Surrogate minimisation in high dimensions

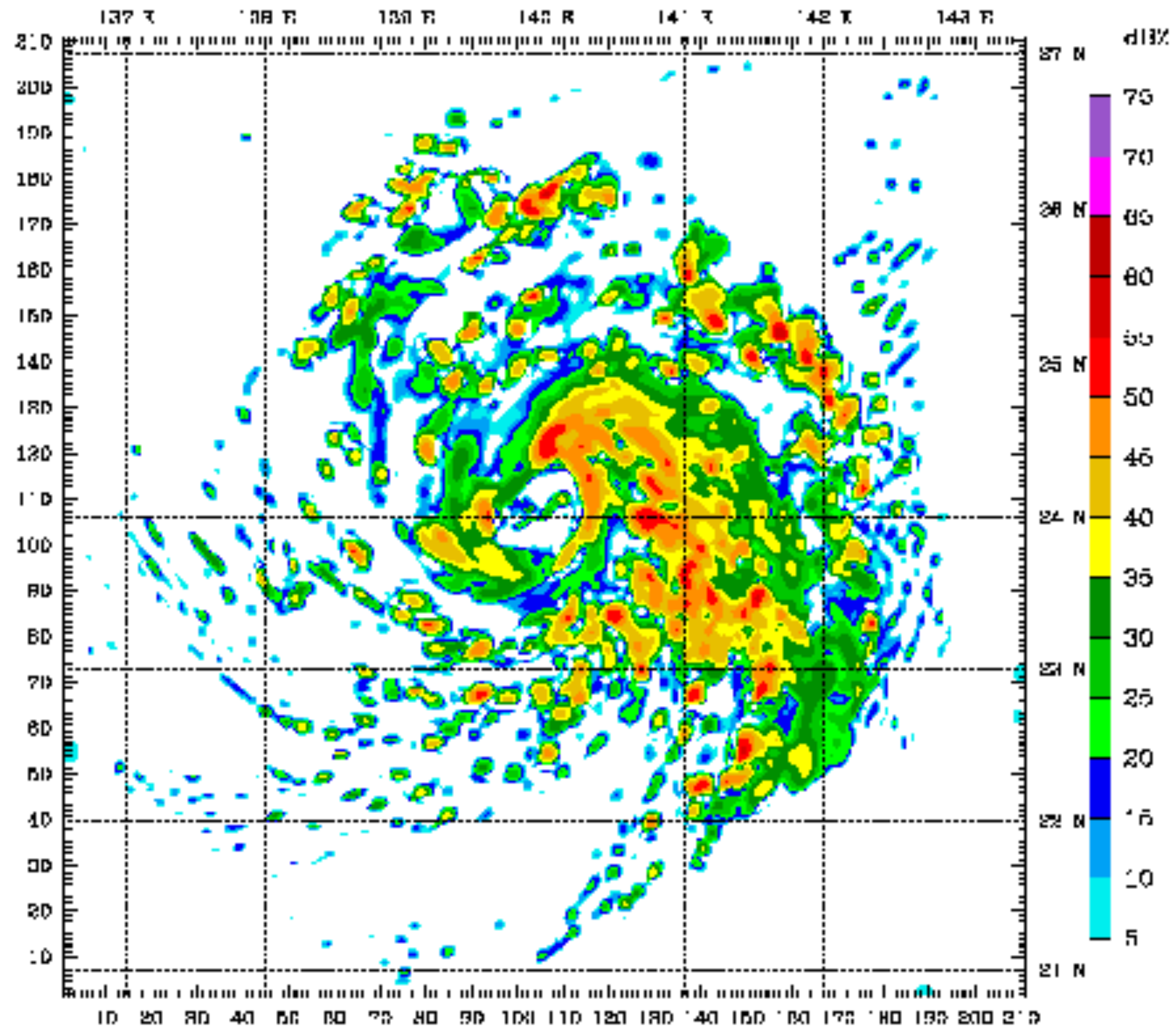
Fabrizia Guglielmetti (ESO)

In collaboration with:
Torsten Enßlin (MPA), Henrik Junklewitz (AIfA), Theo Steininger (MPA)

Outline

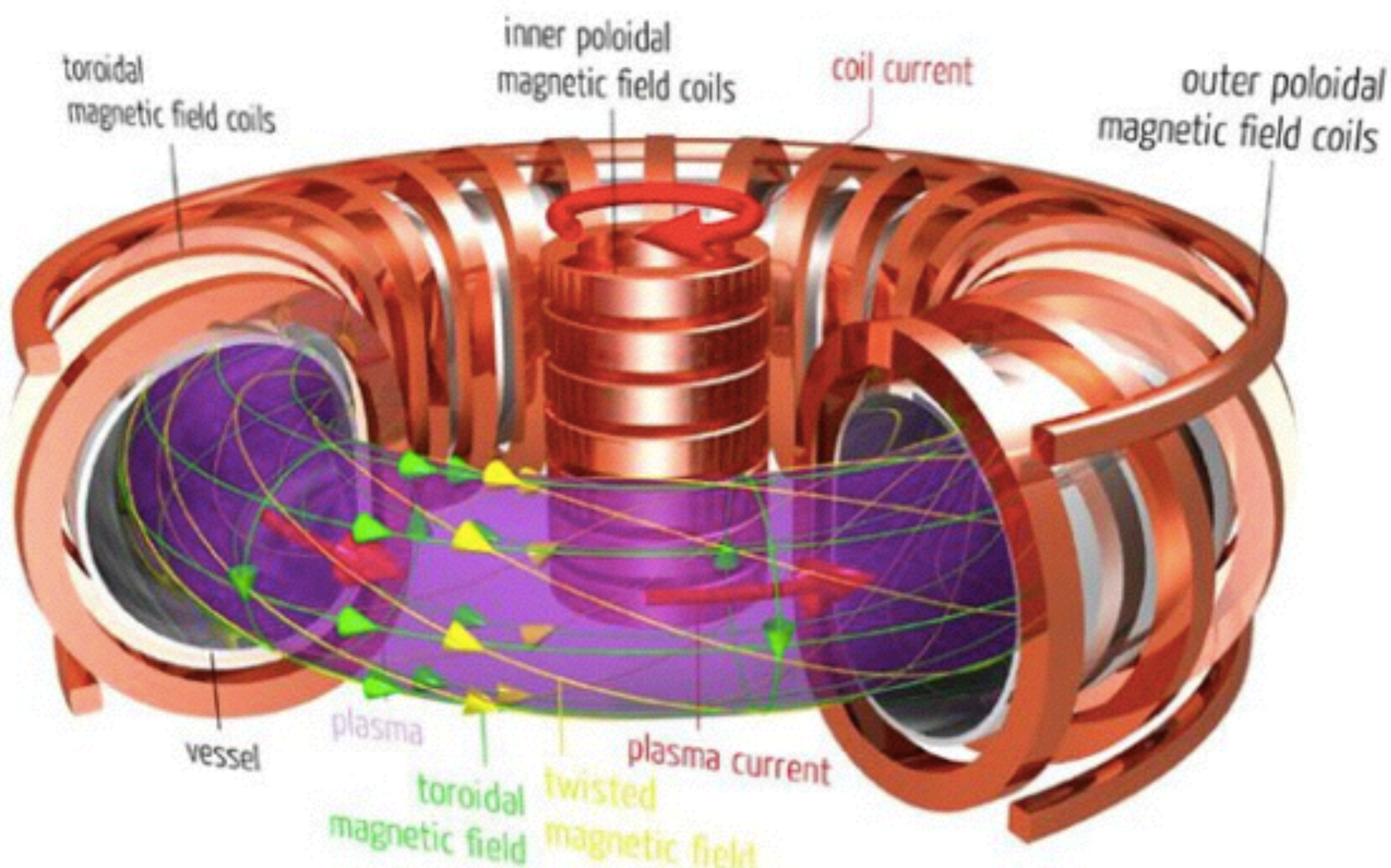
- Motivation
  - Performance enhancement of optimisation schemes
- Global optimisation as a Bayesian decision problem
  - Two connected statistical layers
- Surrogates basic principles
- Application of Kriging Surrogate within NIFTY (Selig, M. et al., 2013, A&A, 554, 26) framework

- Complex computer codes are essentials



A 48-hour computer simulation of Typhoon Mawar using the Weather Research and Forecasting model
credit Wikipedia

