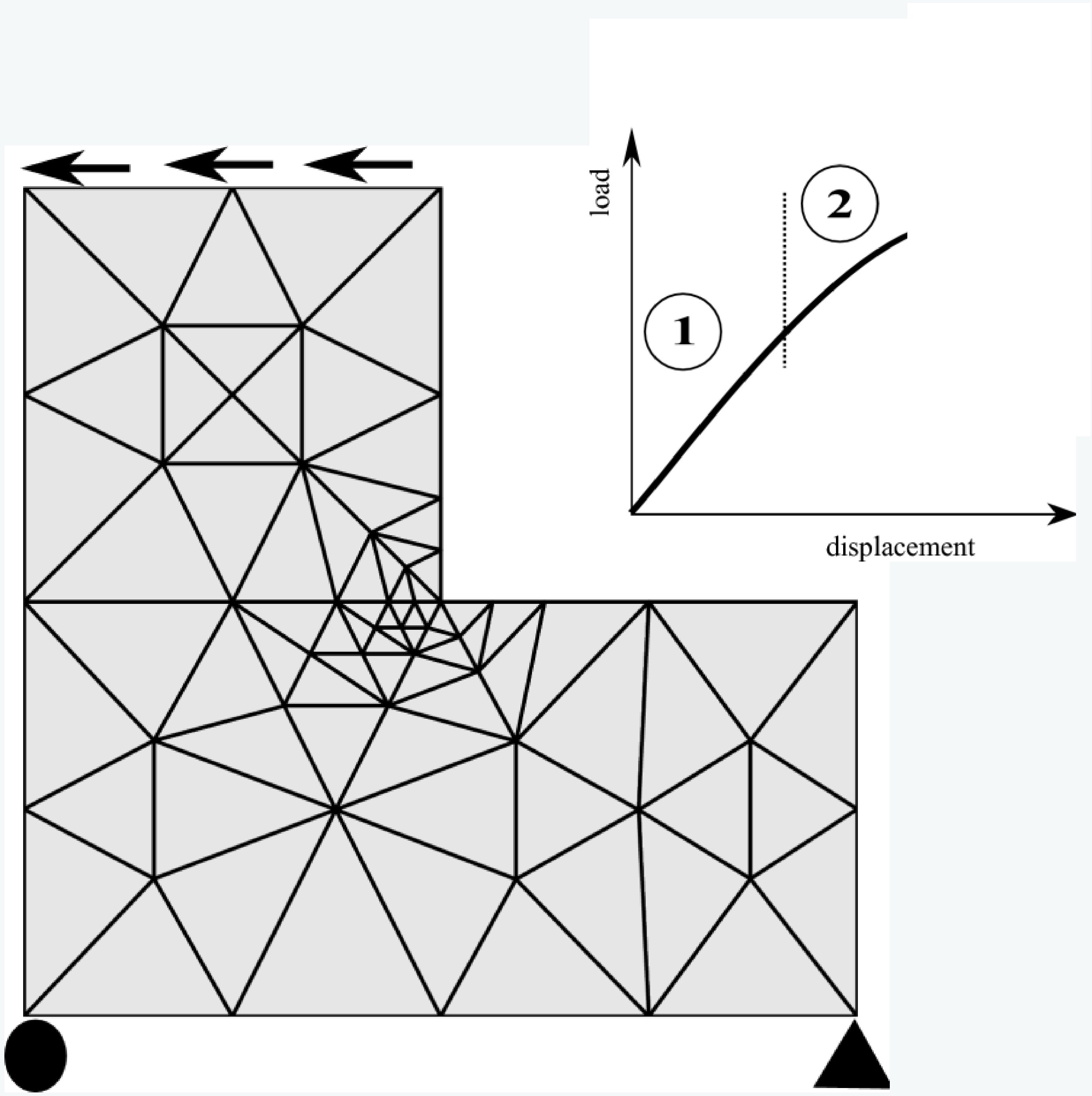
The macroscopic constitutive law is not required
Non-linear material behaviour can be simulated
Microscale behaviour of material is monitored at each load step

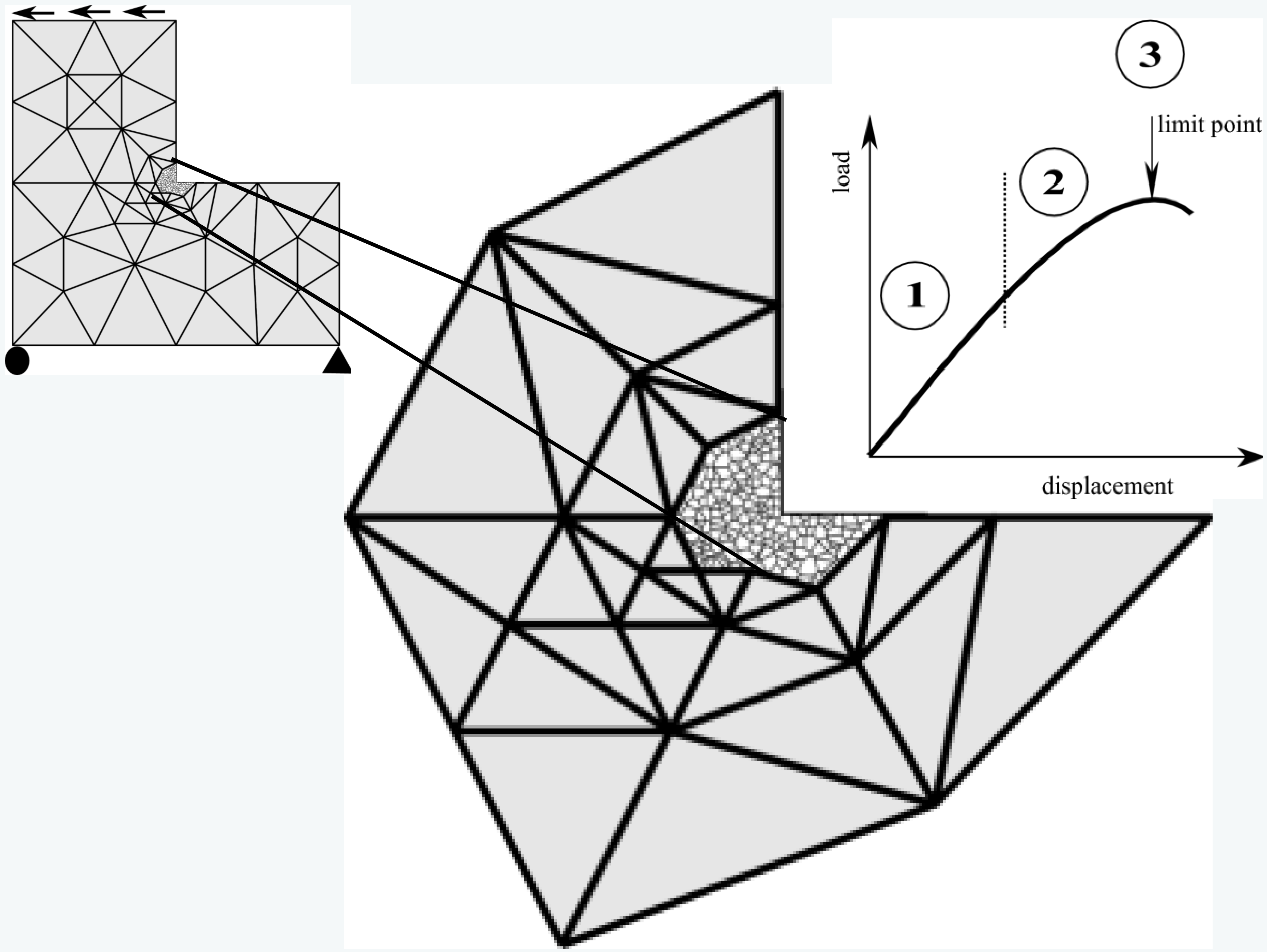
Drawbacks:

In softening regime:

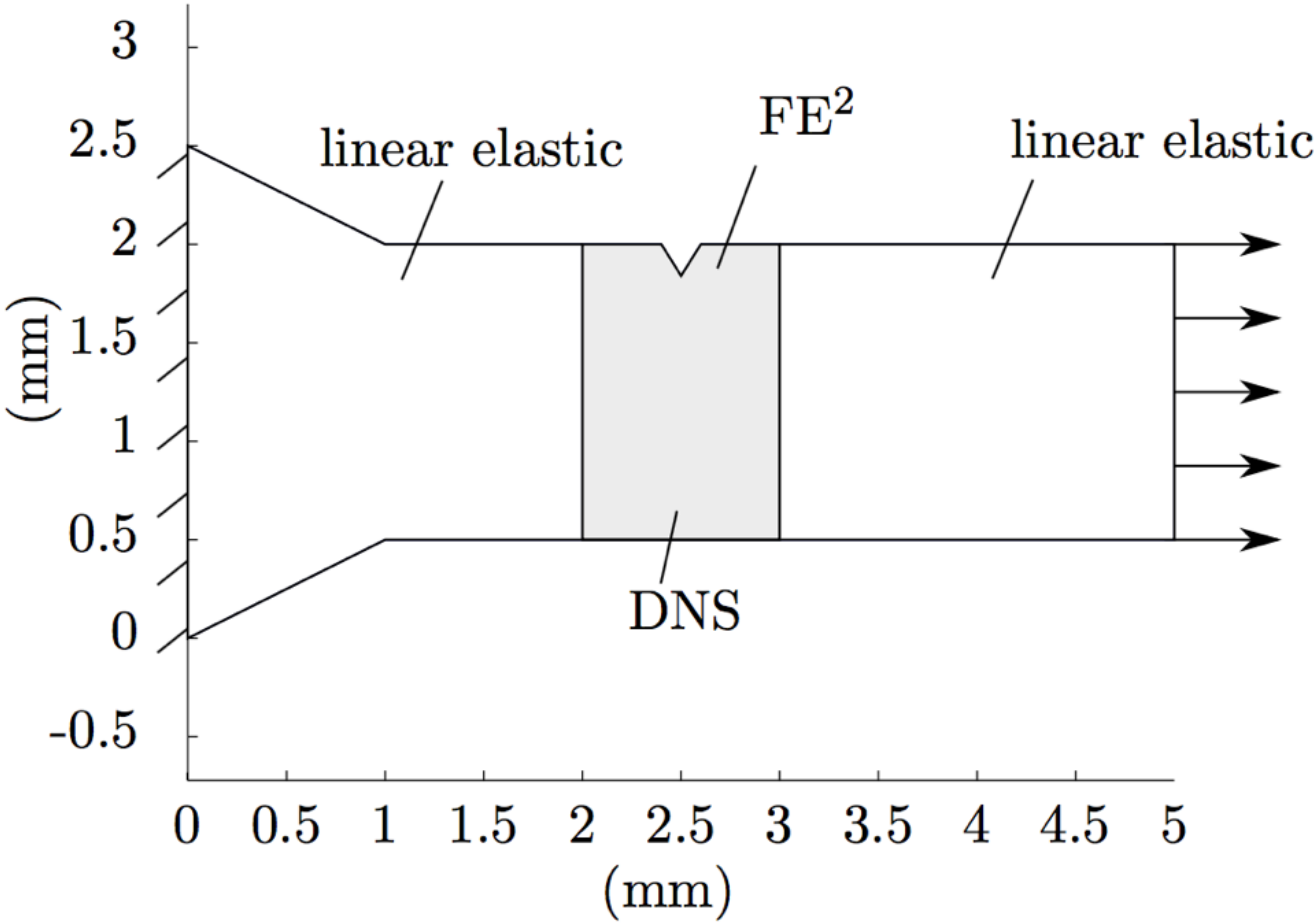
- Lack of scale separation
- Macroscale mesh dependence


load
1
displacement

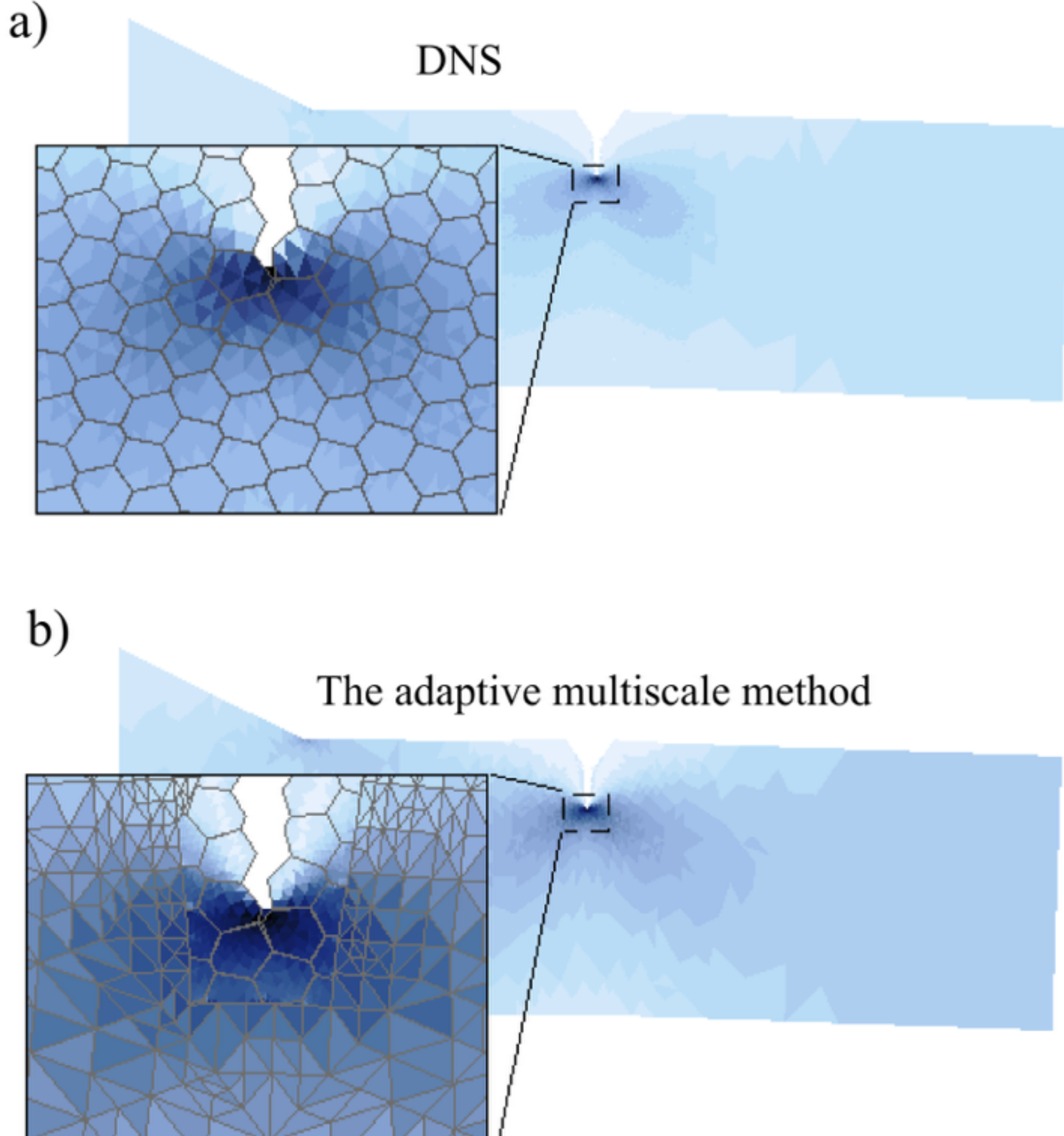

load
displacement
1
2



Details in Phil. Magazine, 2015, Akbari, Kerfriden, Bordas

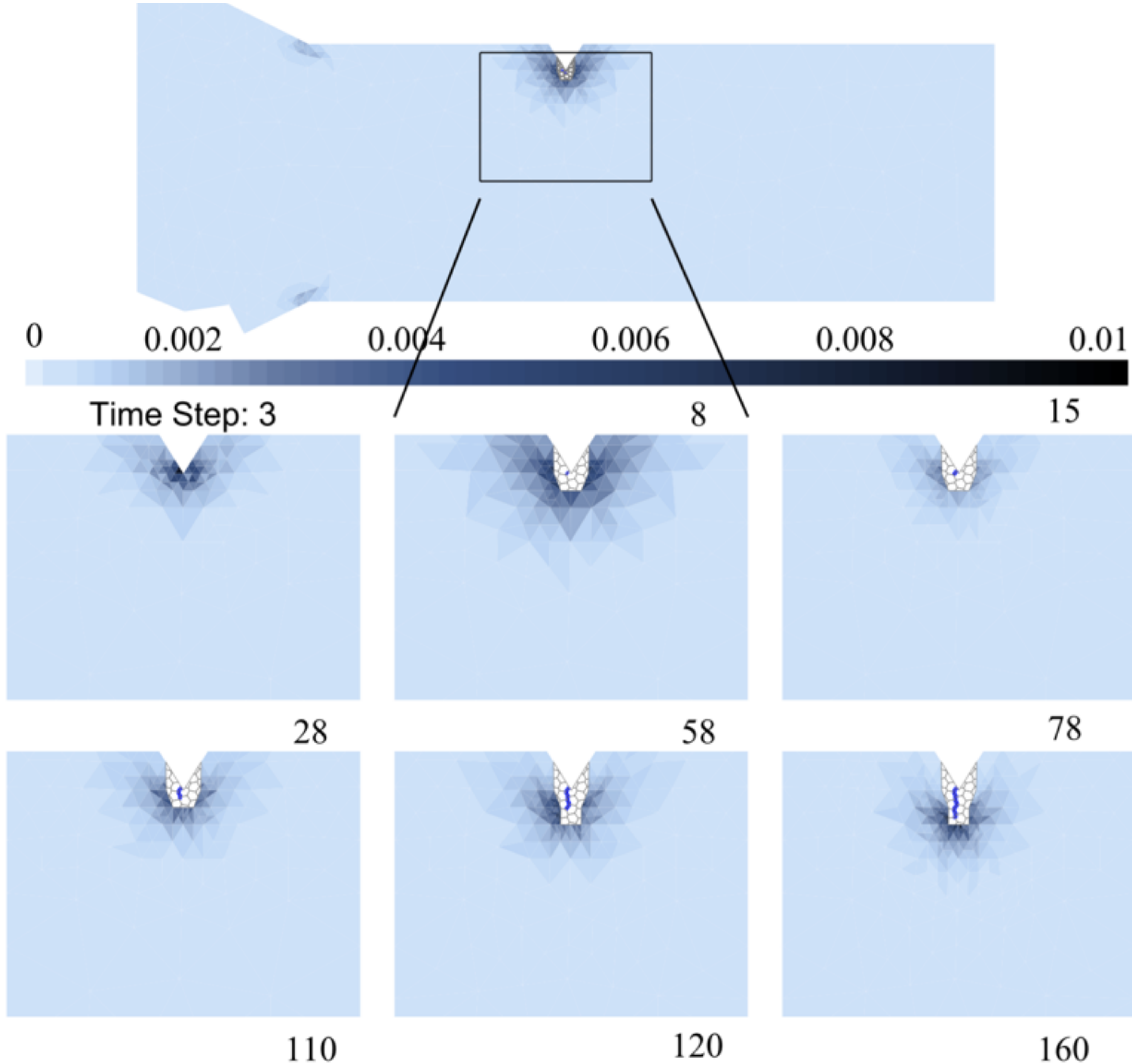


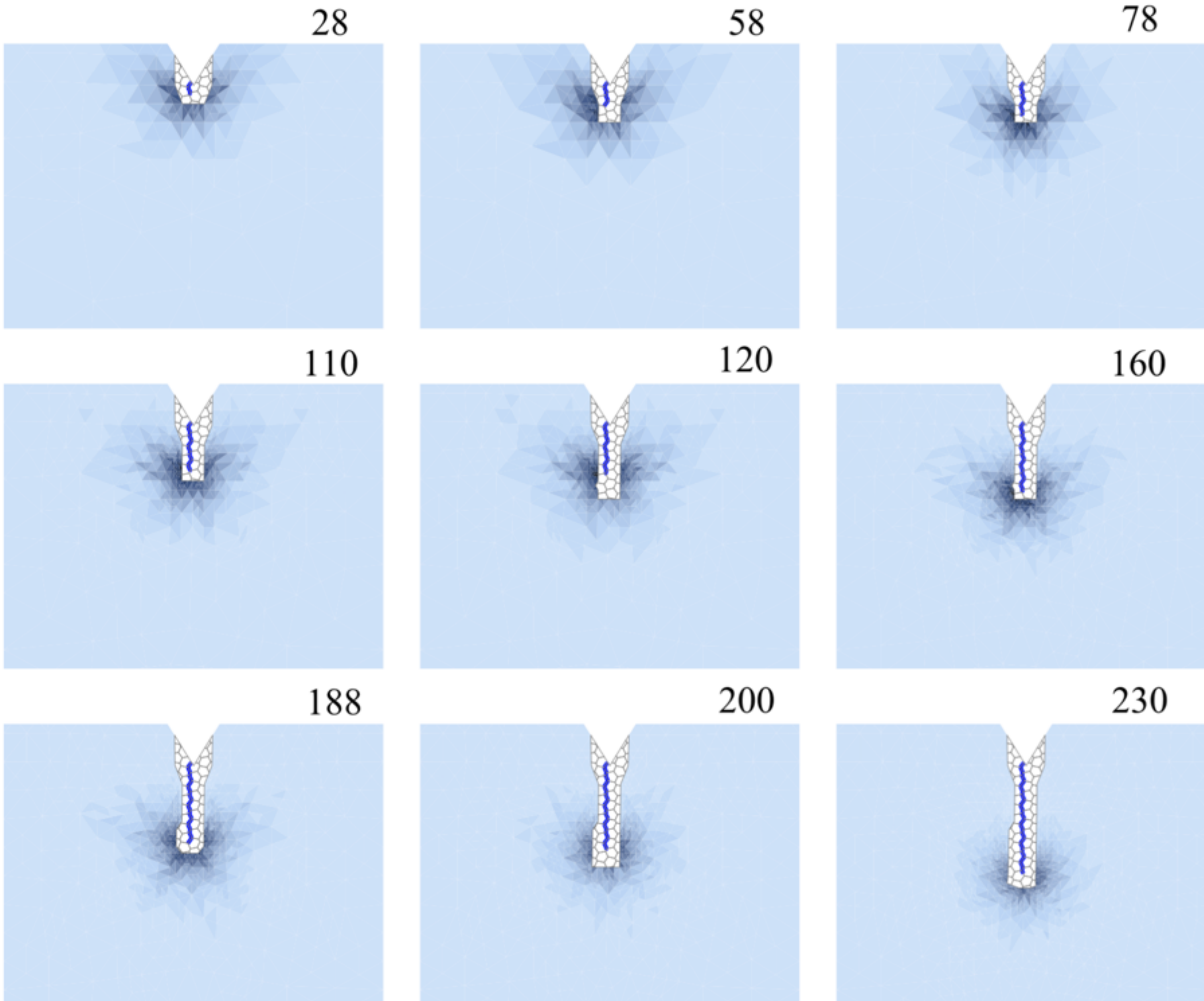
linear elastic
FE^2
linear elastic
DNS
(mm)
(mm)
3
2.5
2
1.5
1
0.5
0
-0.5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5

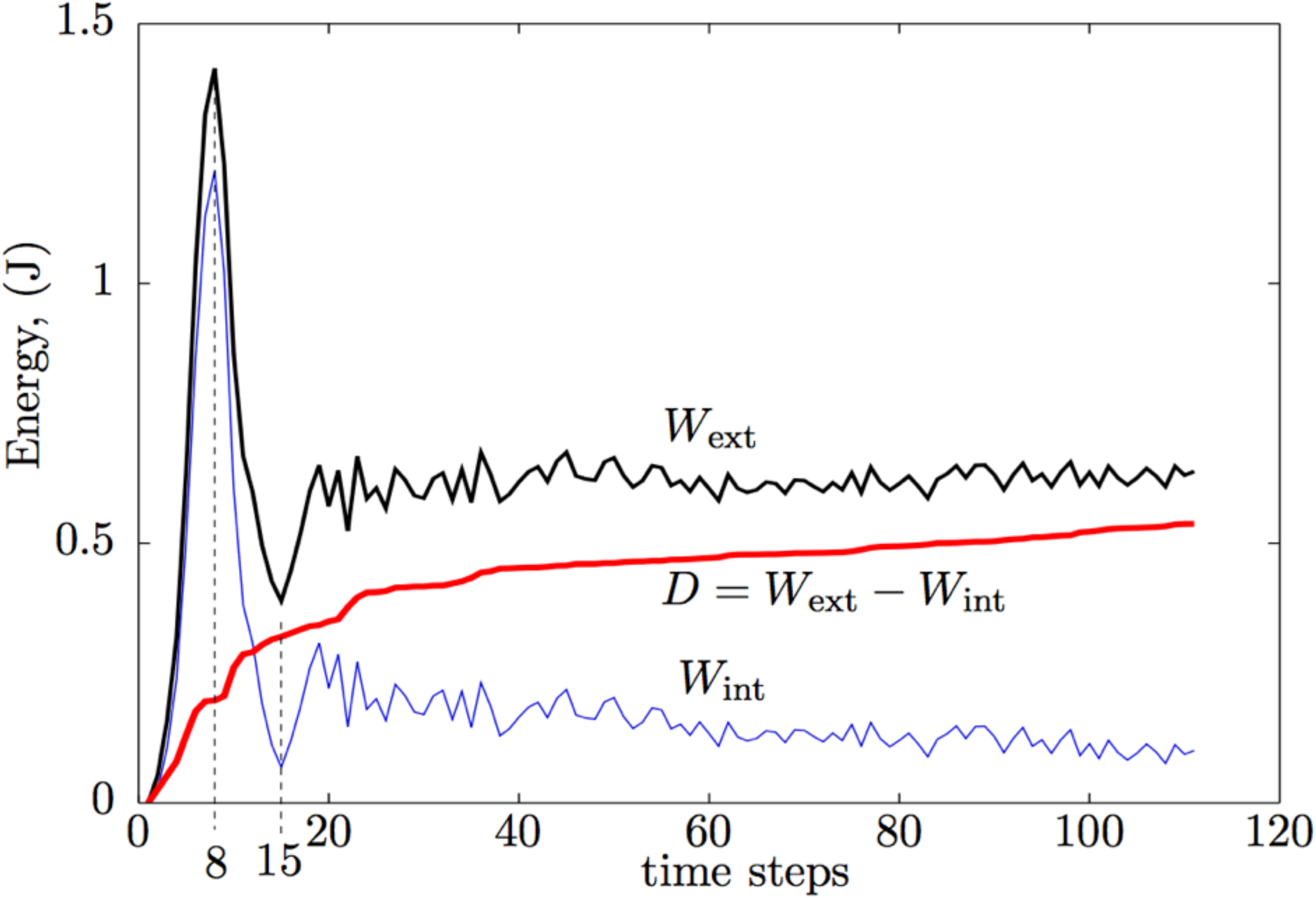


a)
DNS
b)
The adaptive multiscale method

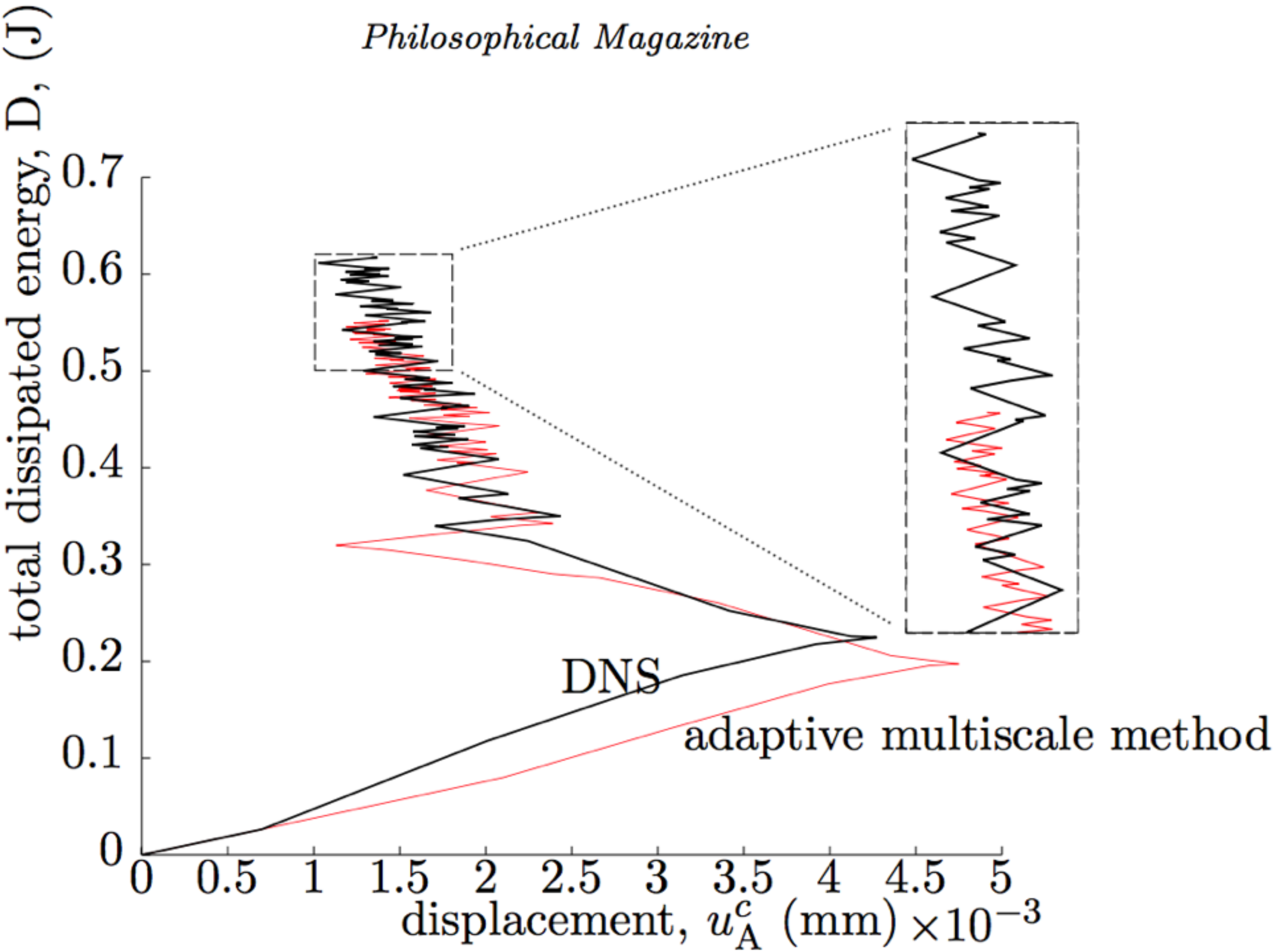
The distribution of strain-gradient sensitivity $L_{\mathcal{V}}||\nabla\nabla\mathbf{u}^c||_e$

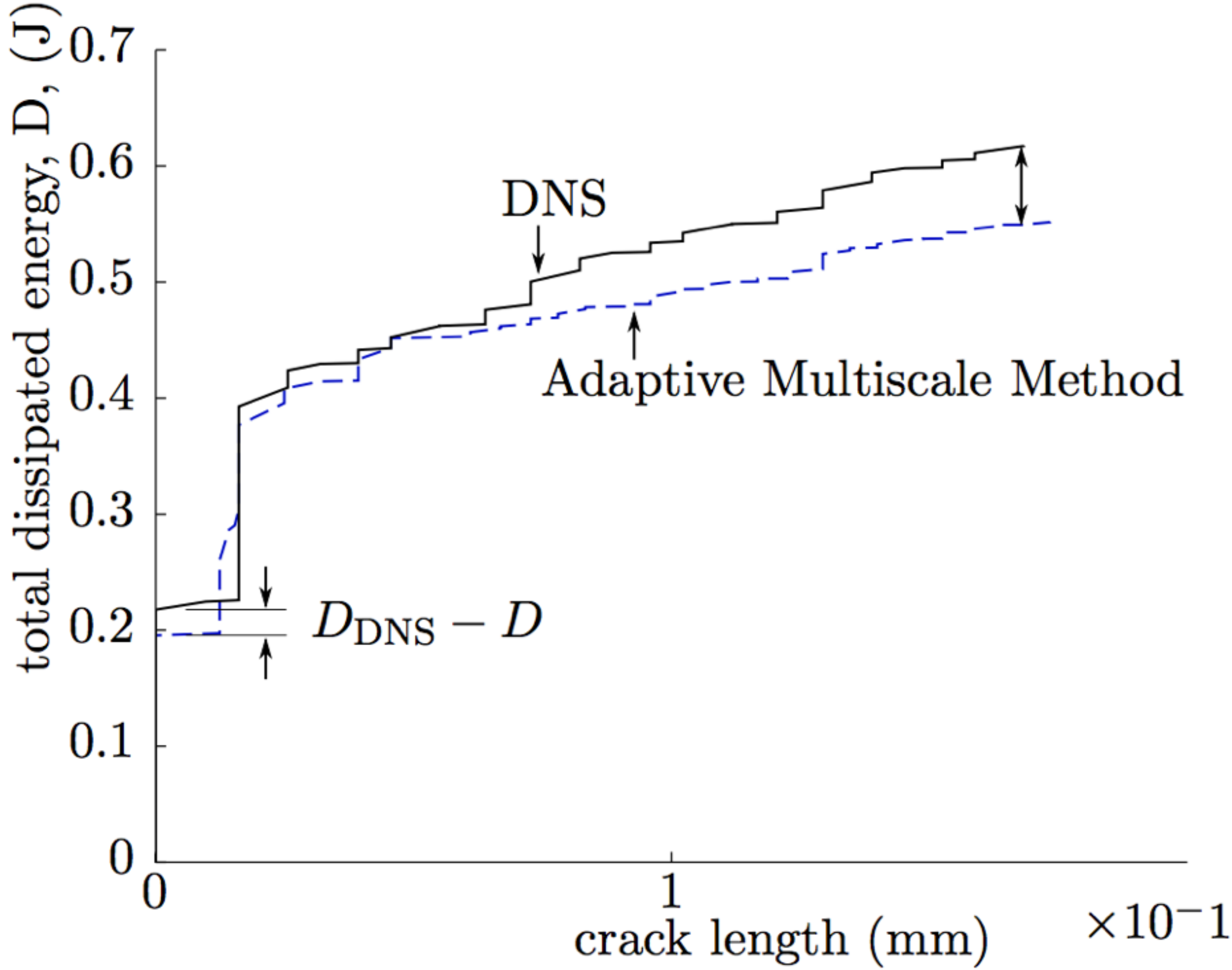



28
58
78
110
120
160
188
200
230



1.5
1
0.5
0
Energy, (J)
$W_{\rm ext}$
$D = W_{\rm ext} - W_{\rm int}$
$W_{\rm int}$
0
20
40
60
80
100
120
8
15
time steps


total dissipated energy, D, (J)
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
displacement, u_A^c (mm) $\times 10^{-3}$
DNS
adaptive multiscale method



total dissipated energy, D, (J)
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
DNS
Adaptive Multiscale Method
$D_{\mathrm{DNS}} - D$
0
1
$\times 10^{-1}$
crack length (mm)

Reduction methods based on algebraic reduction

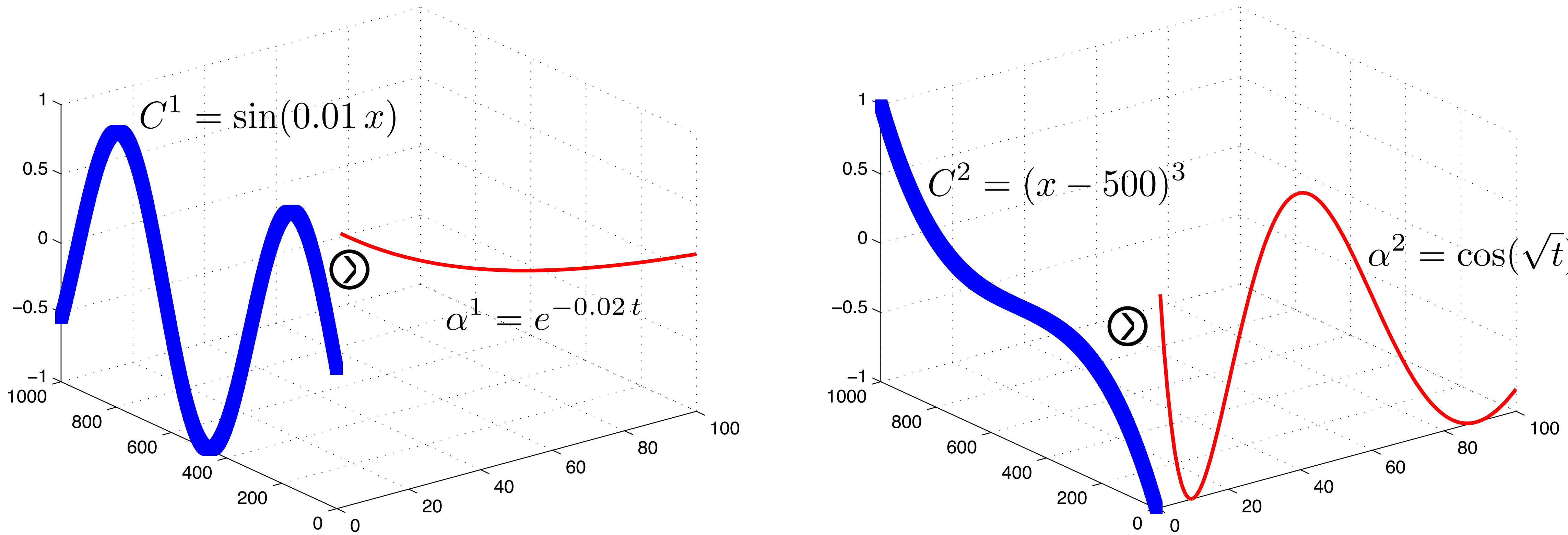

$C^1 = \sin(0.01\,x)$
$\alpha^1 = e^{-0.02\,t}$
$C^2 = (x-500)^3$
$\alpha^2 = \cos(\sqrt{t})$
1
0.5
0
−0.5
−1
1000
800
600
400
200
0
0
20
40
60
80
100

Illustration of the method of separated representation

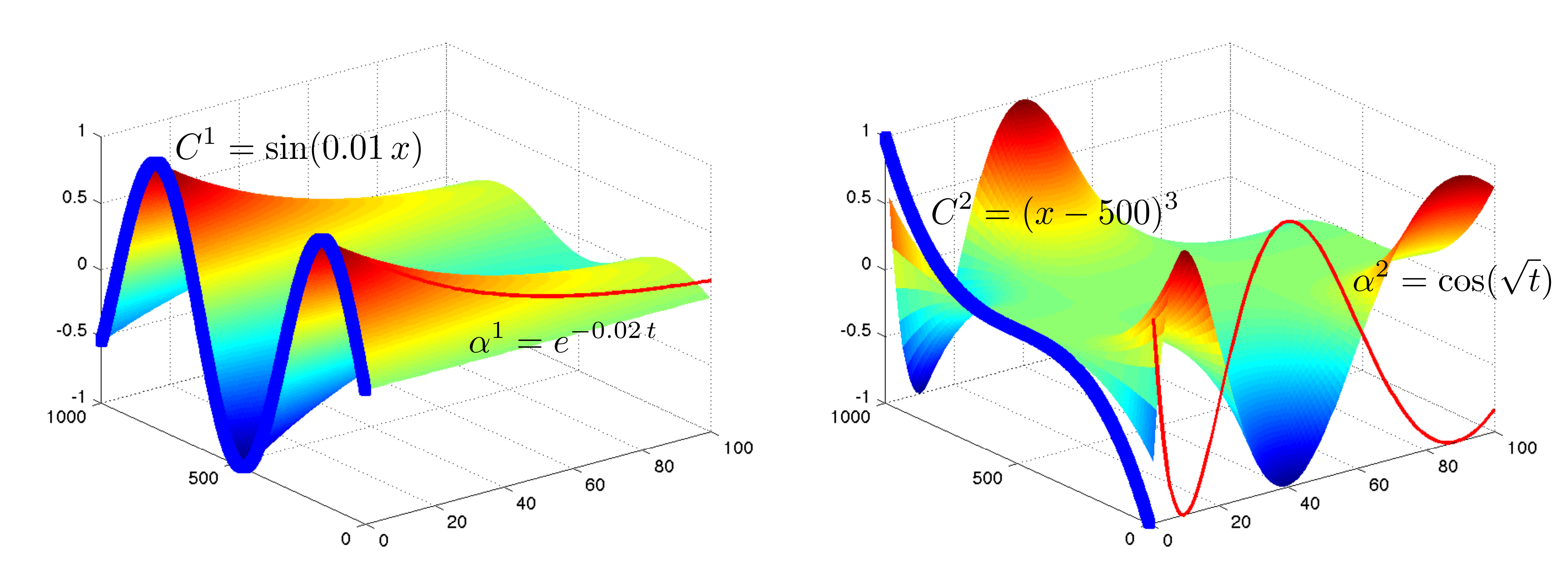
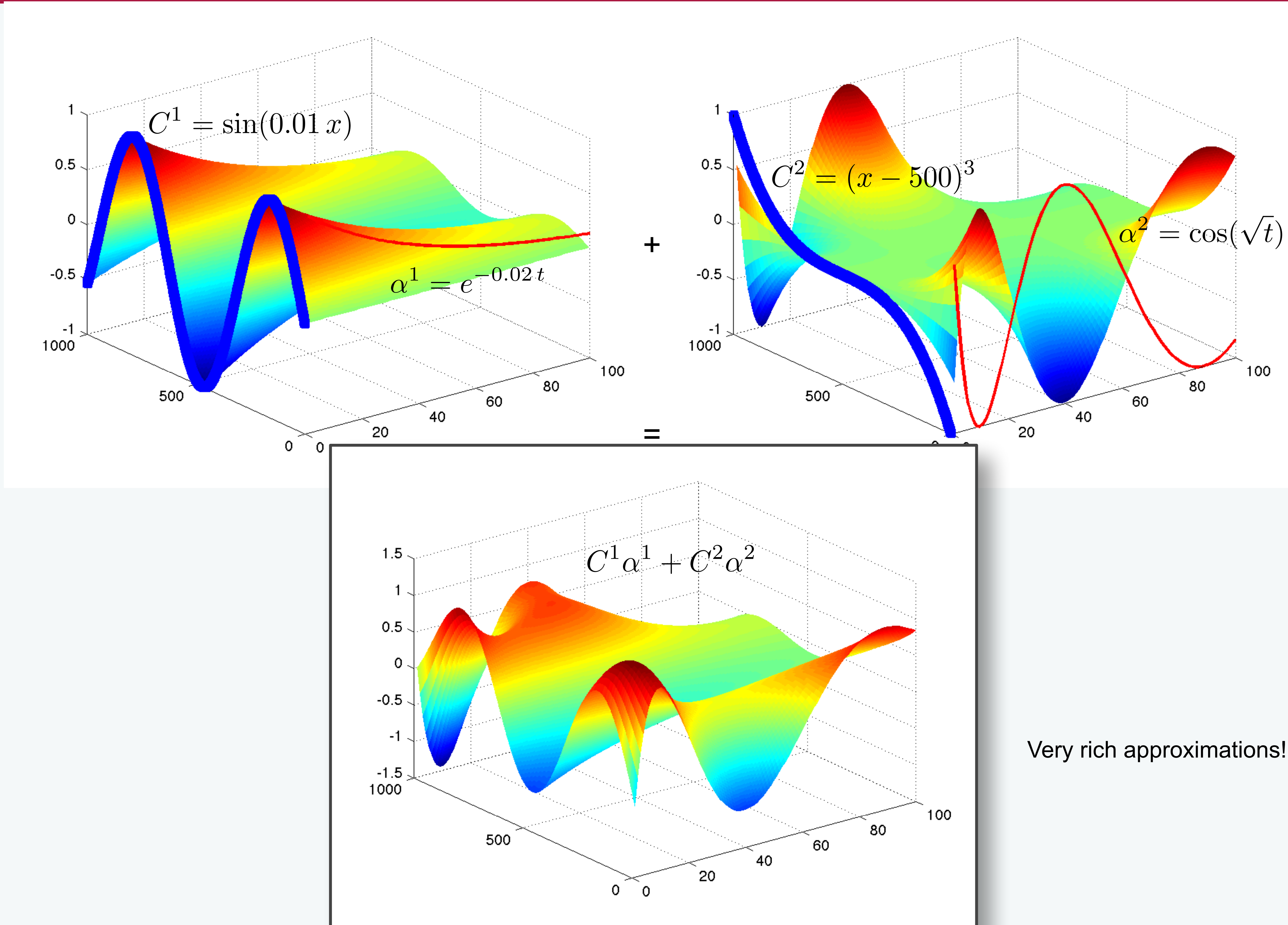
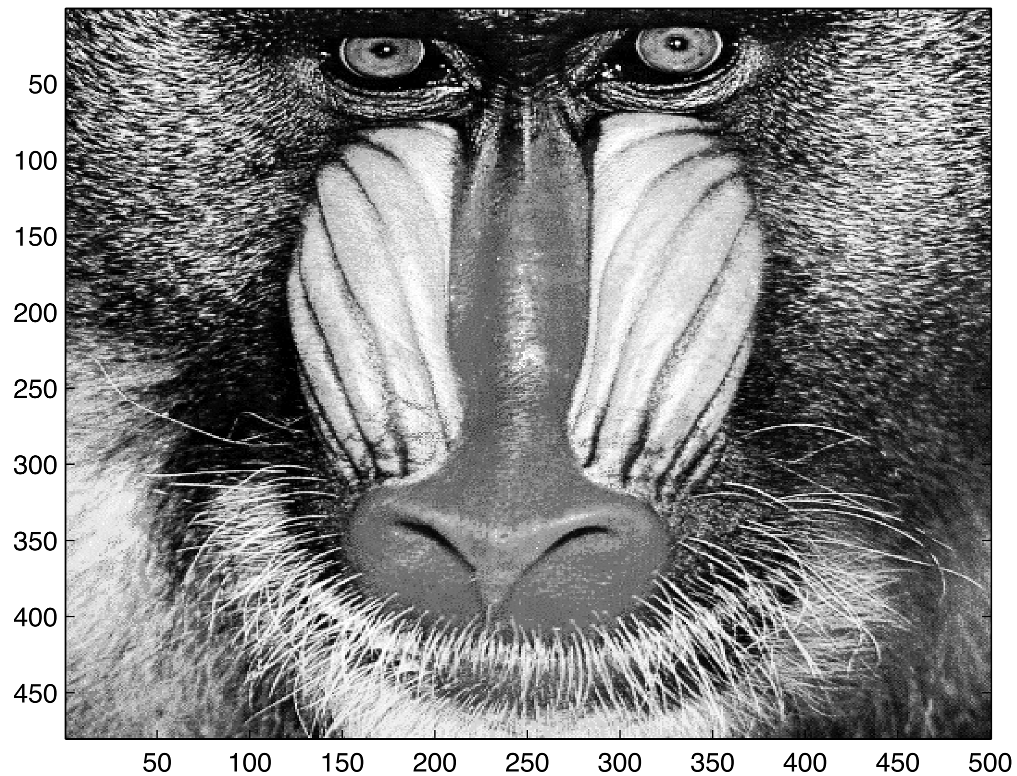
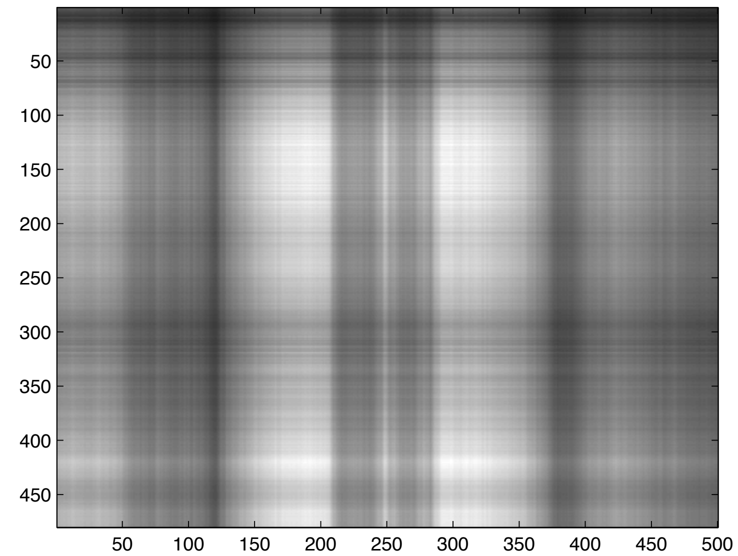


Illustration of the method of separated representation



Very rich approximations!