Introduction

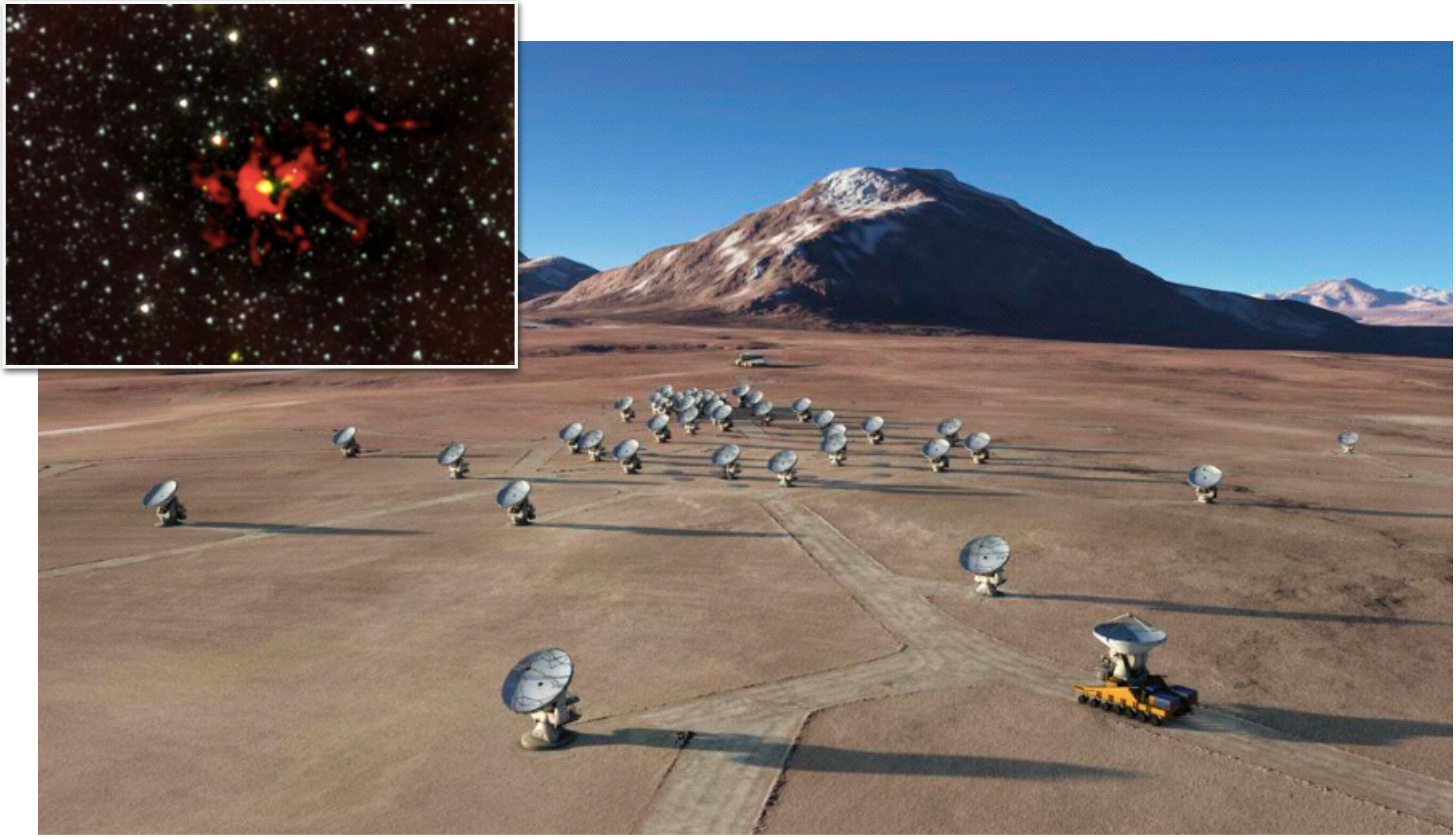


Schematic of the average tokamak. Notice how it has fewer layers than the stellarator and the shape of the magnetic coils is different. Credit: Uploaded by Matthias W Hirsch on Wikipedia

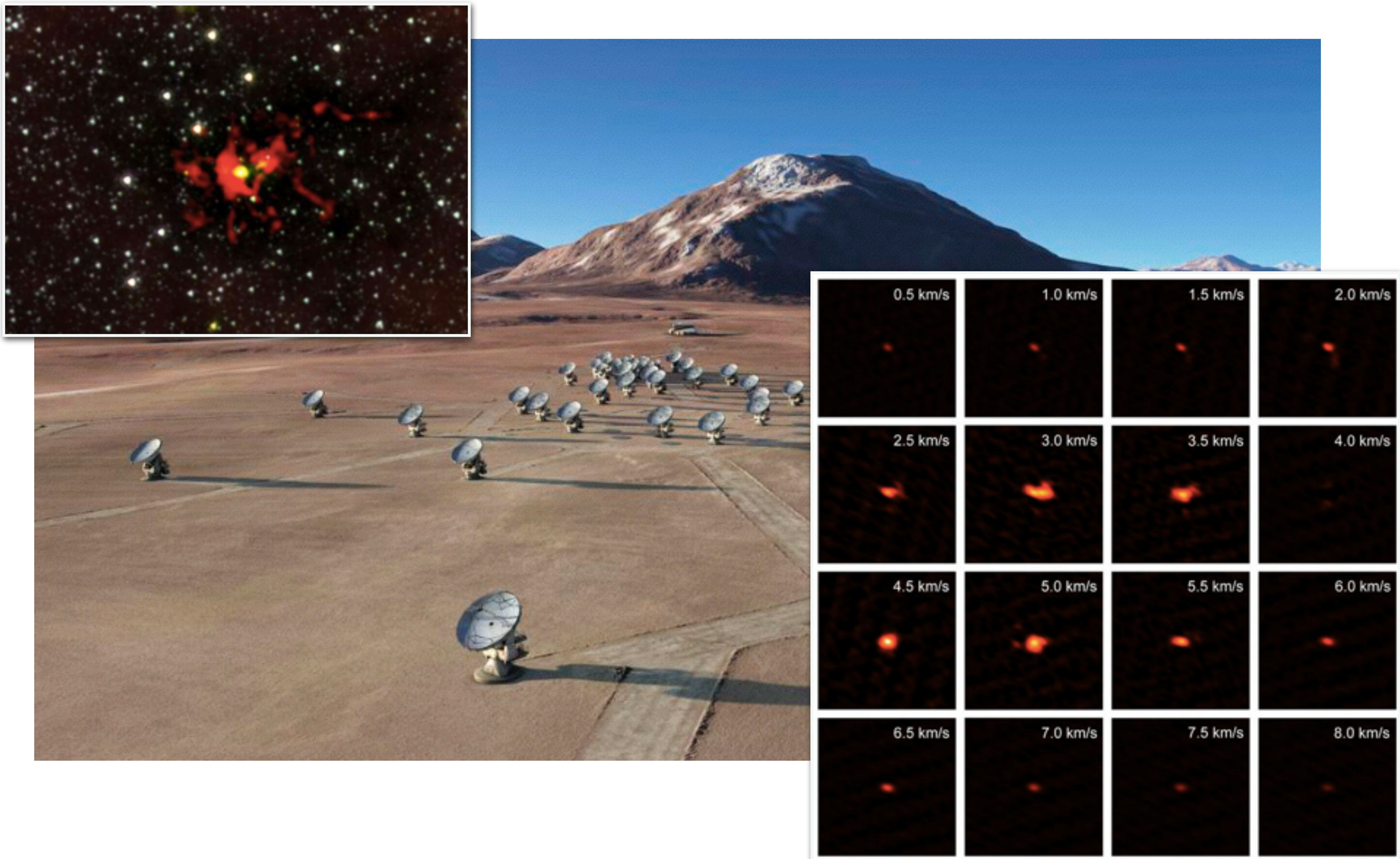
See, e.g., Cox D.D, Park J.S., Singer C.E. (2001)
Preuss, R. & von Toussaint, U. (2016)

Introduction

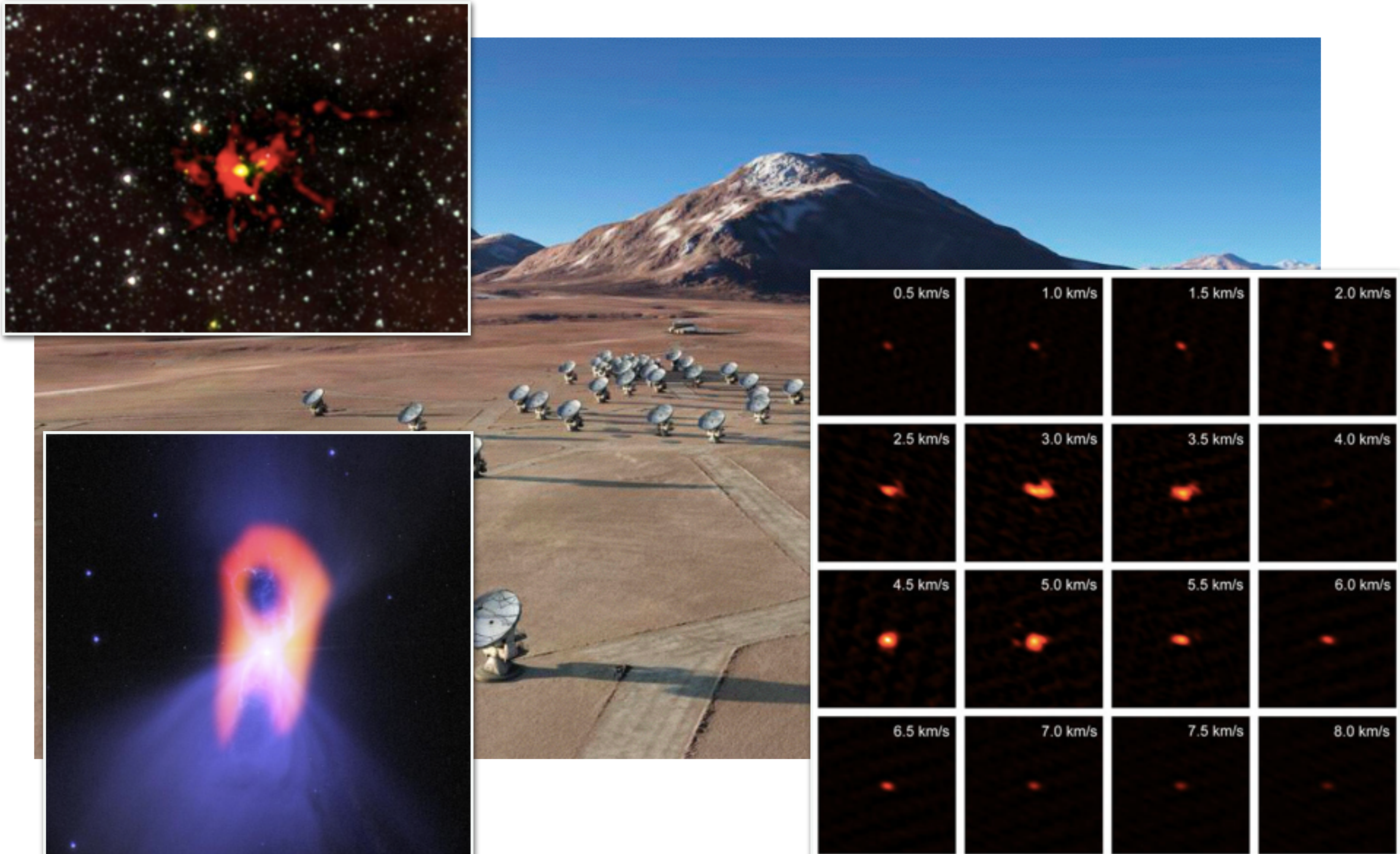
Introduction



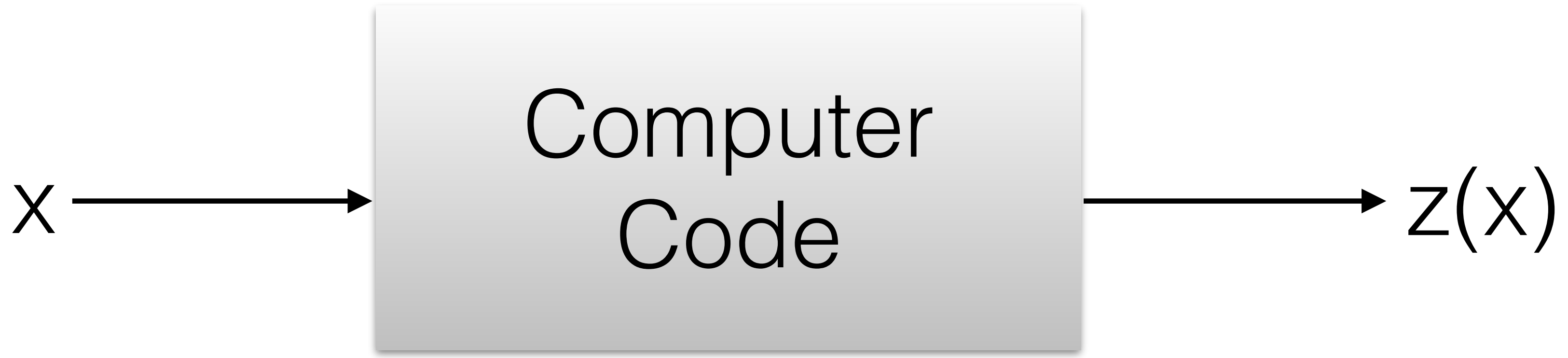
Introduction



Introduction



- Complex computer codes are essentials



Simulator

- Complex computer codes are essentials:

Simulator

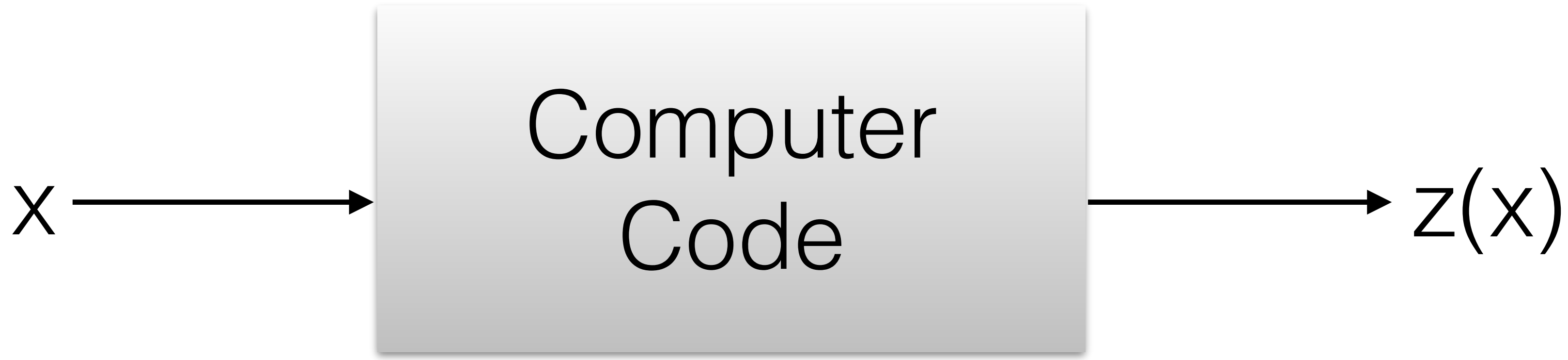
Minimisation of Complex Objective function. For reasonable data size, current estimation techniques:

a. <u>SLOW</u> if full objective function is accounted

b. <u>FAST</u> if do not account for full objective function

Introduction

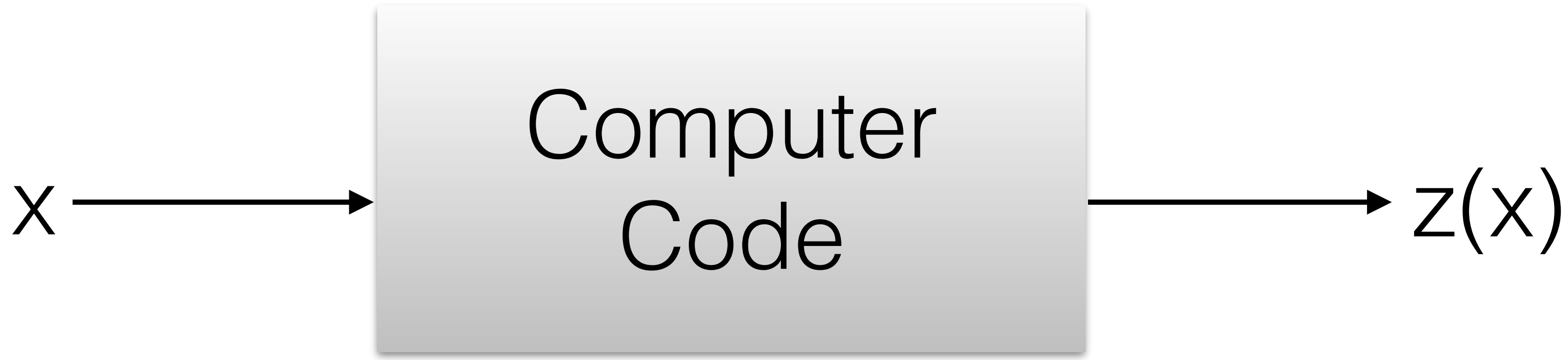
- Complex computer codes are essentials:



Emulator

- statistical representation of x
- expresses knowledge about z(x) at any given x
- built using prior information and training set of model runs