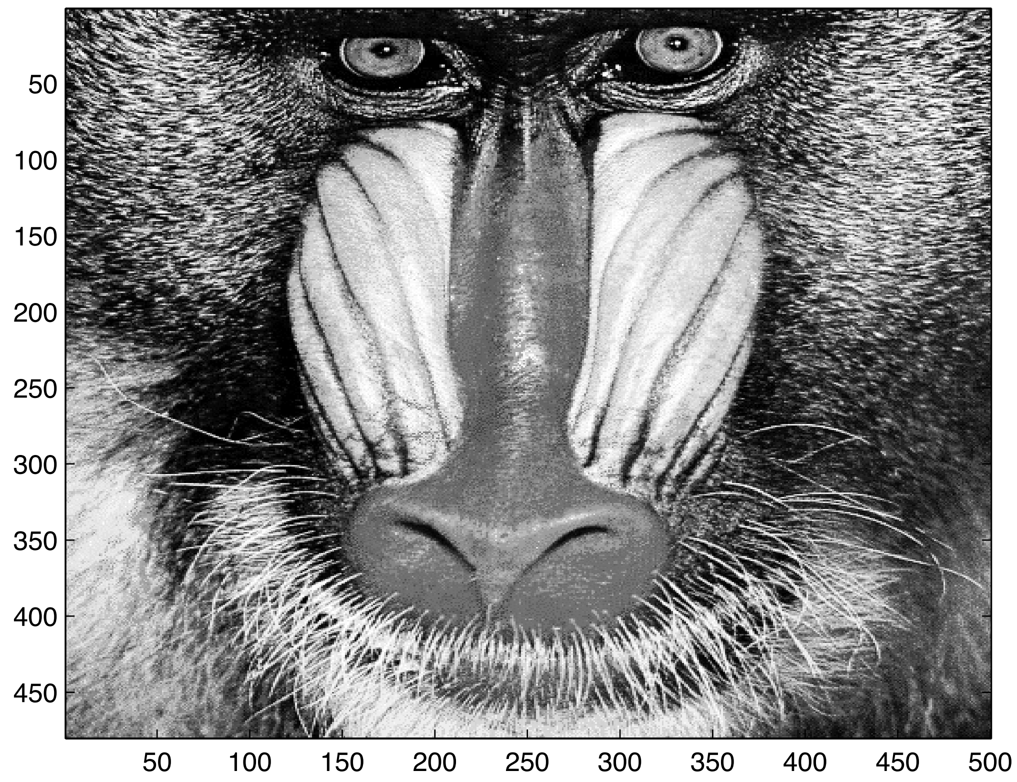


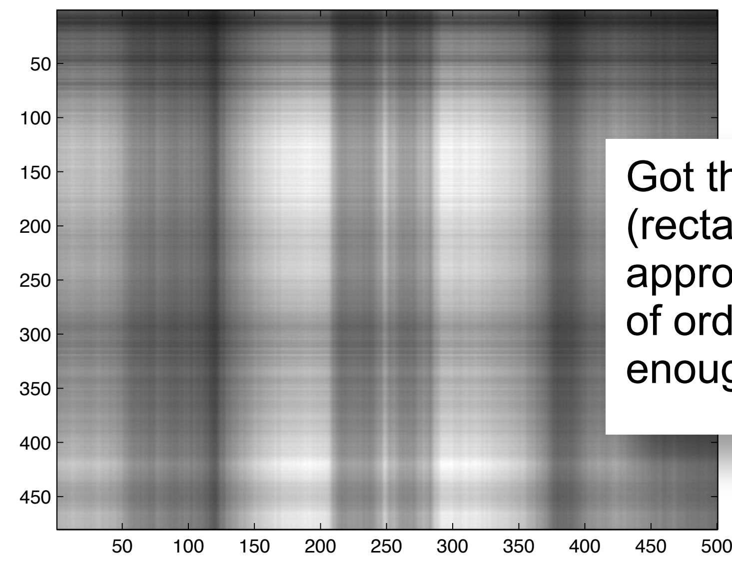
$n_c = 1$

$$\bar{u}(x_i, y_i) = \sum_{i=1}^{n_c} \underline{C}_x^i(x_i)\, \underline{C}_y^i(y_i)$$

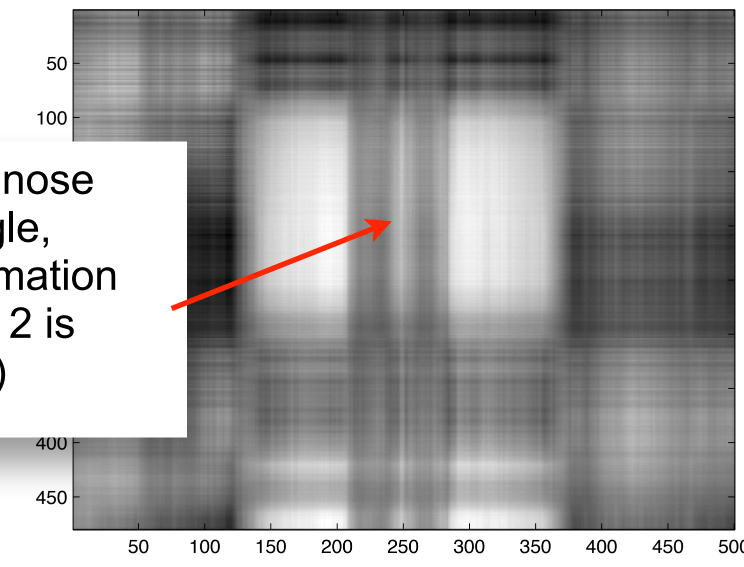
$$(C_x^i, C_y^i)_{i \in [\![1, n_c]\!]} = \operatorname{argmin} \sum_{x_i} \sum_{y_j} (u(x_i, y_j) - \bar{u}(x_i, y_j))^2$$

Data compression: get the nose with the POD!





$n_c = 1$



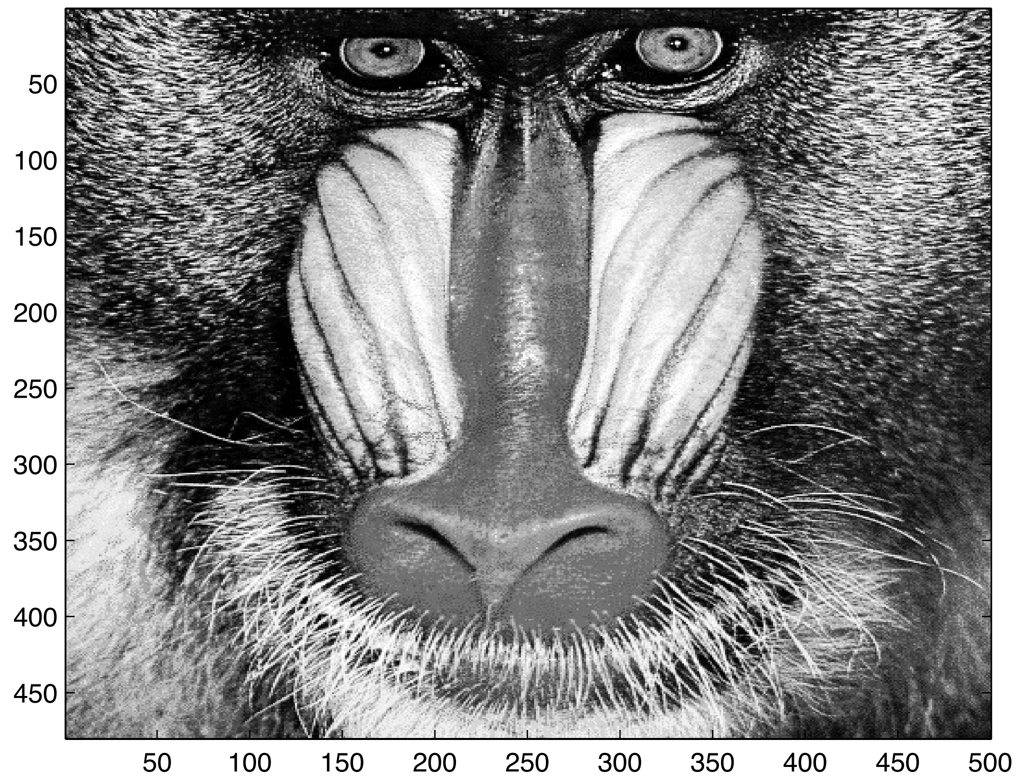
Got the nose (rectangle, approximation of order 2 is enough)

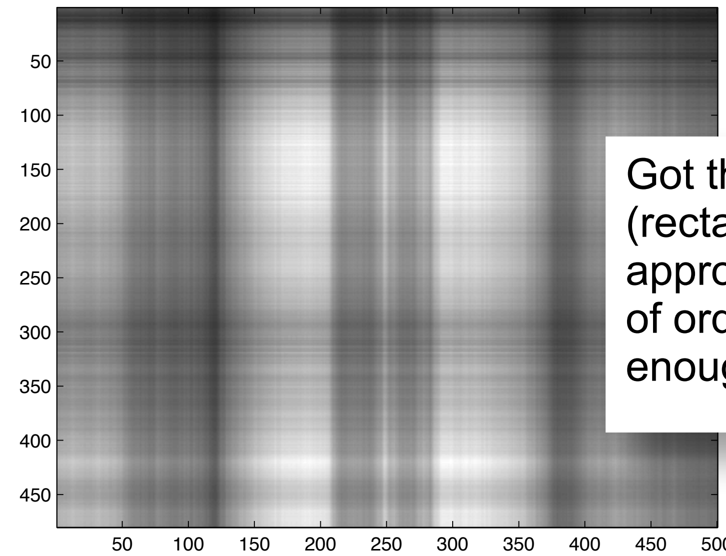
$n_c = 2$

$$\bar{u}(x_i, y_i) = \sum_{i=1}^{n_c} \underline{C_x^i}(x_i)\, \underline{C_y^i}(y_i)$$

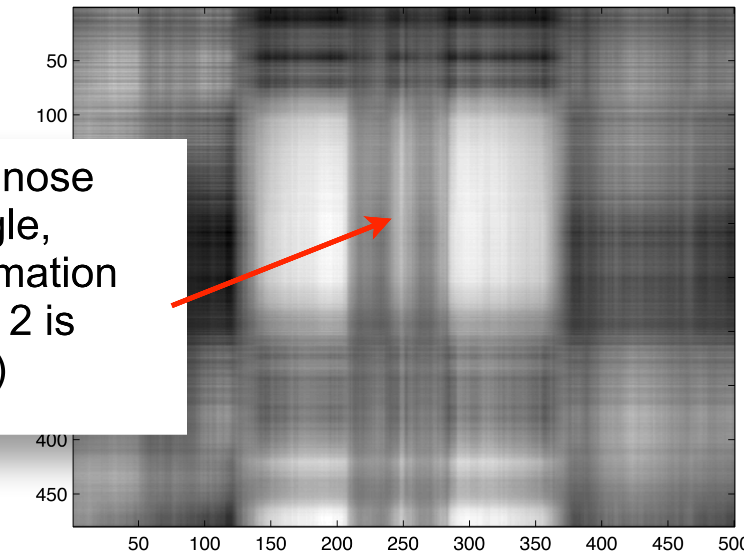
$$(C_x^i, C_y^i)_{i \in [\![1, n_c]\!]} = \operatorname{argmin} \sum_{x_i} \sum_{y_j} (u(x_i, y_j) - \bar{u}(x_i, y_j))^2$$

Data compression: get the nose with the POD!



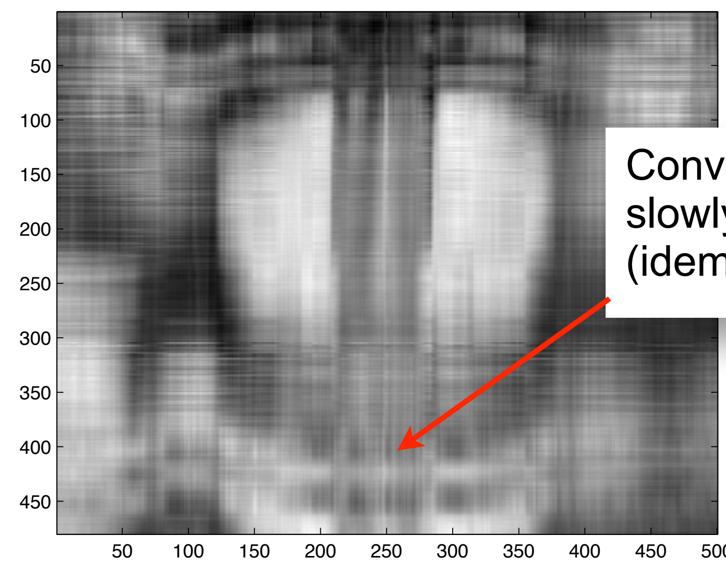


n_c = 1

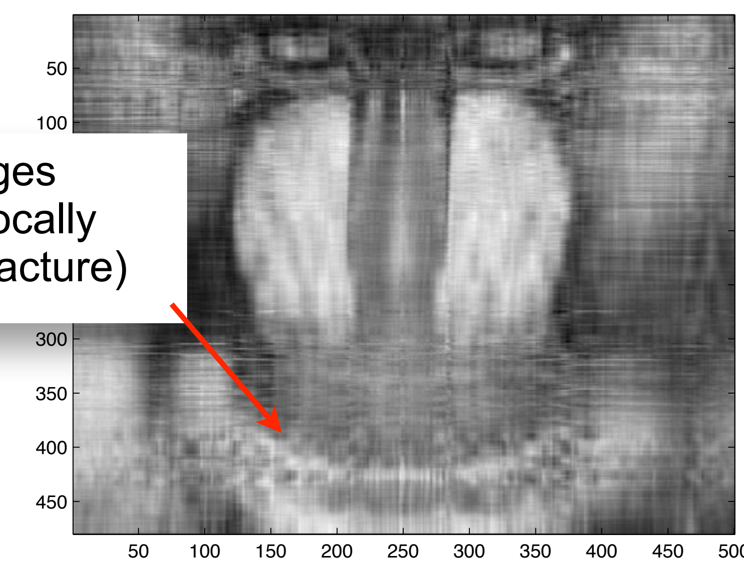


n_c = 2

Got the nose (rectangle, approximation of order 2 is enough)

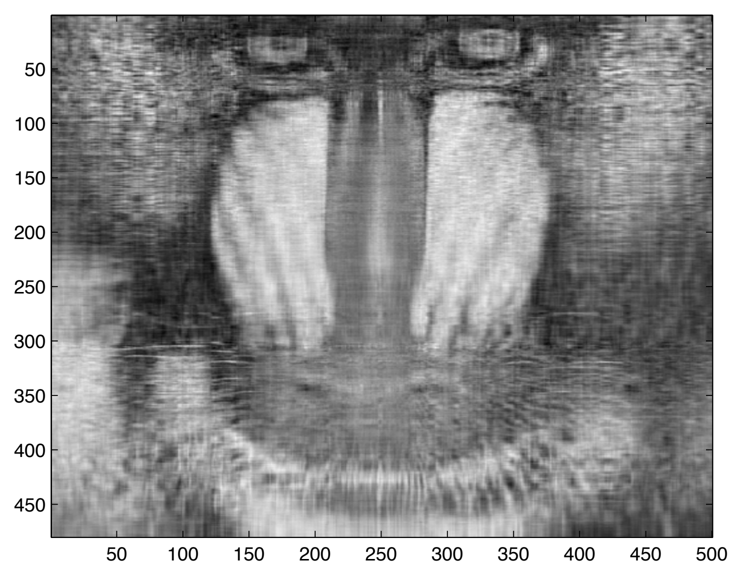


n_c = 5



n_c = 10

Converges slowly locally (idem fracture)



n_c = 20

$$\bar{u}(x_i, y_i) = \sum_{i=1}^{n_c} \underline{C_x^i}(x_i)\, \underline{C_y^i}(y_i)$$

$$(C_x^i, C_y^i)_{i \in [\![1, n_c]\!]} = \operatorname{argmin} \sum_{x_i} \sum_{y_j} (u(x_i, y_j) - \bar{u}(x_i, y_j))^2$$

Method of separated representation

- Search for the solution in space / time / parameter in a product space:

$$\underline{\bar{\mathbf{U}}} : \quad \mathcal{U}_{\text{sep}} = \mathbb{R}^n \times \mathcal{T} \times \mathcal{P} \to \mathbb{R}^n$$

$$\underline{\bar{\mathbf{U}}}(t,\mu) = \sum_{i=1}^{n_C} \underline{\mathbf{C}}_i\, \beta_i(t)\gamma_i(\mu)\,,$$

$$\underline{\mathbf{C}}^i \in \mathbb{R}^n$$
$$\beta^i : \mathcal{T} \to \mathbb{R}, \quad \forall i \in [\![1, n_C]\!]\,,$$
$$\gamma^i : \mathcal{P} \to \mathbb{R}, \quad \forall i \in [\![1, n_C]\!]\,,$$

- Optimality of an expansion of order n_c with respect to a particular metric defined on $\mathcal{U}_{\text{sep}}$
  - ➡ different metrics lead to different methods, which have their pro/cons
  - ➡ Choice strongly dependent on the context
  - Data compression: **POD** (Proper Orthogonal Decomposition) is a classical choice in dimension 2
  - Data compression in many dimensions: **multilinear POD**
  - Solver in many dimensions without *a priori* knowledge of the solution: **PGD**
  - Model order reduction: **Snapshot POD, Snapshot PGD**
  - Initialiser, preconditioners: **low-order POD**, **low-order PGD**, **Snapshot POD**

- One writes the classical POD problem:

find an orthonormal basis $\underline{\underline{\mathbf{C}}} \in \mathbb{R}^{n \times n_c}, \ \underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{C}}} = \underline{\underline{\mathbf{I}}}_d$ *minimising the POD functional:*

$$J_{\text{POD}}(\underline{\underline{\mathbf{C}}}) = \int_{t \in \mathcal{T}} \|\underline{\mathbf{U}}(t) - \underline{\underline{\mathbf{C}}}\,\underline{\underline{\mathbf{C}}}^T \underline{\mathbf{U}}(t)\|_2^2 \, dt$$

- Equivalently, look for a maximum of

$$\bar{J}_{\text{POD}}(\underline{\underline{\mathbf{C}}}) = \int_{t \in \mathcal{T}} \underline{\mathbf{U}}(t)^T \underline{\underline{\mathbf{C}}}\,\underline{\underline{\mathbf{C}}}^T \underline{\mathbf{U}}(t) \, dt = \text{Tr}(\underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{K}}}\,\underline{\underline{\mathbf{C}}})$$

  - Correlation operator:

$$\underline{\underline{\mathbf{K}}} = \int_{t \in \mathcal{T}} \underline{\mathbf{U}}(t)\,\underline{\mathbf{U}}(t) \, dt$$

- Solution: eigenvalue problem

$$\underline{\underline{\mathbf{K}}}\,\underline{\phi}^k = \lambda^k \underline{\phi}^k$$ where $(\lambda^k)_{k \in [\![0,n]\!]}$ in decreasing order

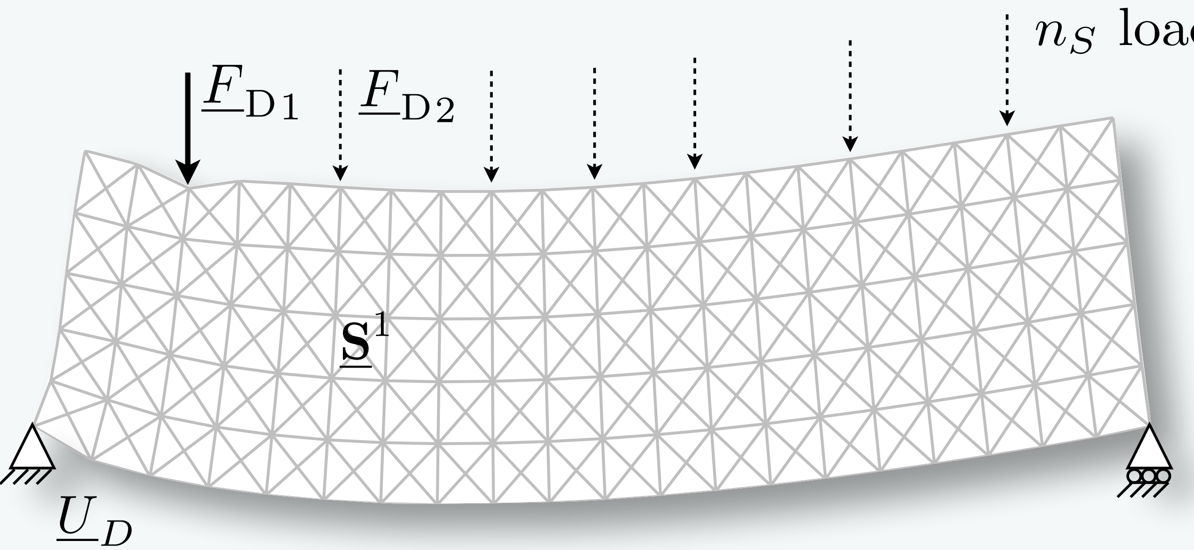
$$\underline{\underline{\mathbf{C}}} = \left(\underline{\phi}^1 \quad \underline{\phi}^2 \quad \dots \quad \underline{\phi}^{n_c} \right)$$

$$\int_{t \in \mathcal{T}} \alpha^i \, \alpha^j \, dt = \delta_{ij} \, \lambda^i$$

- Error

$$J_{\text{POD}}(\underline{\underline{\mathbf{C}}}) = \sum_{k=n_c+1}^{n} \lambda^k$$

a posteriori model order reduction. **Idea**: search for the **solution as a linear combination** of a set of **pre-calculated representative solutions**



(1) Solve FINE for n_S parameters (EXPENSIVE!)

$$\underline{\underline{\mathbf{S}}} = \left(\underline{\mathbf{S}}^1 \quad \underline{\mathbf{S}}^2 \quad \ldots \quad \underline{\mathbf{S}}^{n_S} \right)$$

(2) Singular value decomposition

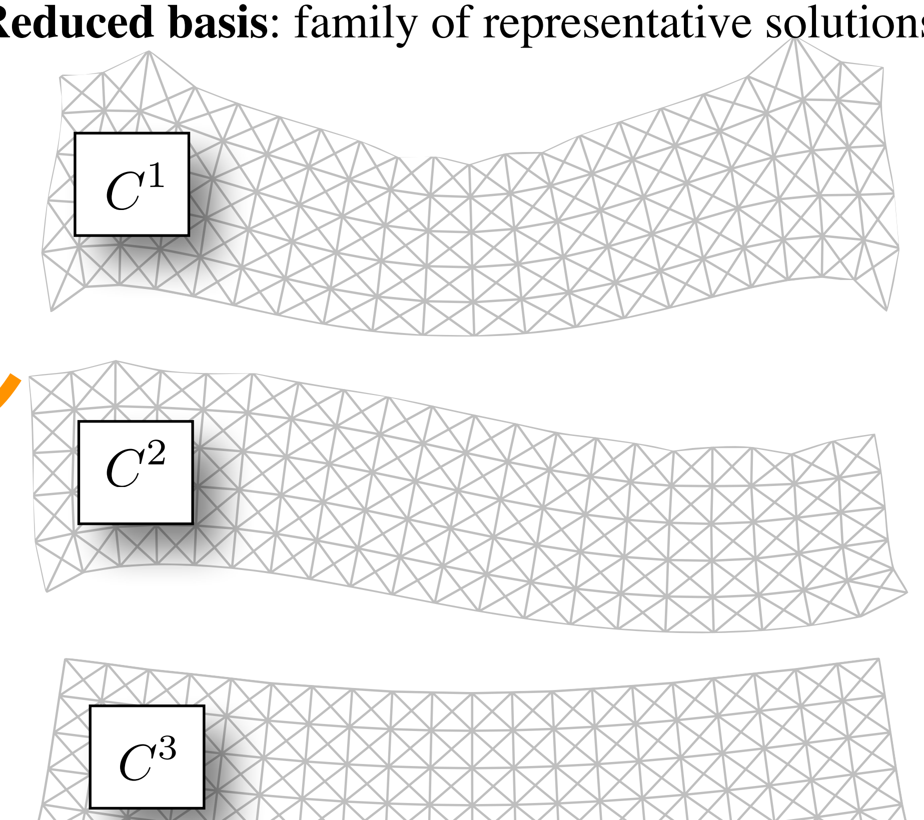
$$\underline{\underline{\mathbf{S}}} = \underline{\underline{\mathbf{U}}}\,\underline{\underline{\mathbf{\Sigma}}}\,\underline{\underline{\mathbf{V}}}^T = \sum_{k=1}^{n_S} \Sigma^k\, \underline{\mathbf{U}}^k\, \underline{\mathbf{V}}^{k\,T}$$

n_S solutions, sorted by relevance

where $(\Sigma^k)_{k \in [\![1\ n_S]\!]}$ in decreasing order

(3) Truncation

Reduced basis: family of representative solutions



$$\underline{\underline{\mathbf{C}}} = \left(\underline{\mathbf{U}}^1 \quad \underline{\mathbf{U}}^2 \quad \ldots \quad \underline{\mathbf{U}}^{n_C} \right)$$

Family of representative solutions

$$\underline{\mathbf{U}} = \underline{\underline{\mathbf{C}}}\,\underline{\alpha}$$

Solution

Coefficients

Initial set of equations

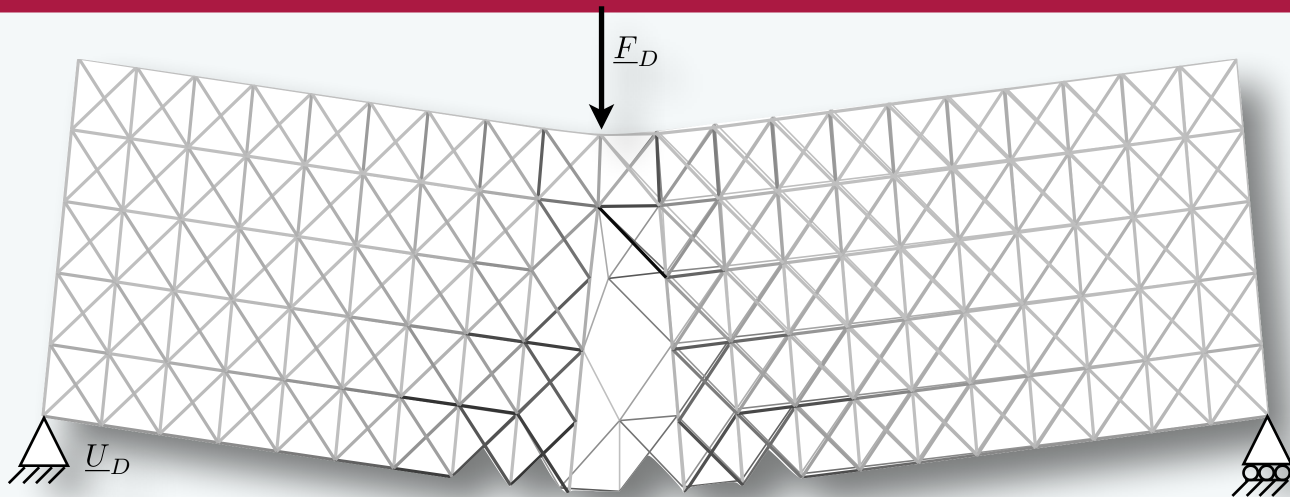
$$\underline{\mathbf{F}}_{\mathrm{Int}}(\underline{\mathbf{U}}) + \underline{\mathbf{F}}_{\mathrm{Ext}} = 0$$

(4) Galerkin orthogonality

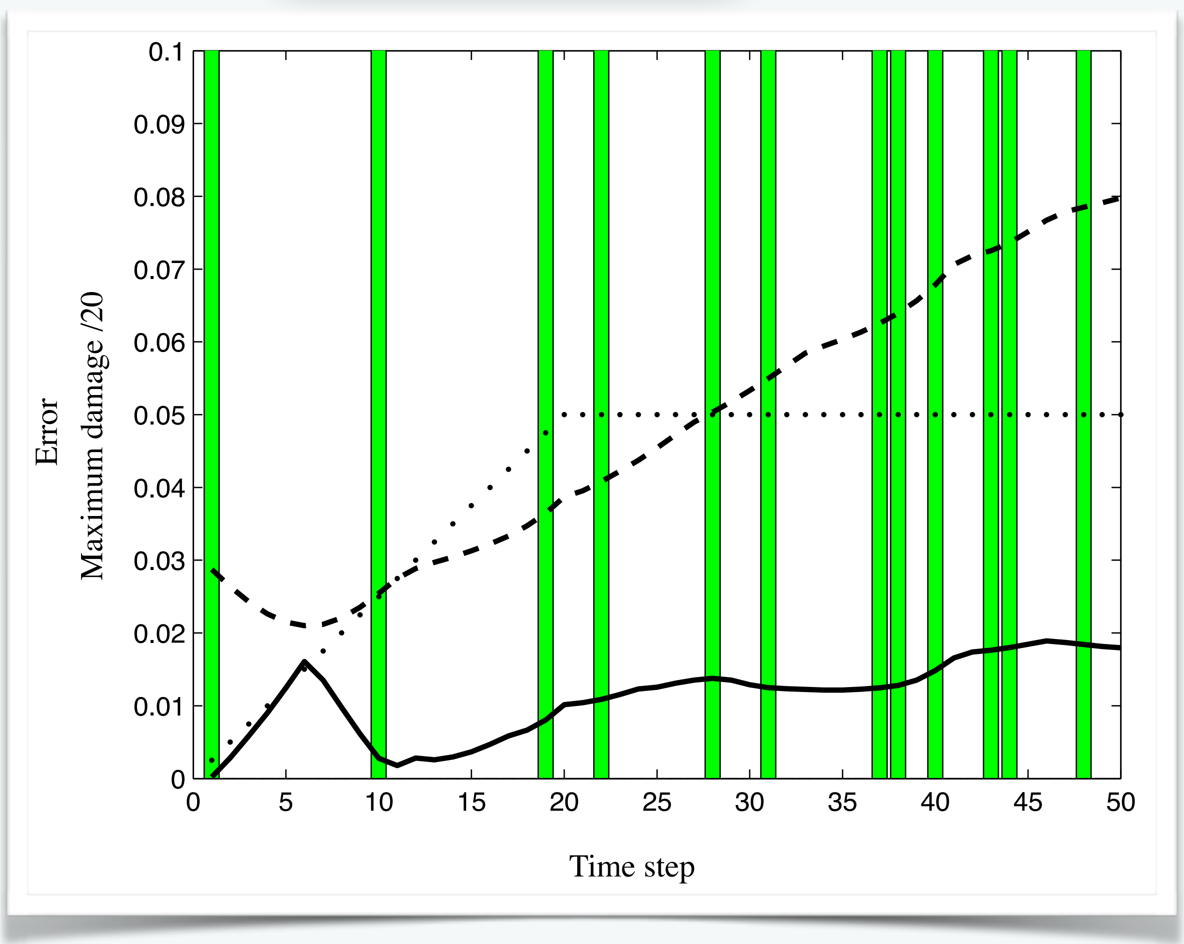
$$\underline{\underline{\mathbf{C}}}^T \underline{\mathbf{F}}_{\mathbf{int}}(\underline{\underline{\mathbf{C}}}\,\underline{\alpha}) + \underline{\underline{\mathbf{C}}}^T \underline{\mathbf{F}}_{\mathbf{ext}} = 0$$

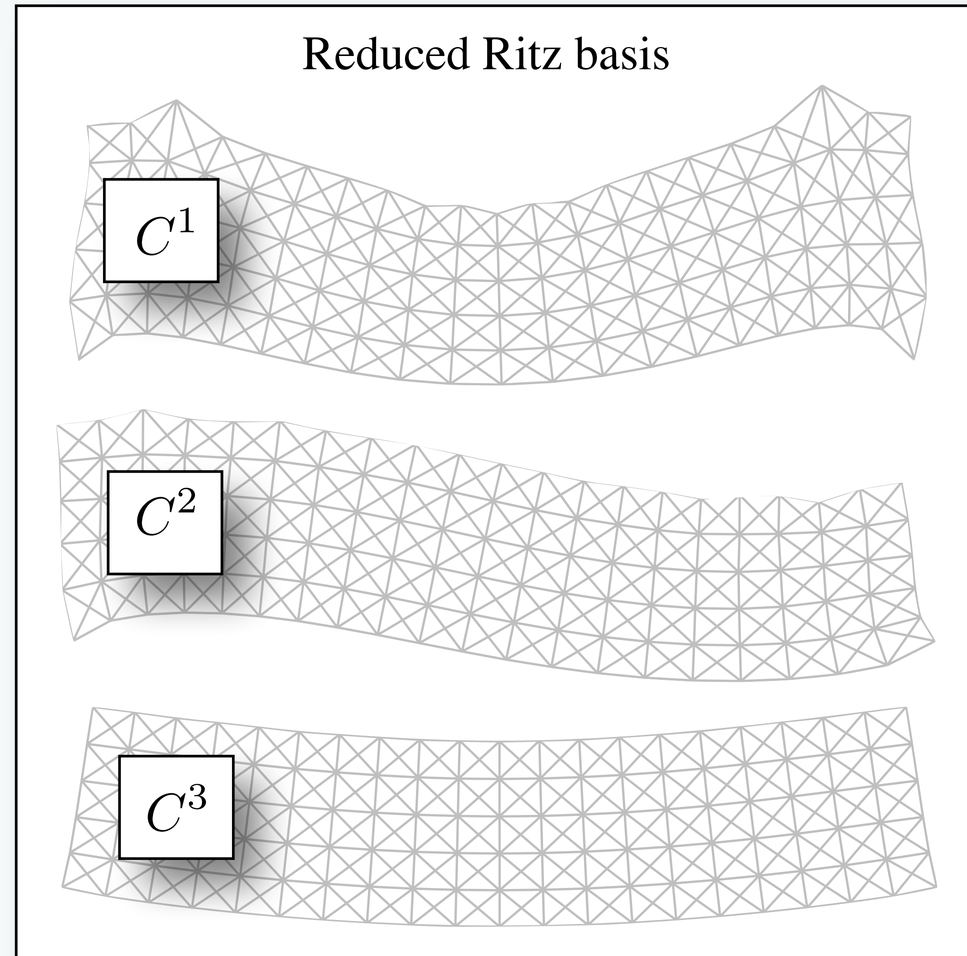
Approximation of the solution in a space of small dimension (n_C)

Limitations: case of highly non-linear fracture mechanics pbs

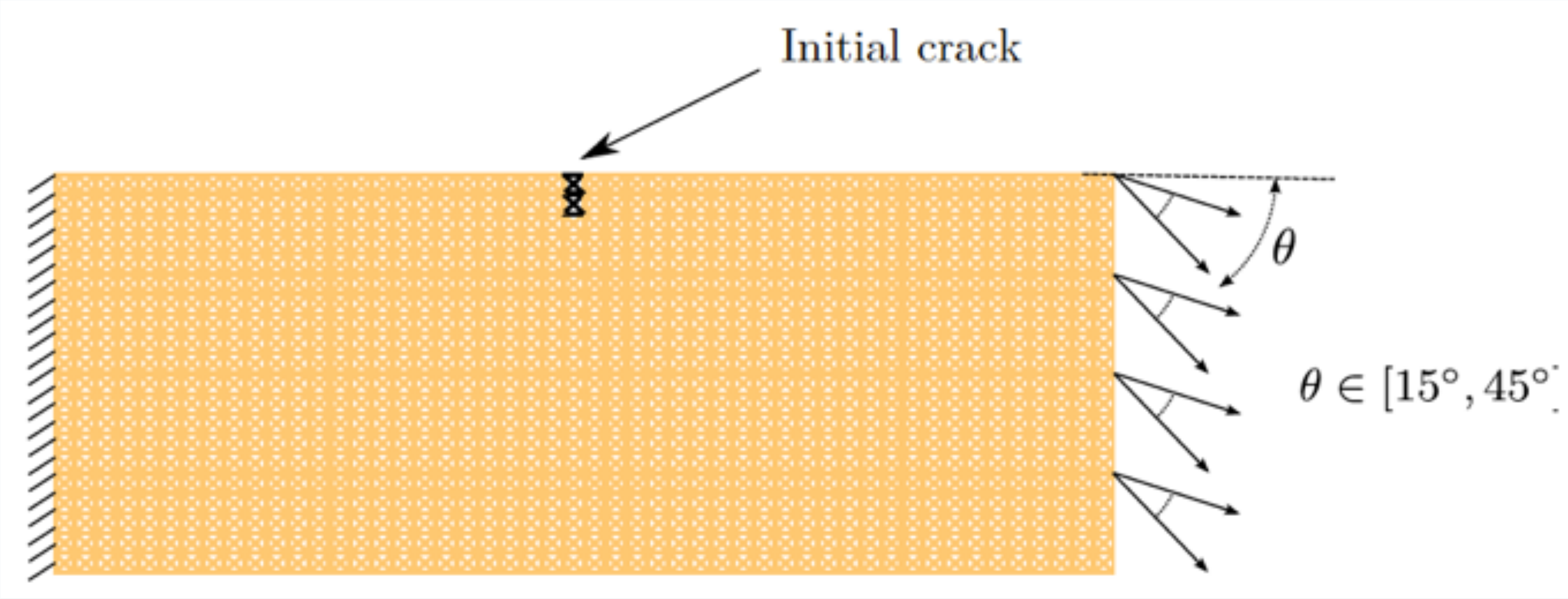


This solution is not in the snapshot !

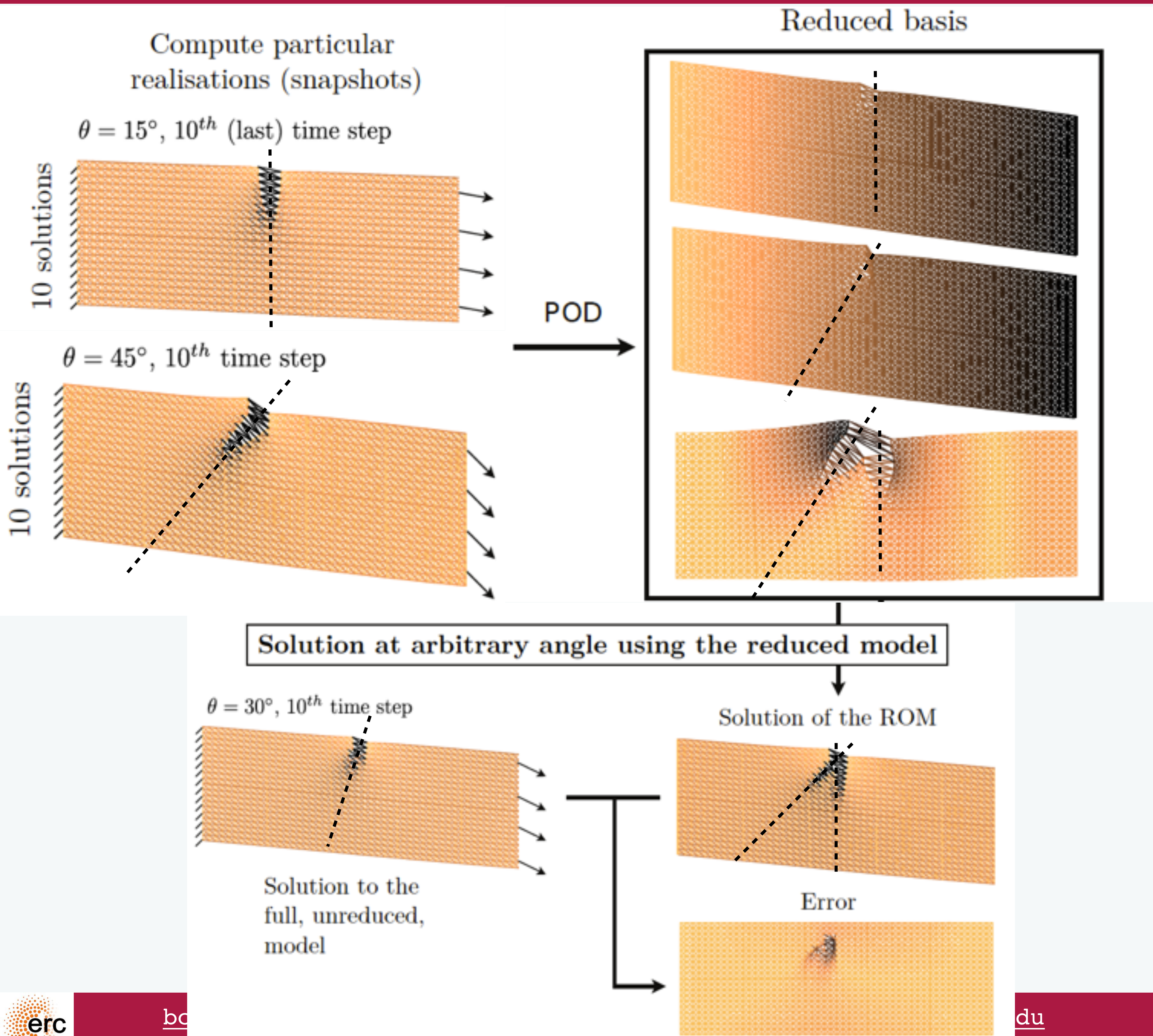




- P. Kerfriden, P. Gosselet, S. Adhikari, and S. Bordas. *Bridging proper orthogonal decomposition methods and augmented Newton-Krylov algorithms: an adaptive model order reduction for highly nonlinear mechanical problems*. Computer Methods in Applied Mechanics and Engineering, 200(5-8):850-866, 2011.