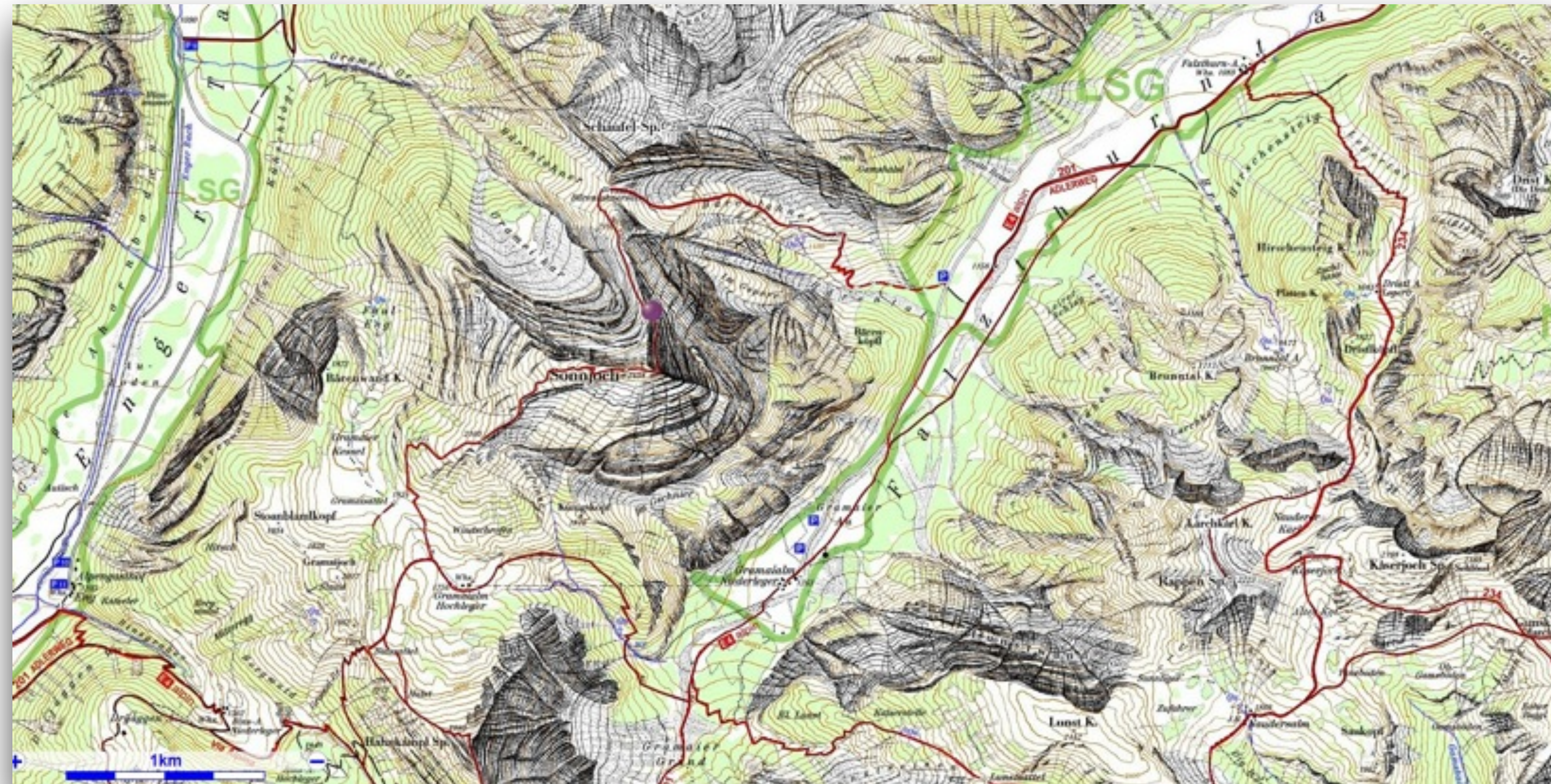
- Complex computer codes are essentials:



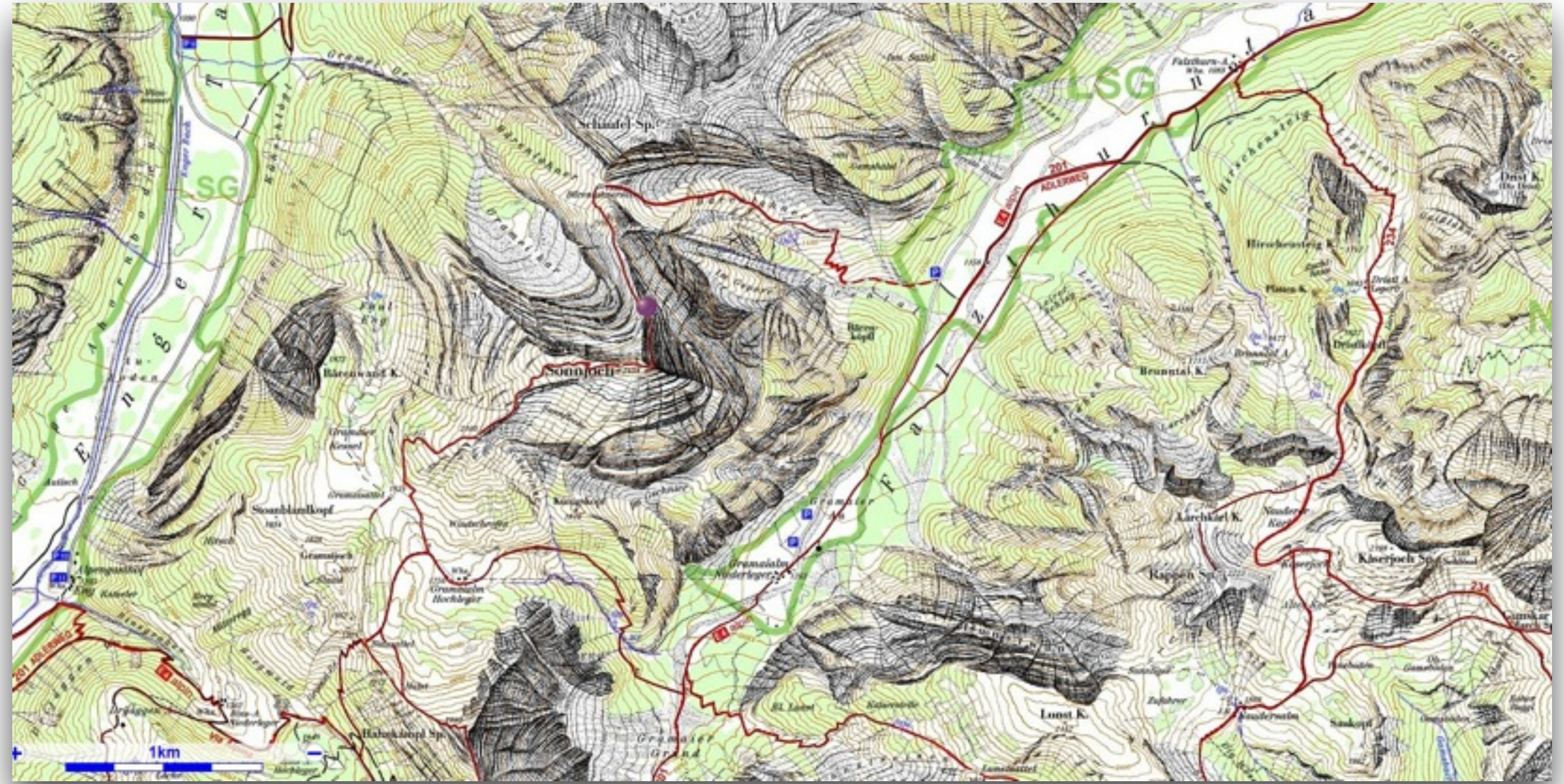
Emulator

Kriging surrogate sampler is used to emulate the original Complex Objective function

Surrogates (or metamodels, emulators, …)



Surrogates (or metamodels, emulators, …)



- A map is a surrogate that predicts terrain elevation
- Prediction of the map allows one to locate the summit without climbing it
- Searching for the highest peak is a form of global optimisation

Surrogates (or metamodels, emulators, ...)

... in the same fashion we can use metamodels to find the optimal signal configuration

Bayesian inference, parametric model

- Data: x, y
- Model: $y = f_k(x) + \epsilon$

Gaussian likelihood: $p(\mathbf{y}|\mathbf{x}, \mathbf{k}, M_i) \propto \prod_j \exp\left(-\frac{1}{2}(\mathrm{y_j} - \mathrm{f_k}(\mathrm{x_j}))^2/\sigma^2\right)$

Prior over the param: $p(\mathbf{k}|M_i)$

Posterior param distr: $p(\mathbf{k}|\mathbf{x}, \mathbf{y}, M_i) = \frac{p(\mathbf{y}|\mathbf{x}, \mathbf{k}, M_i)p(\mathbf{k}|M_i)}{p(\mathbf{y}|\mathbf{x}, M_i)}$

Make predictions:

$$p(y^*|x^*, \mathbf{x}, \mathbf{y}, M_i) = \int p(y^*|\mathbf{k}, x^*, M_i)p(\mathbf{k}|\mathbf{x}, \mathbf{y}, M_i)d\mathbf{k}$$

Bayesian inference, parametric model

- Data: x, y
- Model: $y = f_k(x) + \epsilon$

Gaussian likelihood: $p(\mathbf{y}|\mathbf{x}, \mathbf{k}, M_i) \propto \prod_j \exp\left(-\frac{1}{2}(\mathrm{y_j} - \mathrm{f_k}(\mathrm{x_j}))^2/\sigma^2\right)$

Prior over the param: $p(\mathbf{k}|M_i)$

Posterior param distr: $p(\mathbf{k}|\mathbf{x}, \mathbf{y}, M_i) = \frac{p(\mathbf{y}|\mathbf{x}, \mathbf{k}, M_i)\, p(\mathbf{k}|M_i)}{p(\mathbf{y}|\mathbf{x}, M_i)}$

Make predictions:

$$p(y^*|x^*, \mathbf{x}, \mathbf{y}, M_i) = \int p(y^*|\mathbf{k}, x^*, M_i)\, p(\mathbf{k}|\mathbf{x}, \mathbf{y}, M_i)\, d\mathbf{k}$$

Bayesian inference, parametric model

- Data: x, y
- Model: $y = f_k(x) + \epsilon$

Gaussian likelihood: $p(\mathbf{y}|\mathbf{x}, \mathbf{k}, M_i) \propto \prod_j \exp\left(-\frac{1}{2}(\mathrm{y_j} - \mathrm{f_k}(\mathrm{x_j}))^2/\sigma^2\right)$

Prior over the param: $p(\mathbf{k}|M_i)$

Posterior param distr: $p(\mathbf{k}|\mathbf{x}, \mathbf{y}, M_i) = \frac{p(\mathbf{y}|\mathbf{x}, \mathbf{k}, M_i)\, p(\mathbf{k}|M_i)}{p(\mathbf{y}|\mathbf{x}, M_i)}$

Make predictions:

$$p(y^*|x^*, \mathbf{x}, \mathbf{y}, M_i) = \int p(y^*|\mathbf{k}, x^*, M_i)\, p(\mathbf{k}|\mathbf{x}, \mathbf{y}, M_i)\, d\mathbf{k}$$

IFT for signal inference

- Information Field Theory (Enßlin, T. et al. 2009), information theory for fields
- Signal field (s) estimation:

$$P(s|d) = \frac{P(d|s)P(s)}{P(d)} \equiv \frac{e^{-H(d,s)}}{Z_d}$$

IFT for signal inference

- Information Field Theory (Enßlin, T. et al. 2009), information theory for fields
- Signal field (s) estimation:

Information Hamiltonian

$$P(s|d) = \frac{P(d|s)P(s)}{P(d)} \equiv \frac{e^{-H(d,s)}}{Z_d}$$

IFT for signal inference

- Information Field Theory (Enßlin, T. et al. 2009), information theory for fields
- Signal field (s) estimation:

Information Hamiltonian

$$P(s|d) = \frac{P(d|s)P(s)}{P(d)} \equiv \frac{e^{-H(d,s)}}{Z_d}$$

Partition function

IFT for signal inference

- Information Field Theory (Enßlin, T. et al. 2009), information theory for fields
- Signal field (s) estimation:

$$P(s|d) = \frac{P(d|s)P(s)}{P(d)} \equiv \frac{e^{-H(d,s)}}{Z_d}$$

$d = (d_1, d_2, \ldots, d_n)^T \qquad n \in \mathbb{N} \qquad$ finite dataset

IFT exploits known or inferred correlation structures of the field of interest $s = s(x)$ over some domain $\Omega = \{x\}$ to regularise the ill-posed inverse problem of determining an ∞ number of dof from a finite dataset

IFT for signal inference

- Information Field Theory (Enßlin, T. et al. 2009), information theory for fields
- Signal field (s) estimation:

$$P(s|d) = \frac{P(d|s)P(s)}{P(d)} \equiv \frac{e^{-H(d,s)}}{Z_d}$$

most probable signal configuration

IFT for signal inference

- Information Field Theory (Enßlin, T. et al. 2009), information theory for fields
- Signal field (s) estimation:

$$P(s|d) = \frac{P(d|s)P(s)}{P(d)} \equiv \frac{e^{-H(d,s)}}{Z_d}$$

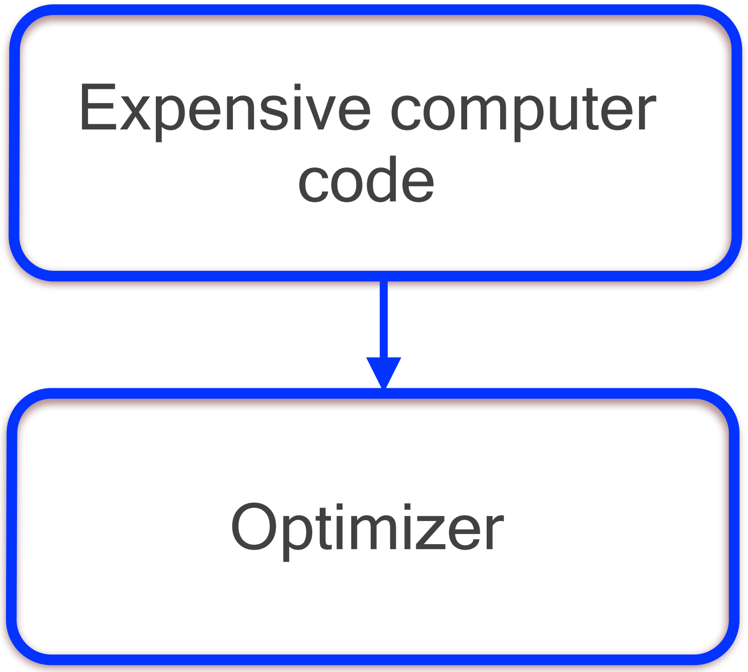
most probable signal configuration

$$\langle s \rangle = \mathrm{argmin}_{\langle \mathrm{s}|\mathrm{d} \rangle} \mathrm{H}(\mathrm{d}, \mathrm{s})$$

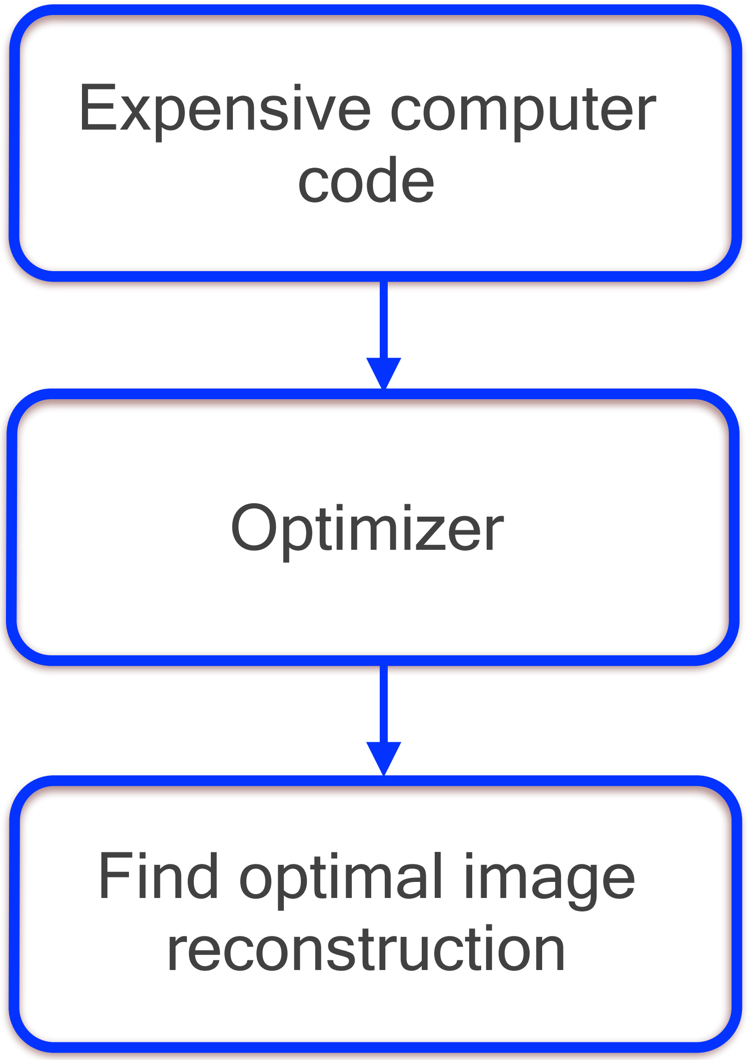
Minimization of (complex) energy function

Expensive computer code

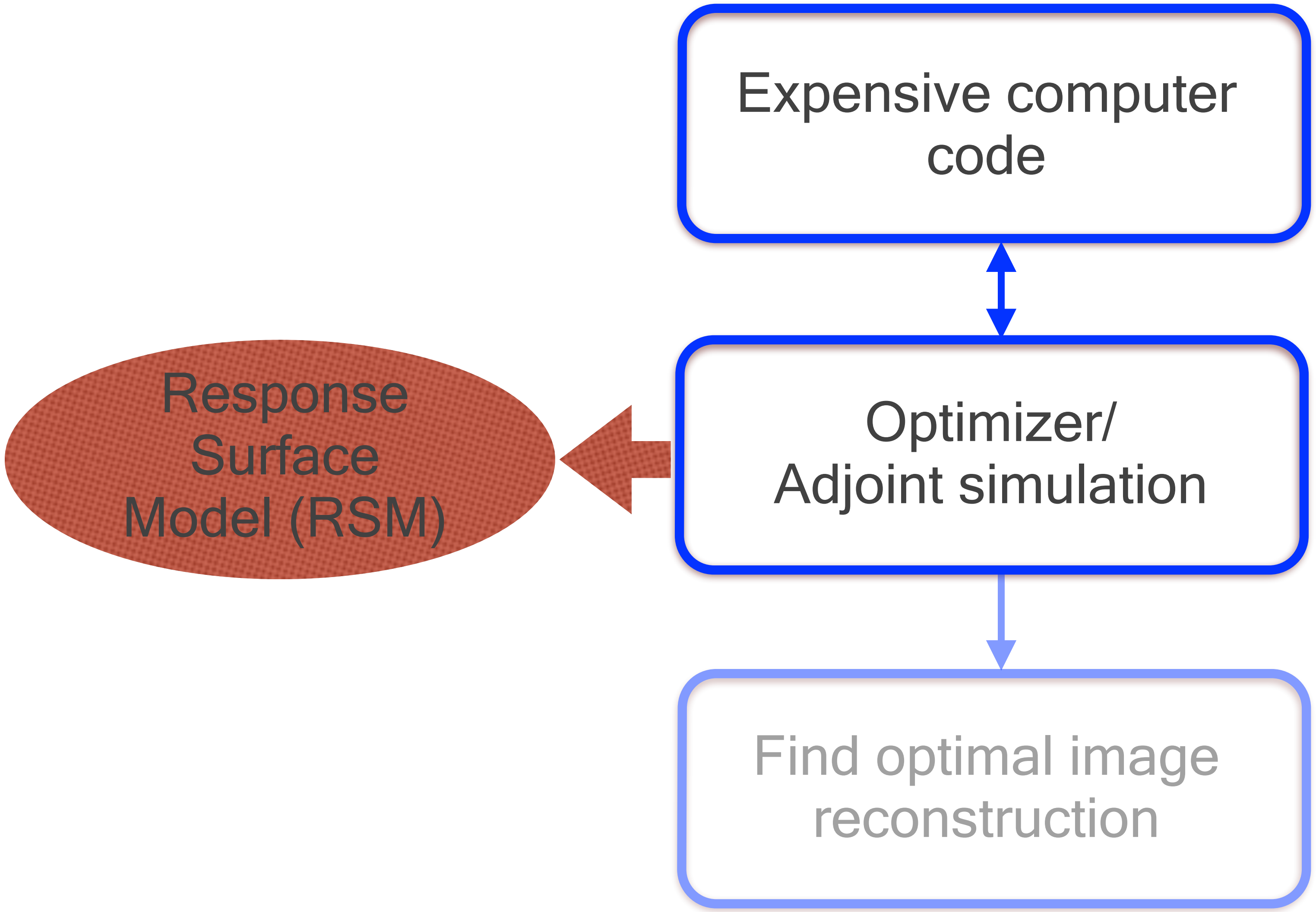
Minimization of (complex) energy function