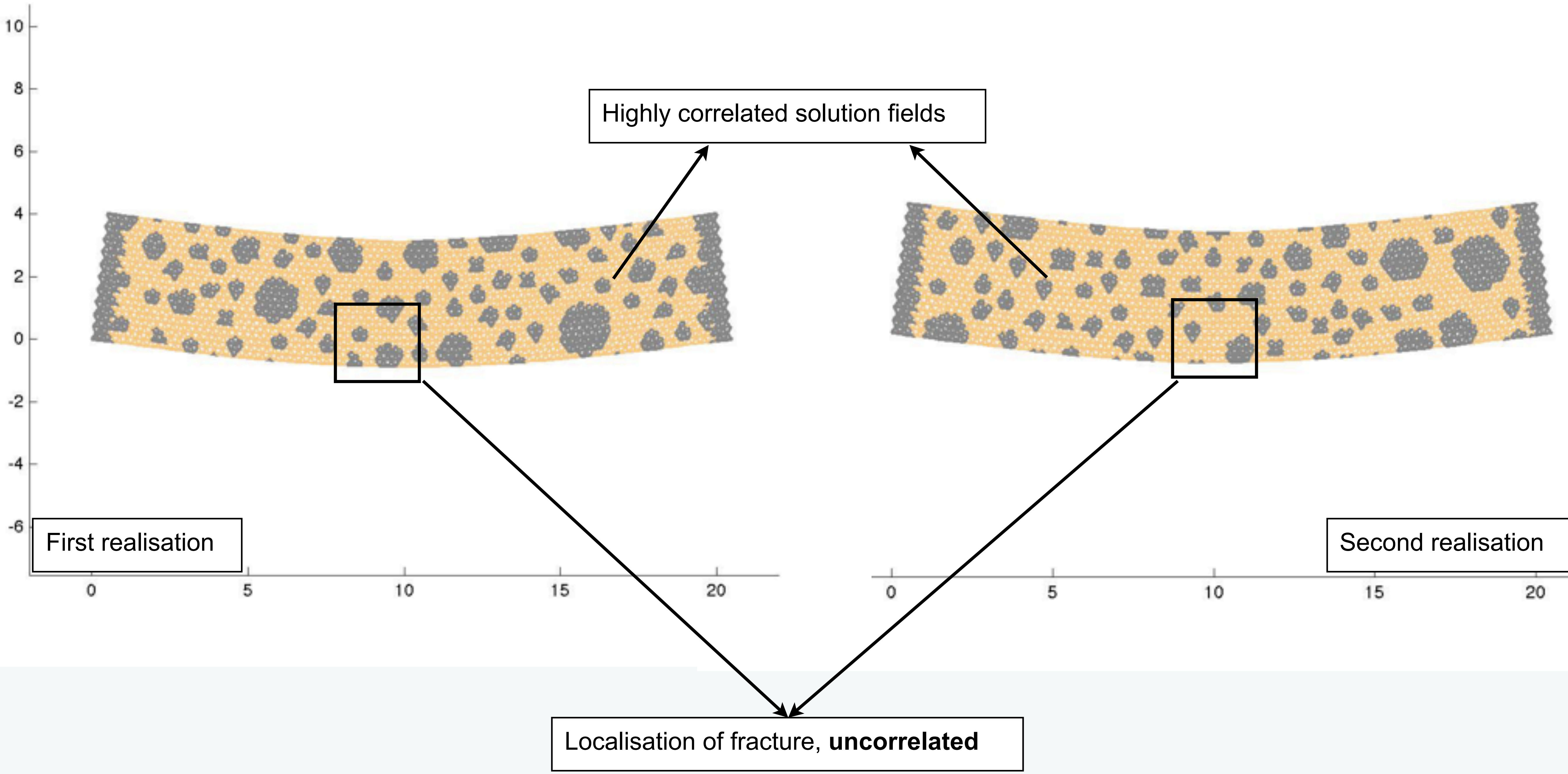

Initial crack
θ
θ ∈ [15°, 45°]

Application to a parametric fracture problem



- The POD solution is not able to reproduce the solution in the cracked area
- Due to lack of correlation introduced by crack growth
- Leads to a local projection error

Parametric / stochastic multiscale fracture mechanics

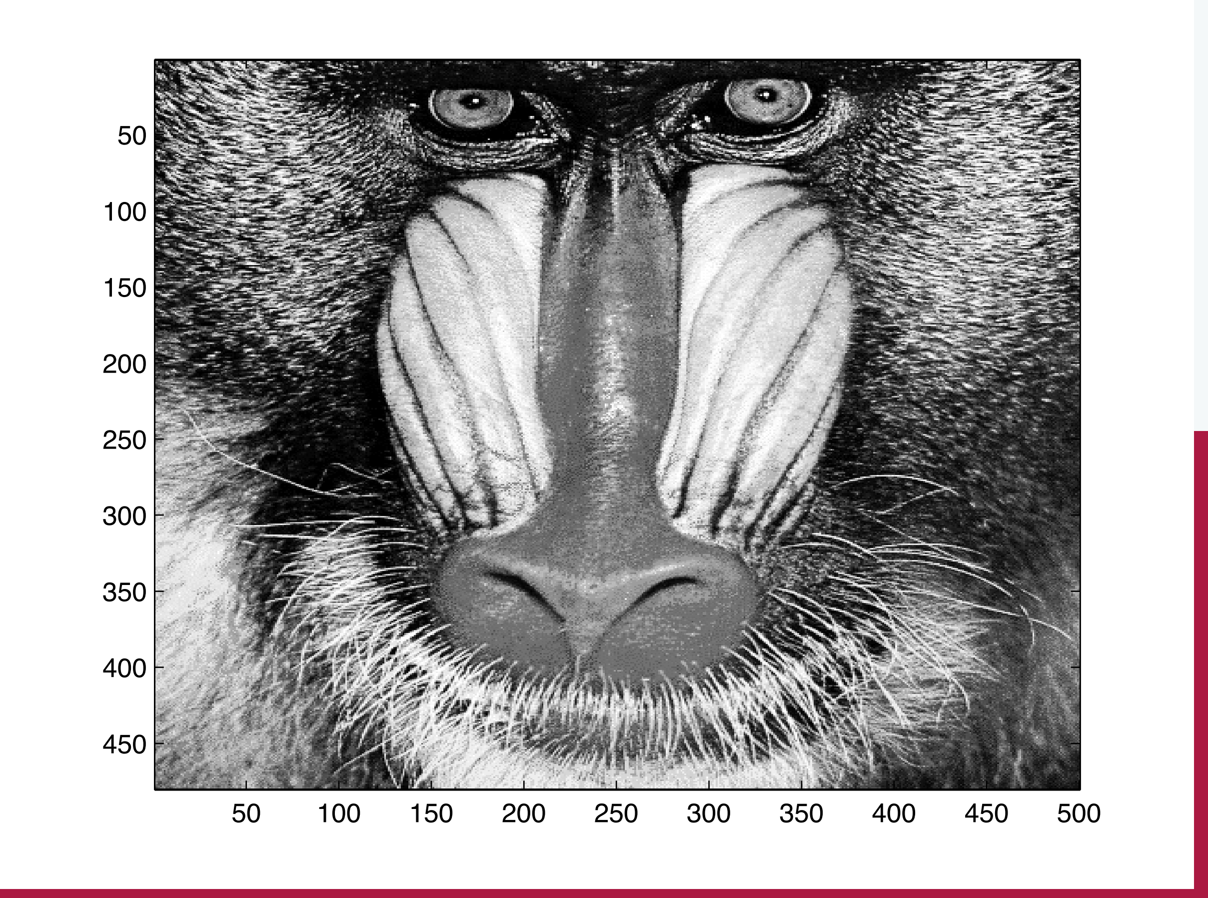


➡ Direct numerical simulation: efficient preconditioner?

➡ Reduced order modelling?

➡ Adaptive coupling?

THE RETURN OF THE MONKEY!



What can we do to address this lack of separation of scales/reducibility?

P. Kerfriden, P. Gosselet, S. Adhikari, and S. Bordas. Bridging proper orthogonal decomposition methods and augmented Newton-Krylov algorithms: an adaptive model order reduction for highly nonlinear mechanical problems. *Computer Methods in Applied Mechanics and Engineering*, 200(5-8):850–866, 2011.

P. Kerfriden, J.C. Passieux, and S. Bordas. Local/global model order reduction strategy for the simulation of quasi-brittle fracture. *International Journal for Numerical Methods in Engineering*, 89(2):154–179, 2011.

P. Kerfriden, K.M. Schmidt, T. Rabczuk, and Bordas S.P.A. Statistical extraction of process zones and representative subspaces in fracture of random composites. *Accepted for publication in International Journal for Multiscale Computational Engineering, arXiv:1203.2487v2*, 2012.

http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3672853/

http://orbilu.uni.lu/bitstream/10993/12454/2/presentationUSNC http://orbilu.uni.lu/bitstream/10993/18015/1/presentationWCCM.pdf

https://orbilu.uni.lu/bitstream/10993/12452/1/Extended_abstract_ACME-UK_2012_OlivierGouryUpdated.pdf

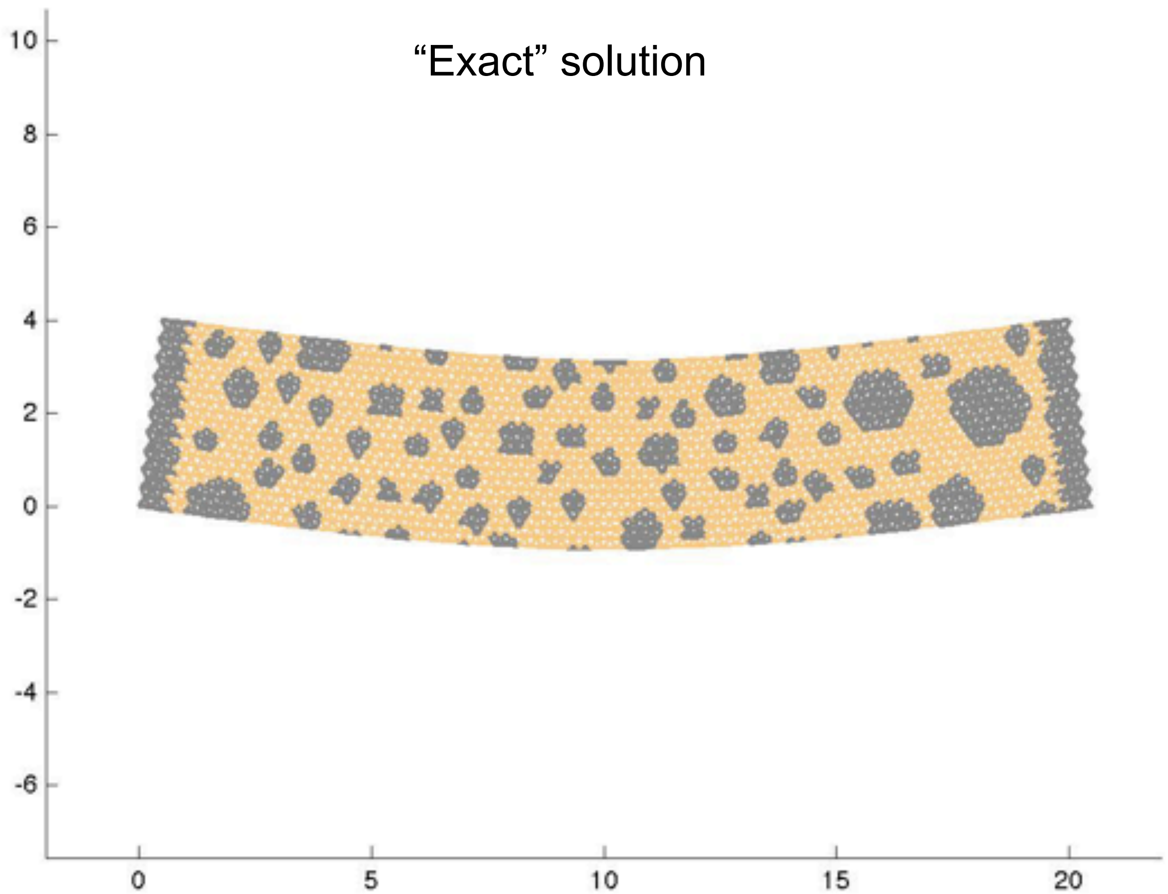
https://hal.inria.fr/docs/00/99/49/23/PDF/multiscaleMOR.pdf

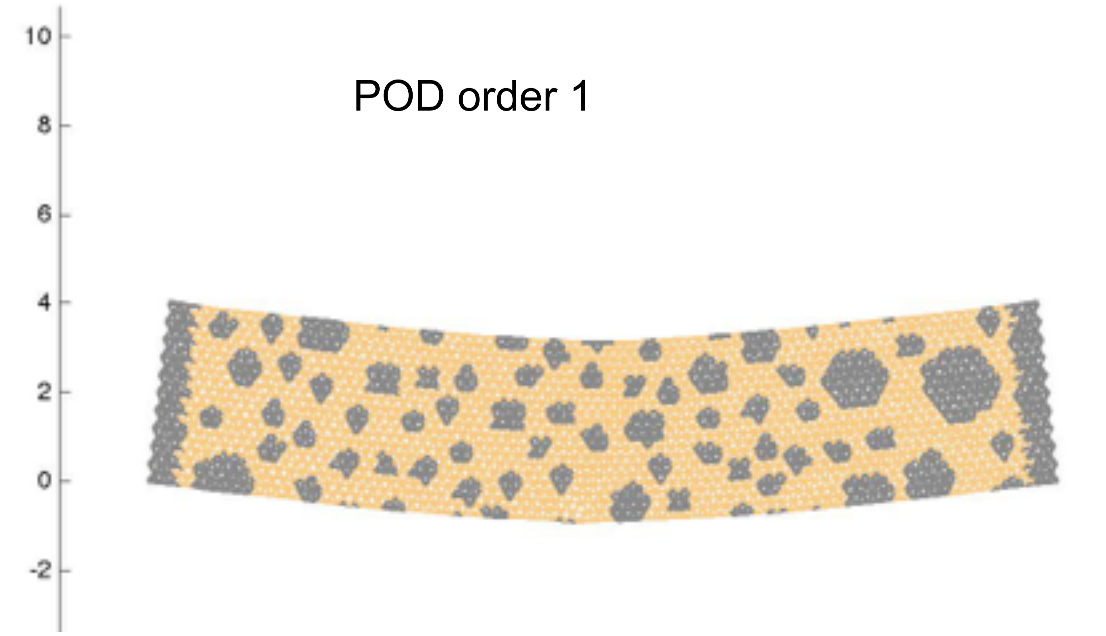
http://www.researchgate.net/profile/Olivier_Goury/publication/273832418_Dealing_with_interfaces_in_partitioned_model_order_reduction_for_application_to_nonlinear_problems/links/550e95c40cf212874168fe76.pdf

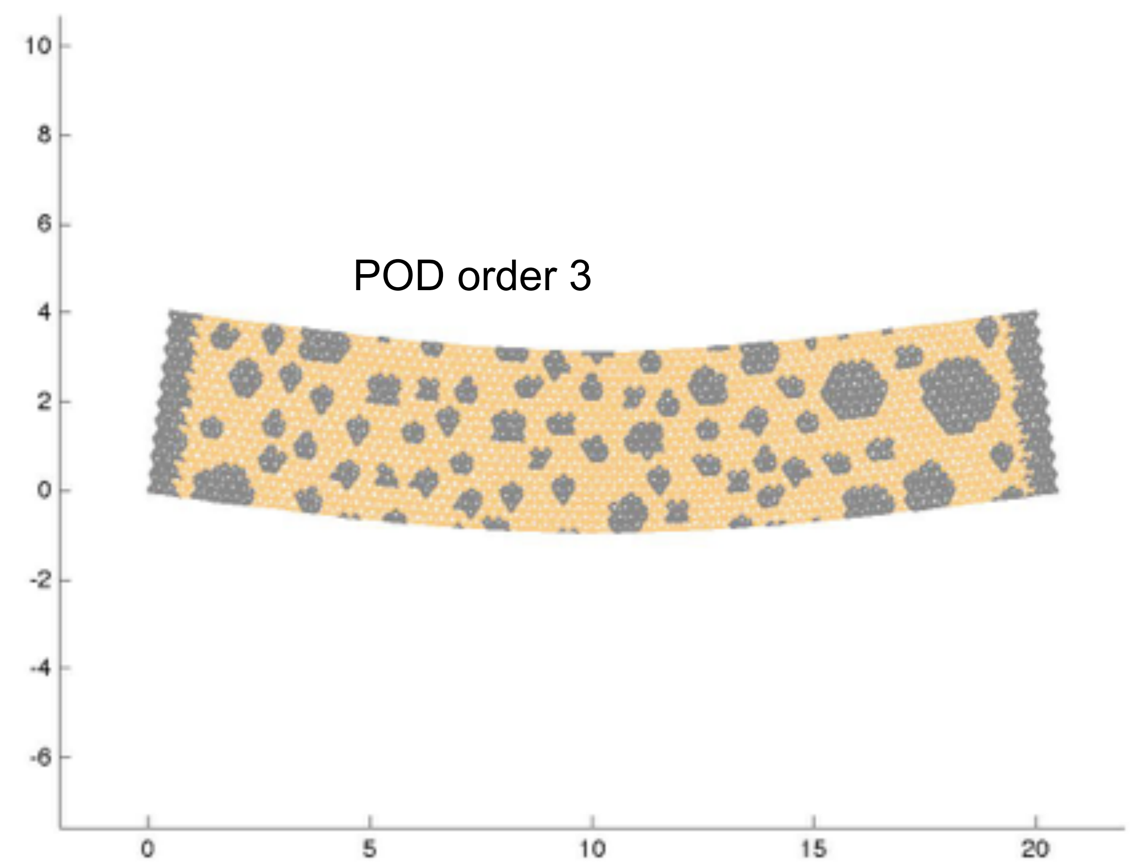
http://www.researchgate.net/profile/Ahmad_Akbari_R/pub
280083497_Scale_selection_in_nonlinear_fracture_mechanics_o
us_materials/links/55cdb5f308aebd6b88e06691.pd

Data compression: fracture

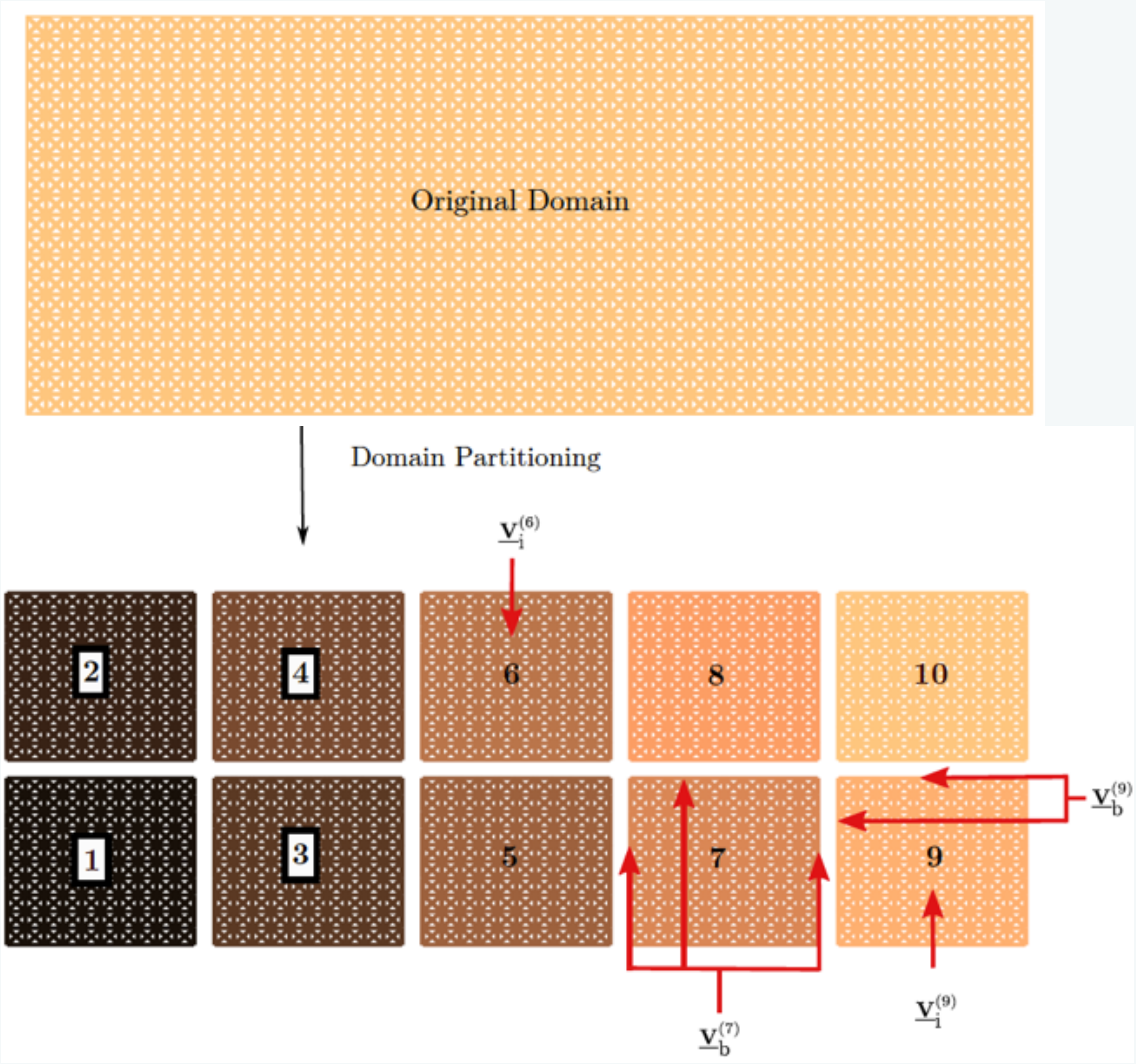
Snapshot POD (snapshot space is spanned by the ensemble of solutions at all time steps)



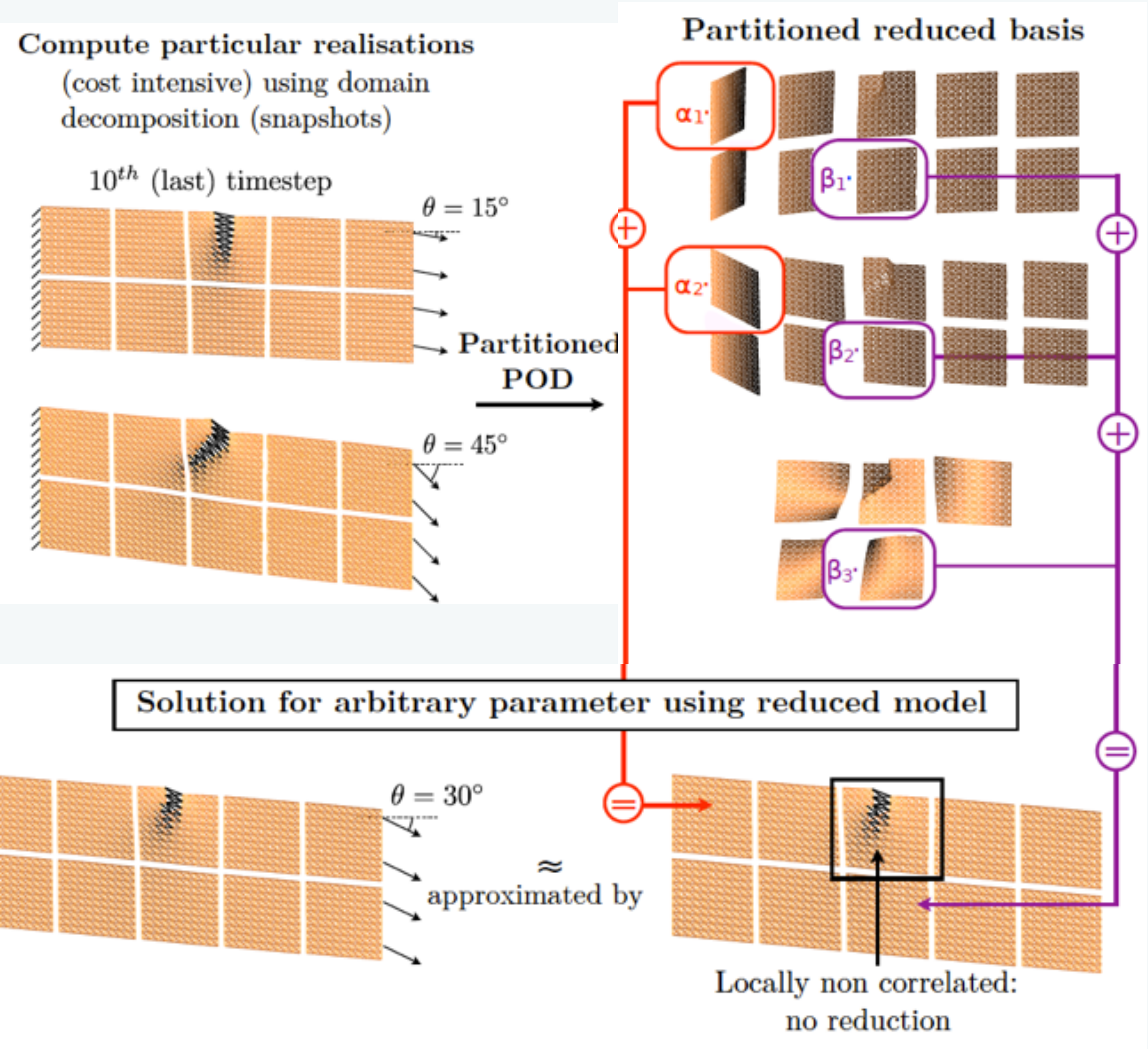




Partitioned POD/DDM

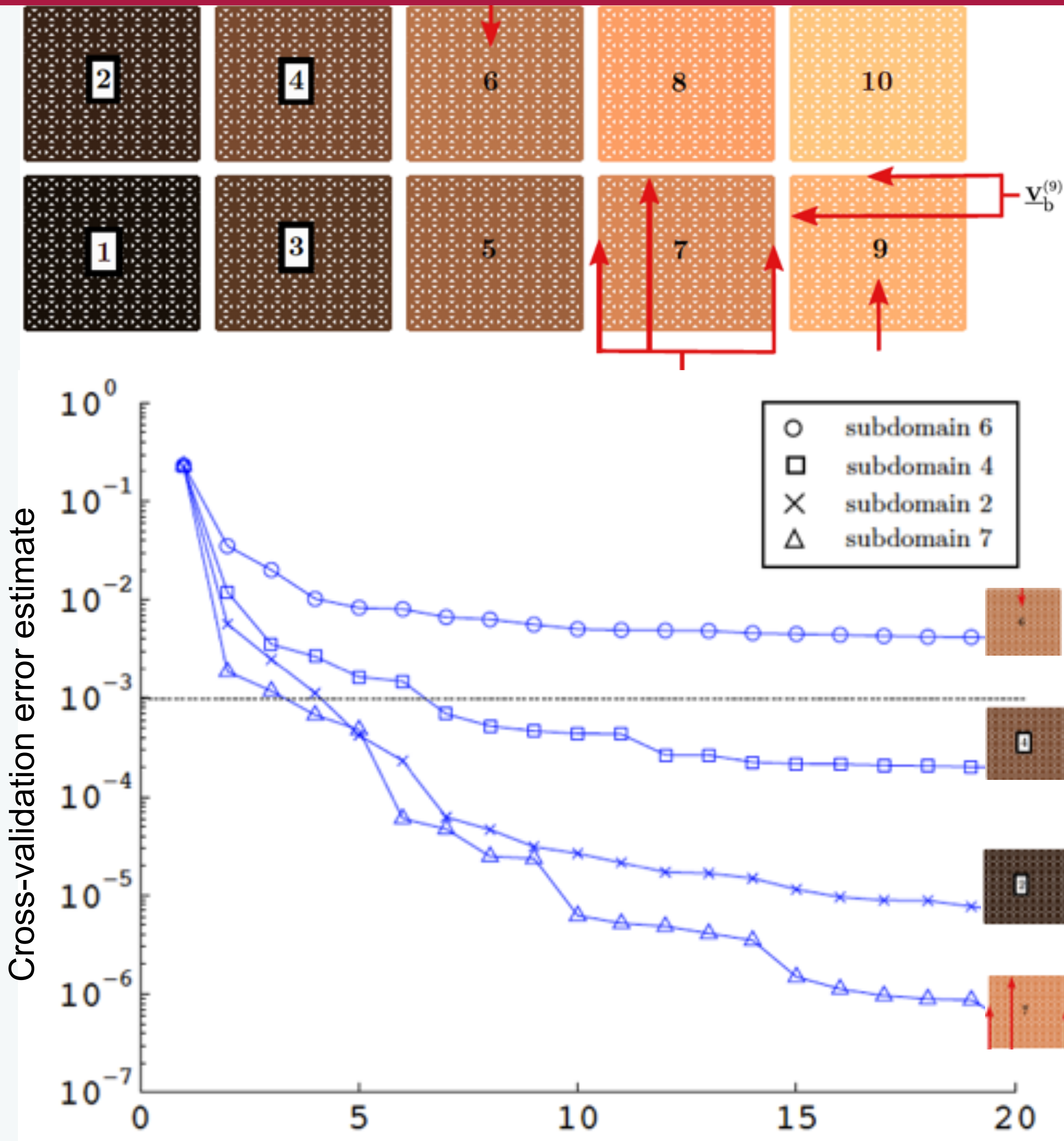


Reduced DDM-POD



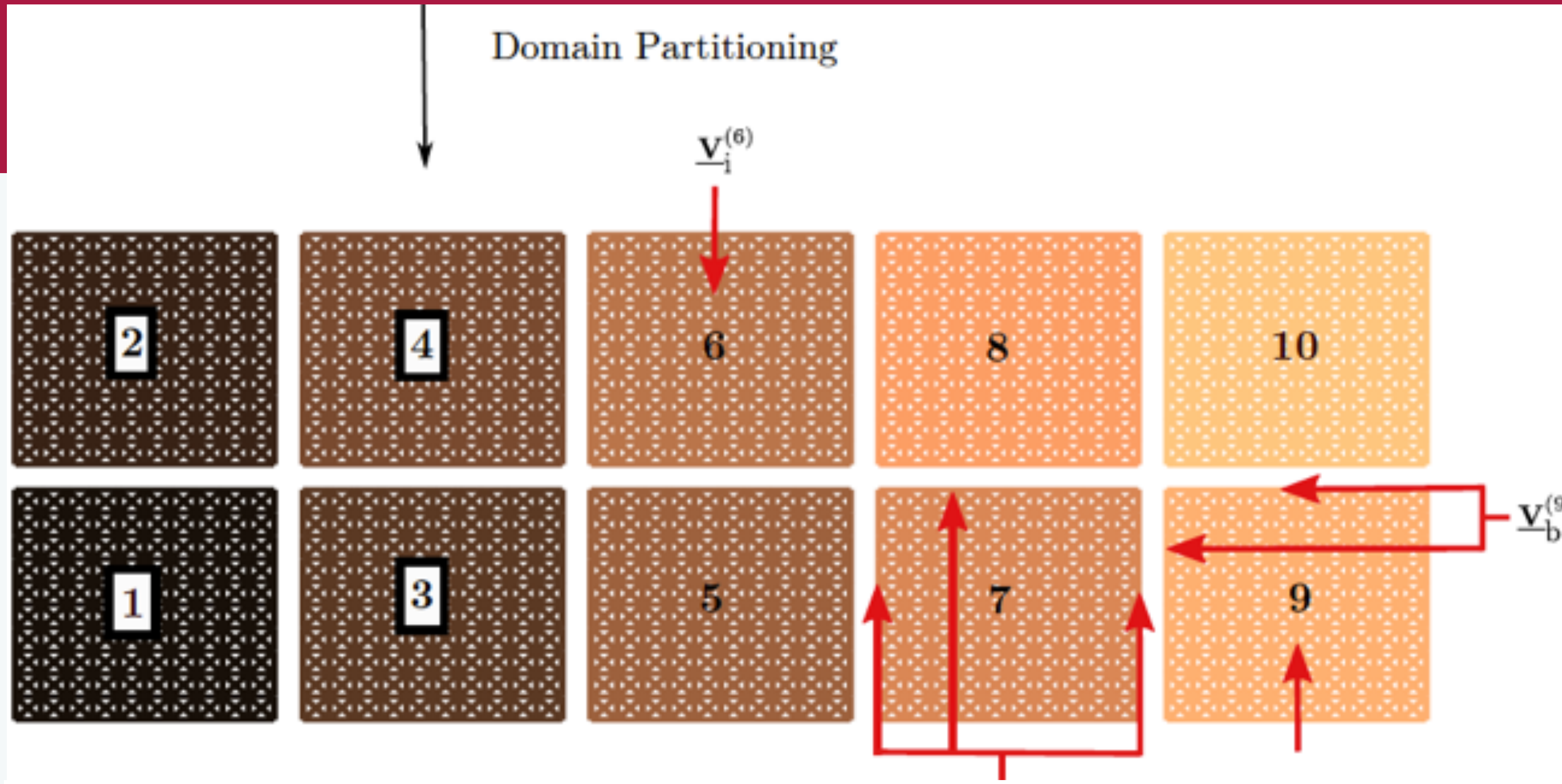
- Decompose the structure into subdomains
- Perform a reduction in the highly correlated region
- Couple the reduced to the non-reduced region by a primal Schur complement

Choice of the reduced subdomains: local error estimation by "leave one out cross-validation" (LOOCV)

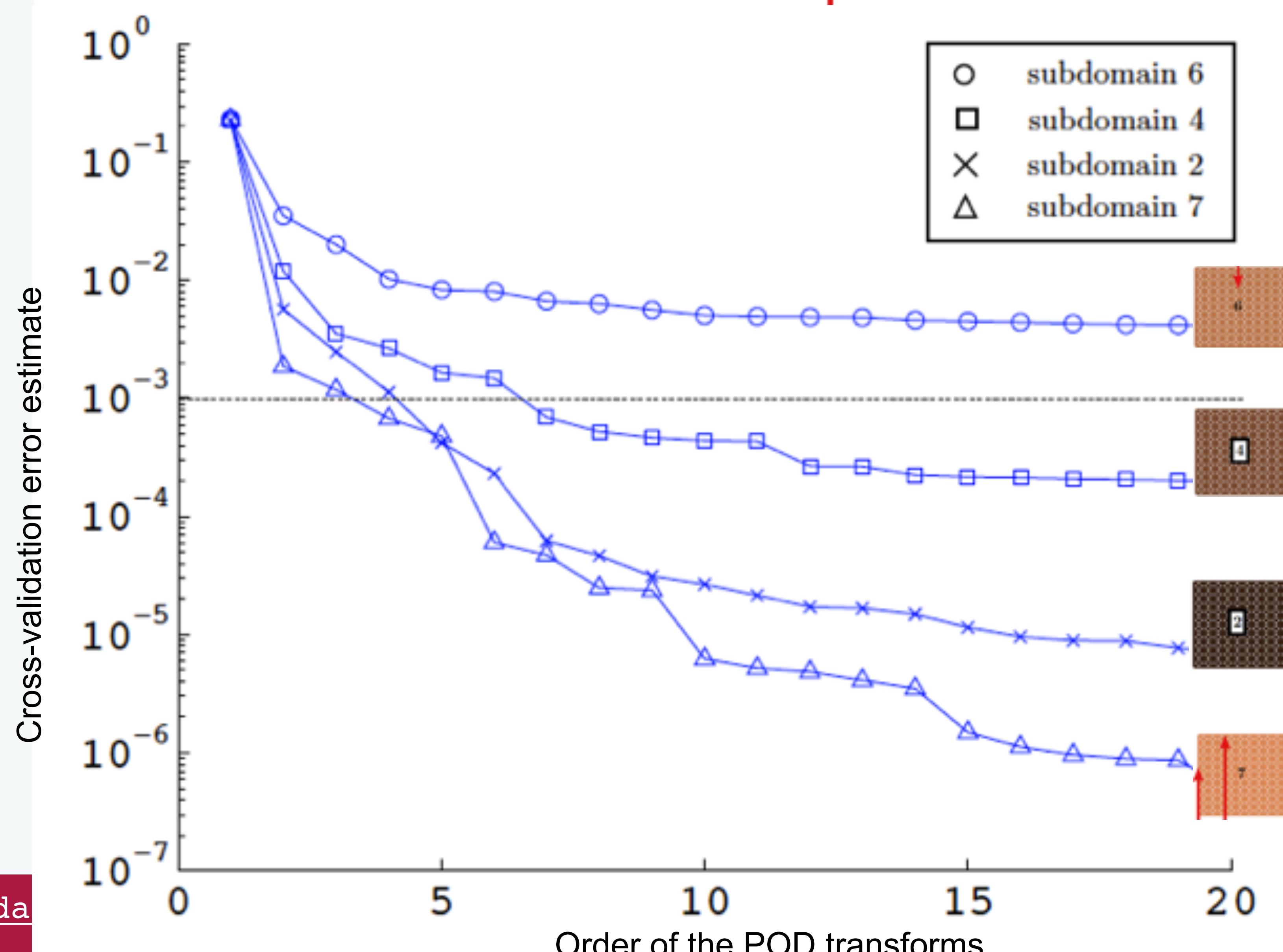


$$\left(\tilde{\nu}^{(e)}_{\text{snap}}\right)^2 = \frac{\sum_{\mu\in\mathcal{P}^s}\sum_{t_n\in\mathcal{T}^h}\left\|\underline{\mathbf{U}}_{\mathrm{i}}(t_n,\mu) - \sum_{j=1}^{n_c^{(e)}}\left(\tilde{\underline{\mathbf{C}}}_{\mathrm{i},j}^{(e),(\mu)\,T}\,\underline{\mathbf{U}}_{\mathrm{i}}(t_n,\mu)\right)\tilde{\underline{\mathbf{C}}}_{\mathrm{i},j}^{(e),(\mu)}\right\|_2^2}{\sum_{t_n\in\mathcal{T}^h}\sum_{\mu\in\mathcal{P}^s}\|\underline{\mathbf{U}}_{\mathrm{i}}(t_n,\mu)\|_2^2}$$

- Reduced subspaces are independent and we assume a snapshot is *a priori* available
  - (1) Dimension of the local space for each subdomain?
  - (2) Is a given subdomain is reducible?
- (1) and (2) will be treated by cross-validation (e.g. W. J. Krzanowski. Cross-validation in principal component analysis. Biometrics, 43(3):575-584, 1987.)
  - **Training set**: snapshot
  - **Validation set**: set of additional finescale solutions
  - Independent training/validation avoids overfitting
  - Cross validation **emulates independence**. Error calculated using the local reduced basis obtained by a snapshot POD transform of all the available snapshot solutions except the one corresponding to the value of the summation variable.
- **NOTE:** If the snapshot is not assumed *a priori* then
  - Assess whether the snapshot contains sufficient information, and generate additional, suitable, data if required
  - Most analysis (mostly by statisticians) assume the snapshot is known *a priori*. Recent review: Hervé Abdi and Lynne J. Williams. Principal component analysis. Wiley Interdisciplinary Reviews: Computational Statistics, 2(4):433{459, 2010.