Minimization of (complex) energy function

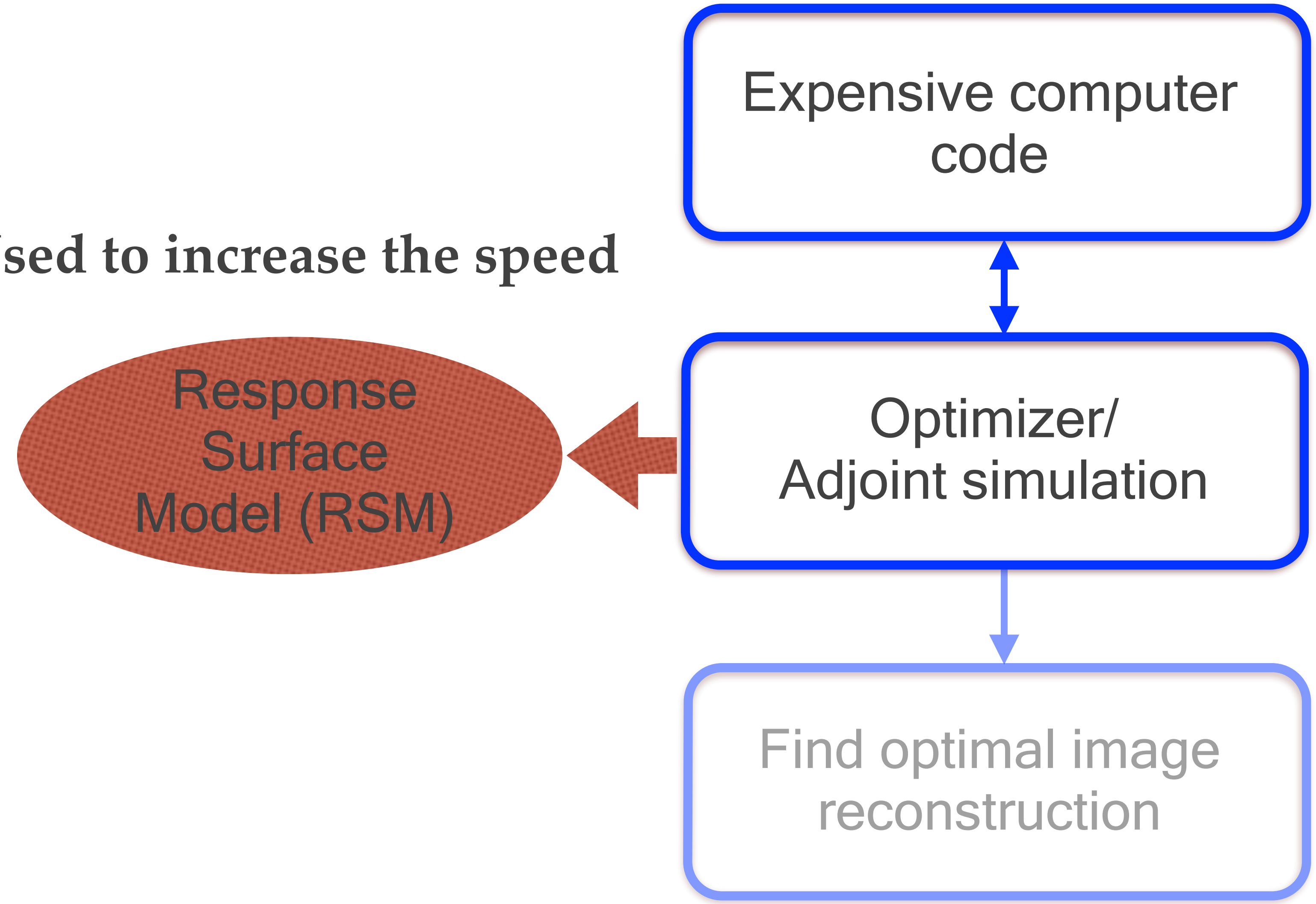


Minimization of (complex) energy function



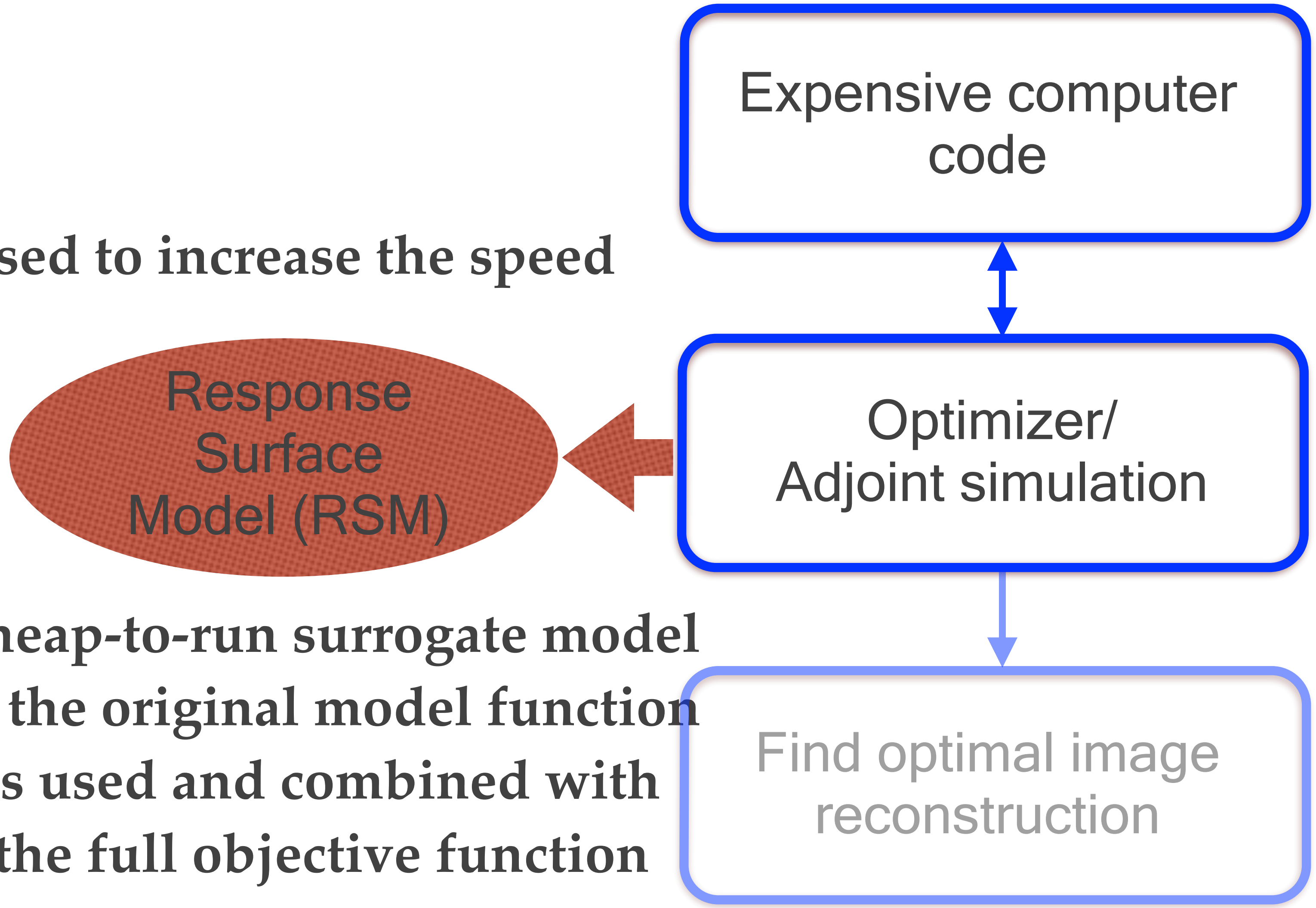
Minimization of (complex) energy function