Domain Partitioning
$\underline{v}_i^{(6)}$
2
4
6
8
10
1
3
5
7
9
$\underline{v}_b^{(9)}$



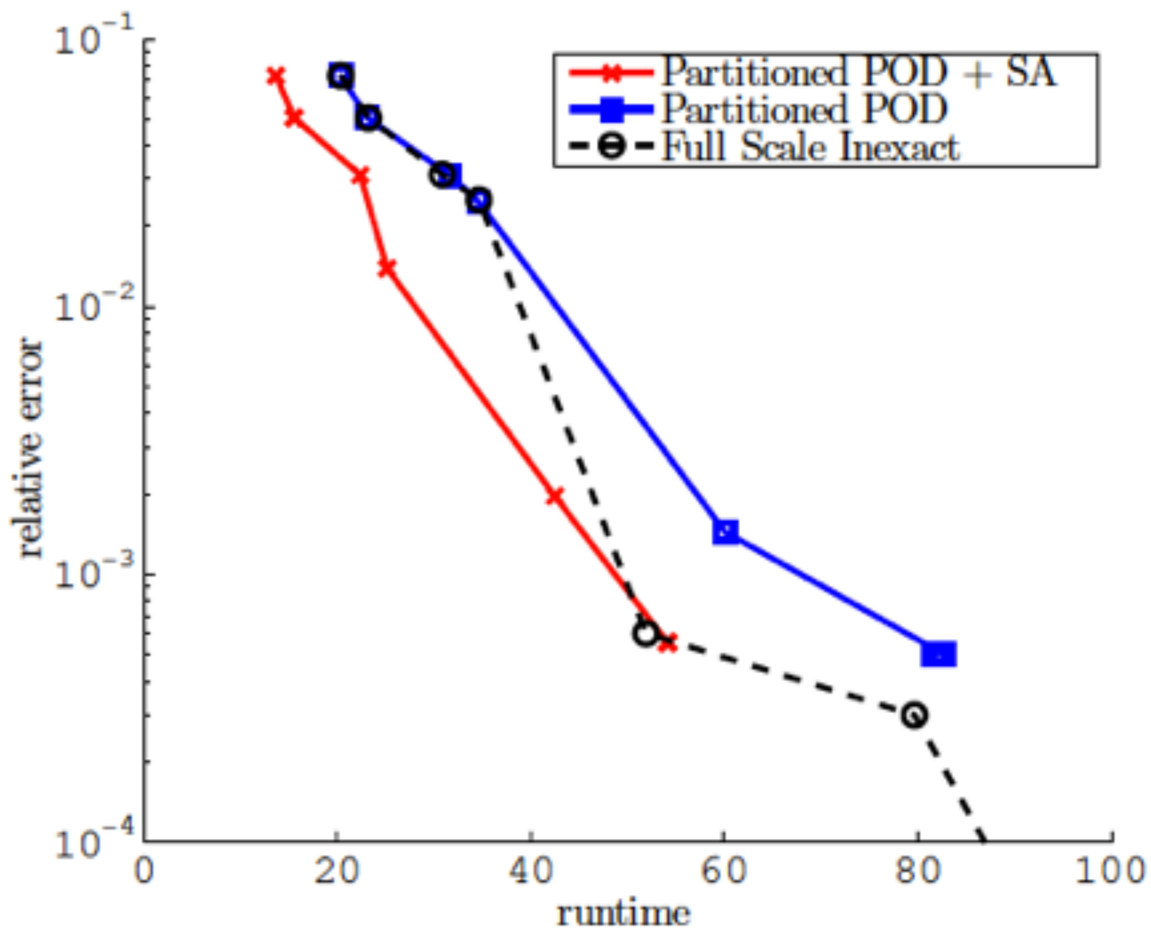
Cross-validation error estimate
Order of the POD transforms
subdomain 6
subdomain 4
subdomain 2
subdomain 7
10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
0
5
10
15
20

Performance: load angle 40 | 27 - 121 nodes

- Relative error

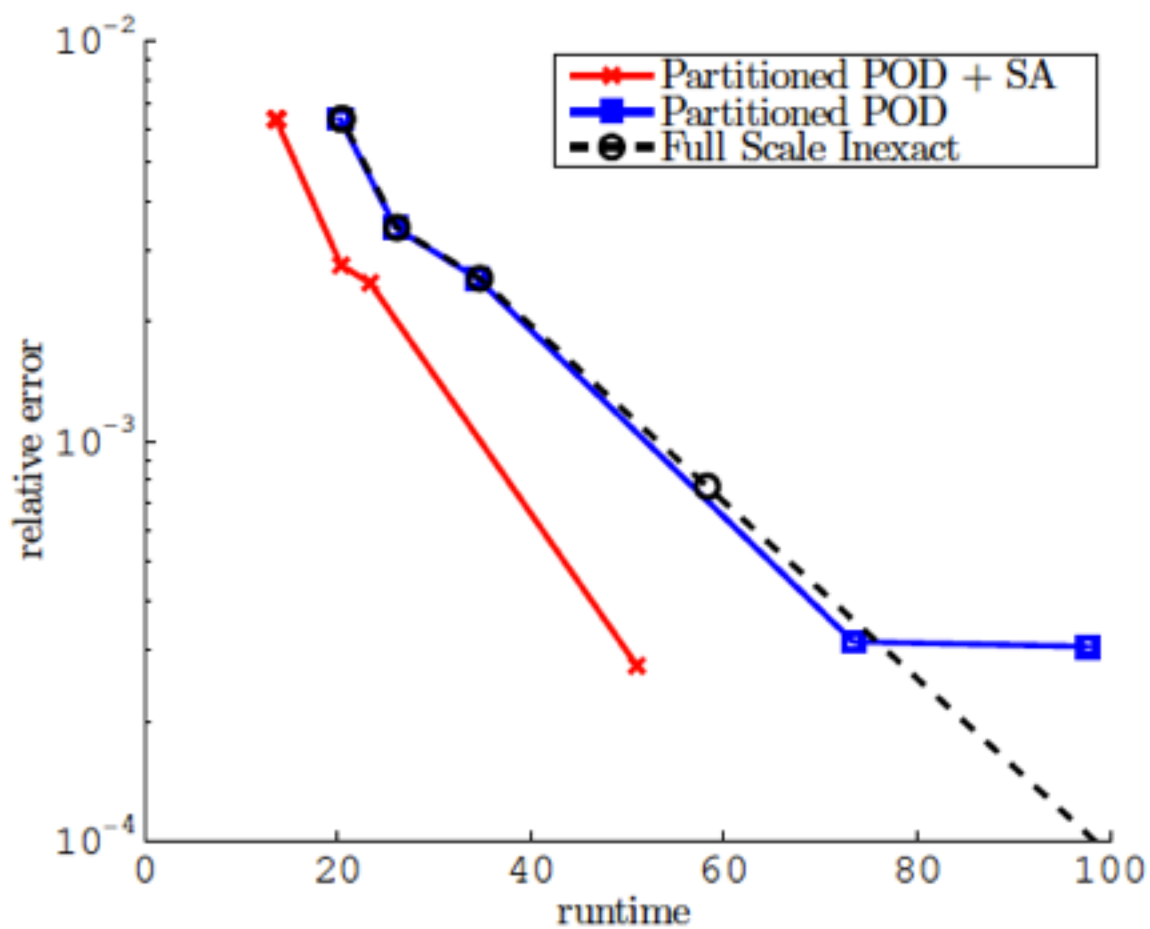
$$\nu^{\mathrm{app},(\mu)}\left(\underline{\mathbf{U}}^{\mathrm{app}}\right)^2 = \frac{\sum\limits_{t_n \in \mathcal{T}^h} \|\underline{\mathbf{U}}^{\mathrm{app}}(t_n,\mu) - \underline{\mathbf{U}}^{\mathrm{ex}}(t_n,\mu)\|_2^2}{\sum\limits_{t_n \in \mathcal{T}^h} \|\underline{\mathbf{U}}^{\mathrm{ex}}(t_n,\mu)\|_2^2}$$

40^o



(a) Relative error for the different models using 121 nodes per subdomain

27^o



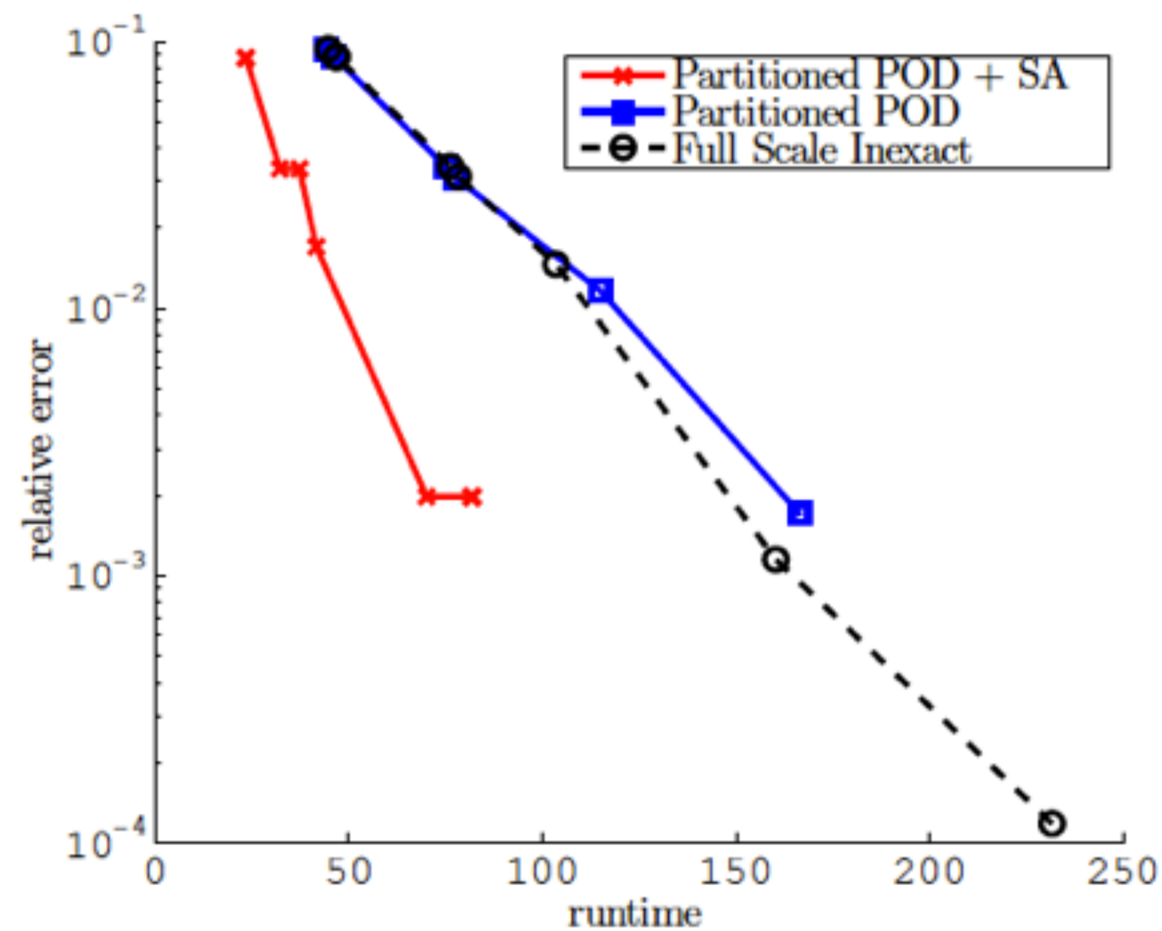
(a) Relative error for the different models using 121 no per subdomain

Performance: load angle 40 | 27 - 256 nodes

- Relative error

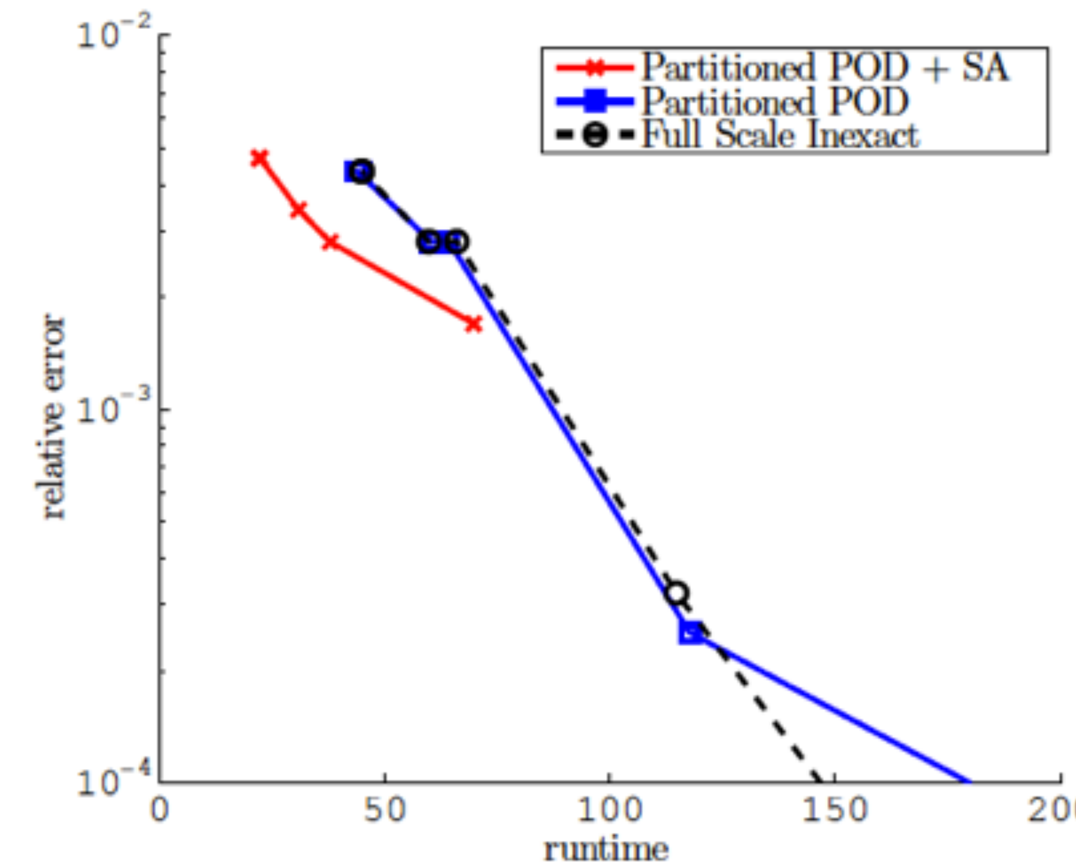
$$\nu^{\mathrm{app},(\mu)}\left(\underline{\mathbf{U}}^{\mathrm{app}}\right)^2 = \frac{\sum_{t_n \in \mathcal{T}^h} \|\underline{\mathbf{U}}^{\mathrm{app}}(t_n,\mu) - \underline{\mathbf{U}}^{\mathrm{ex}}(t_n,\mu)\|_2^2}{\sum_{t_n \in \mathcal{T}^h} \|\underline{\mathbf{U}}^{\mathrm{ex}}(t_n,\mu)\|_2^2}$$

40^o



(b) Relative error for the different models using 256 nodes per subdomain

27^o



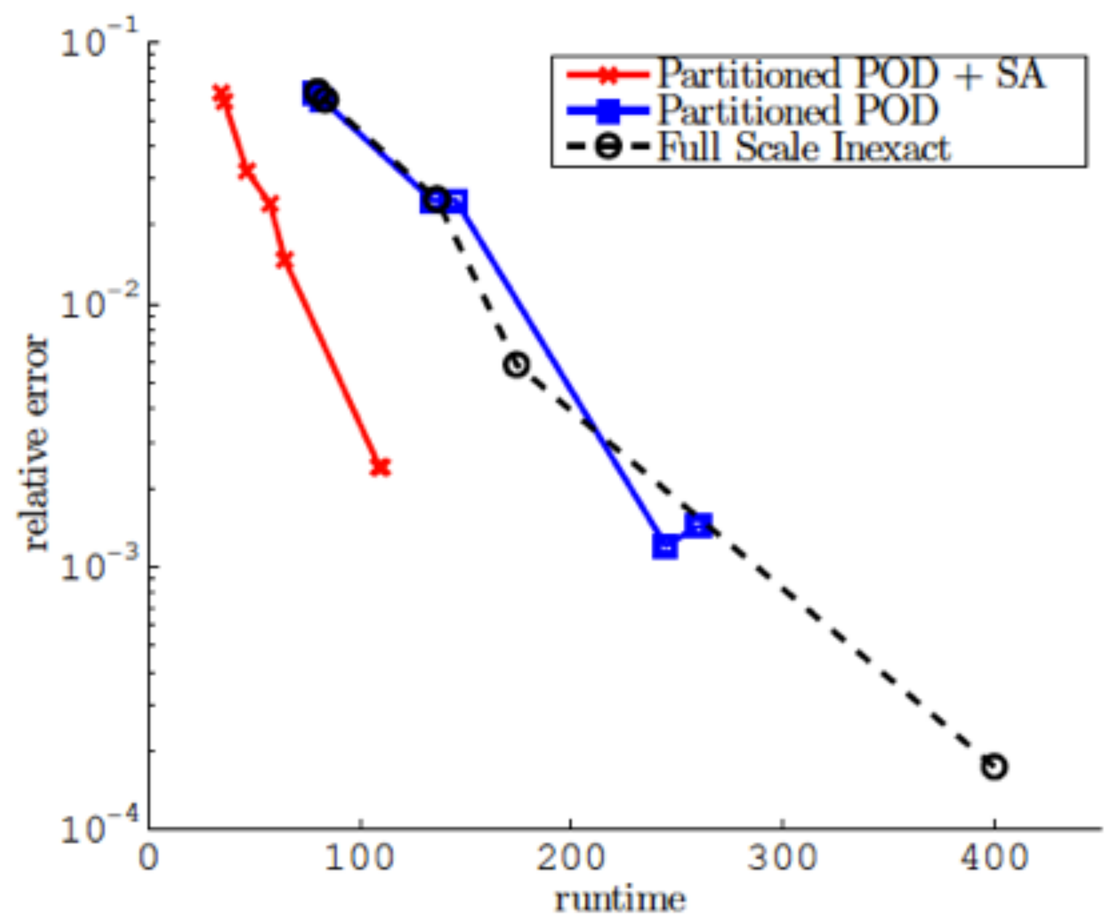
(b) Relative error for the different models using 256 nodes per subdomain

Performance: load angle 40 | 27 - 441 nodes

- Relative error

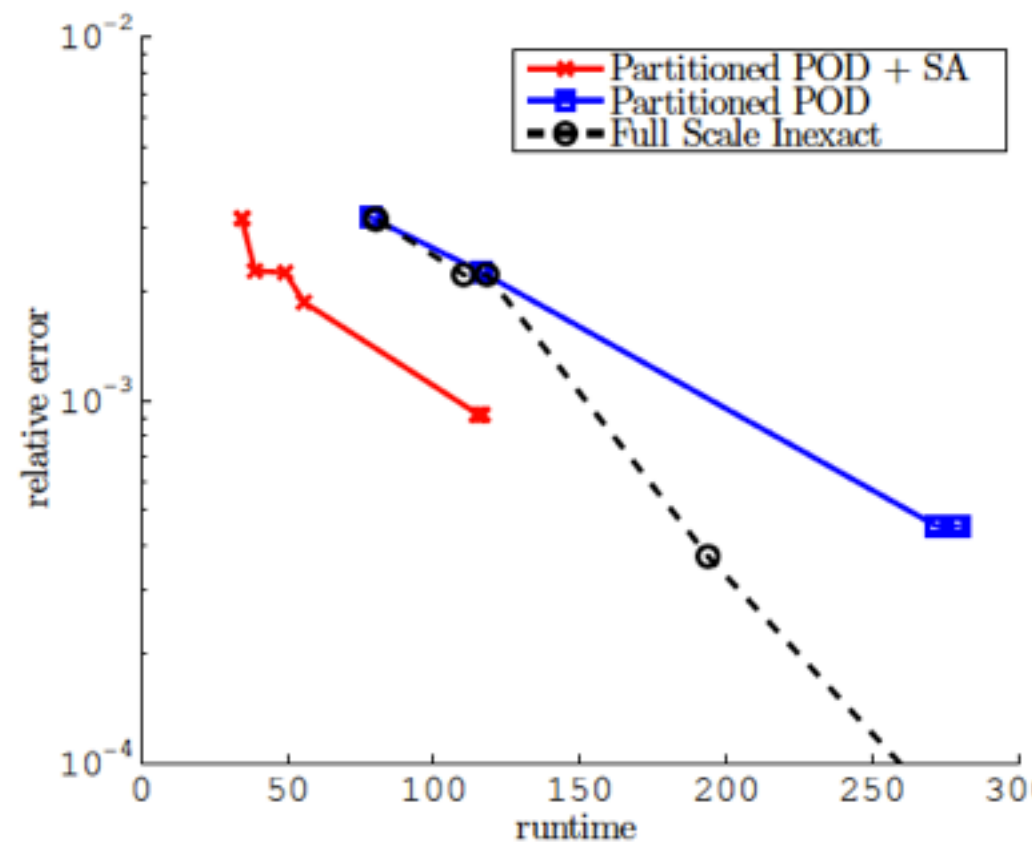
$$\nu^{\mathrm{app},(\mu)}\left(\underline{\mathbf{U}}^{\mathrm{app}}\right)^2 = \frac{\sum_{t_n \in \mathcal{T}^h} \|\underline{\mathbf{U}}^{\mathrm{app}}(t_n,\mu) - \underline{\mathbf{U}}^{\mathrm{ex}}(t_n,\mu)\|_2^2}{\sum_{t_n \in \mathcal{T}^h} \|\underline{\mathbf{U}}^{\mathrm{ex}}(t_n,\mu)\|_2^2}$$

40^o



(c) Relative error for the different models using 441 nodes per subdomain

27^o



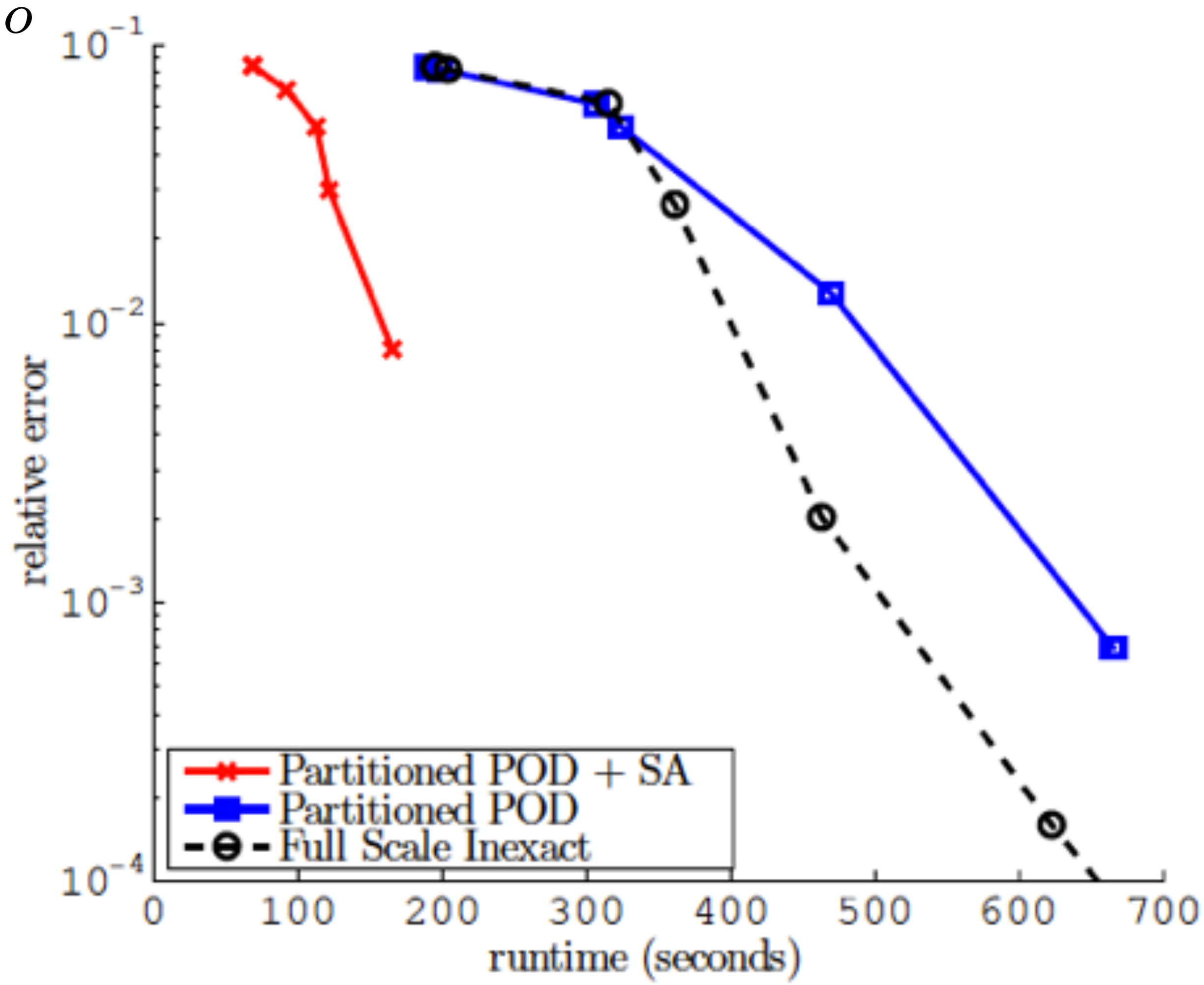
(c) Relative error for the different models using 441 nodes per subdomain

Performance: load angle 40 | 27 - 961 nodes

- Relative error

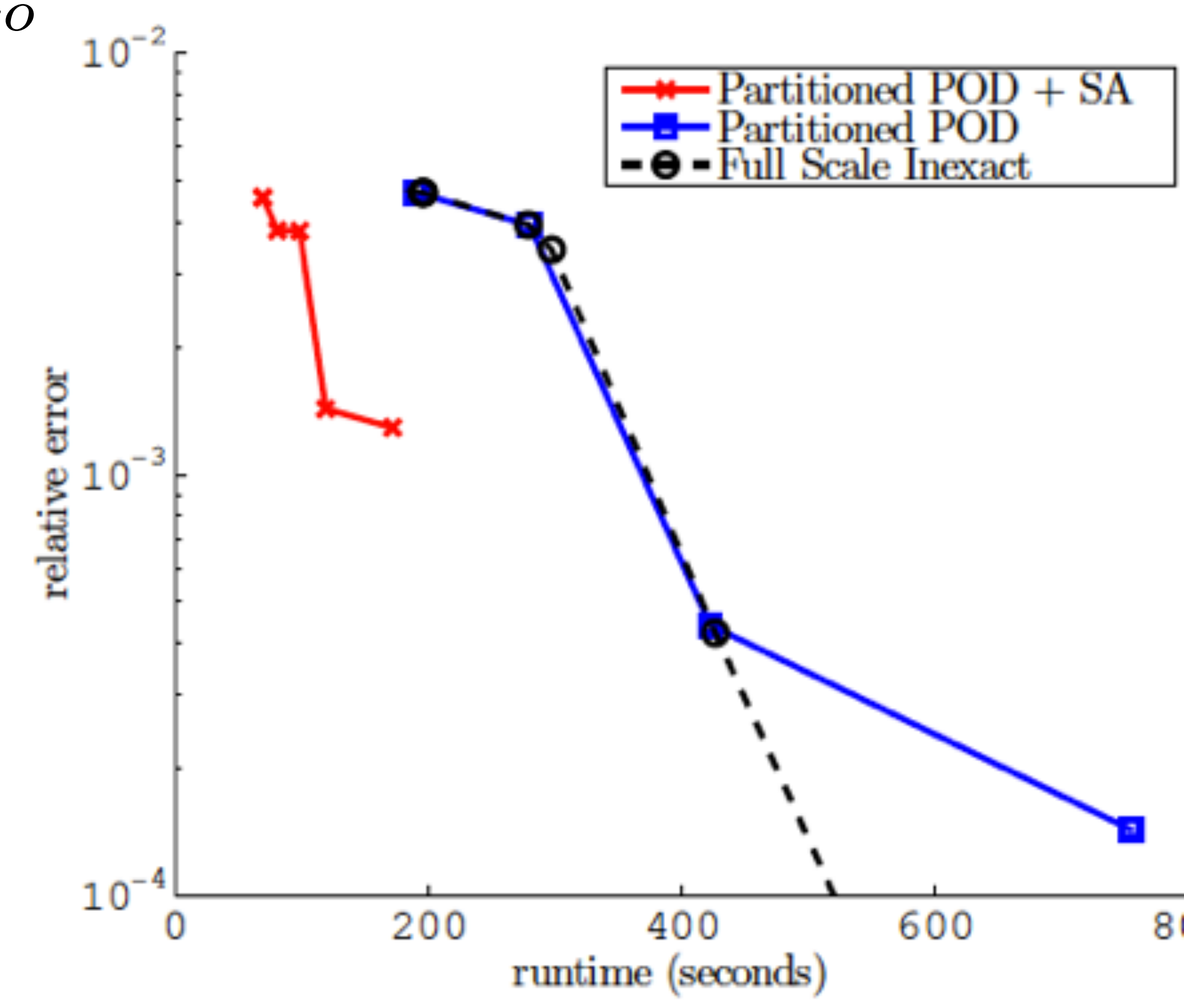
$$\nu^{\mathrm{app},(\mu)}\left(\underline{\mathbf{U}}^{\mathrm{app}}\right)^2 = \frac{\sum_{t_n \in \mathcal{T}^h} \|\underline{\mathbf{U}}^{\mathrm{app}}(t_n,\mu) - \underline{\mathbf{U}}^{\mathrm{ex}}(t_n,\mu)\|_2^2}{\sum_{t_n \in \mathcal{T}^h} \|\underline{\mathbf{U}}^{\mathrm{ex}}(t_n,\mu)\|_2^2}$$

40^o

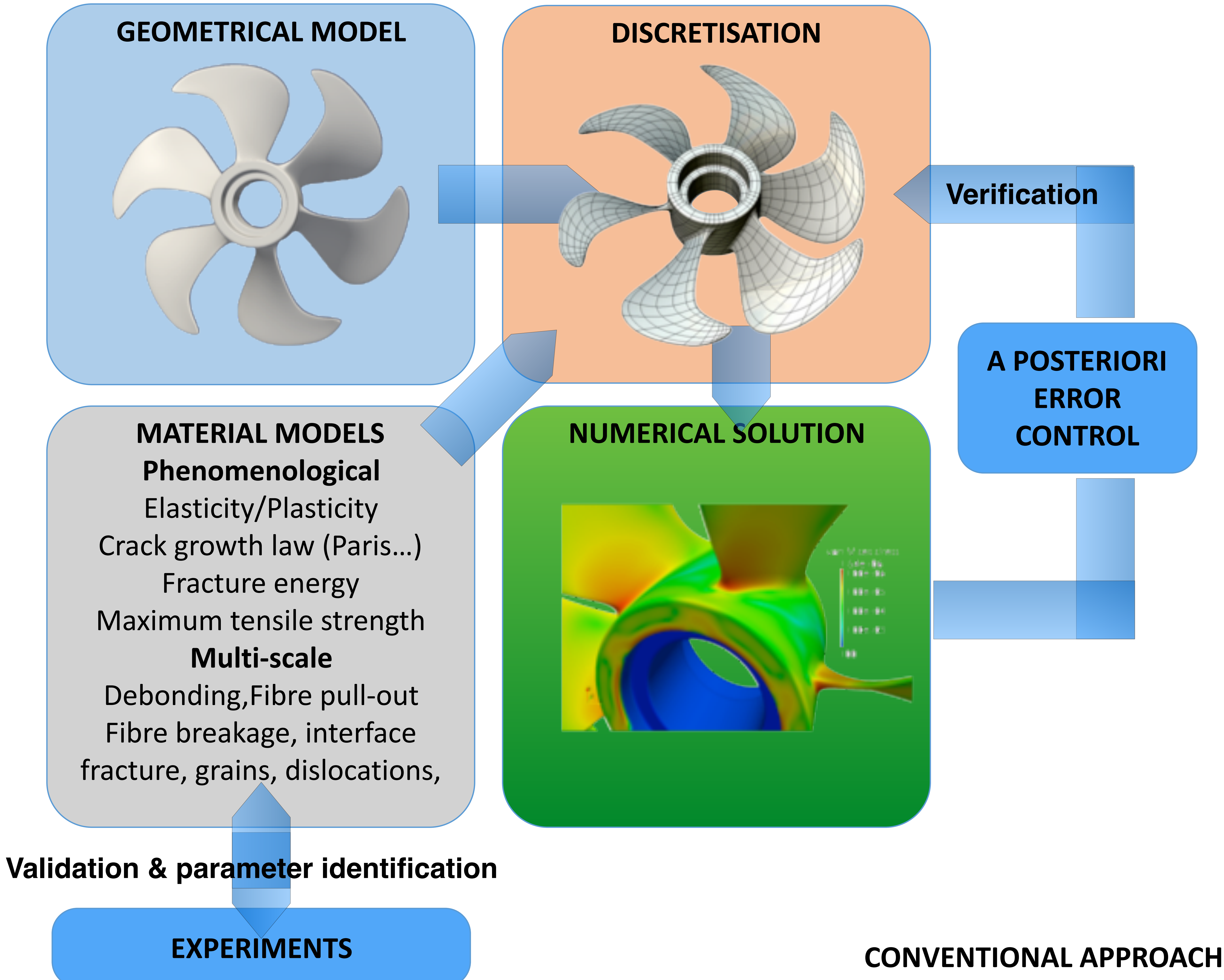


(d) Relative error for the different models using 961 nodes per subdomain

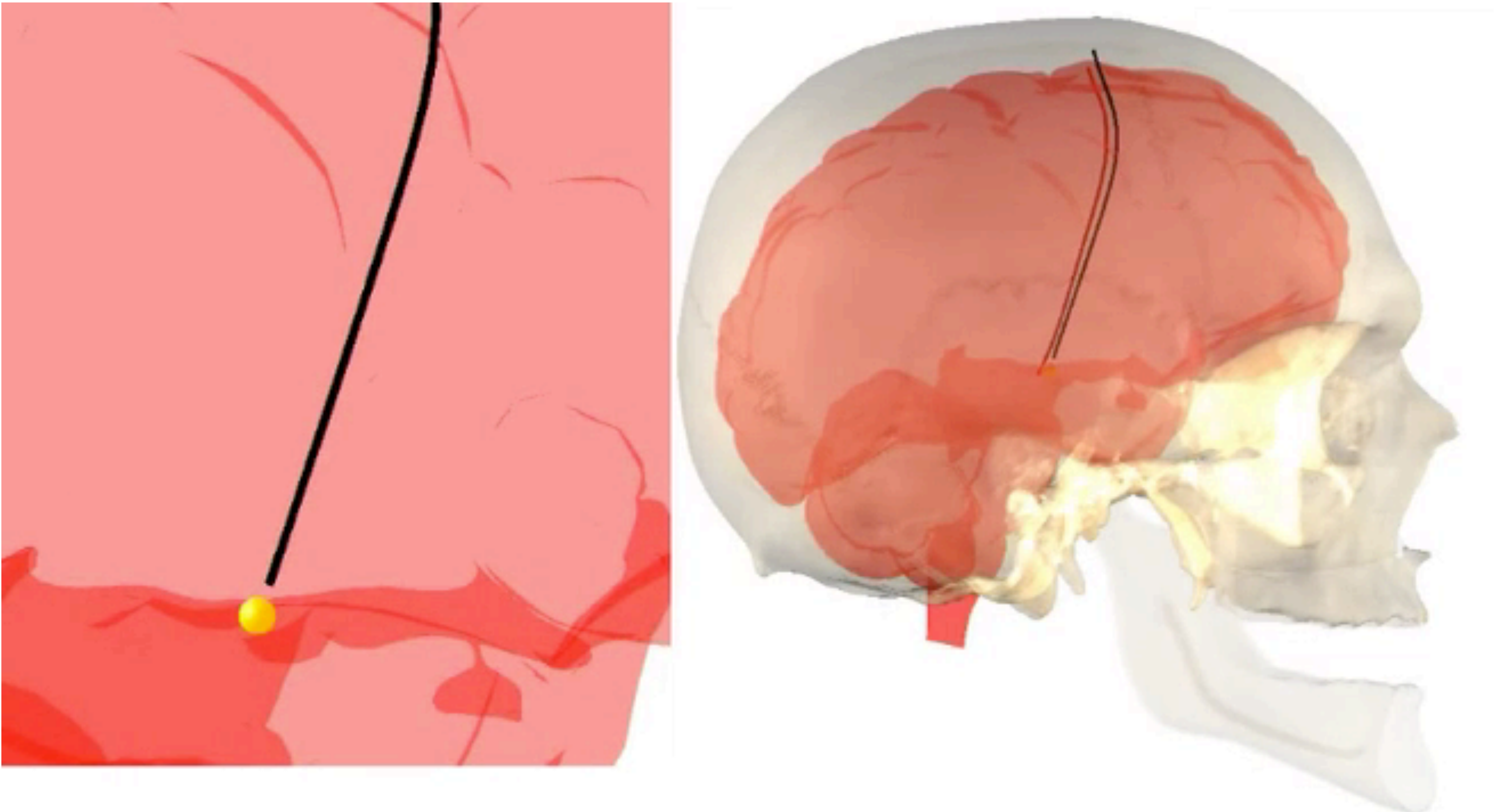
27^o



(d) Relative error for the different models using 961 per subdomain

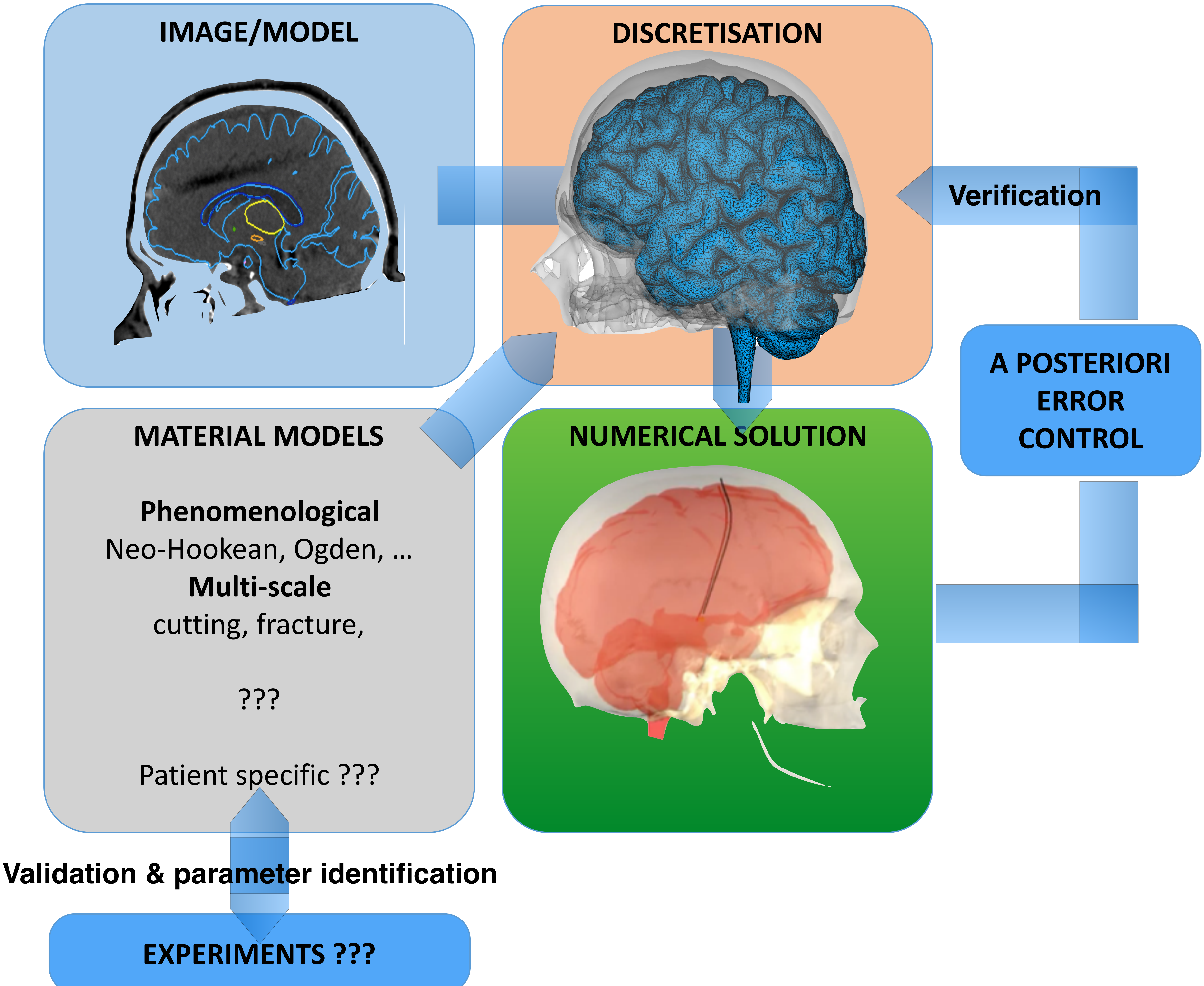

GEOMETRICAL MODEL
DISCRETISATION
Verification
A POSTERIORI ERROR CONTROL
MATERIAL MODELS
Phenomenological
Elasticity/Plasticity
Crack growth law (Paris...)
Fracture energy
Maximum tensile strength
Multi-scale
Debonding,Fibre pull-out
Fibre breakage, interface
fracture, grains, dislocations,
NUMERICAL SOLUTION
Validation & parameter identification
EXPERIMENTS
CONVENTIONAL APPROACH

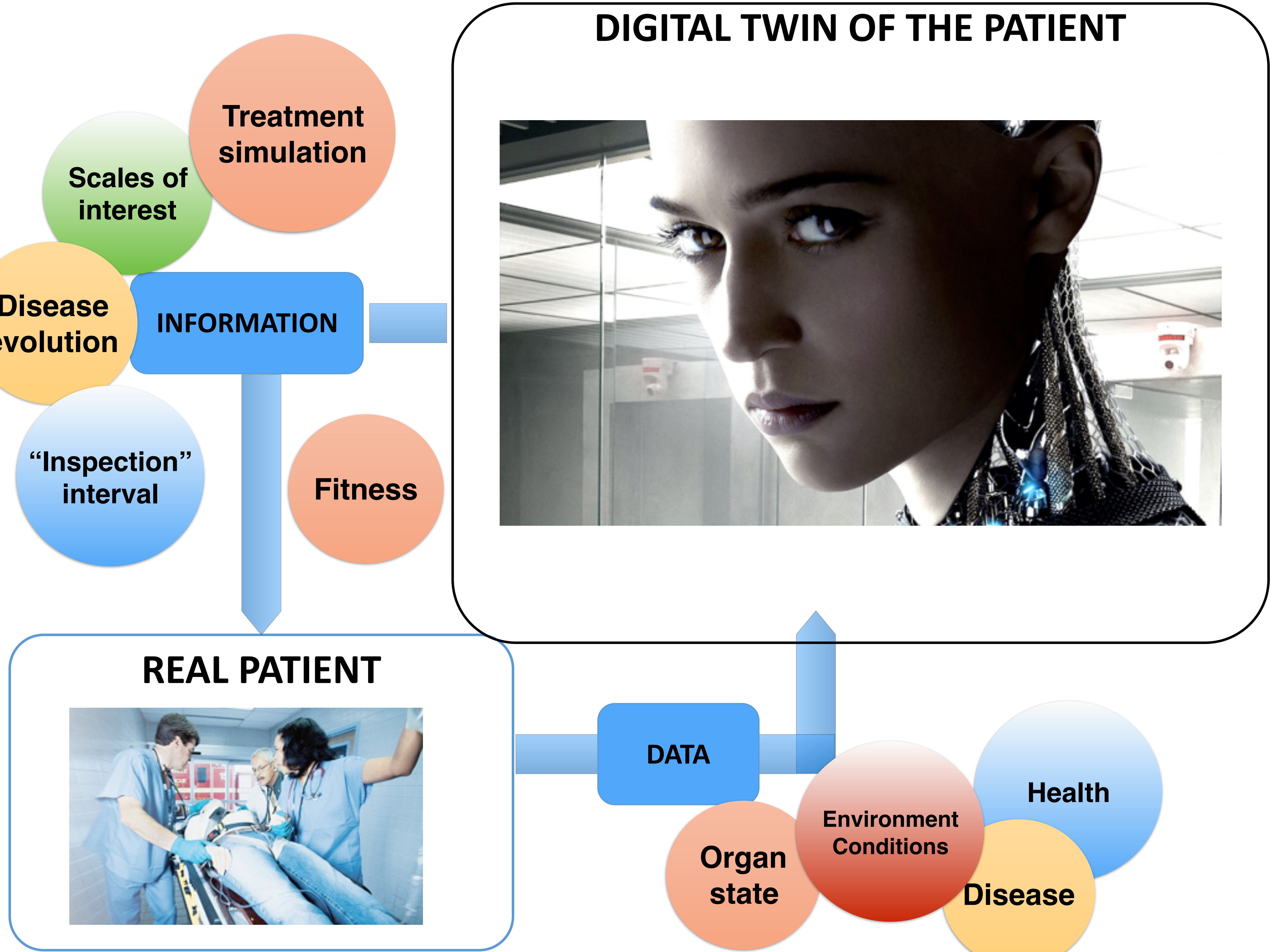
When the material model is not known, this conventional approach is inadequate



Deep-brain stimulation

Courtesy Alexandre Bilger, PhD thesis, Inria, 2014


IMAGE/MODEL
DISCRETISATION
Verification
A POSTERIORI ERROR CONTROL
MATERIAL MODELS
Phenomenological
Neo-Hookean, Ogden, …
Multi-scale
cutting, fracture,
???
Patient specific ???
NUMERICAL SOLUTION
Validation & parameter identification
EXPERIMENTS ???



DIGITAL TWIN OF THE PATIENT
Treatment simulation
Scales of interest
Disease evolution
INFORMATION
"Inspection" interval
Fitness
REAL PATIENT
DATA
Health
Environment Conditions
Organ state
Disease

news from legato

http://legato-team.eu

Thank you for your attention!

International Centre
for Mechanical Sciences

CISM

motus est vita

CISM-ECCOMAS International Summer School on “Modelling, Simulation and Characterization of Multi-Scale Heterogeneous Materials” September 28, 2015 — October 2, 2015

OPEN SOURCE CODES

PERMIX: Multiscale, XFEM, large deformation, coupled 2 LAMMPS, ABAQUS, OpenMP -

MATLAB Codes: XFEM, 3D ISOGEOMETRIC XFEM, 2D ISOGEOMETRIC BEM, 2D MESHLESS
DOWNLOAD @ http://cmechanicsos.users.sourceforge.net/

COMPUTATIONAL MECHANICS DISCUSSION GROUP Request membership @
http://groups.google.com/group/computational_mechanics_discussion/about

Conclusions and perspectives

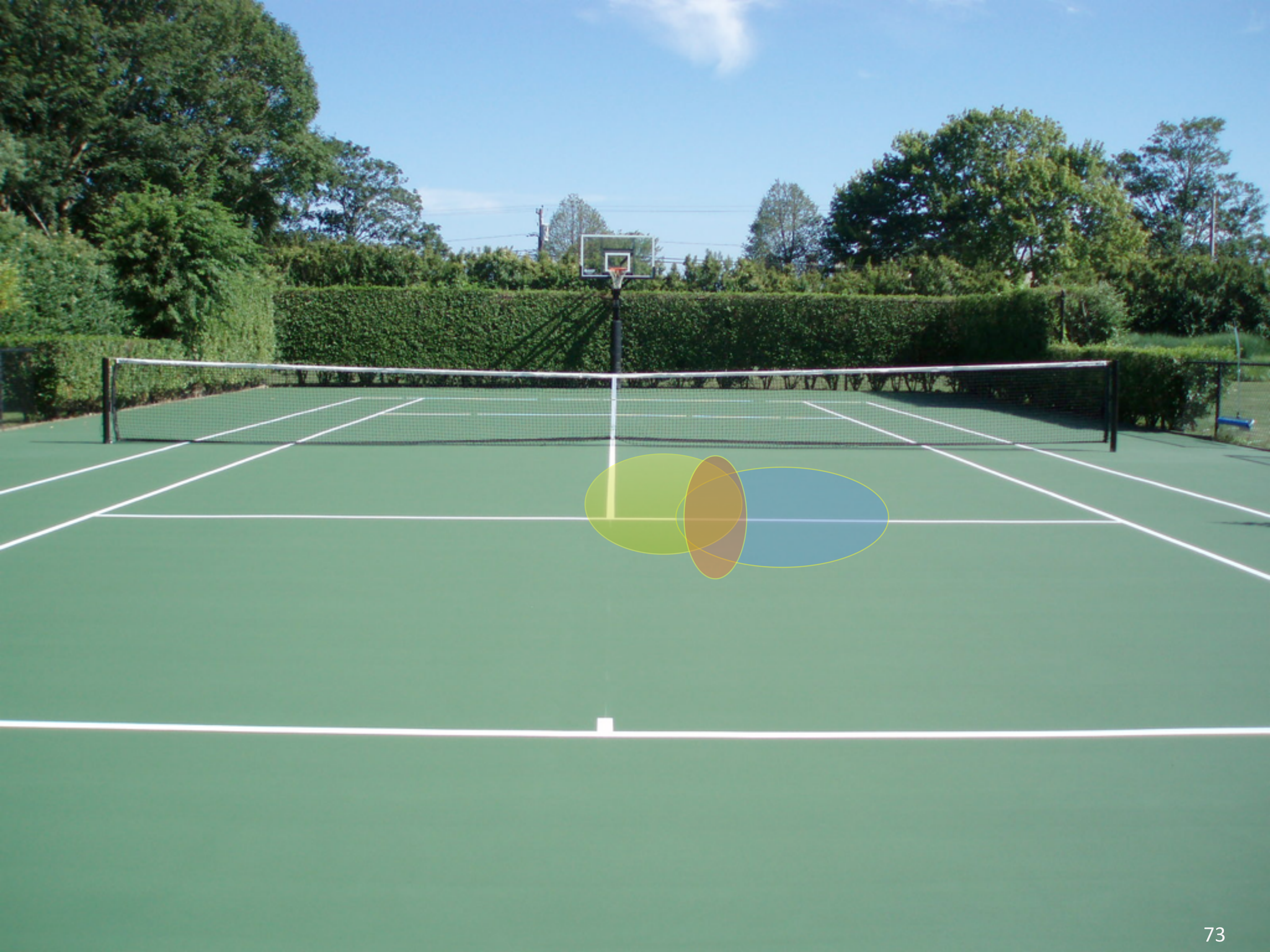
- Domain coupling using the primal Schur-complement domain decomposition method.
- Local subproblems have been reduced by projection in low-dimensional subspaces obtained by the snapshot POD.
- This approach permits to flexibly reduce the computational cost associated with highly nonlinear problems. In particular:
  - the **local reduced spaces are generated independently**, and have independent dimensions, which allows us to focus the numerical effort where it is most needed.
  - subdomains that are close to highly damaged zones need a richer model to account for the effect of topological changes. The local **POD transforms automatically generate local reduced spaces of larger dimension in these zones**.
  - the domain decomposition framework enables us to **switch from reduced local solvers to full local solvers** in a transparent manner. This is particularly useful for the subdomains that contain process zones, as a solution obtained by projection would be more expensive than a direct solution for a desirable accuracy.
  - the transition between ``offline'' and ``online'' computations becomes flexible. The **reduced models can be used in the zones where the local reduced spaces converge quickly** when enriching the snapshot space, while still computing snapshots and refining the reduced models via a direct local solver in the remaining subdomains.