Used to increase the speed



Minimization of (complex) energy function

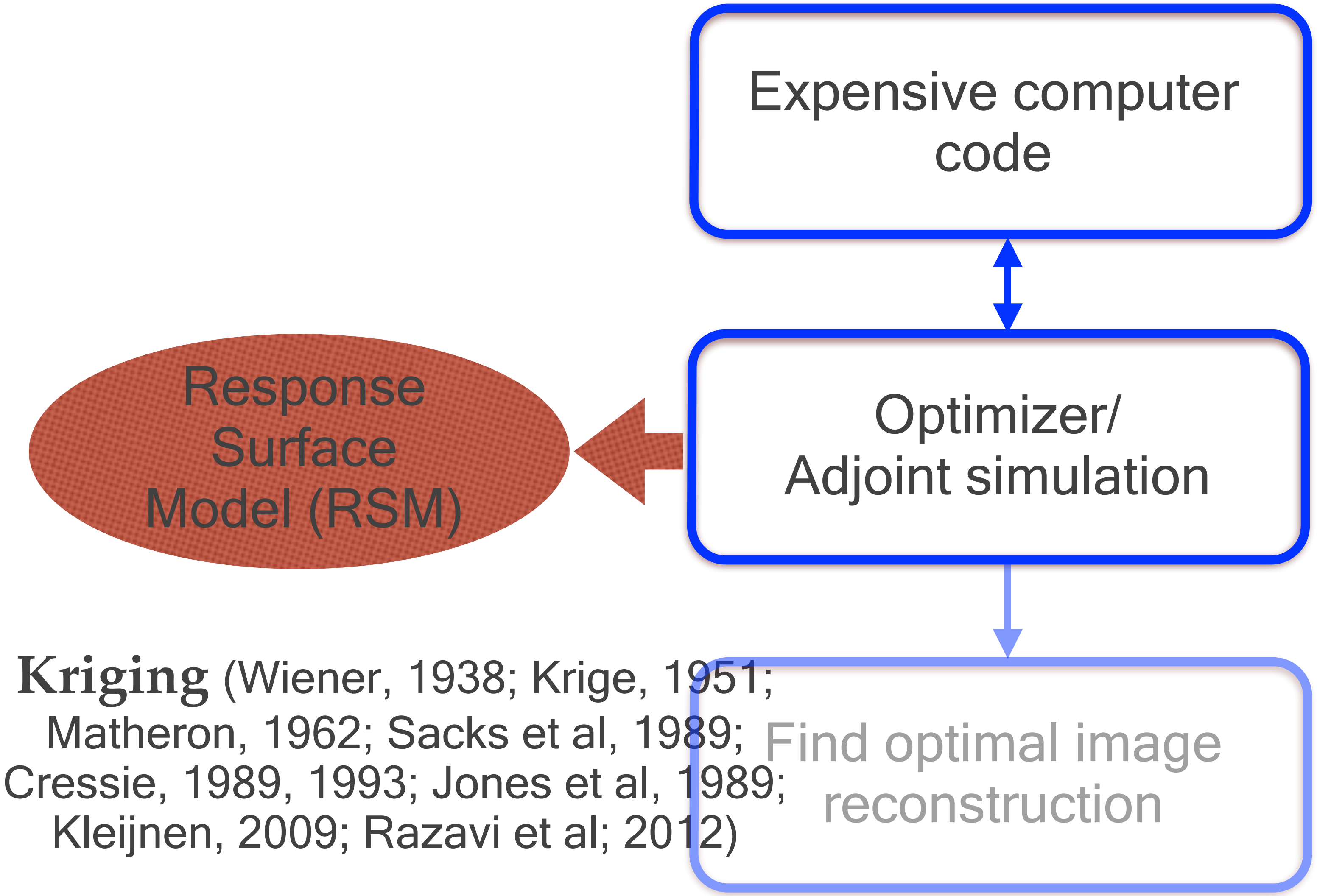
Used to increase the speed



Cheap-to-run surrogate model of the original model function is used and combined with the full objective function

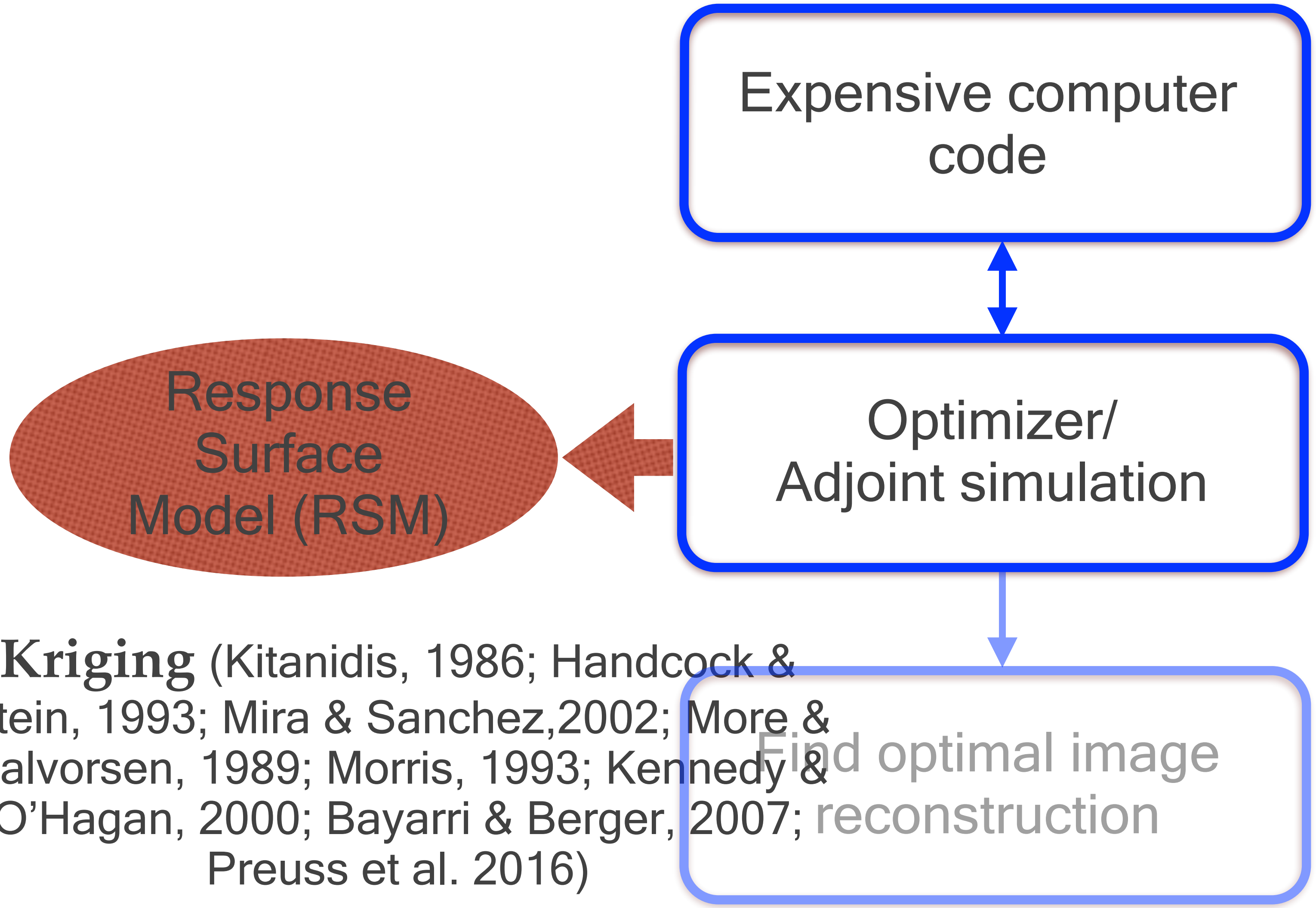
(same model is used throughout the optimisation process)