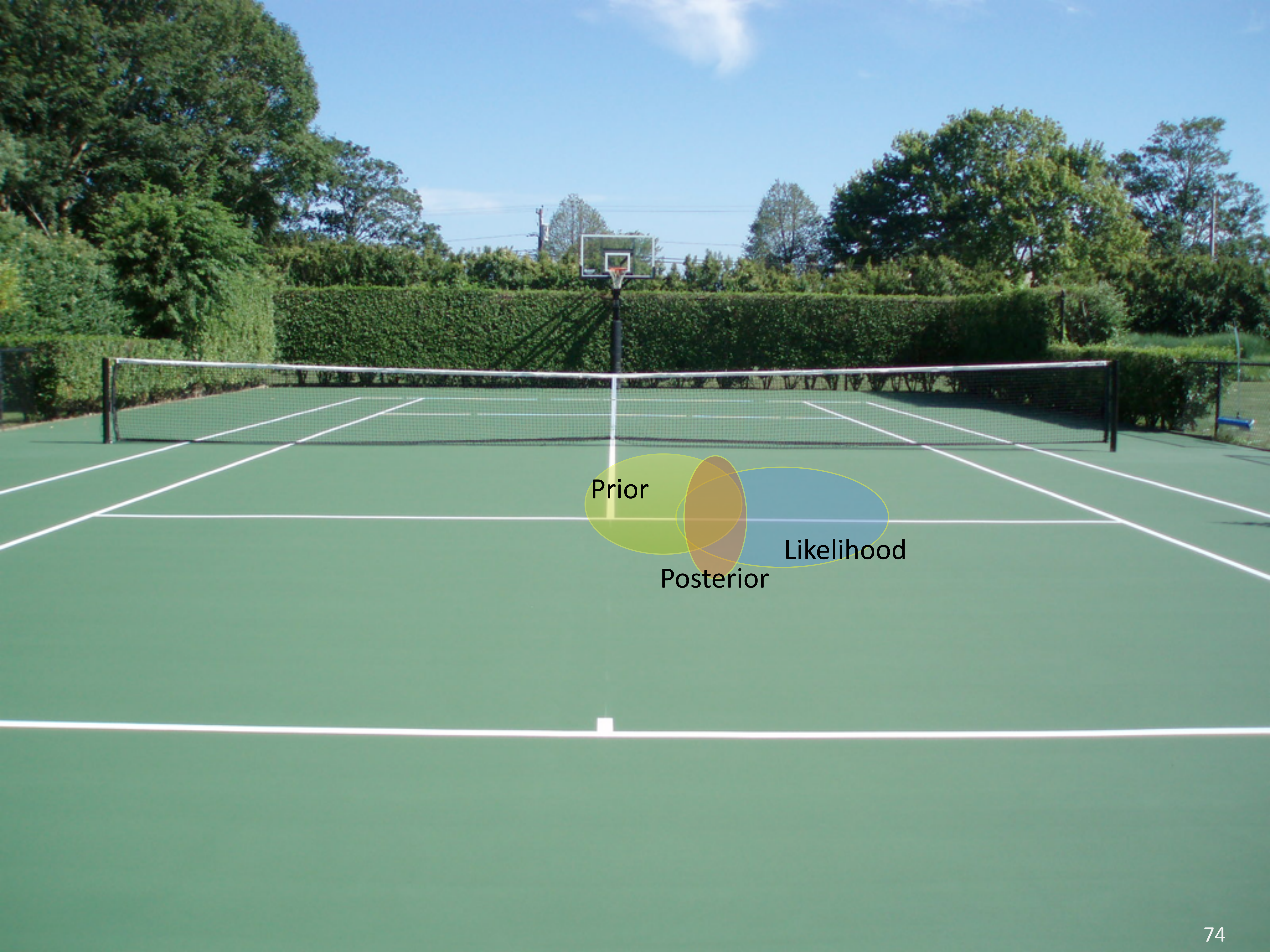
Perspectives

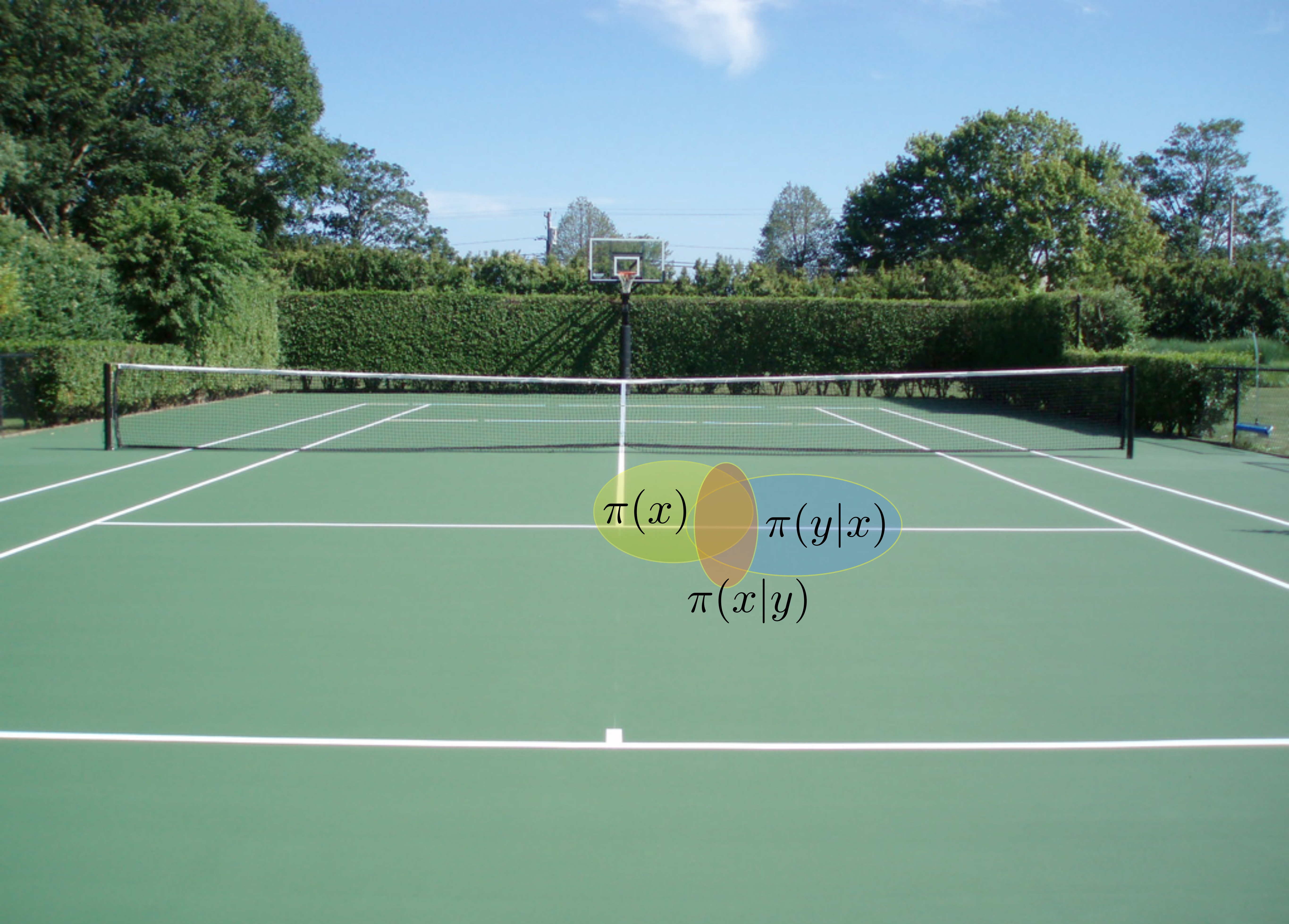
- Further work related to domain decomposition
  - **load balancing** mismatch would occur when using such a strategy in parallel. CPUs which support domains that are not reduced, or domains for which the corresponding subproblems need to be projected in a space of relatively high dimension, would require to perform more operations. The domain partitioning itself should be performed jointly with the model reduction in order to distribute the load evenly.
  - **the interface problem itself was not reduced** here, to guarantee the interface kinematic compatibility.
    - ➡ Suboptimal reduced order model. Would generate expensive communications in parallel
    - ➡ A reduction of the interface problem using the POD can be done but is neither elegant nor easy
    - ➡ Dual Schur-complement domain decomposition method would allow the kinematic approximation of the subproblems to include the interface. However, this would only deflect the difficulty to the necessary reduction of the interface Lagrange multiplier space. This issue is our current direction of research.

Bayesian inference

Primer




Prior
Likelihood
Posterior

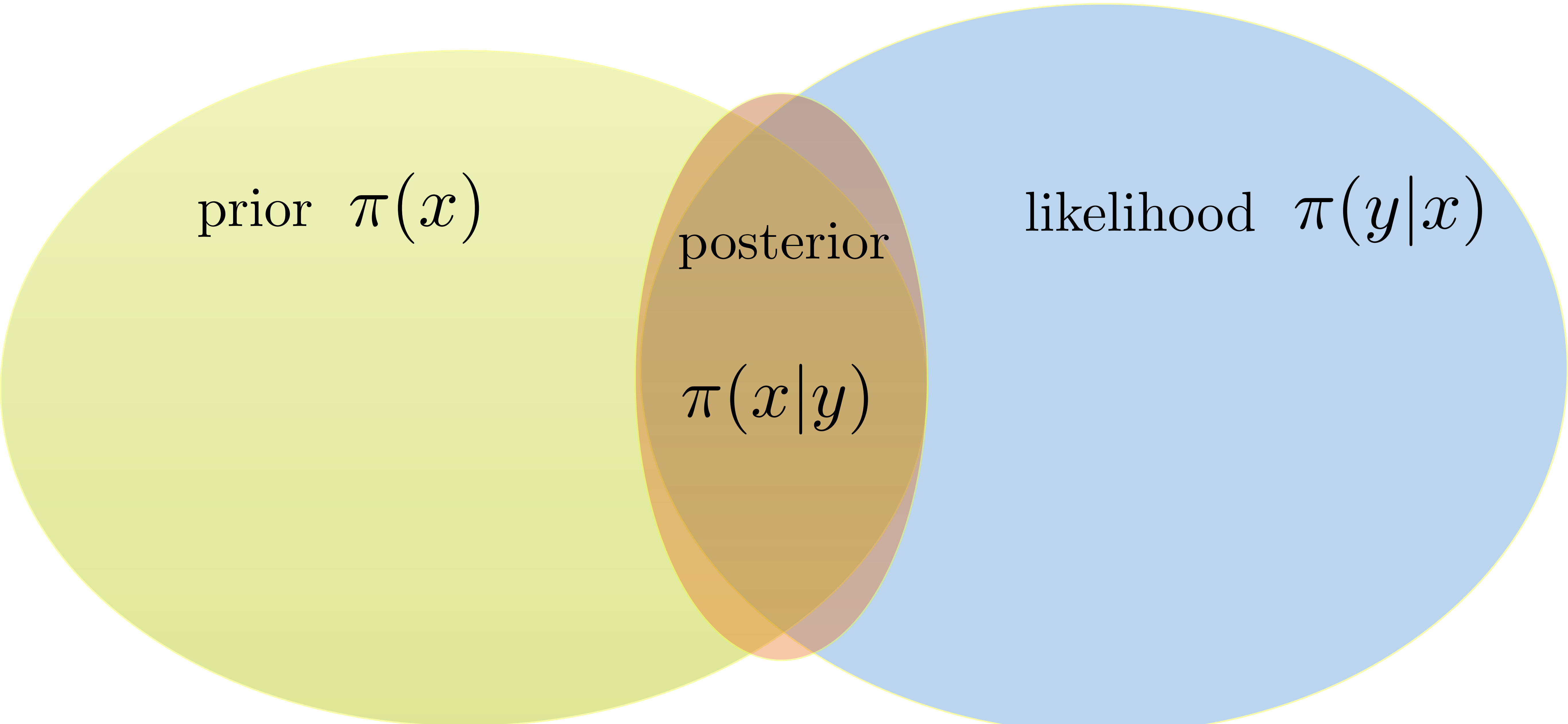


$\pi(x)$
$\pi(y|x)$
$\pi(x|y)$

Bayes' theorem

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

$$\pi(x|y) = \frac{\pi(x)\pi(y|x)}{\int \pi(x)\pi(y|x)dx}$$



Parameter identification: Bayesian approach

Bayes' theorem

$$\pi(x|y) = \frac{\pi(x)\pi(y|x)}{\int \pi(x)\pi(y|x)dx}$$

$\pi(.)$: probability distribution function

$\pi(.|.)$: conditional probability distribution function

x : material parameter

y : observations

Parameter identification: Bayesian approach

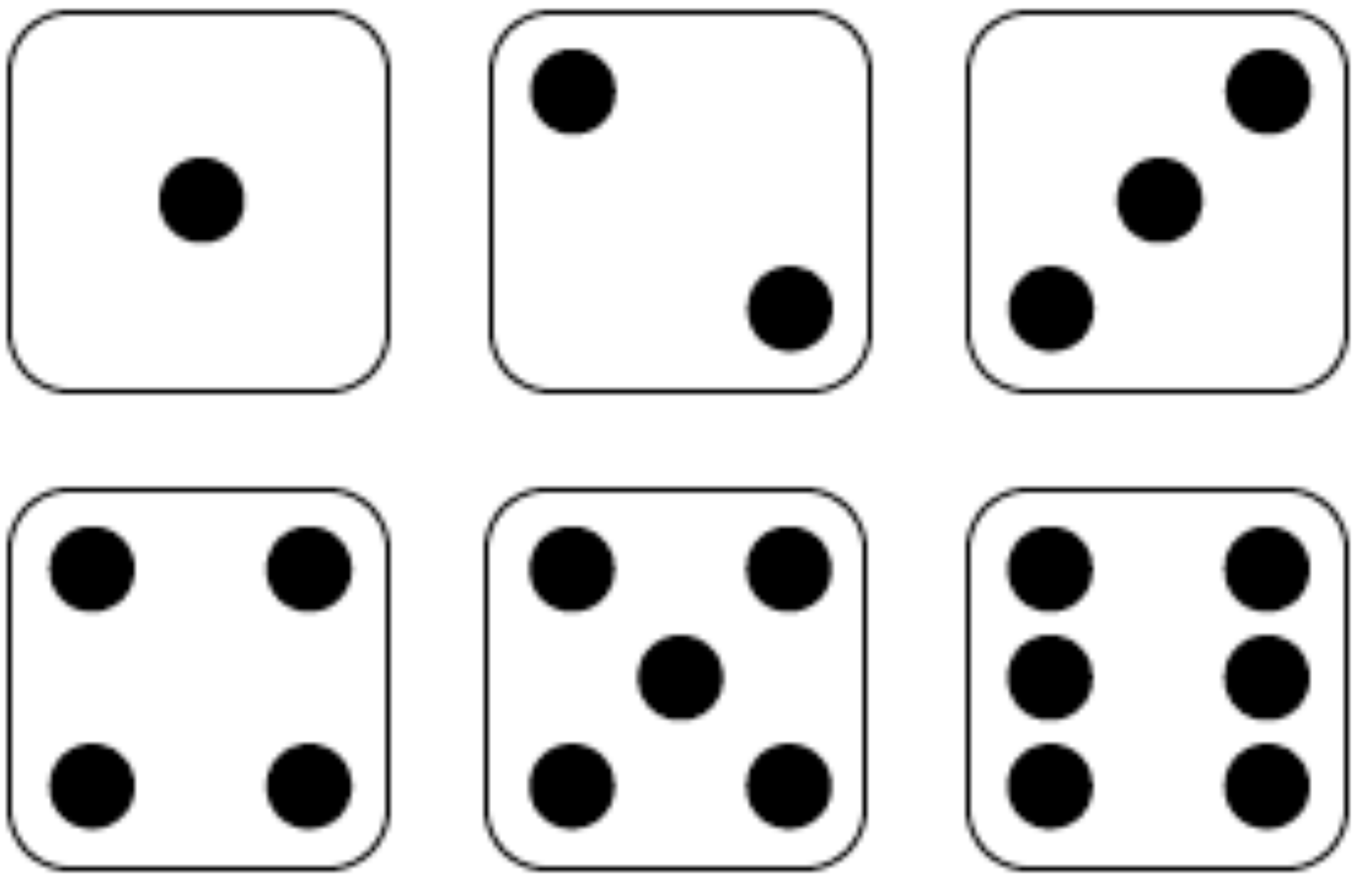
Bayes' theorem

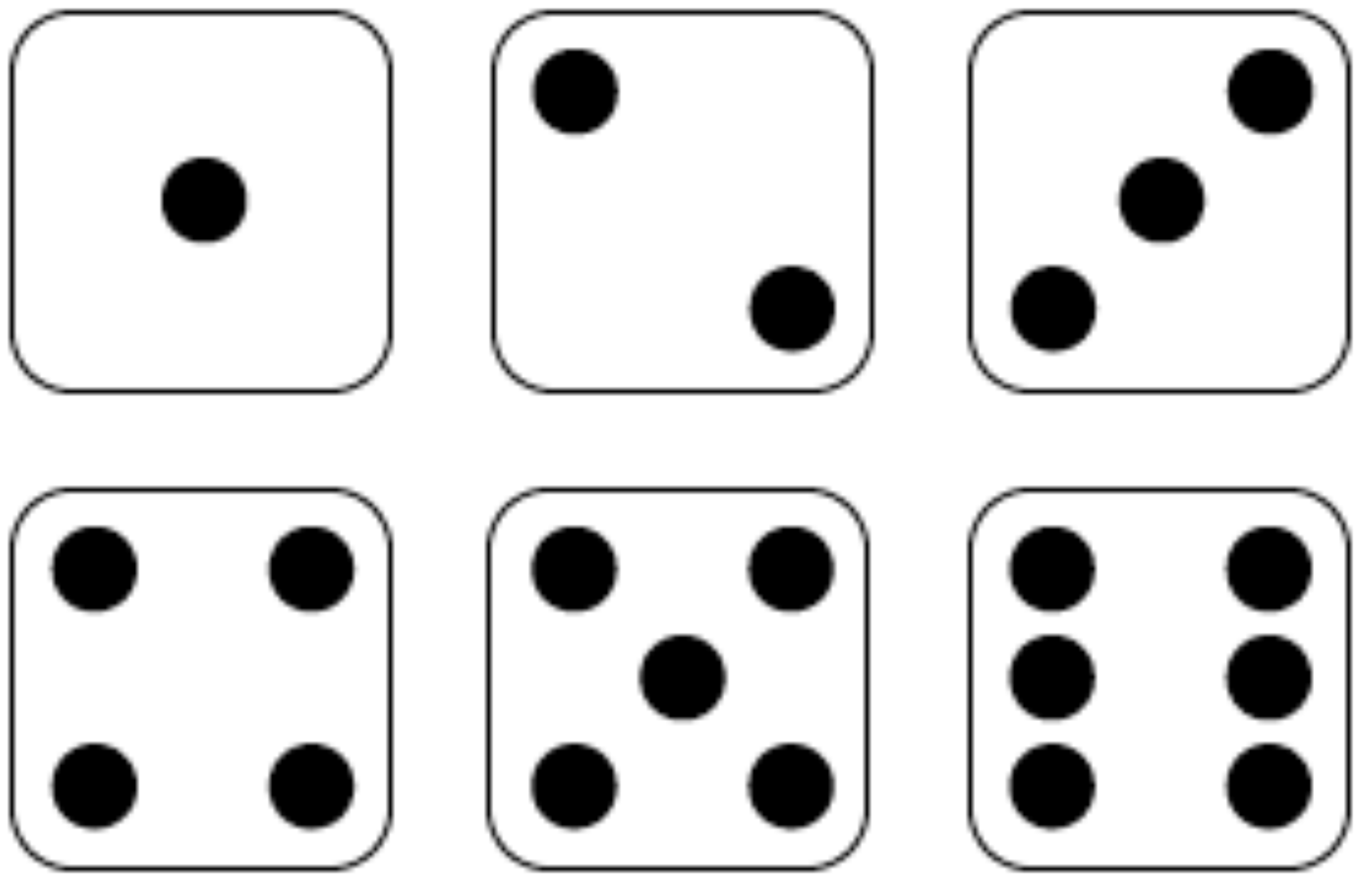
$$\pi(x|y) = \frac{\pi(x)\pi(y|x)}{\int \pi(x)\pi(y|x)dx}$$

Descriptive formula

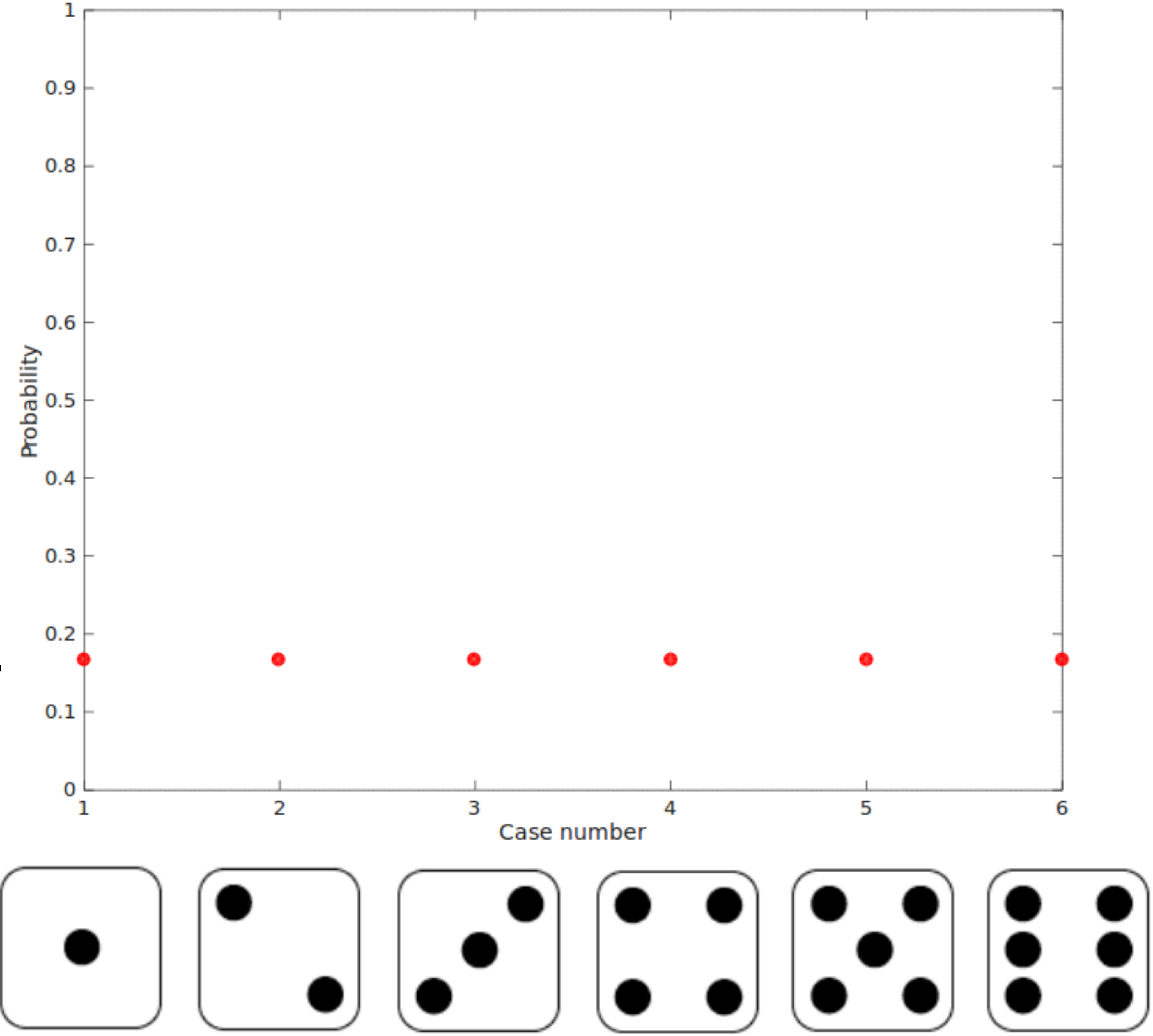
$$\text{Posterior} = \frac{\text{Prior} \times \text{Likelihood}}{\text{Evidence}}$$

A discrete example of Bayes' theorem





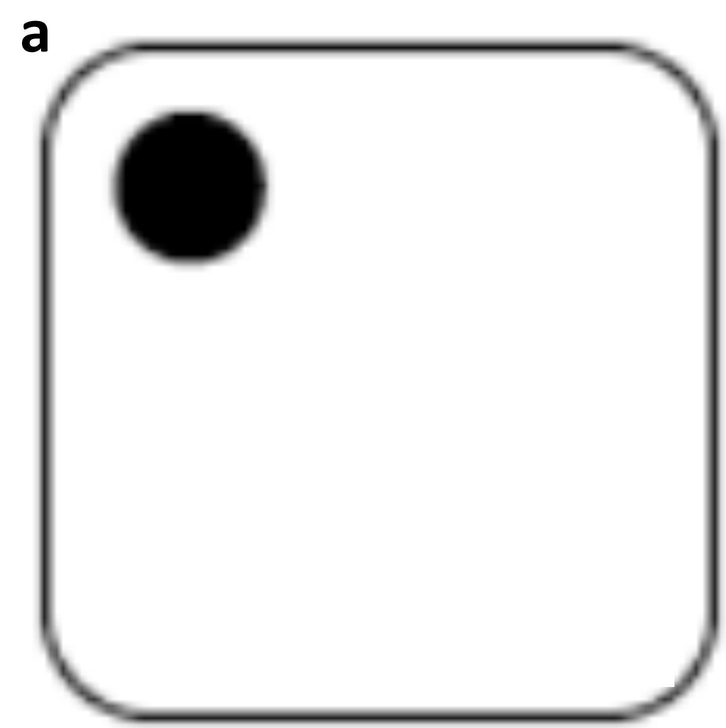
This is our prior information for the probability of each face: 1/6

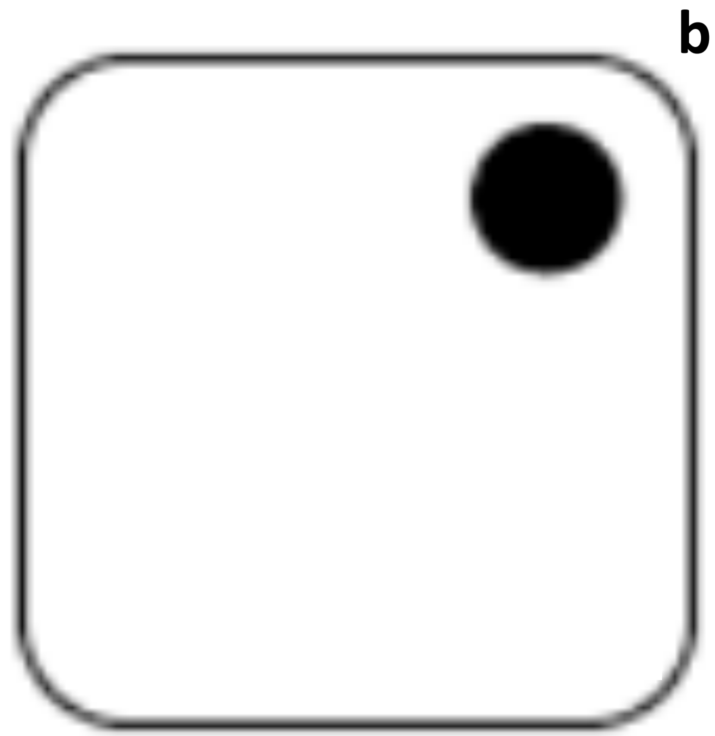


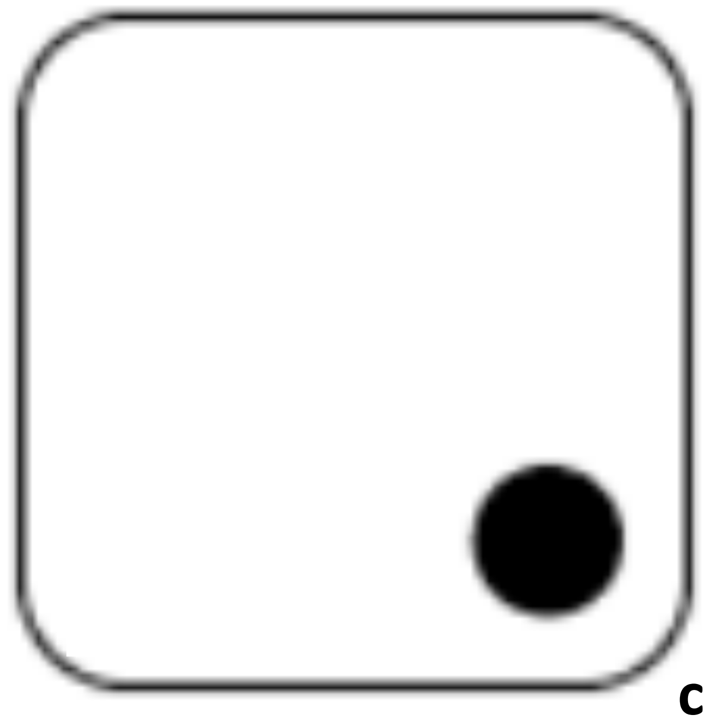
P=1/6

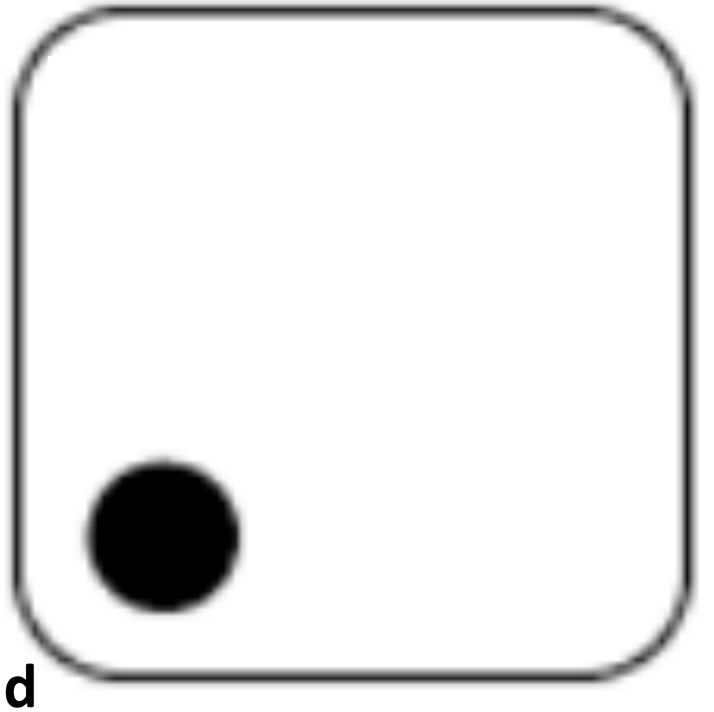
Assume that after throwing the dice, you see the above evidence

Goal: determine the probability of this evidence for each face of the dice

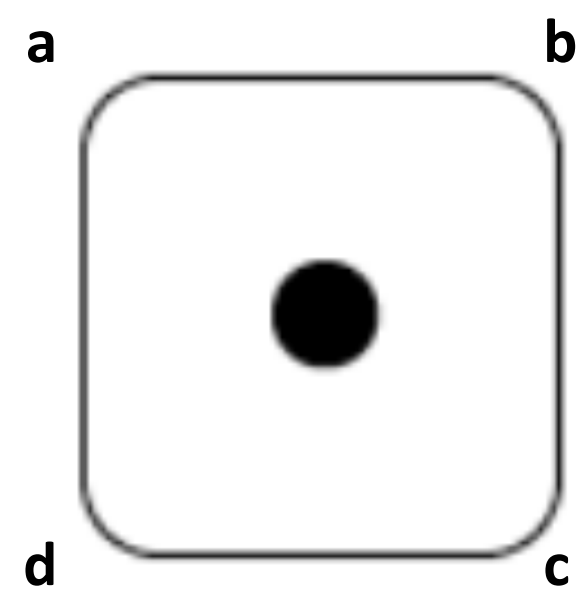

a

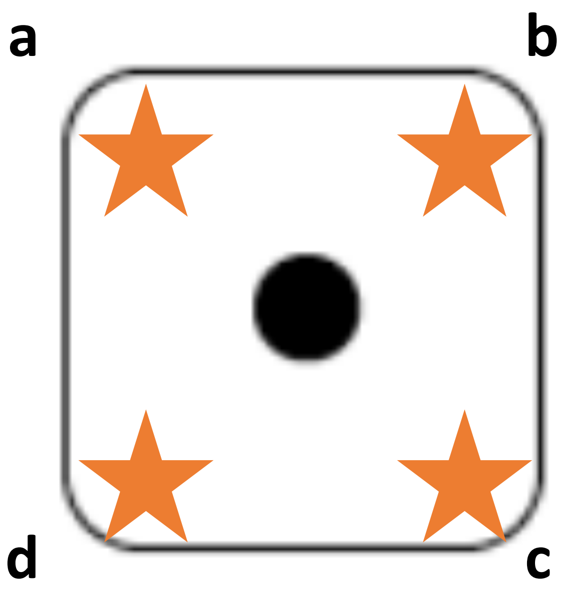

b

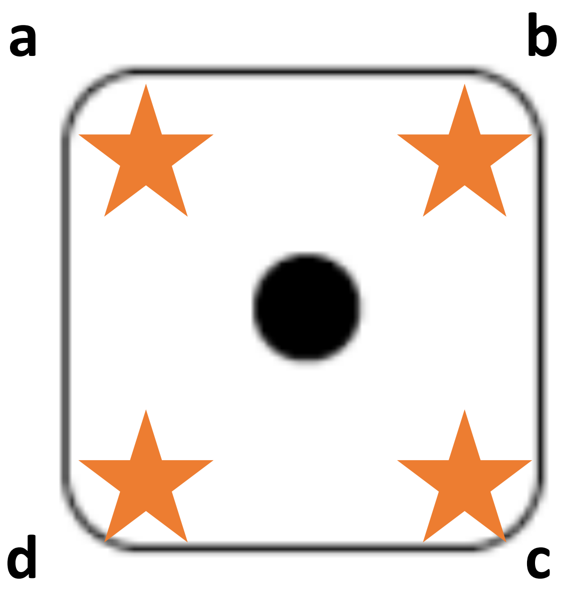

c



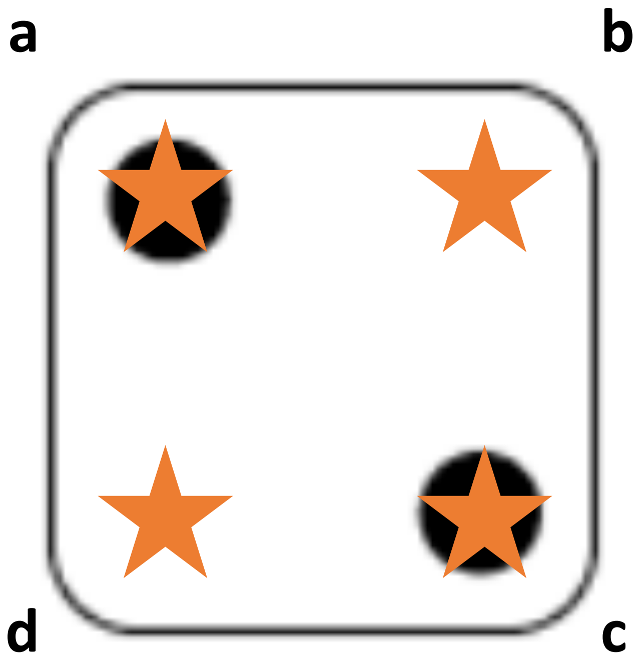
d

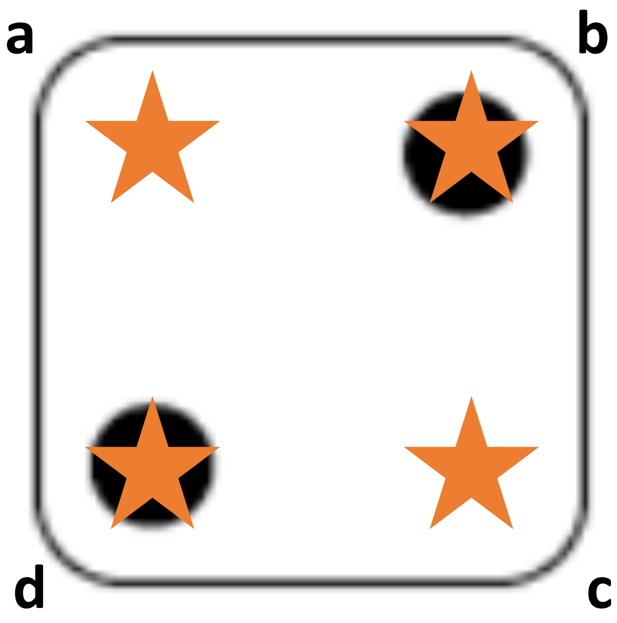

a
b
d
c


a
b
d
c



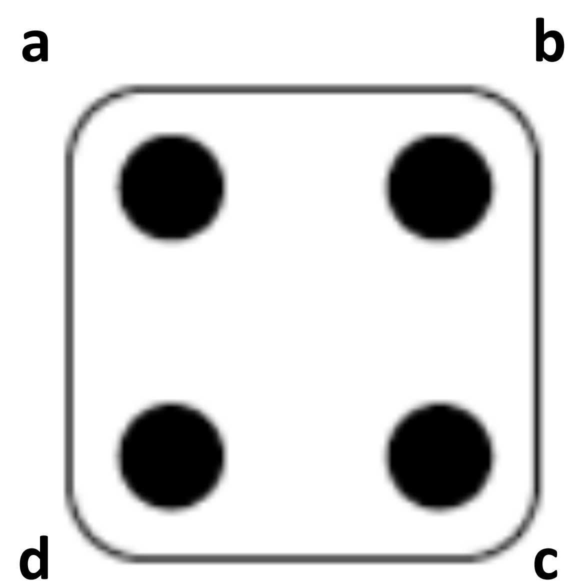
One would never see a dot at the star positions for this face
The probability of the evidence is *zero*

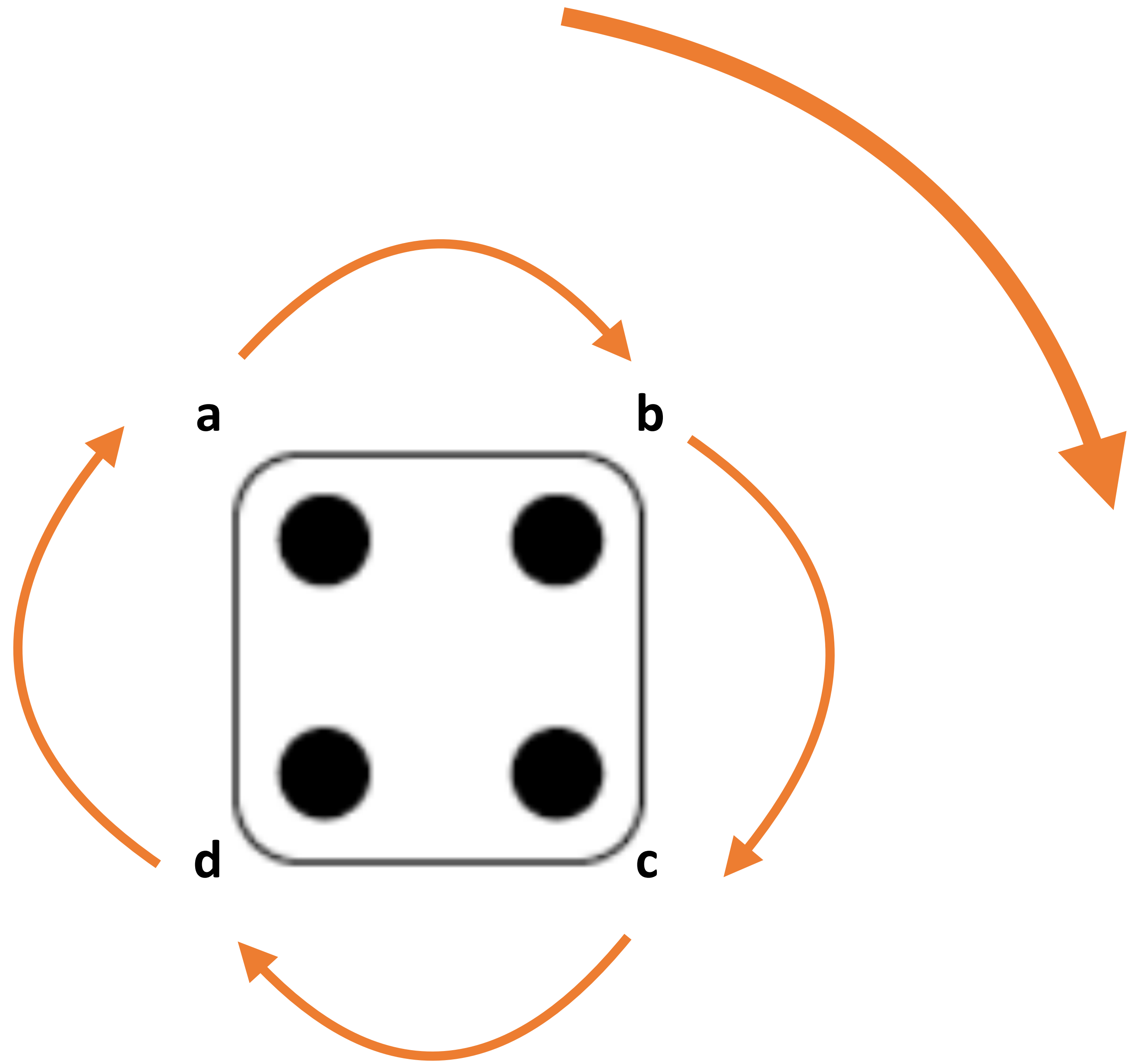


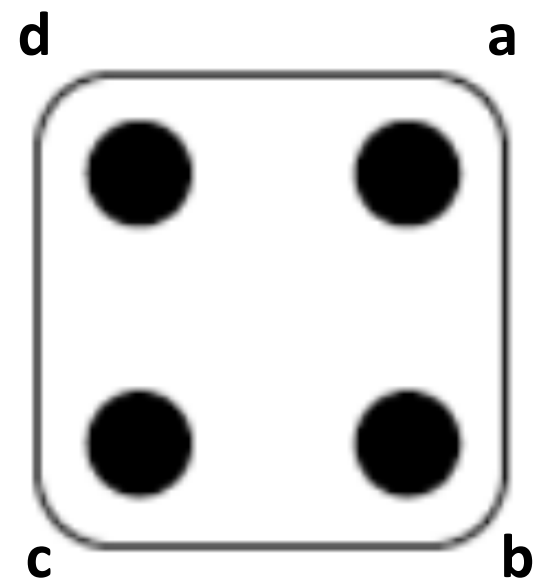


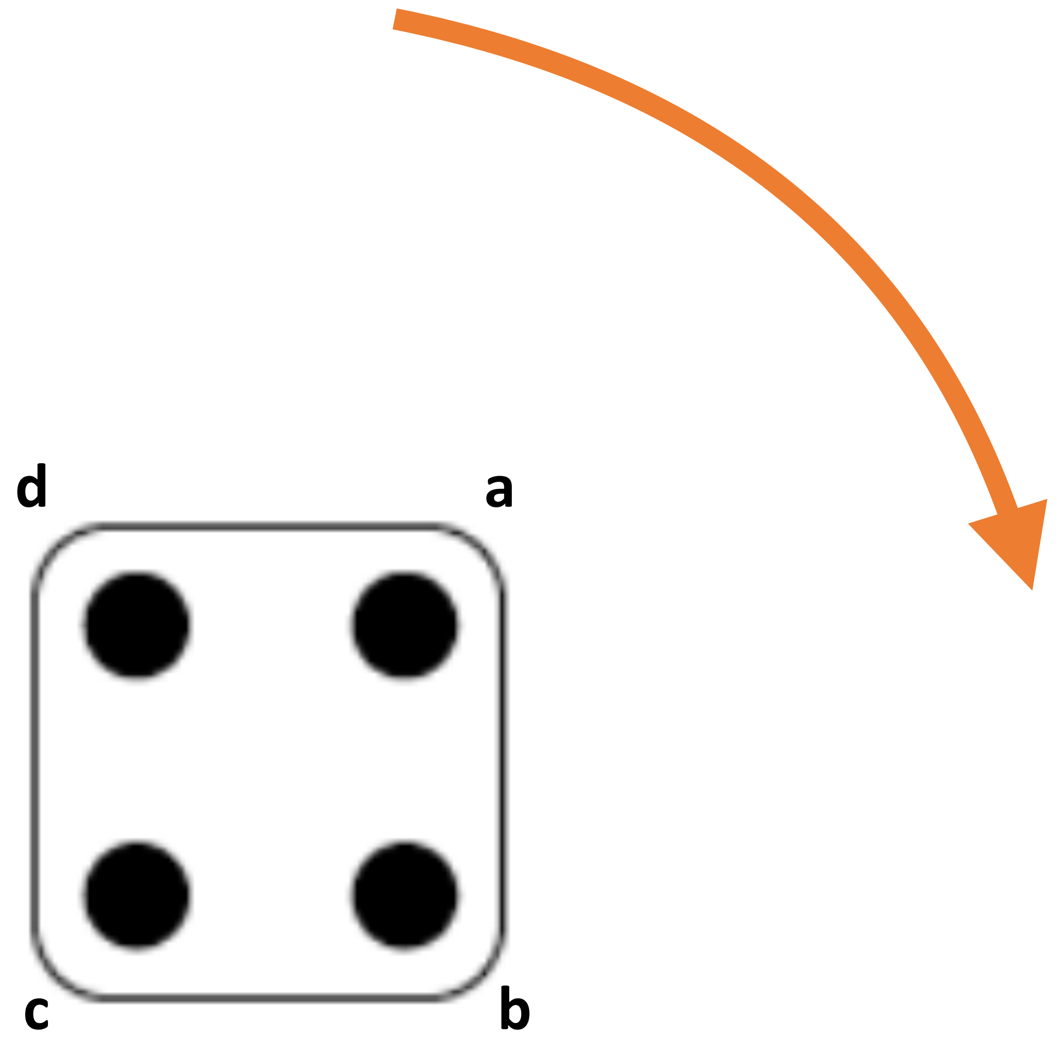
Two possibilities (a,c) and (b,d)

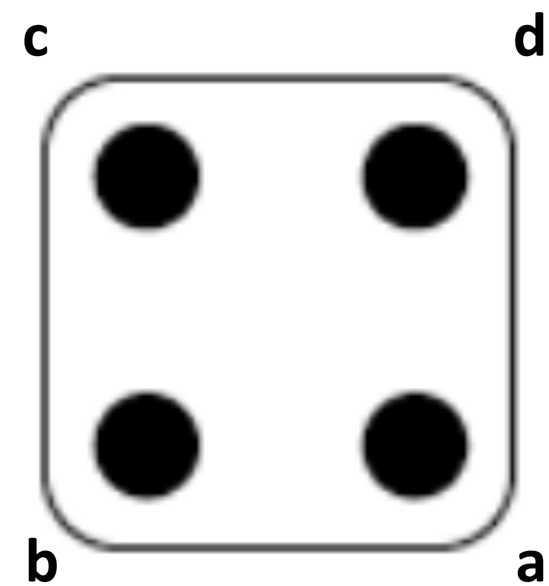
Also two possibilities (a,c) and (b,d)

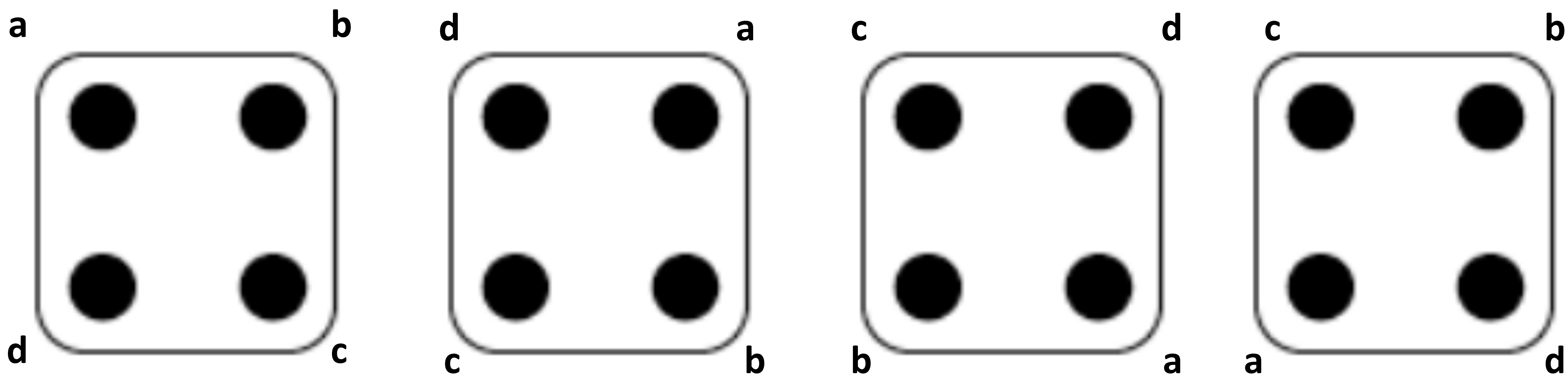

a
b
d
c

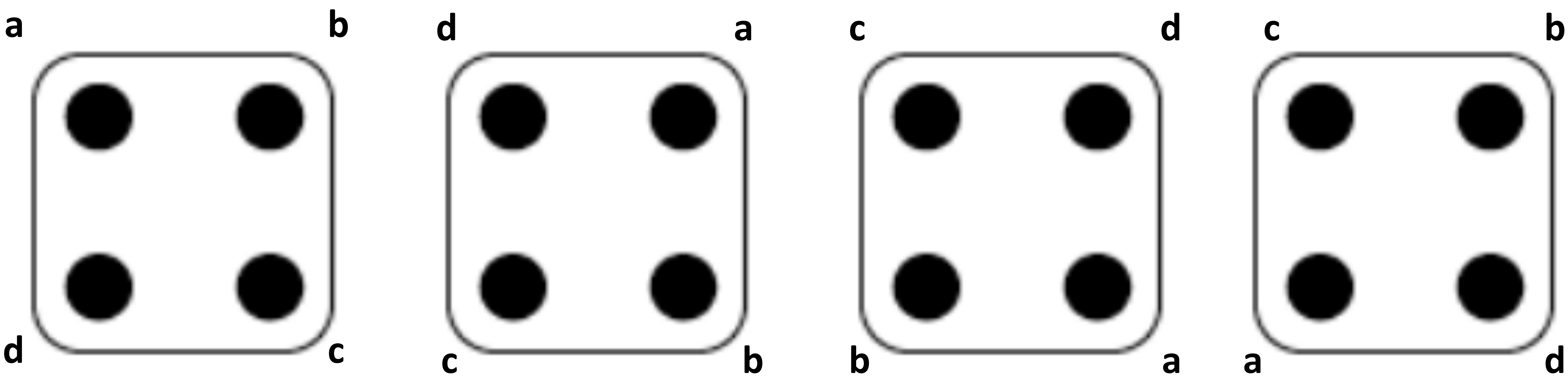

a
b
c
d


d
a
c
b

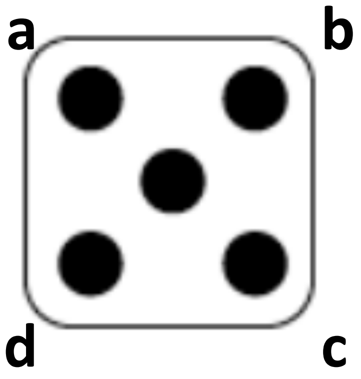

d
a
c
b


c
d
b
a

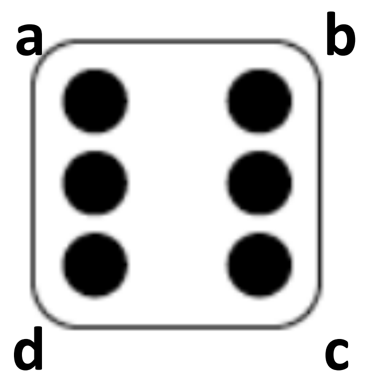

a
b
d
c
d
a
c
b
c
d
b
a
c
b
a
d



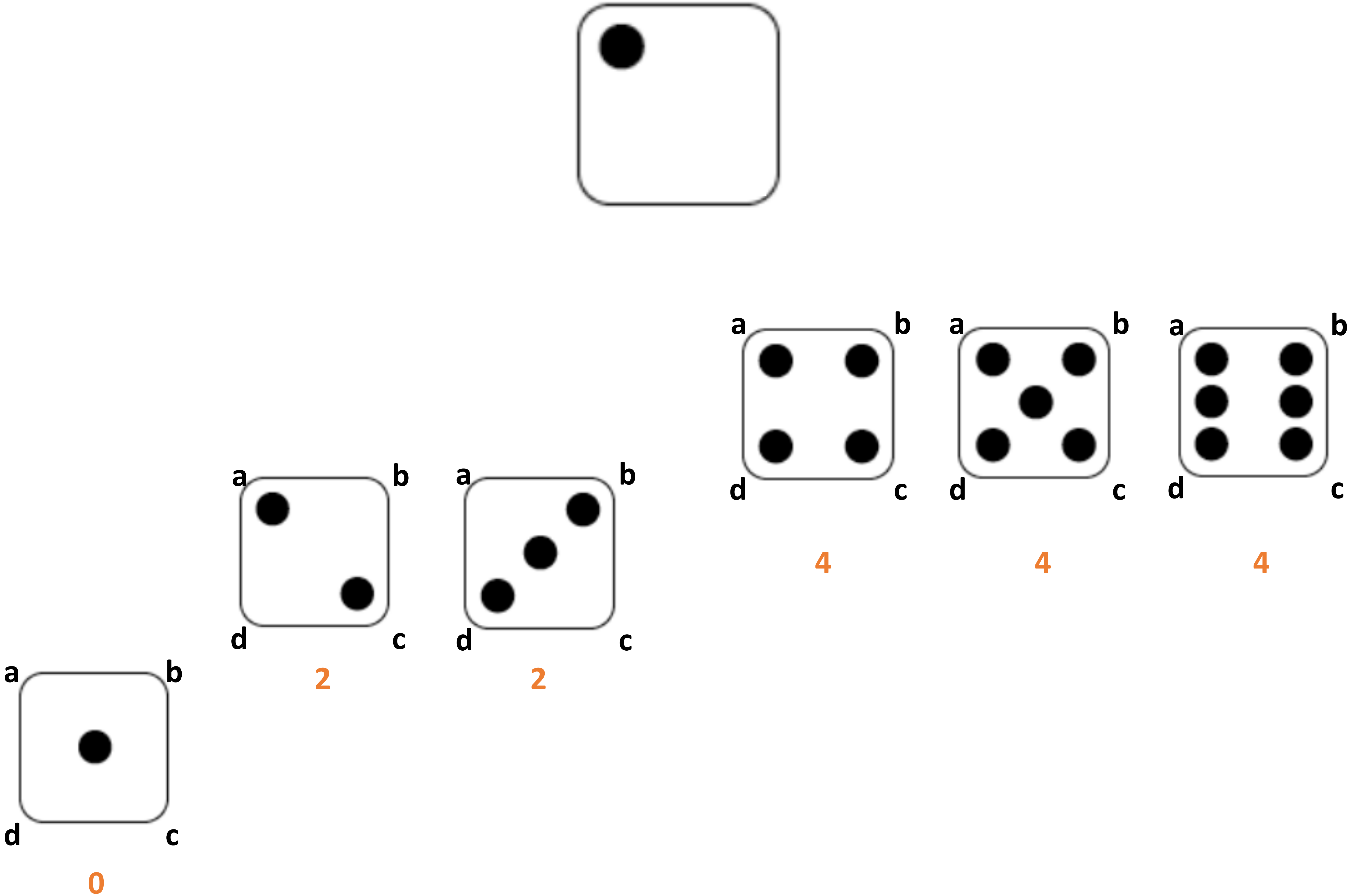
Four possibilities

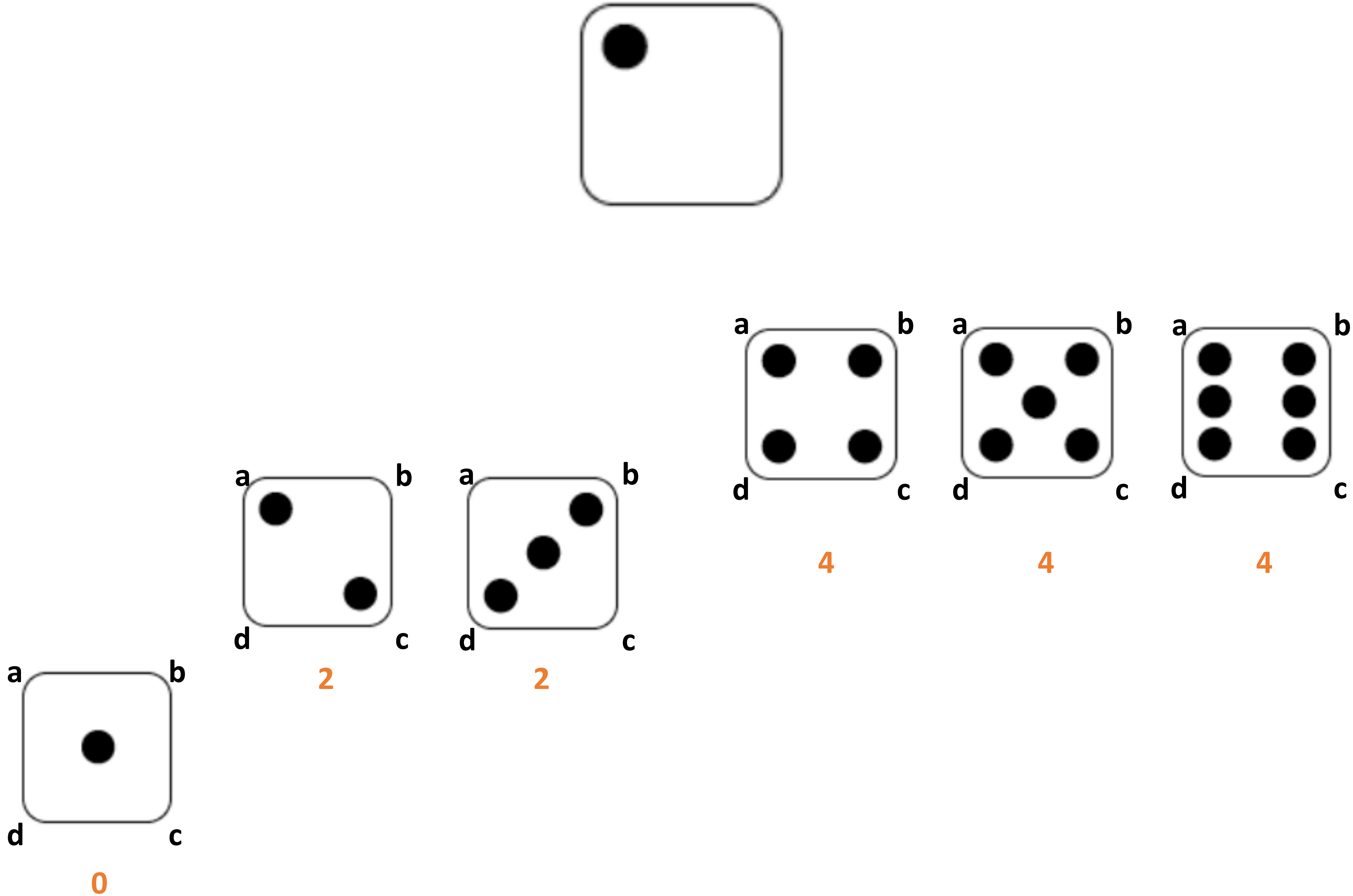


Four possibilities



Four possibilities

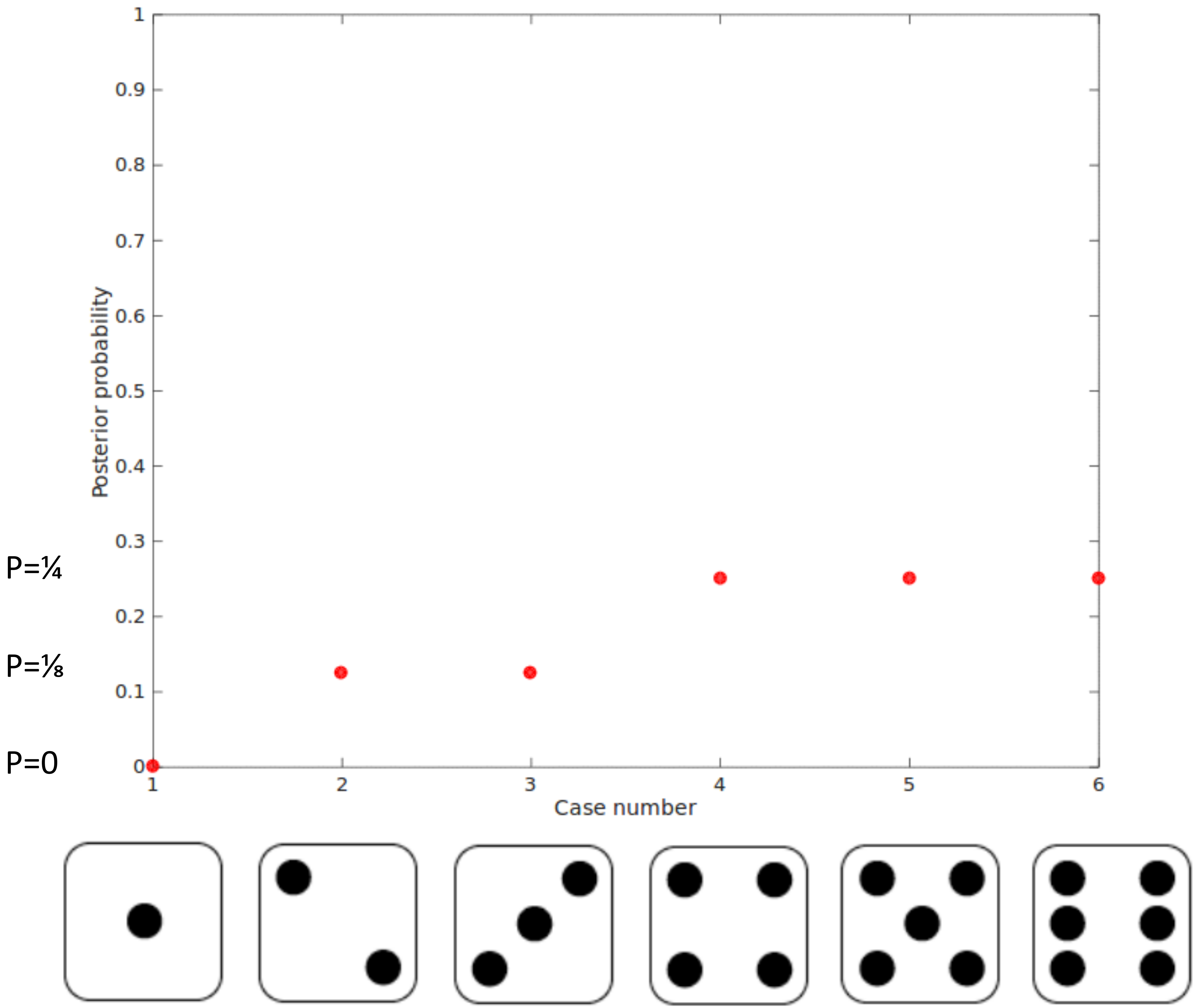

a
b
d
c
0
a
b
d
c
2
a
b
d
c
2
a
b
d
c
4
a
b
d
c
4
a
b
d
c
4



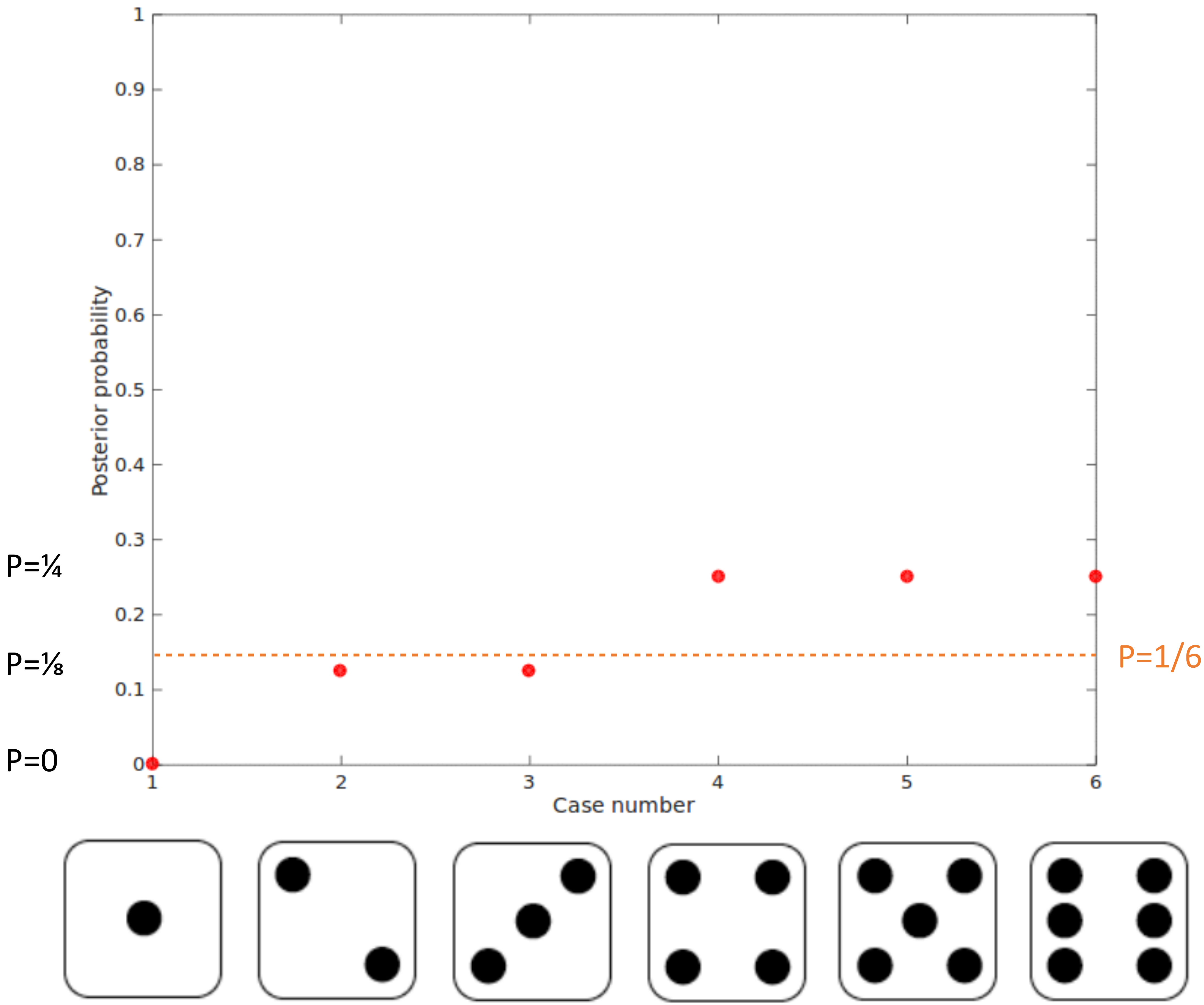
$$\pi(y) = \frac{0+2+2+4+4+4}{6 \times 4} = \frac{16}{24}$$

$$\pi(x|y) = \frac{\text{Prior} \times \text{Likelihood}}{\text{Evidence}} = \frac{\frac{1}{6} \times \frac{1}{2}}{\frac{16}{24}} = 0.125$$

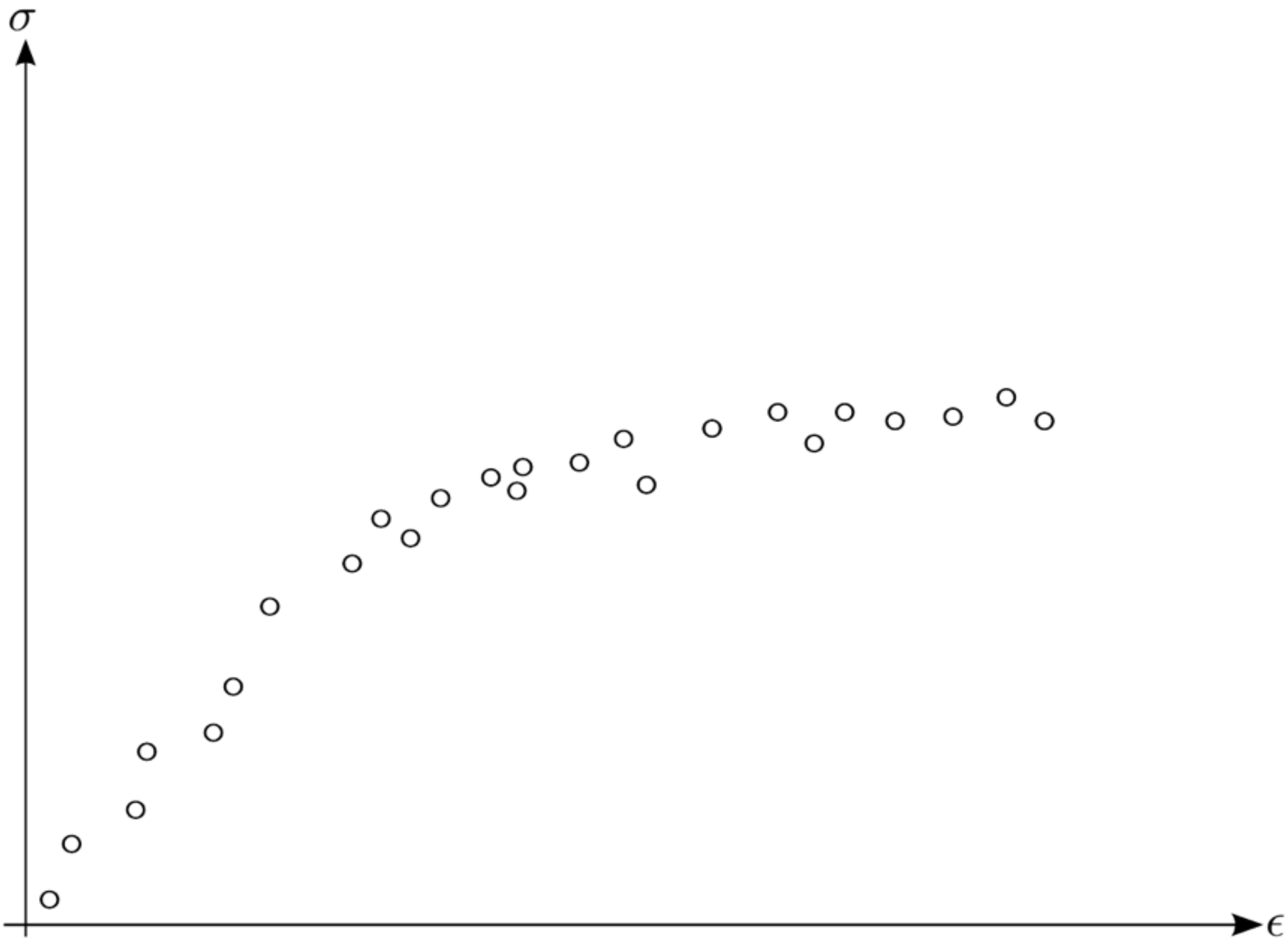
Probability that  **was the face of the dice knowing**



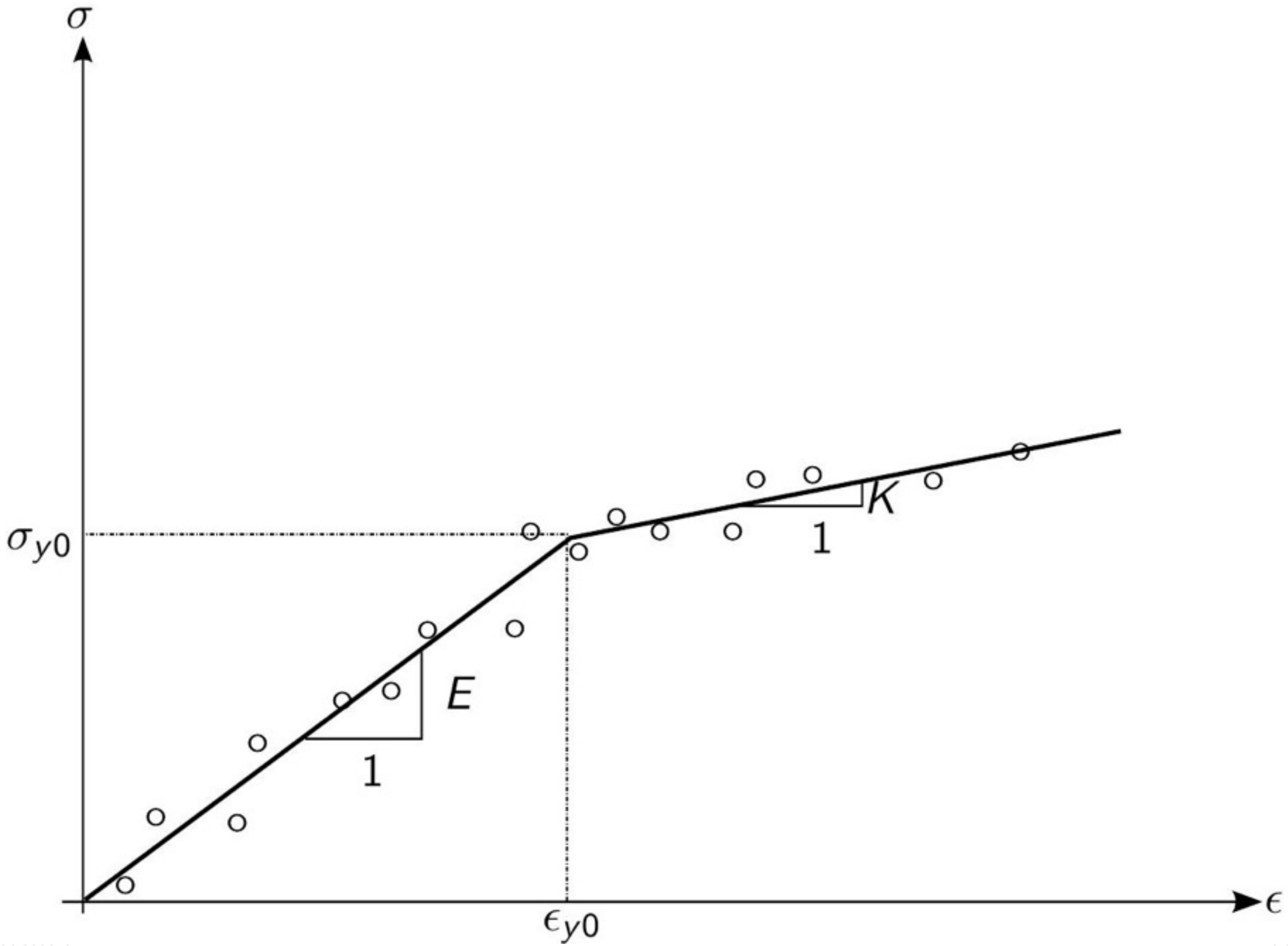
P=¼
P=⅛
P=0
Posterior probability
Case number


P=¼
P=⅛
P=0
P=1/6
Posterior probability
Case number
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1
2
3
4
5
6

Stress-strain data



Identify the parameters



Construct the likelihood function

Model

$Y = f(X, \Omega)$ observations=f(parameters, error)

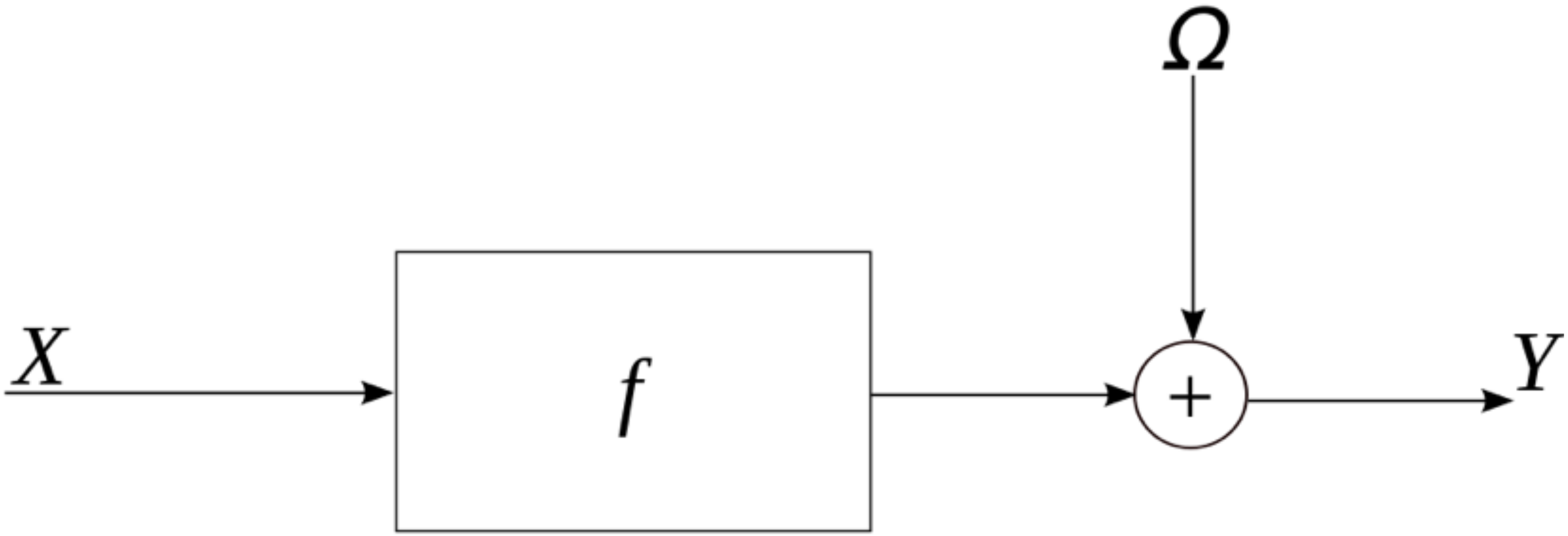
Ω : Error

X : Material parameter

Noise model

Additive noise model

$$Y = f(X) + \Omega$$



Likelihood function

Likelihood function for additive model

$$\pi(y|x) = \pi(\omega) = \pi(y - f(x))$$

$$Y = f(X) + \Omega$$

Constitutive law: linear elasticity

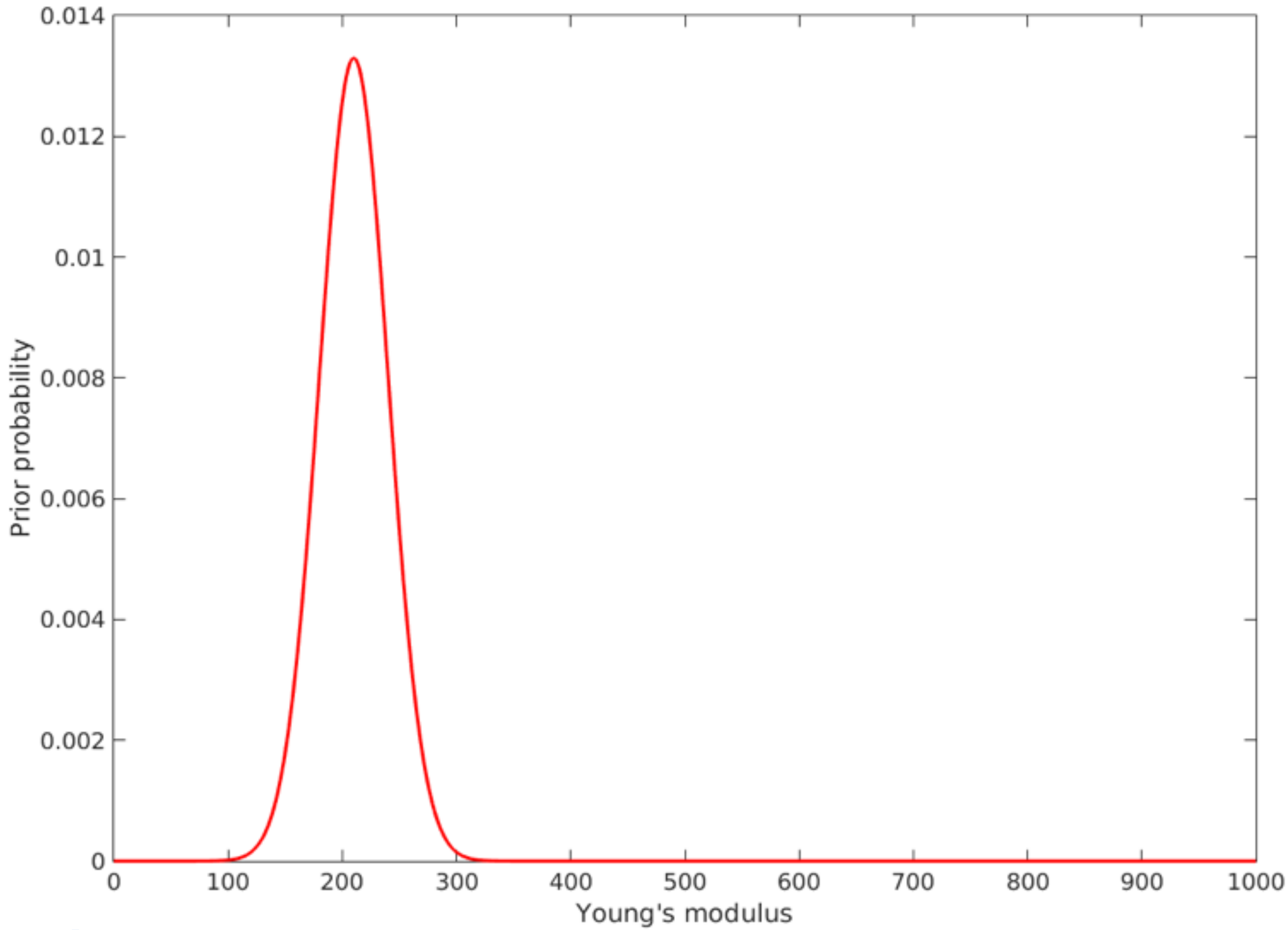
Constitutive model

$$\sigma = E\epsilon \text{ or } \sigma = x\epsilon$$

Observed data

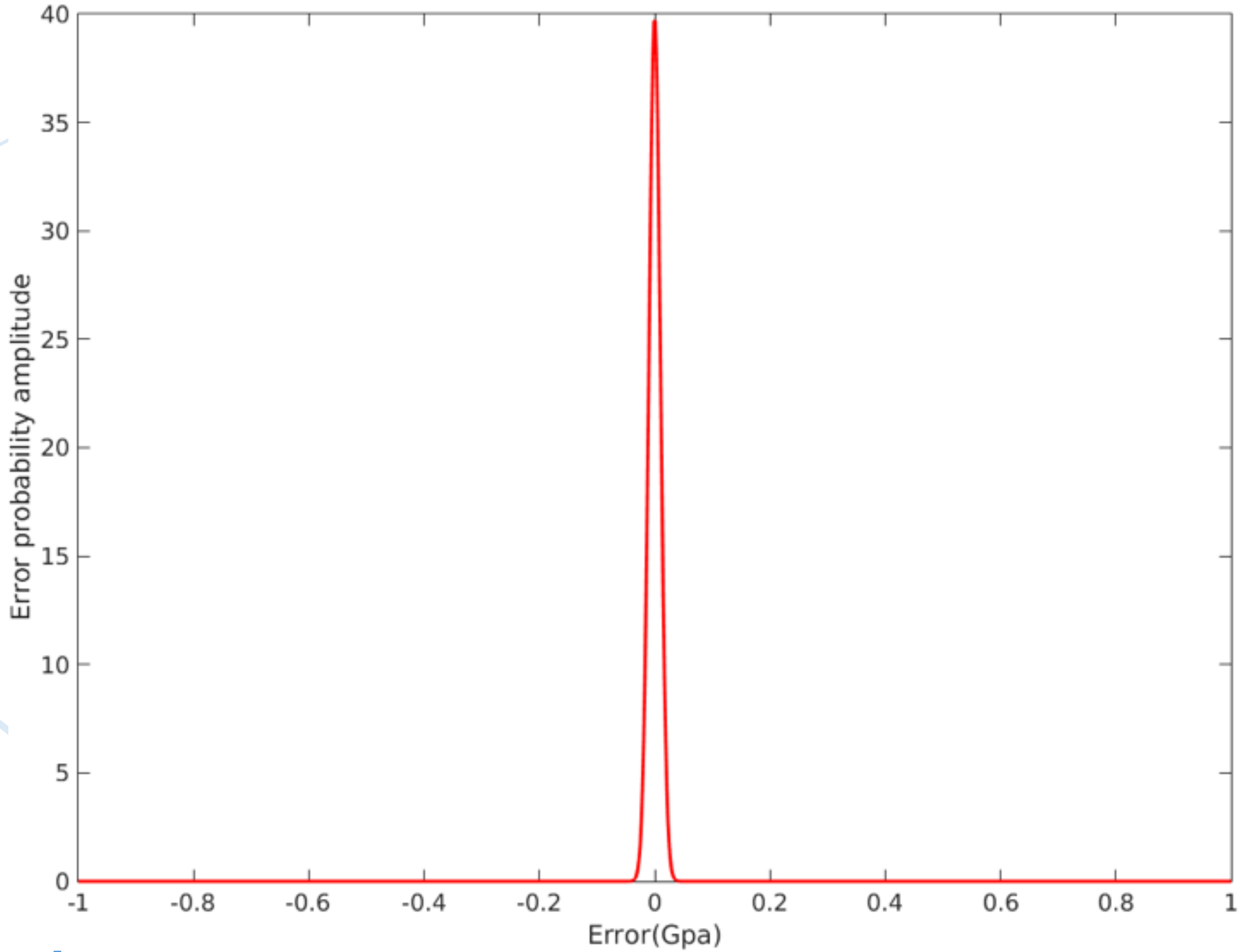
$$Y = X\epsilon + \Omega$$

Prior information on Young's modulus



$$\pi_{prior}(x) = N(210, 900)$$

Error model (noise)



$$\pi(e)_{error} = N(0, 0.0001)$$

Likelihood function

Likelihood function

$$\pi(y|x) = N(y - x\epsilon, 0.0001)$$

$$\pi(y|x) = \pi(\omega) = \pi(y - f(x))$$

Bayes' theorem: calculate the posterior

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

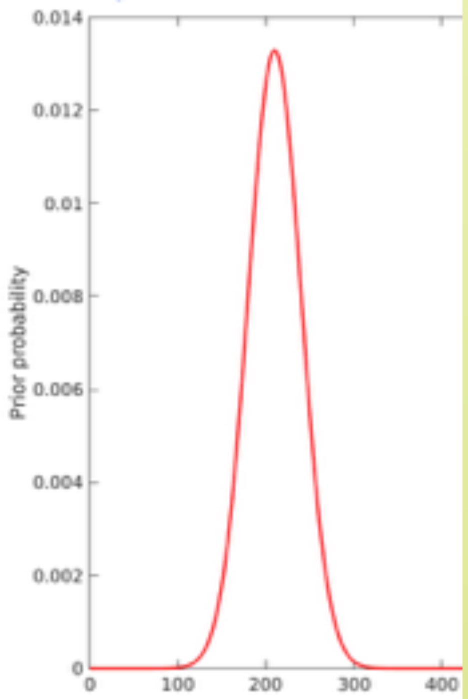
$$\pi(x|y) = \frac{\pi(x)\pi(y|x)}{\int \pi(x)\pi(y|x)dx}$$

Bayes' theorem: calculate the posterior

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

$$\pi(x|y) = \frac{\pi(x)\pi(y|x)}{\int \pi(x)\pi(y|x)dx}$$

prior $\pi(x)$

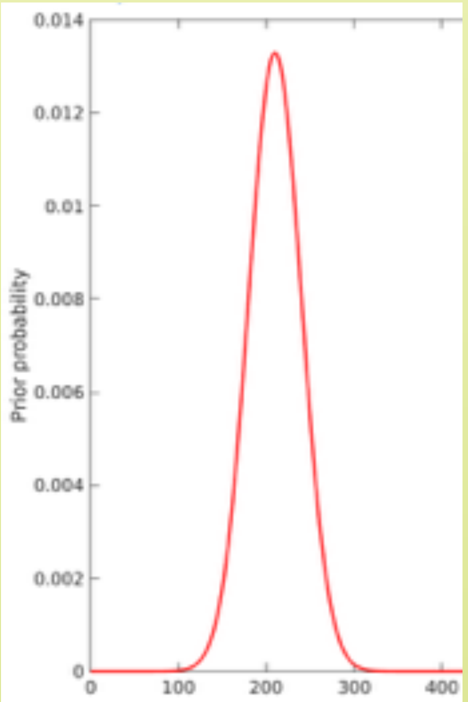


Bayes' theorem: calculate the posterior

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

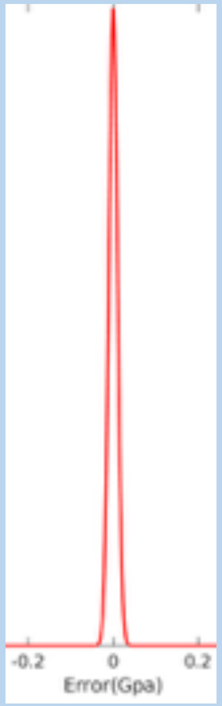
$$\pi(x|y) = \frac{\pi(x)\pi(y|x)}{\int \pi(x)\pi(y|x)dx}$$

prior $\pi(x)$



likelihood $\pi(y|x)$

$$\pi(y|x) = N(y - x\epsilon, 0.0001)$$
$$\pi(y|x) = \pi(\omega) = \pi(y - f(x))$$

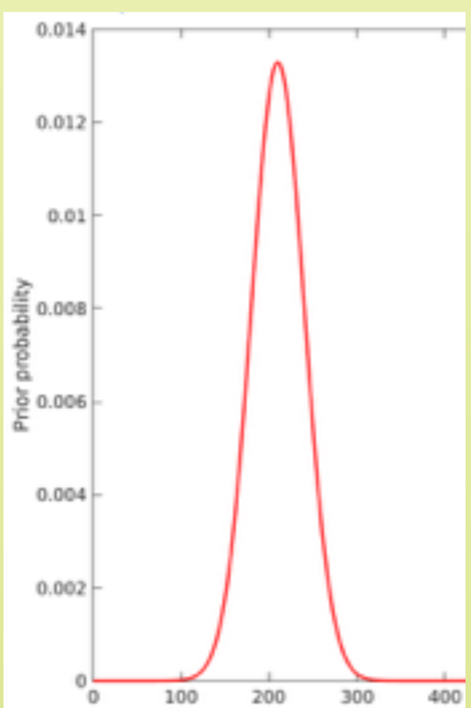


Bayes' theorem: calculate the posterior

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

$$\pi(x|y) = \frac{\pi(x)\pi(y|x)}{\int \pi(x)\pi(y|x)dx}$$

prior $\pi(x)$

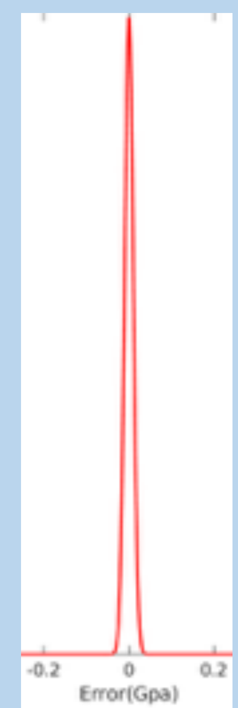


posterior

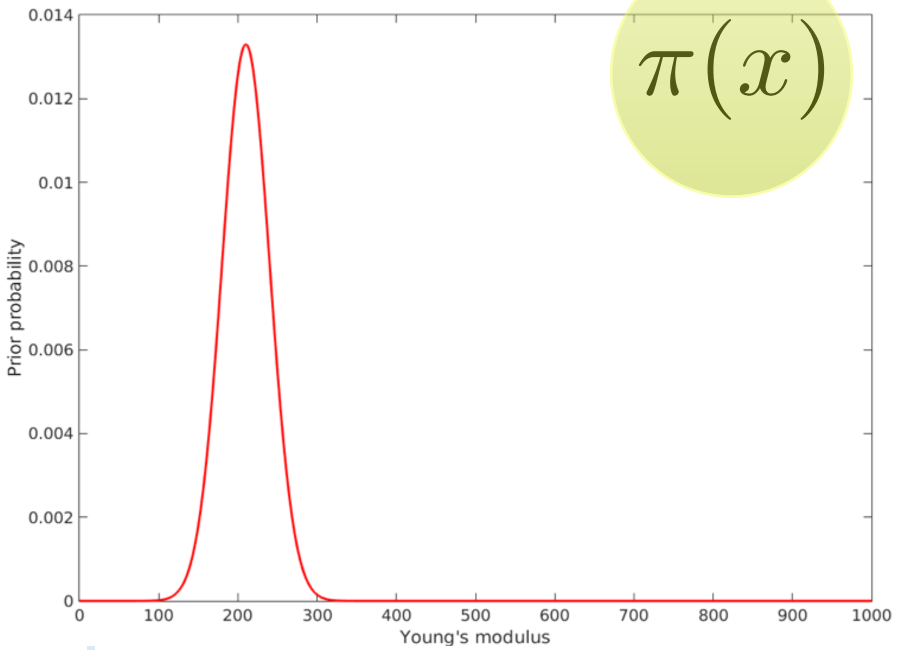
$\pi(x|y)$

likelihood $\pi(y|x)$

$$\pi(y|x) = N(y - x\epsilon, 0.0001)$$
$$\pi(y|x) = \pi(\omega) = \pi(y - f(x))$$

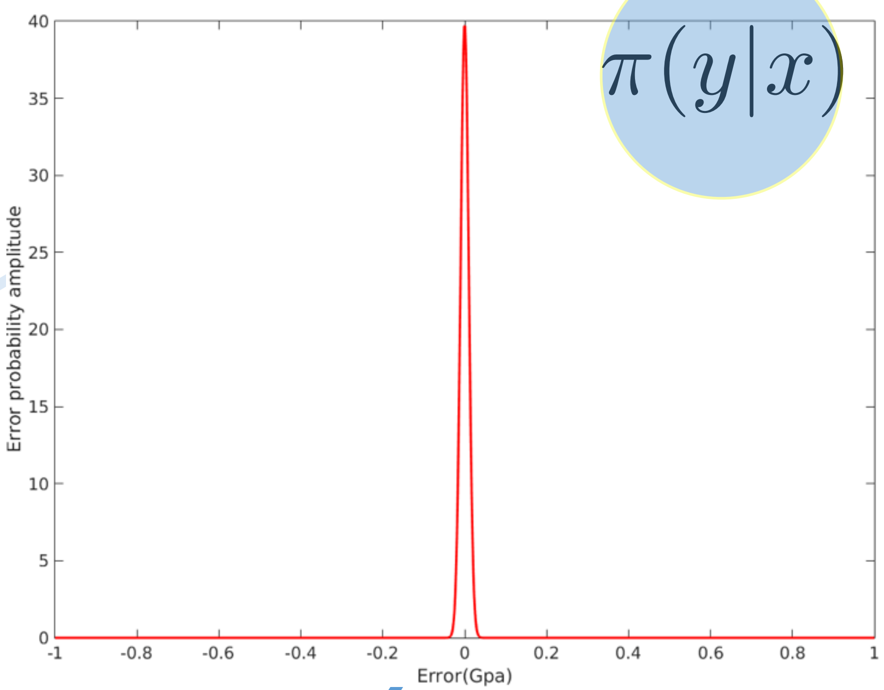


Posterior probability



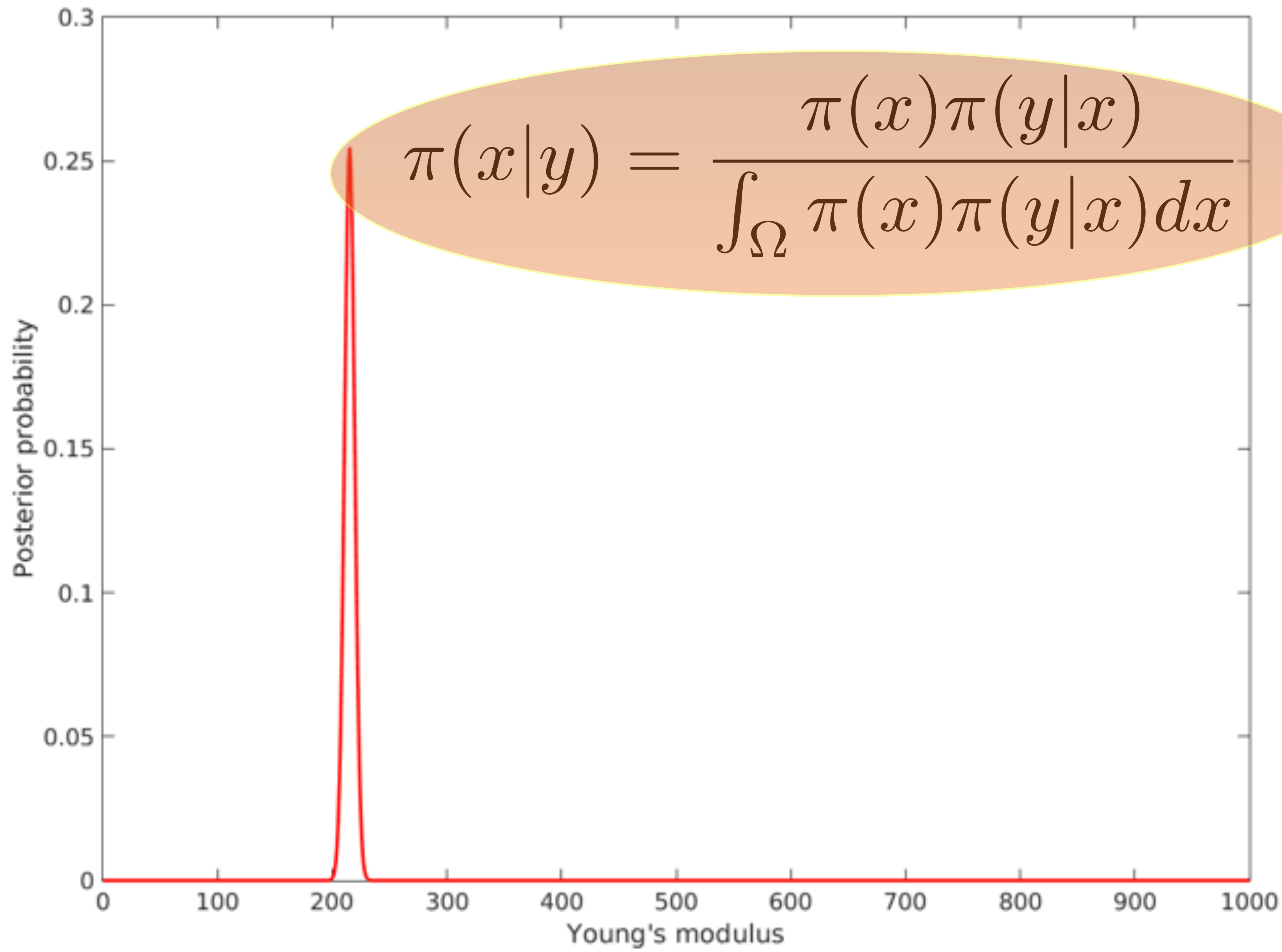
$$\pi(x)$$

$$\pi_{prior}(x) = N(210, 900)$$



$$\pi(y|x)$$

$$\pi(e)_{error} = N(0, 0.0001)$$

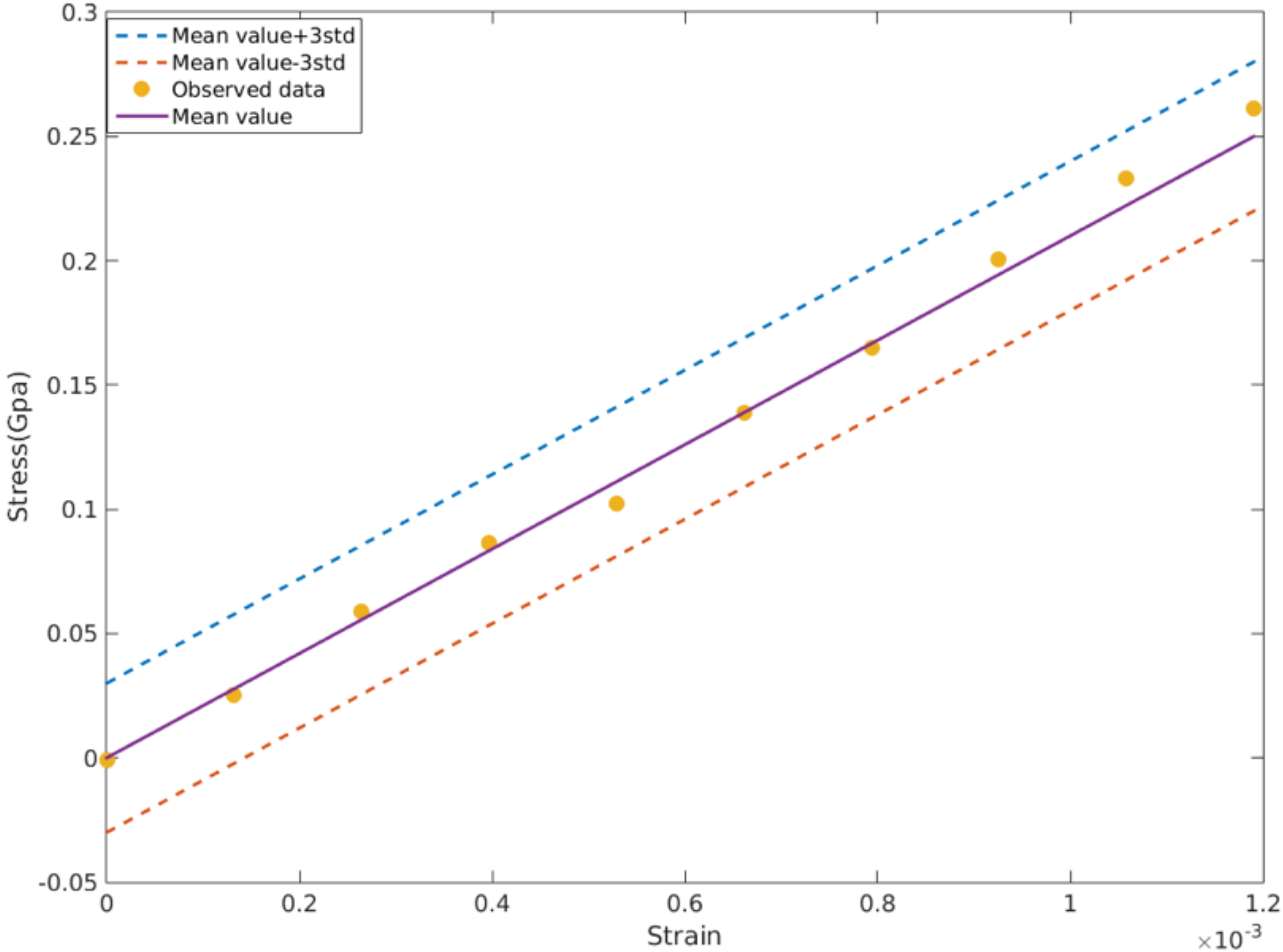


$$\pi(x|y) = \frac{\pi(x)\pi(y|x)}{\int_\Omega \pi(x)\pi(y|x)dx}$$

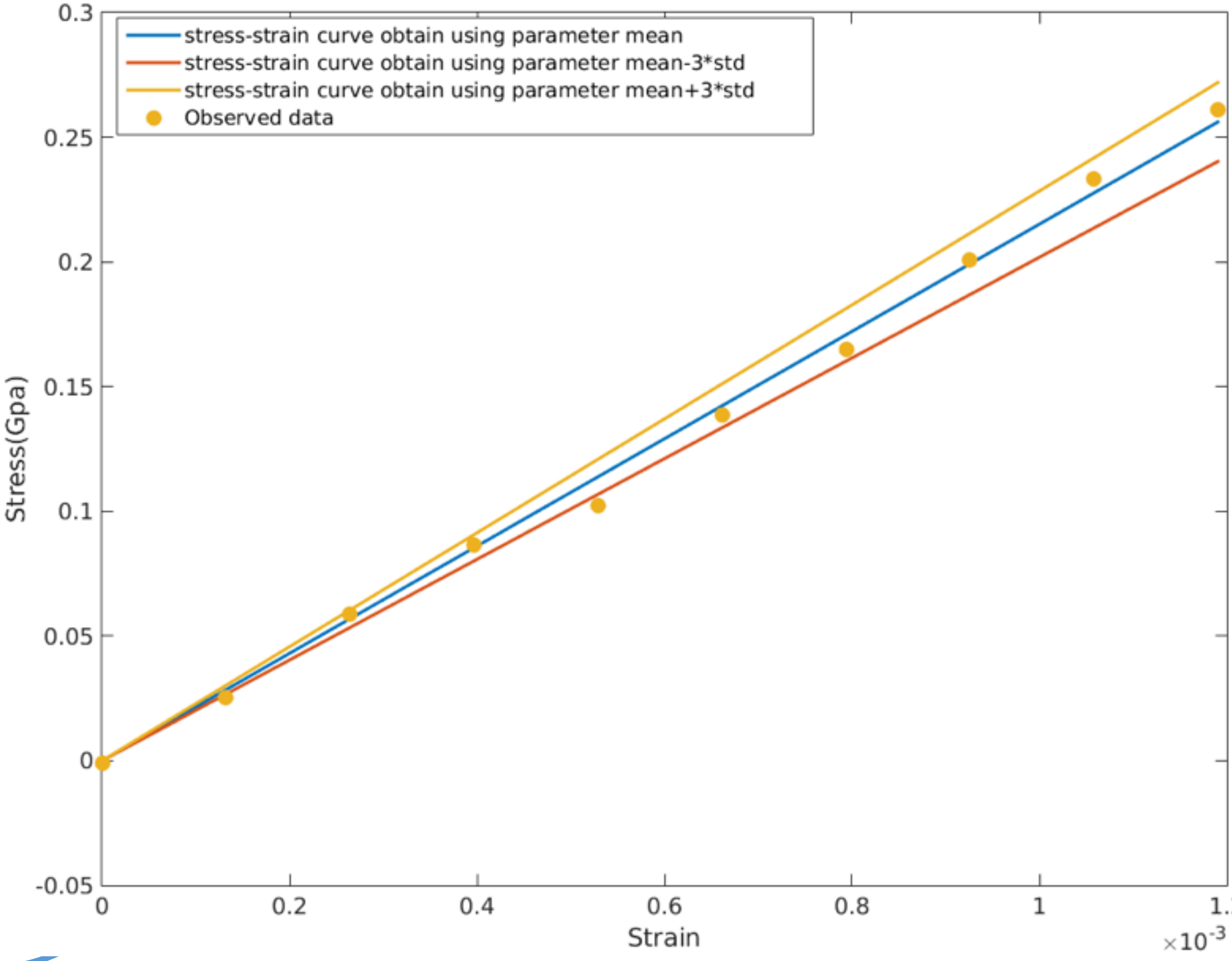
$$\pi_{posterior} = N(215.1533, 19.6168)$$

$$N_{sample} = 10$$

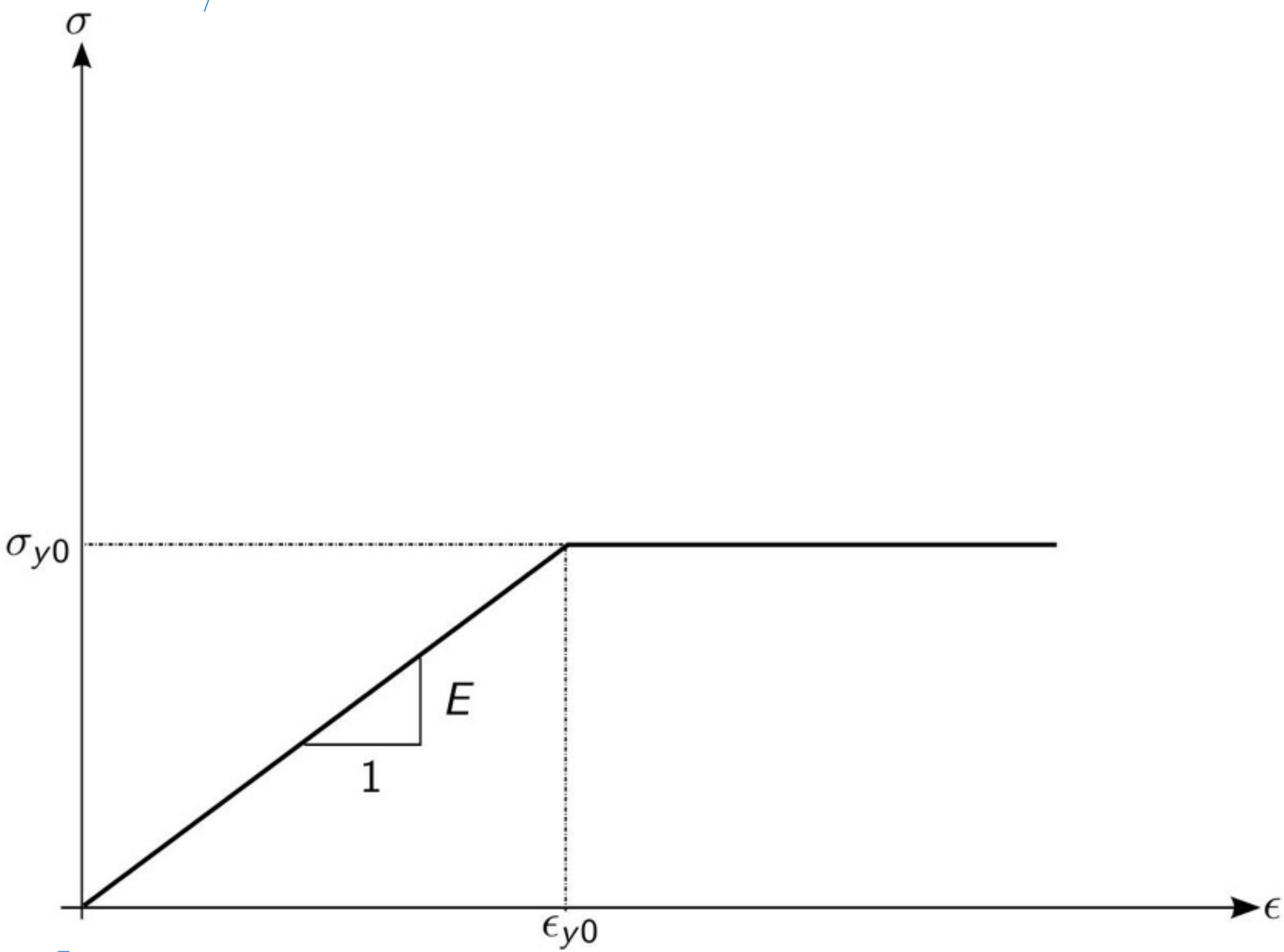
The 99.73% rule: observations



Propagation of the uncertainty to the constitutive model



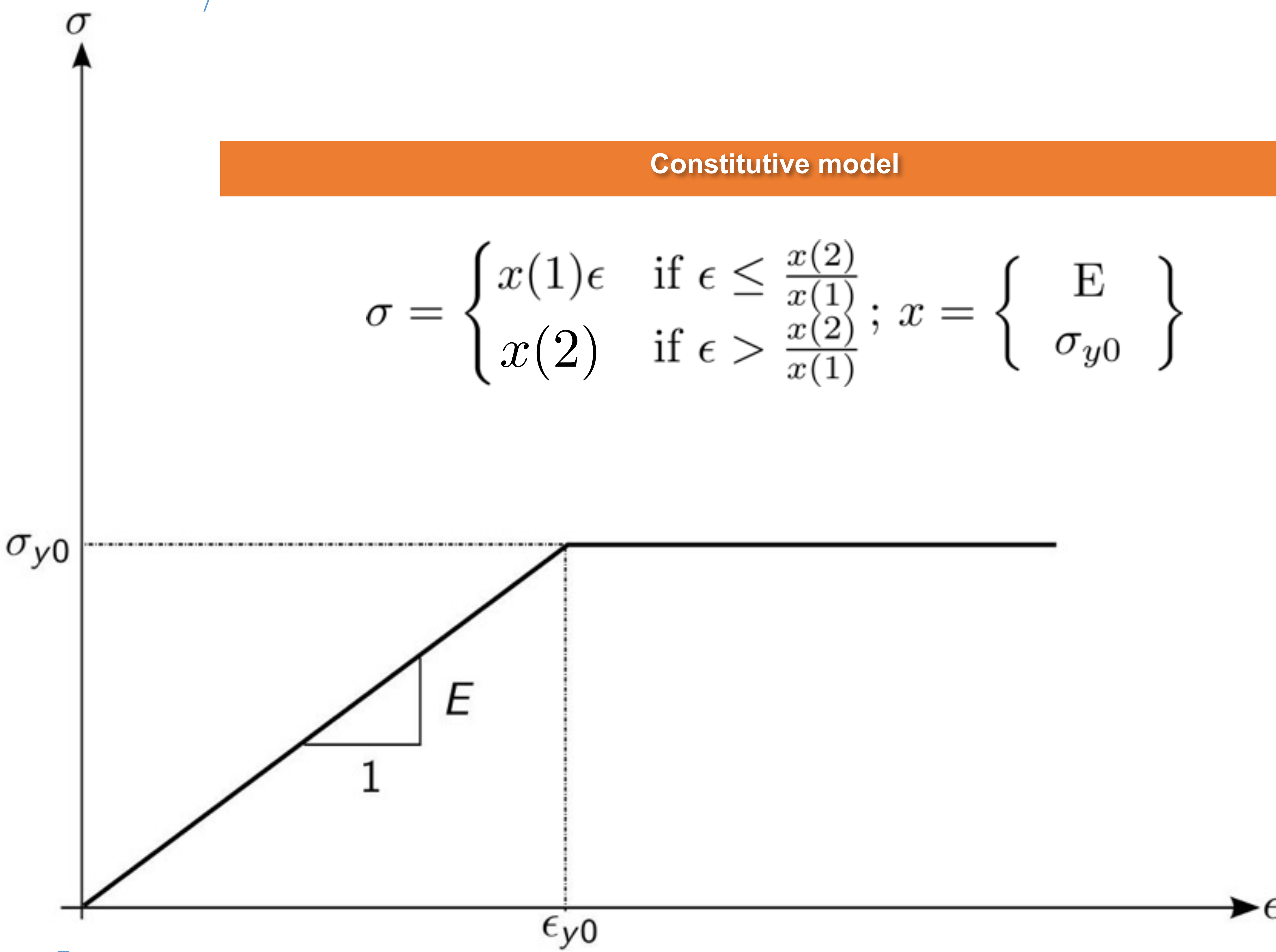
Perfect plasticity



Perfect plasticity

Constitutive model

$$\sigma = \begin{cases} x(1)\epsilon & \text{if } \epsilon \leq \frac{x(2)}{x(1)} \\ x(2) & \text{if } \epsilon > \frac{x(2)}{x(1)} \end{cases} ; \; x = \left\{ \begin{array}{c} \mathrm{E} \\ \sigma_{y0} \end{array} \right\}$$



Perfect plasticity

Modified form for constitutive model

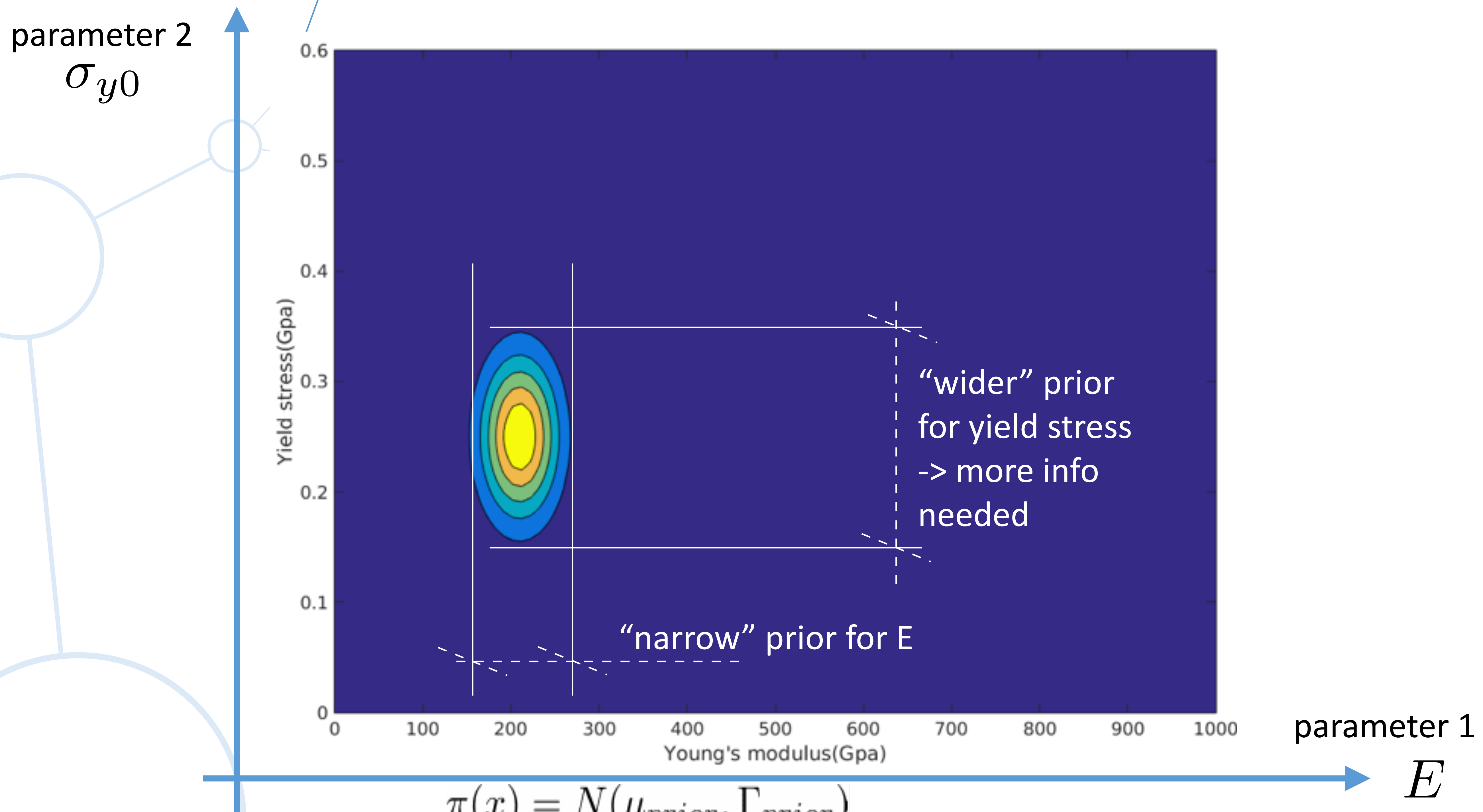
$$\sigma = x(1)\epsilon(1 - h(\sigma - x(2))) + x(2)h(\sigma - x(2))$$

$$h : \text{heaviside function}$$

Observed data

$$Y = \sigma + \Omega$$

Perfect plasticity: contour plot of prior in parameter space



$$\pi(x) = N(\mu_{prior}, \Gamma_{prior})$$

$$\mu_{prior} = \begin{bmatrix} 210 \\ 0.25 \end{bmatrix}; \Gamma_{prior} = \begin{bmatrix} 900 & 0 \\ 0 & 0.0025 \end{bmatrix}$$

Posterior probability

$$\pi(x|y_{N_{obs}}) \propto exp\left(-\frac{1}{2}\left((x-\mu_{prior})^T \Gamma_{prior}^{-1} (x-\mu_{prior}) + \frac{\sum_{i=1}^{N_{obs}} (y_i - F_i)^2}{\sigma_\Omega^2} \right) \right)$$

σ_Ω : Error standard deviation

Posterior probability

$$(x-\mu)^2$$

$$\pi(x|y_{N_{obs}}) \propto exp\left(-\frac{1}{2}\left((x-\mu_{prior})^T\Gamma_{prior}^{-1}(x-\mu_{prior}) + \frac{\sum_{i=1}^{N_{obs}}(y_i - F_i)^2}{\sigma_\Omega^2}\right)\right)$$

σ_Ω : Error standard deviation

$$1/\sigma^2$$

likelihood for each observation

Perfect plasticity

Posterior probability

stress measurement   stress model

$$(x-\mu)^2$$

$$\pi(x|y_{N_{obs}}) \propto exp\left(-\frac{1}{2}\left((x-\mu_{prior})^T\Gamma_{prior}^{-1}(x-\mu_{prior}) + \frac{\sum_{i=1}^{N_{obs}}(y_i - F_i)^2}{\sigma_\Omega^2}\right)\right)$$

σ_Ω : Error standard deviation

$$1/\sigma^2$$

likelihood for each observation

Perfect plasticity

Posterior probability

stress measurement — stress model

$$(x-\mu)^2$$

$$\pi(x|y_{N_{obs}}) \propto exp\left(-\frac{1}{2}\left((x-\mu_{prior})^T\Gamma^{-1}_{prior}(x-\mu_{prior}) + \frac{\sum_{i=1}^{N_{obs}}(y_i - F_i)^2}{\sigma_\Omega^2}\right)\right)$$

σ_Ω : Error standard deviation

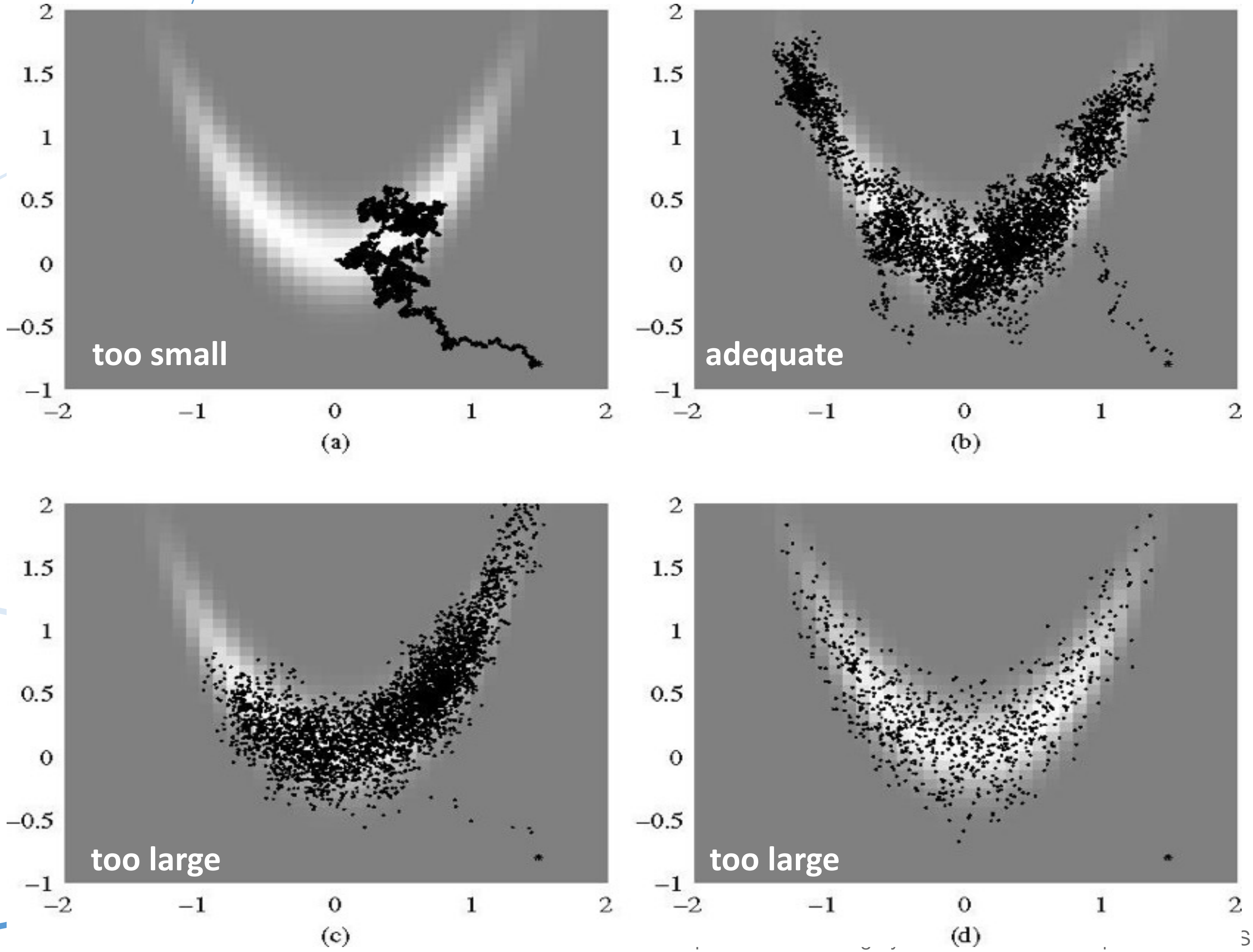
$$1/\sigma^2$$

likelihood for each observation

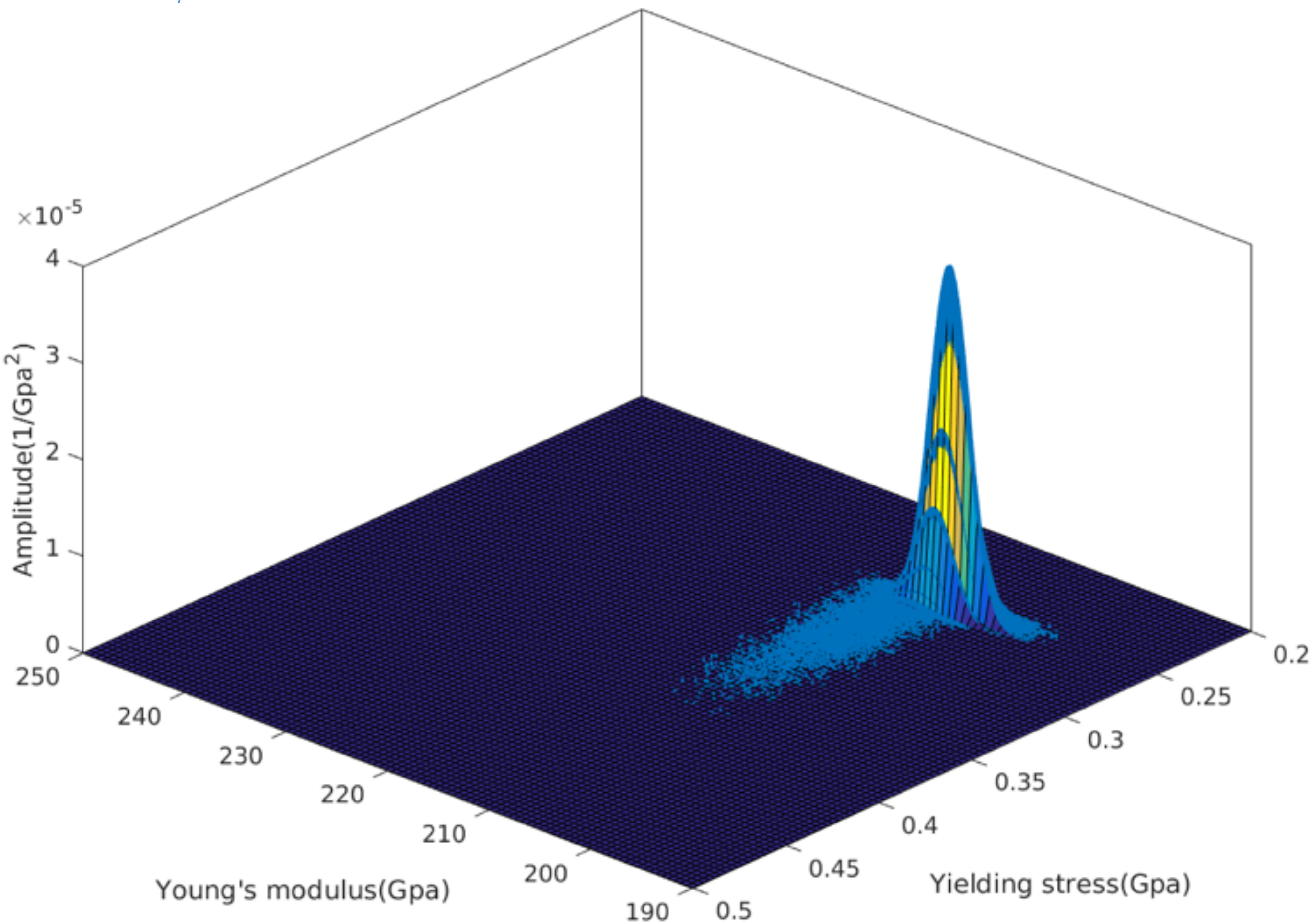
$$f(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Difficult to compute the evidence probability: use MCMC

Markov-Chain Monte Carlo (MCMC) method: parameter space



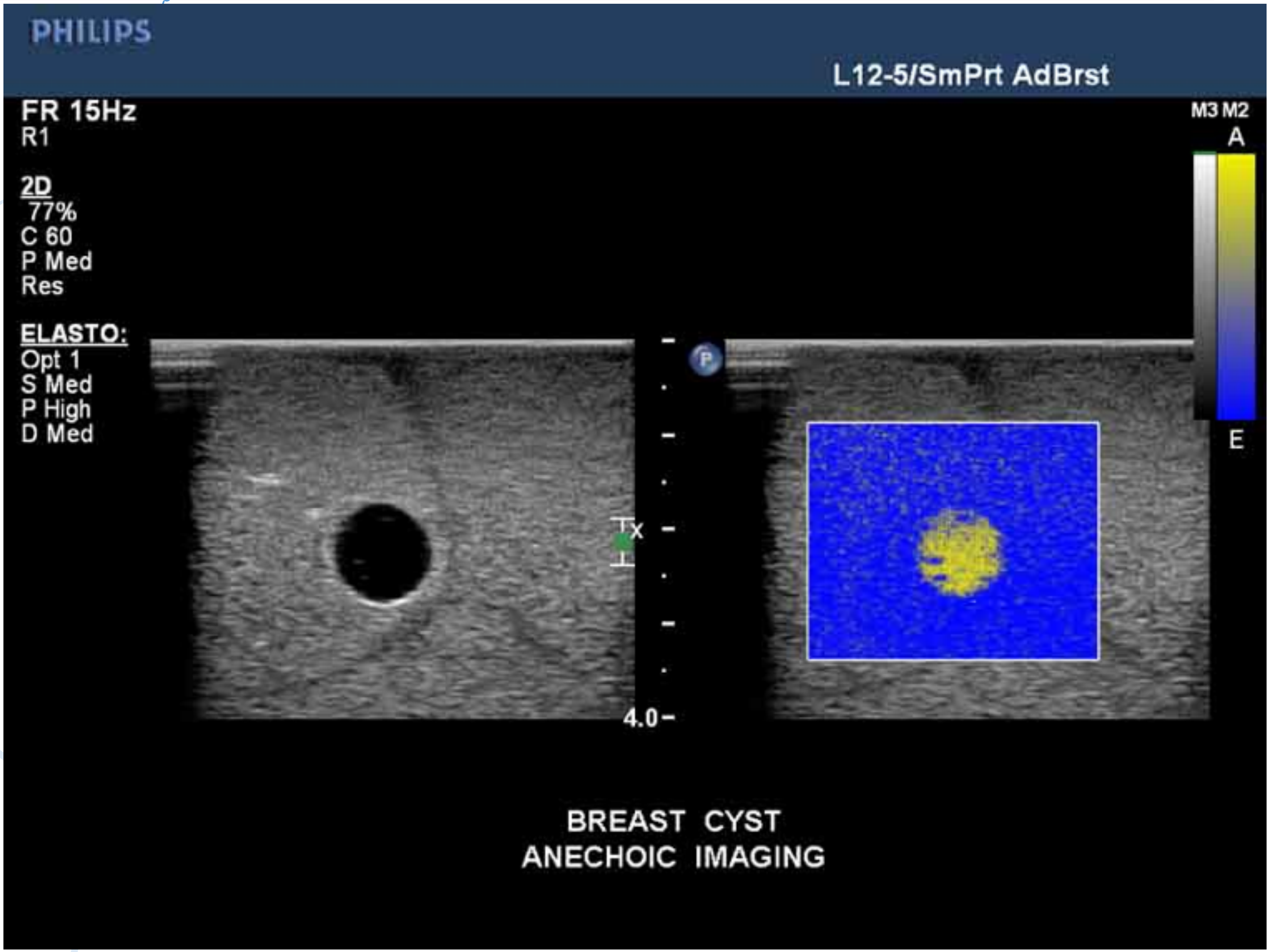
Perfect plasticity: amplitude plot



$$\mu_{posterior} = [208.6690.2603] \; ; \Gamma_{posterior} = \begin{bmatrix} 4.0918 & 0.0044 \\ 0.0044 & 0.0001 \end{bmatrix}$$

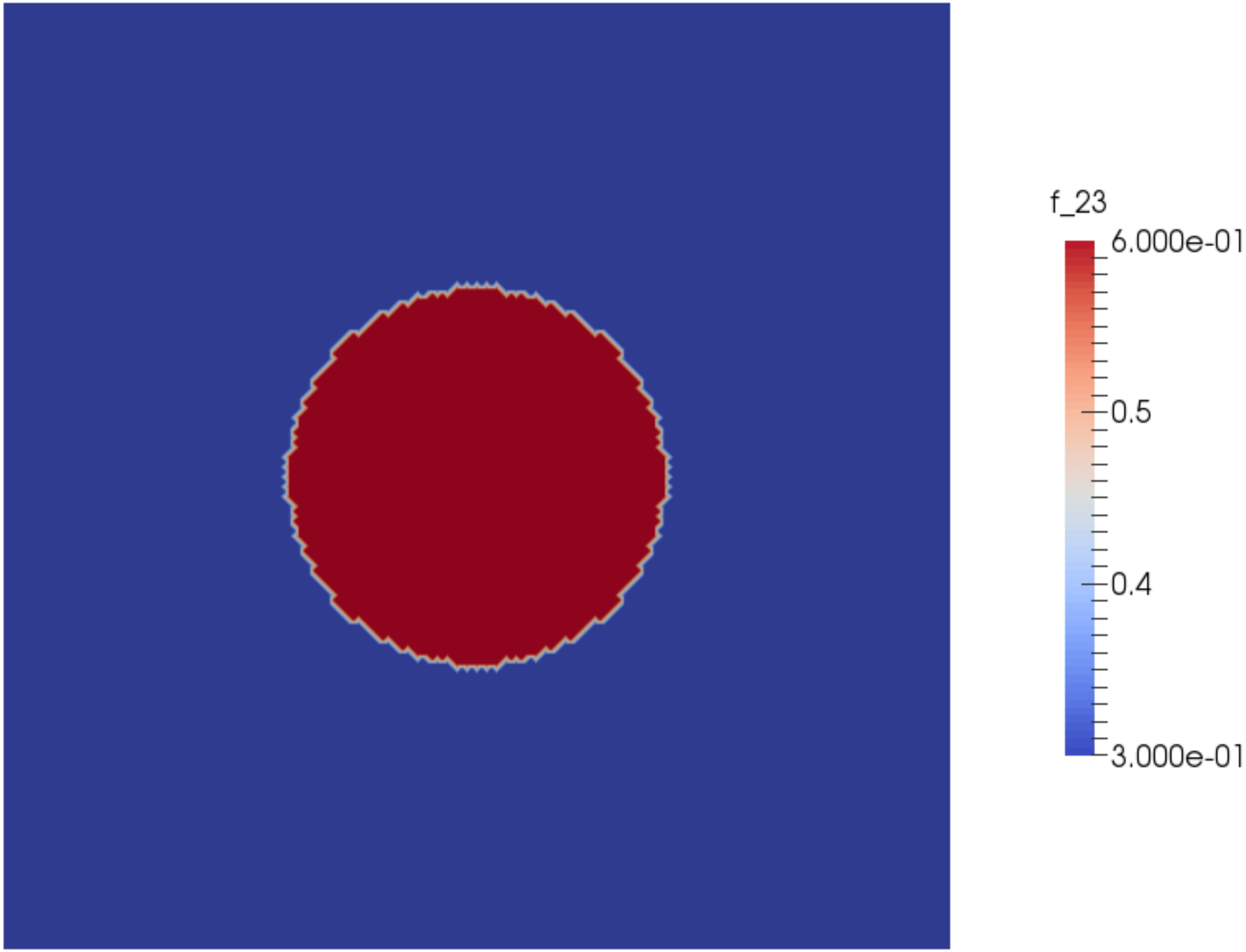
$$N_{obs} = 39$$

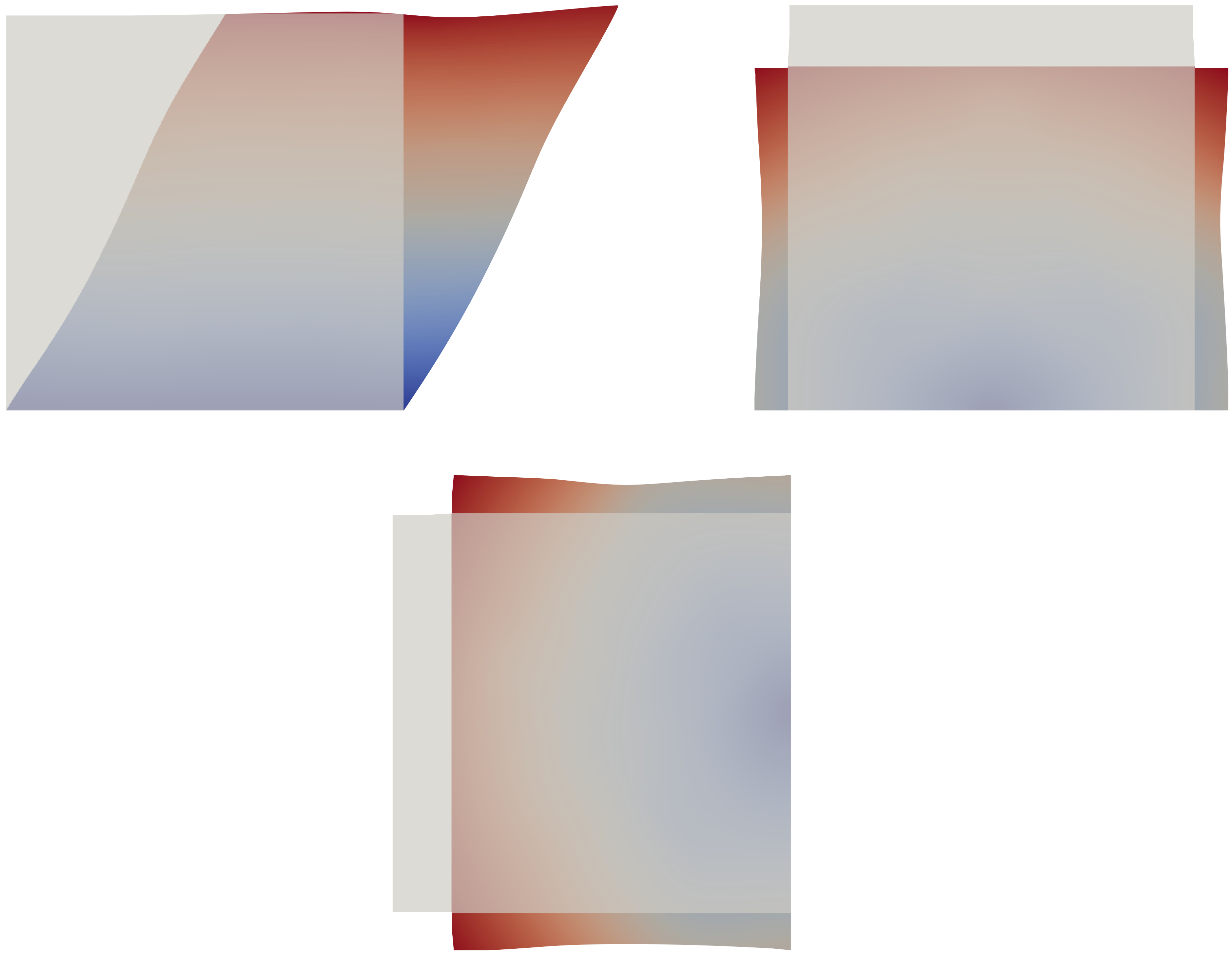
Application to cyst localisation

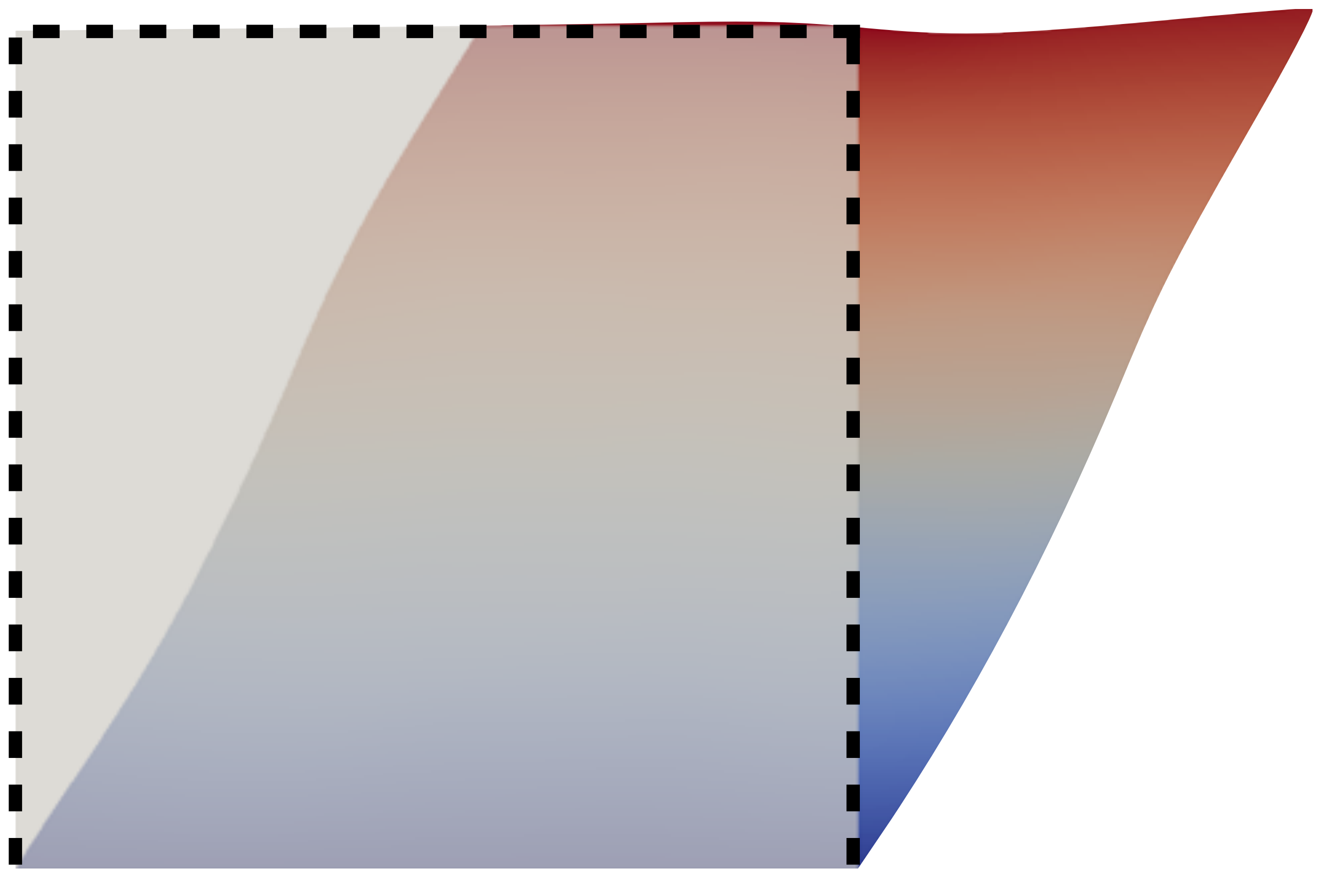


Source: Phillips

Application to cyst localisation

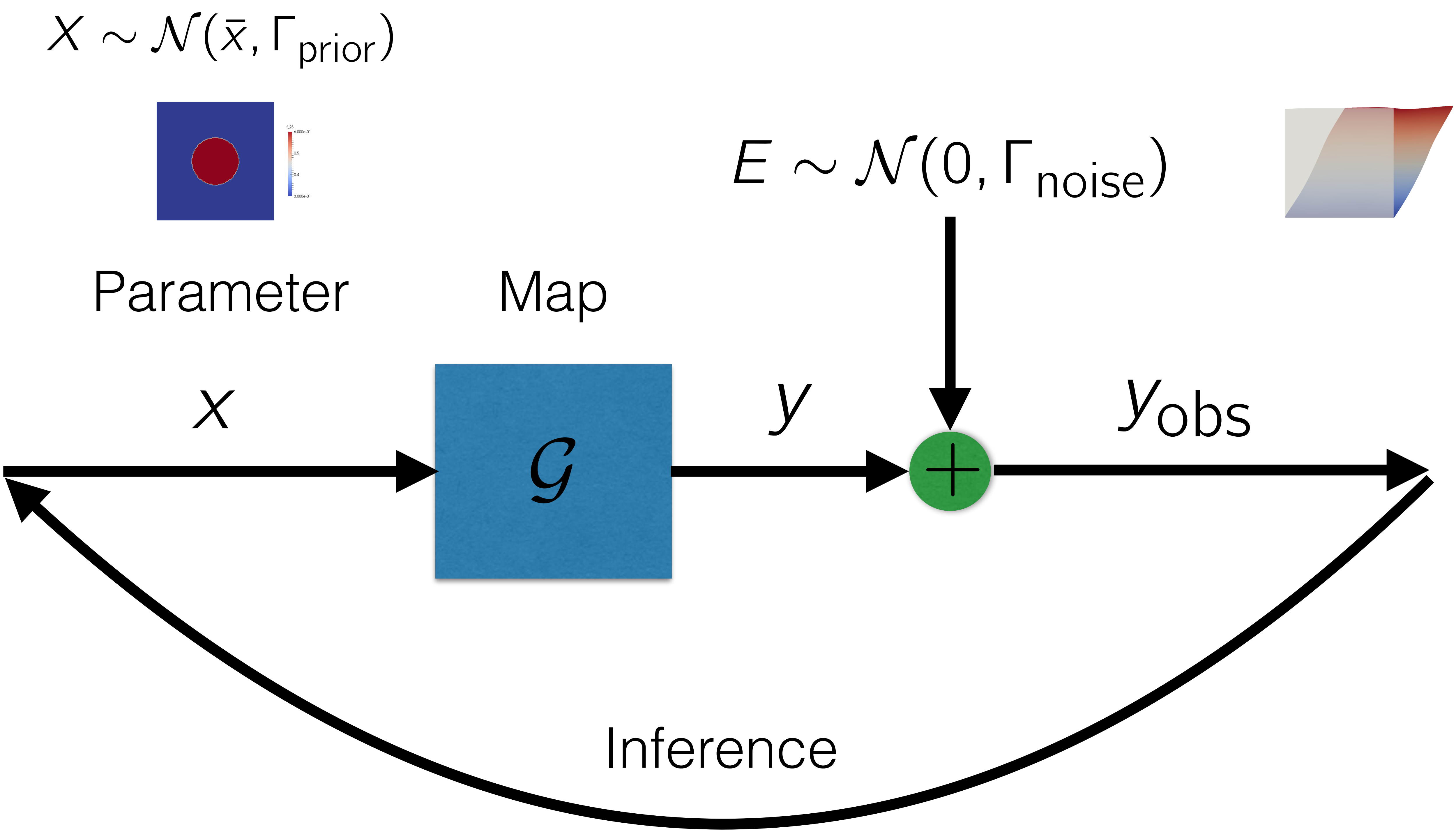






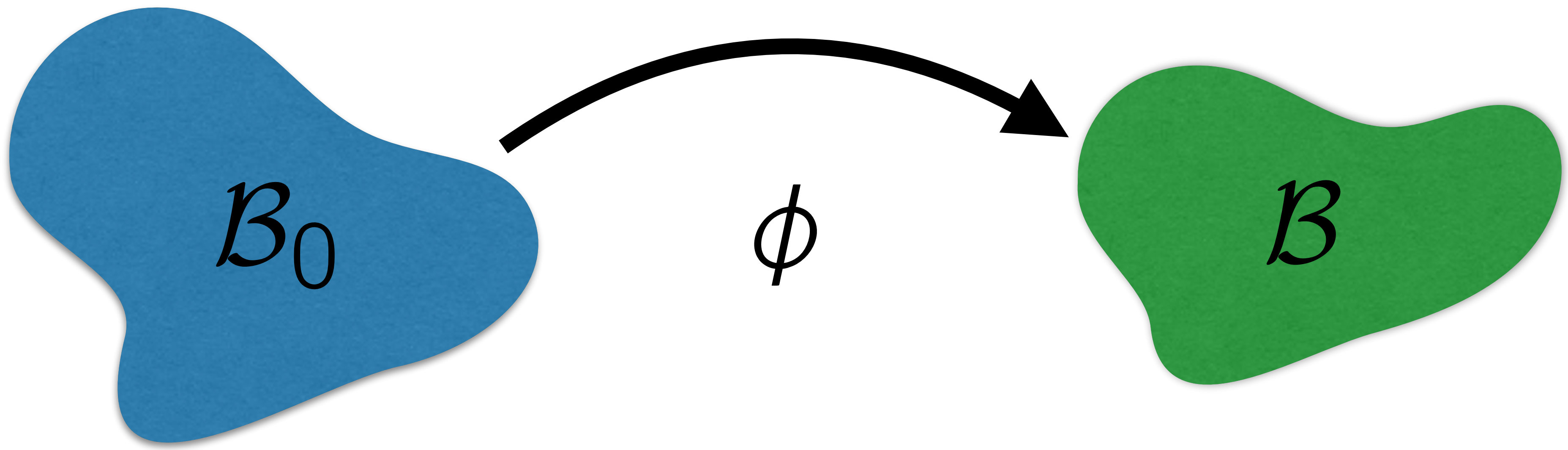
Q: What can we infer about the parameters inside the domain, just from displacement observations on the outside?

Q: Which parameters am I most uncertain about?



$$\pi_{\text{posterior}}(x \mid y) \propto \pi_{\text{likelihood}}(y \mid x)\pi_{\text{prior}}(x)$$

$$\pi_{\text{posterior}}(x|y) \propto \exp\left(-\frac{1}{2}||y - \mathcal{G}(u)||^2_{\Gamma^{-1}_{\text{noise}}} - \frac{1}{2}||x - \bar{x}||_{\Gamma^{-1}_{\text{prior}}}\right)$$



The displacements y for a given material parameter x are defined by a the minimum point of the following Lagrangian:

$$\mathcal{L}(y, x) = \int_\Omega \psi(y, x)\, dx - \int_\Gamma t \cdot y\, ds$$

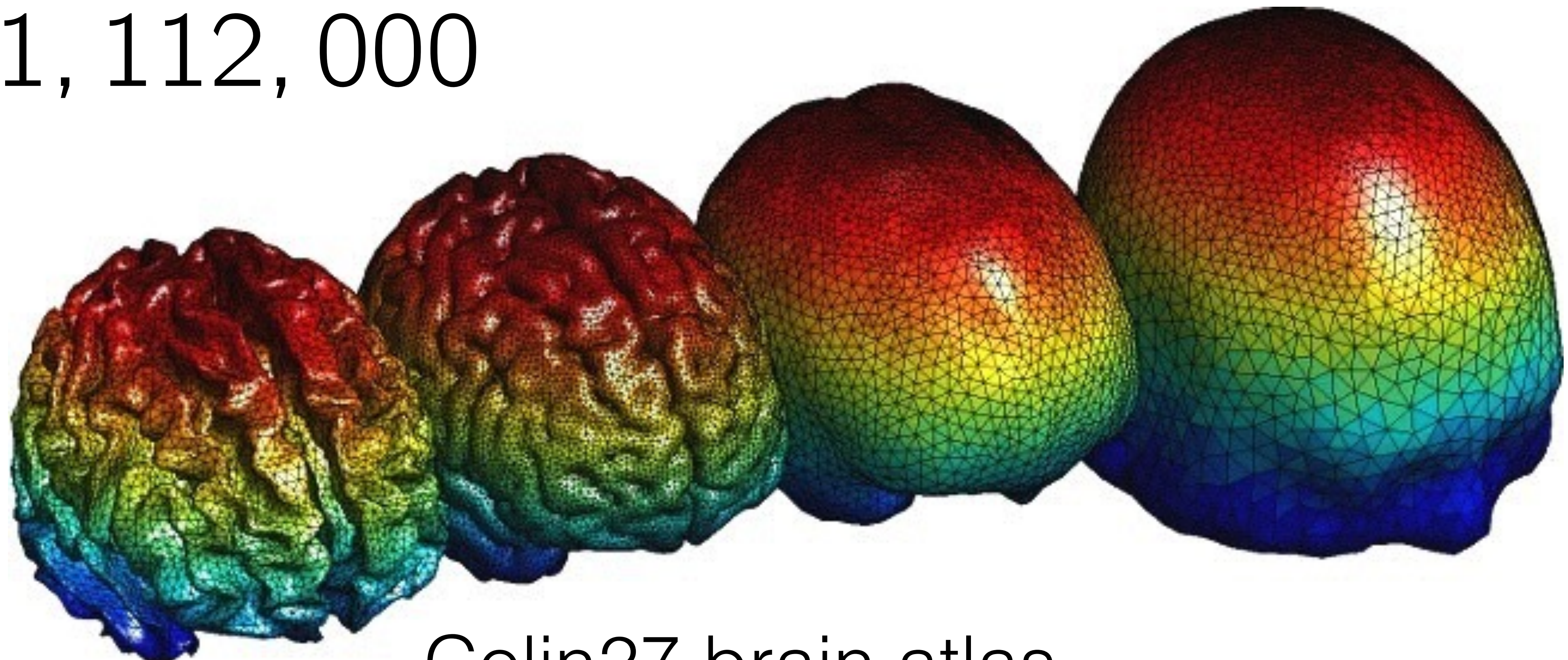
where the energy density functional ψ is defined through the following equations:

$$\begin{aligned}
\psi(u, x) &= \frac{x}{2}(I_c - d) - x \ln(J) + \frac{\lambda}{2} \ln(J)^2, \\
\mathbf{F} &= \frac{\partial \phi}{\partial X} = \mathbf{I} + \nabla y, \\
\mathbf{C} &= \mathbf{F}^T \mathbf{F}, \\
\mathbf{I_C} &= \mathrm{tr}(C), \\
J &= \det \mathbf{F}.
\end{aligned}$$

Even once discretised (Finite Element Method)

$$\mathcal{G}_h : \mathbb{R}^n \to \mathbb{R}^m$$

$$n = 1,112,000$$



Colin27 brain atlas

20% extension test, 16 Core Xeon, 1.12 million cells, ~29 secs

on our Luxembourg Cluster

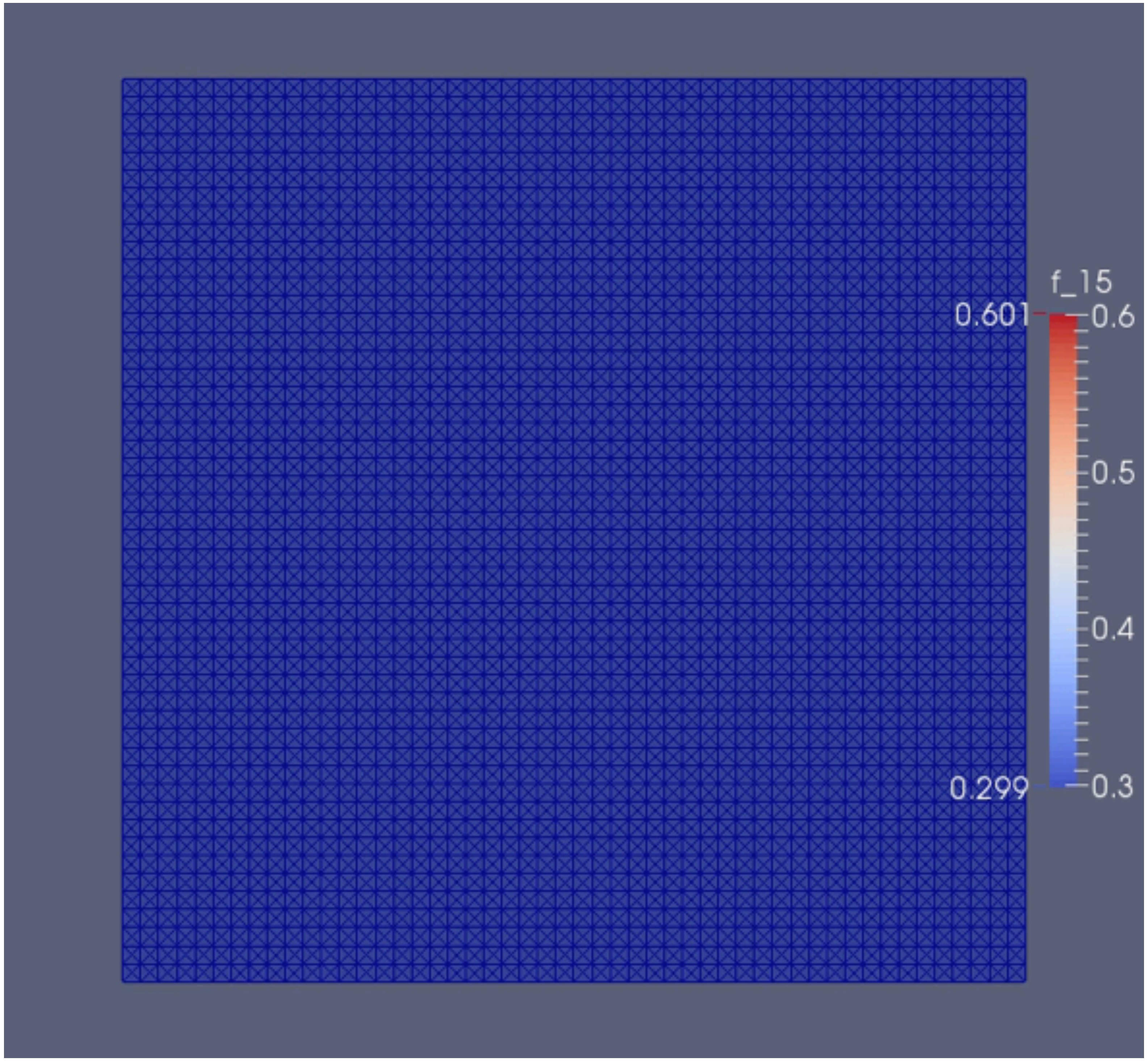
Problems

- Evaluating parameter-to-observable map is *very* expensive.
- Discretised parameter space can be *very* large.
- *Outcome:* Exploring posterior with ‘traditional sampling’ is not going to work.

Solutions

1. Connect Bayesian approach to ideas from classical optimisation. Using derivatives of posterior in parameter-space (Girolami).

2. Exploiting low-rank structure of prior to posterior covariance updates (Flath 2012, Spantini 2015).



MAP estimate. Bound-constrained Quasi-Newton BLMVM with More-Thuente line search and 'correct' Riesz map.

Tools

- The FEniCS Project is collection of free software for the automated, efficient solution of differential equations using the finite element method.

- dolfin-adjoint automatically derives the discrete adjoint, tangent linear and higher-order adjoint models from a high-level description of the forward model.

FEniCS Project

http://fenicsproject.org

Wells, Logg, Rognes, Kirby and many, many others…

dolfin-adjoint

http://www.dolfin-adjoint.org

Farrell, Funke, Ham and Rognes.
2015 Wilkinson Prize for Numerical Software.

The displacements y for a given material parameter x are defined by a the minimum point of the following Lagrangian:

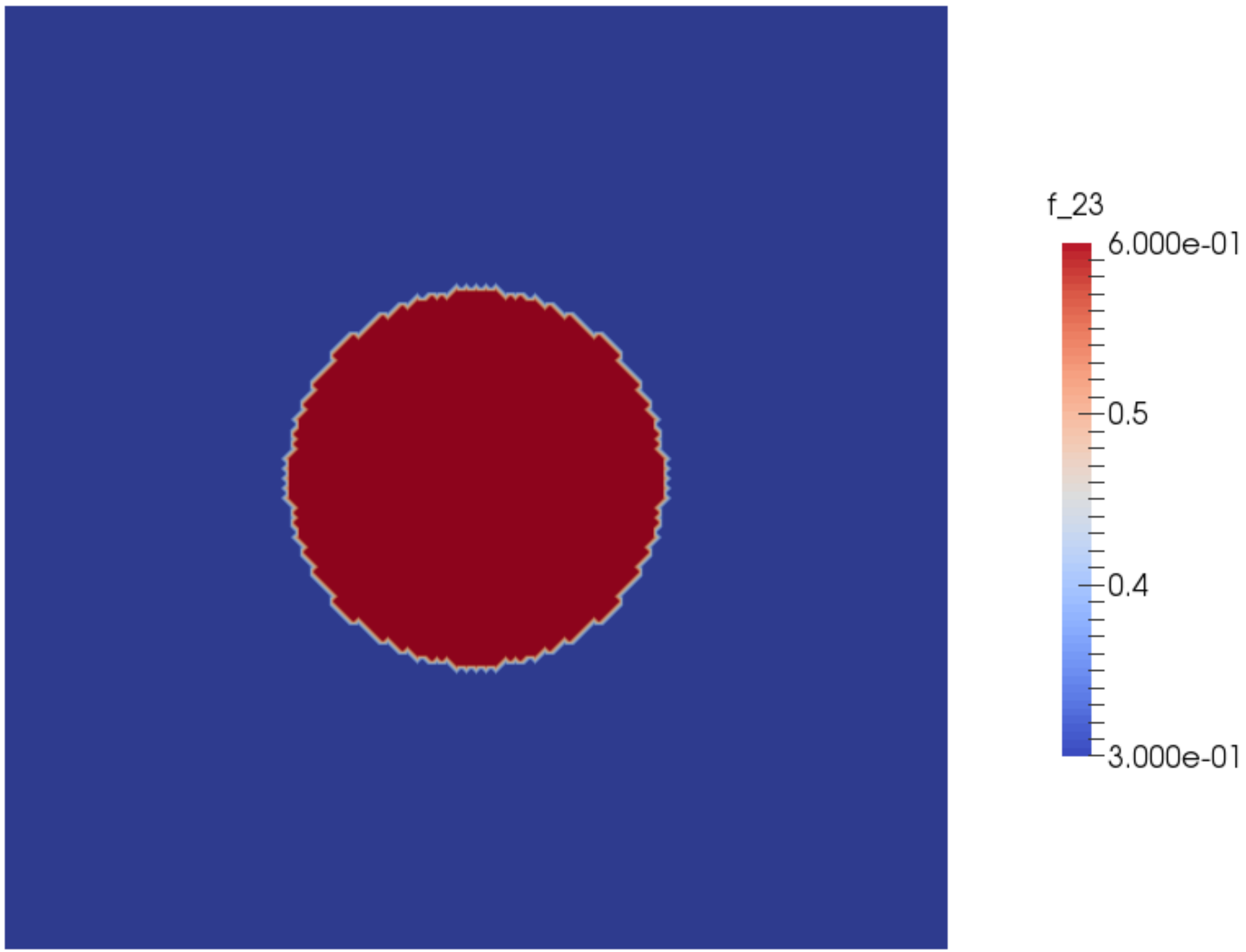
$$\mathcal{L}(y, x) = \int_{\Omega} \psi(y, x)\ dx - \int_{\Gamma} t \cdot y\ ds$$

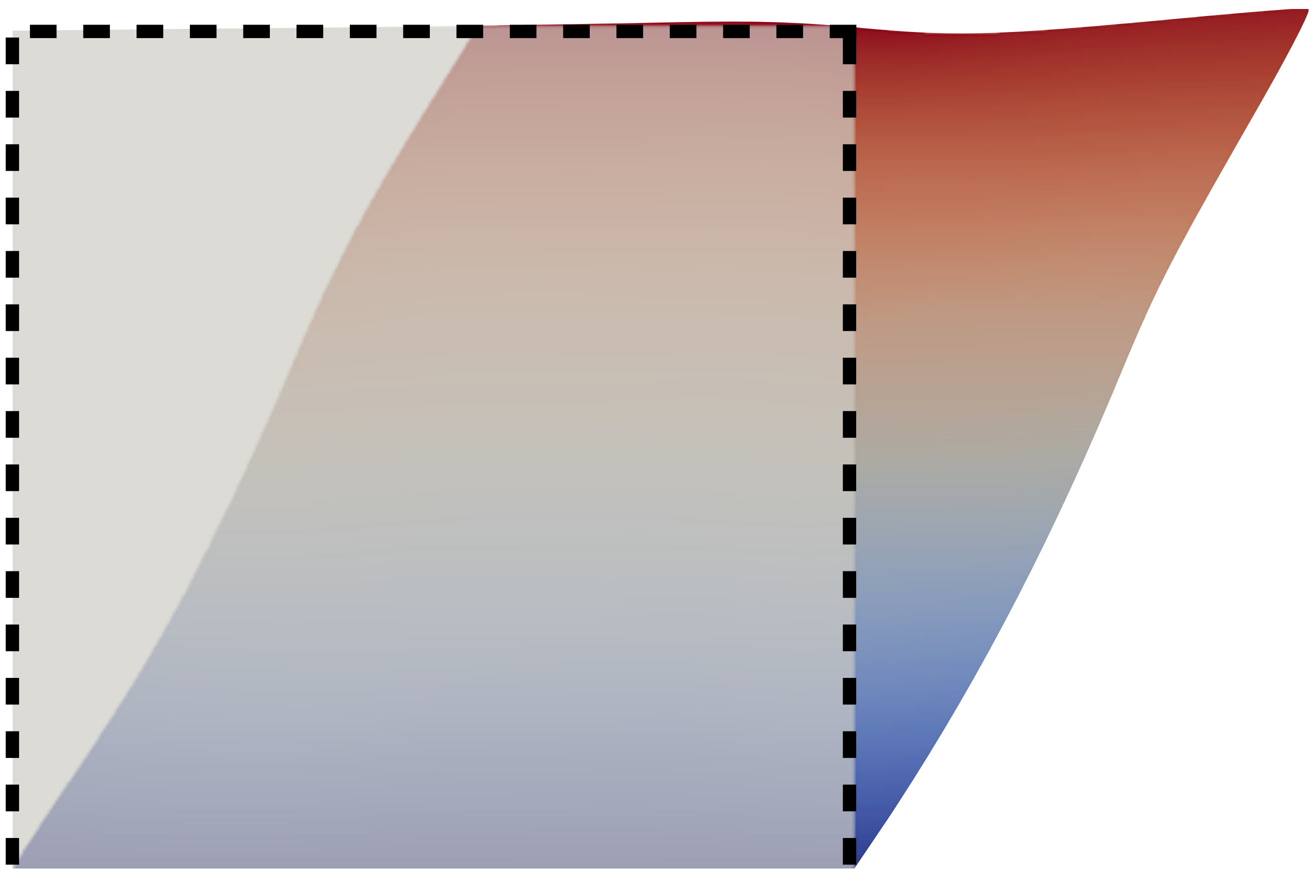
where the energy density functional ψ is defined through the following equations:

$$\psi(u, x) = \frac{x}{2}(I_c - d) - x \ln(J) + \frac{\lambda}{2}\ln(J)^2,$$
$$\mathbf{F} = \frac{\partial \phi}{\partial X} = \mathbf{I} + \nabla y,$$
$$\mathbf{C} = \mathbf{F}^T\mathbf{F},$$
$$\mathbf{I_C} = \mathrm{tr}(C),$$
$$J = \det\mathbf{F}.$$

```
from dolfin import *
mesh = UnitSquareMesh(32, 32)

U = VectorFunctionSpace(mesh, "CG", 1)
V = FunctionSpace(mesh, "CG", 1)
# solution
u = Function(U)
# test functions
v = TestFunction(U)
# incremental solution
du = TrialFunction(U)
x = interpolate(Constant(1.0), V)
lmbda = interpolate(Constant(100.0), V)

dims = mesh.type().dim()
I = Identity(dims)
F = I + grad(u)
C = F.T*F
J = det(F)
Ic = tr(C)
```

```
phi = (x/2.0)*(Ic - dims) - x*ln(J) + (lmbda/
2.0)*(ln(J))**2
Pi = phi*dx
# gateaux derivative with respect to u in
direction v
F = derivative(Pi, u, v)
# and with respect to u in direction du
J = derivative(F, u, du)

u_h = Function(U)
F_h = replace(F, {u: u_h})
J_h = replace(J, {u: u_h})
solve(F_h == 0, u_h, bcs, J=J_h)
```


f_23
6.000e-01
0.5
0.4
3.000e-01

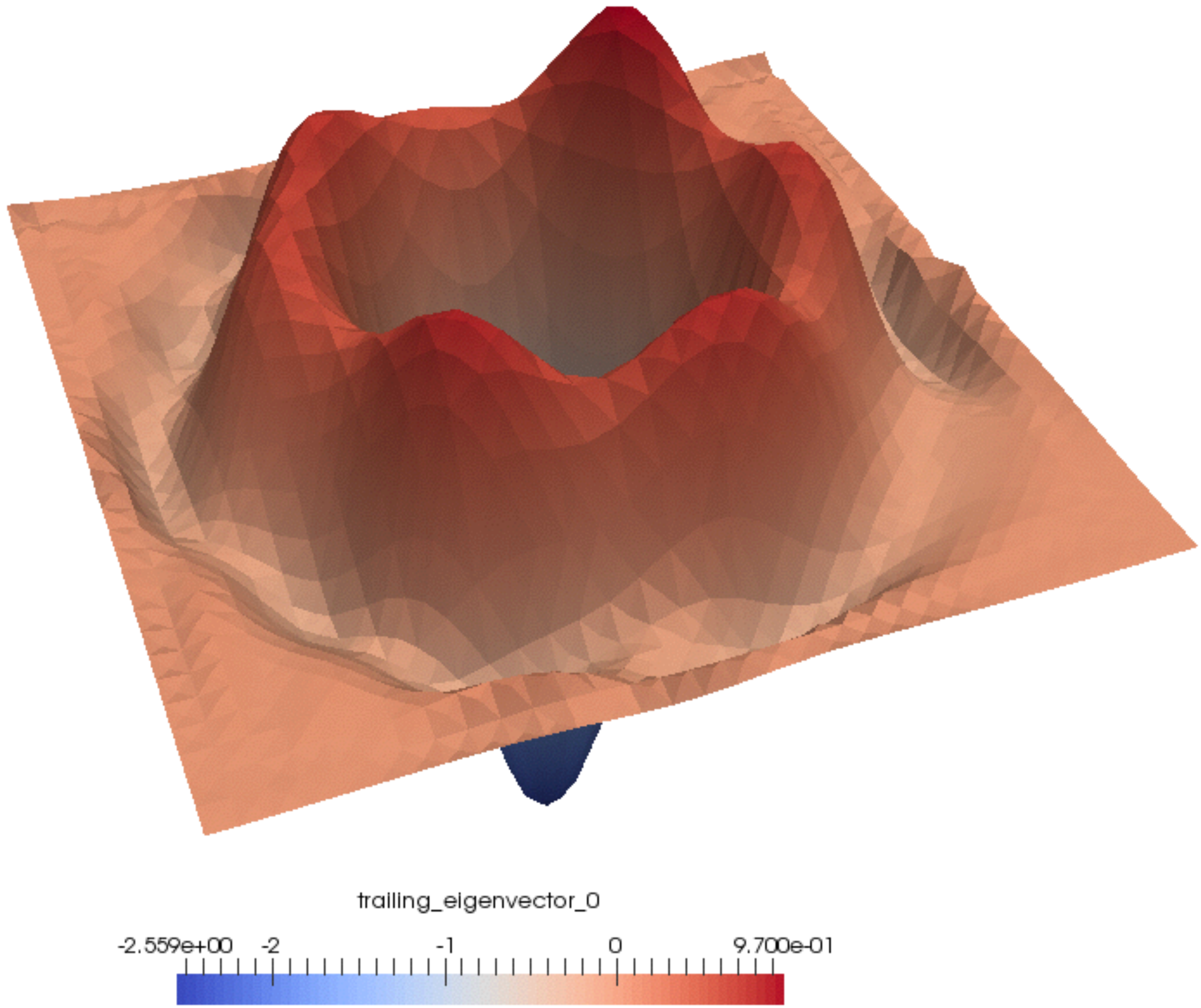


Q: What can we infer about the parameters inside the domain, just from displacement observations on the outside?

Q: Which parameters am I most uncertain about?

Trailing Eigenvector

Direction in parameter space *least* constrained by the observations

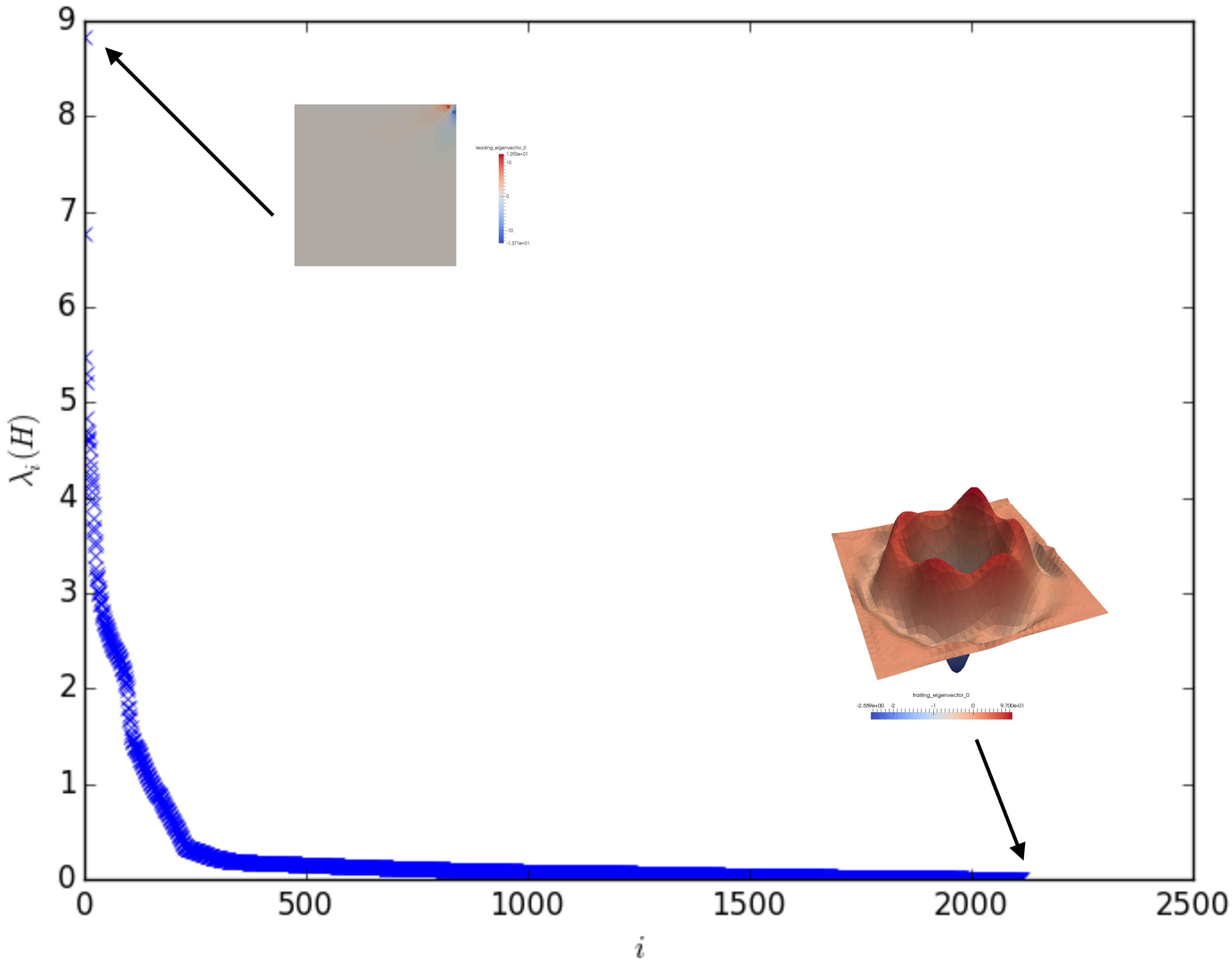

trailing_eigenvector_0
-2.559e+00
-2
-1
0
9.700e-01

Leading Eigenvectors

Direction in parameter space *most* constrained by the observations


leading_eigenvector_0
1.250e+01
10
0
-10
-1.371e+01


leading_eigenvector_1
1.405e+01
10
0
-10
-1.377e+01



leading_eigenvector_0
1.250e+01
10
0
-10
-1.371e+01
trailing_eigenvector_0
-2.559e+00
-2
-1
0
9.700e-01
$\lambda_i(H)$
i
0
1
2
3
4
5
6
7
8
9
0
500
1000
1500
2000
2500

Full Hessian.
4000+ actions.

Low-rank update.
292 actions.

Huge savings in computational cost.
Scales with model dimension because *observations* stay the same.

Bayesian approach: summary

- Quantify and propagate uncertainties
- Select the “best” model (Bayes factor)
- Identify parameters for these models
- Assimilate experimental or other numerical data
- Which parameters are we most uncertain about?
- What additional data would reduce uncertainty?