Minimization of (complex) energy function



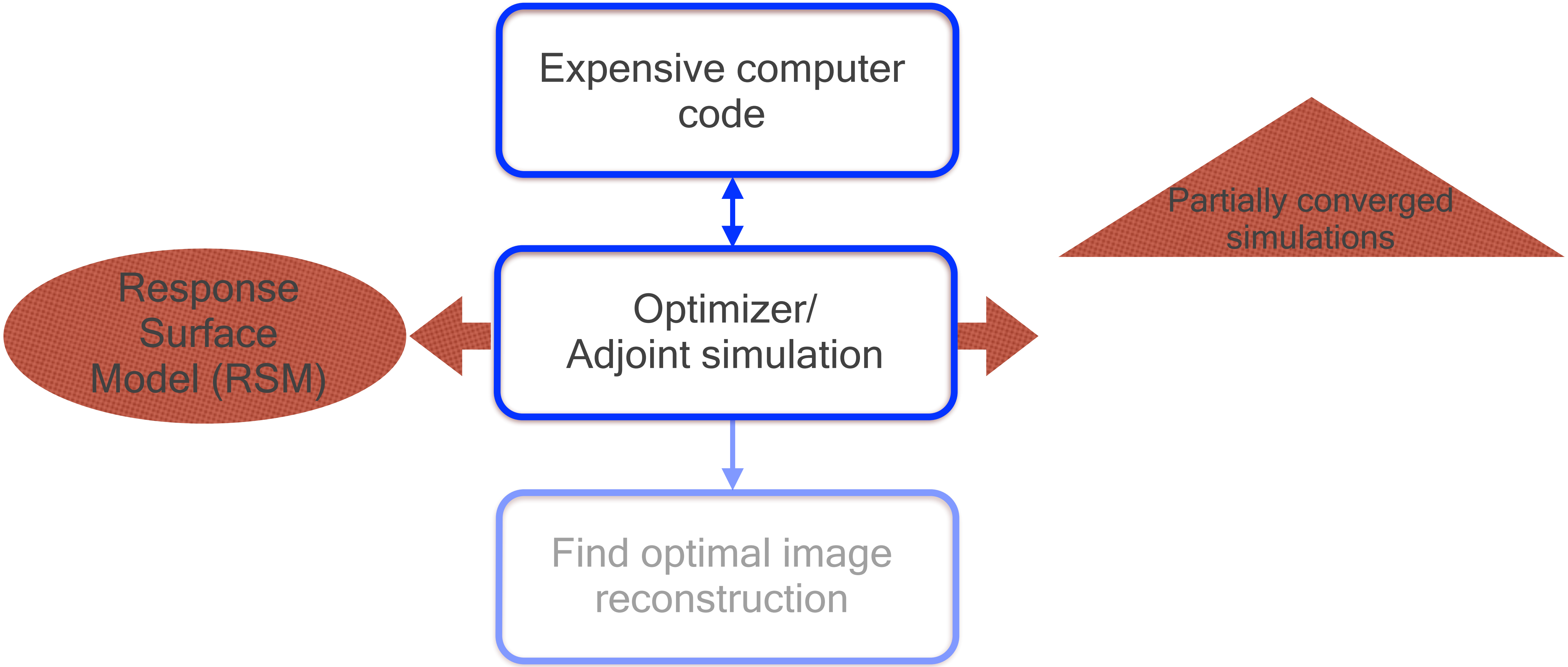
Kriging (Wiener, 1938; Krige, 1951; Matheron, 1962; Sacks et al, 1989; Cressie, 1989, 1993; Jones et al, 1989; Kleijnen, 2009; Razavi et al; 2012)

Minimization of (complex) energy function

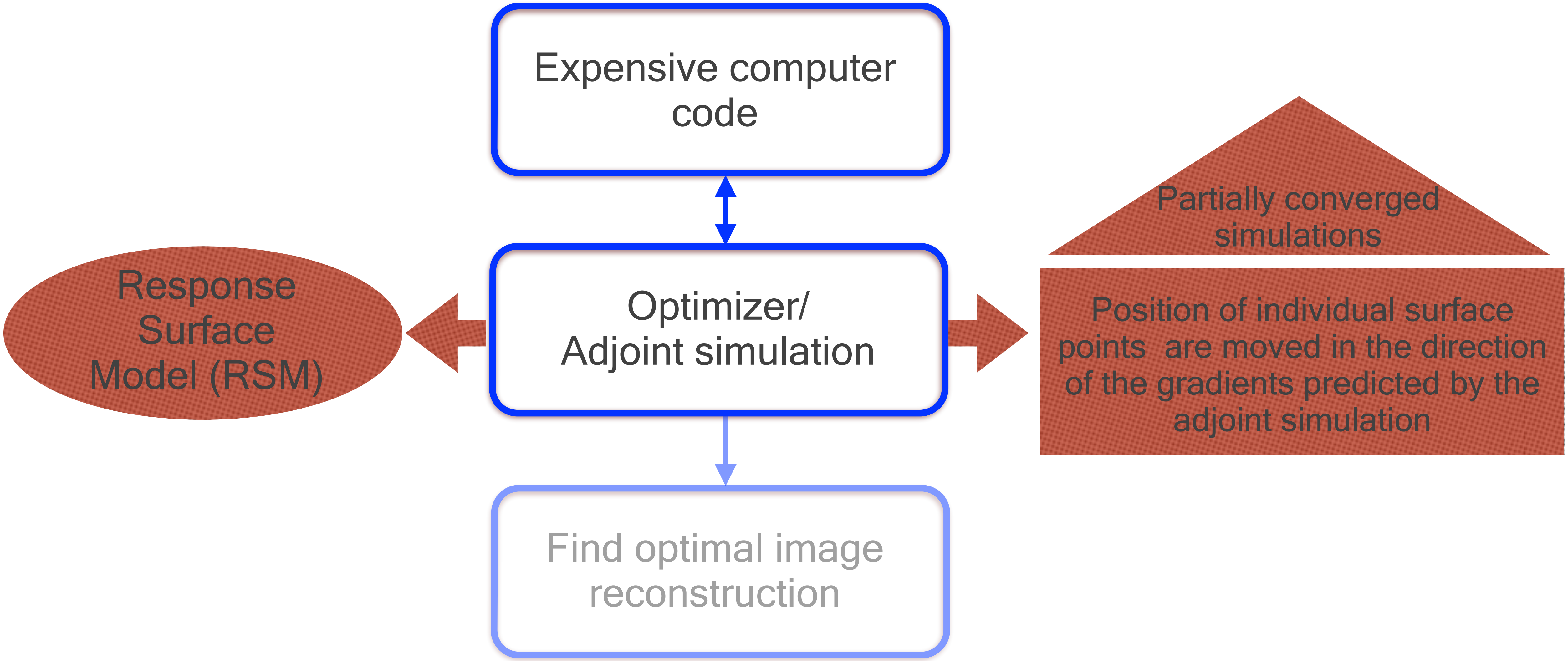


Kriging (Kitanidis, 1986; Handcock & Stein, 1993; Mira & Sanchez,2002; More & Halvorsen, 1989; Morris, 1993; Kennedy & O'Hagan, 2000; Bayarri & Berger, 2007; Preuss et al. 2016)

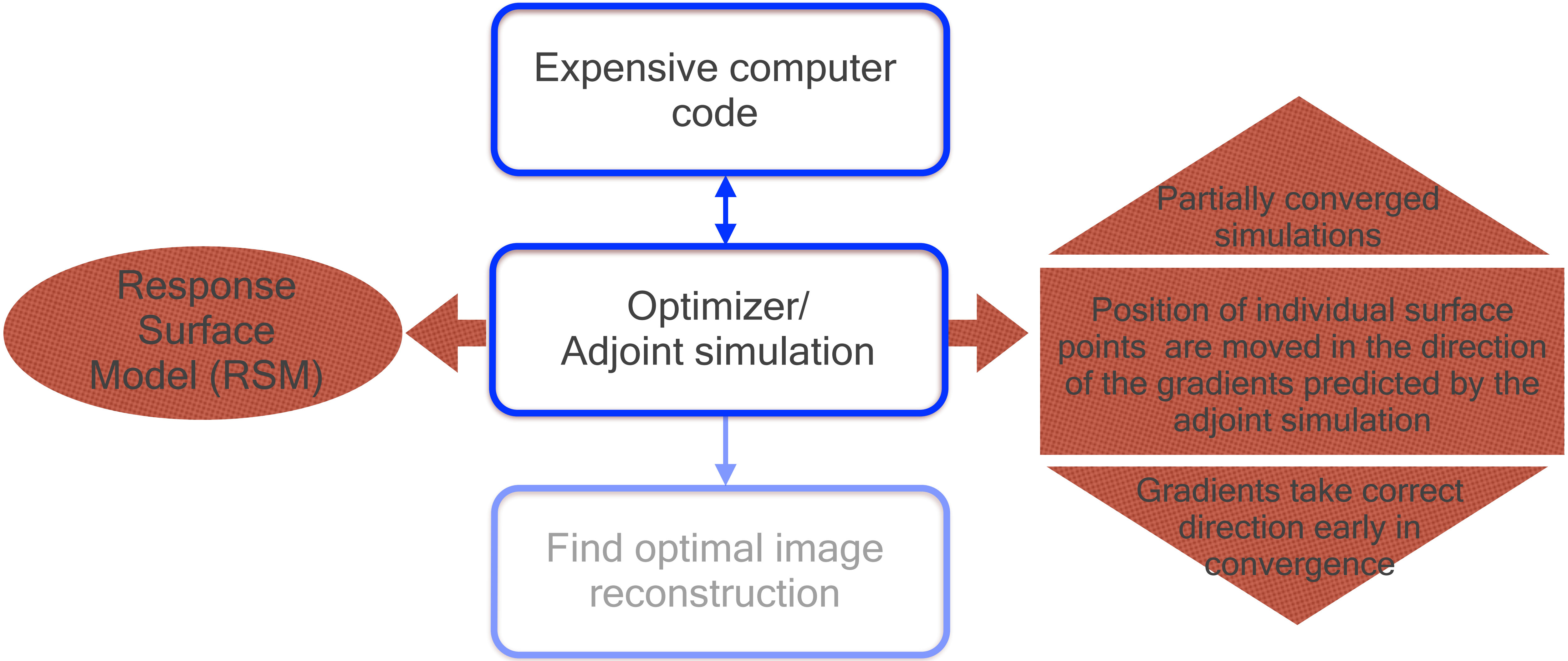
Minimization of (complex) energy function



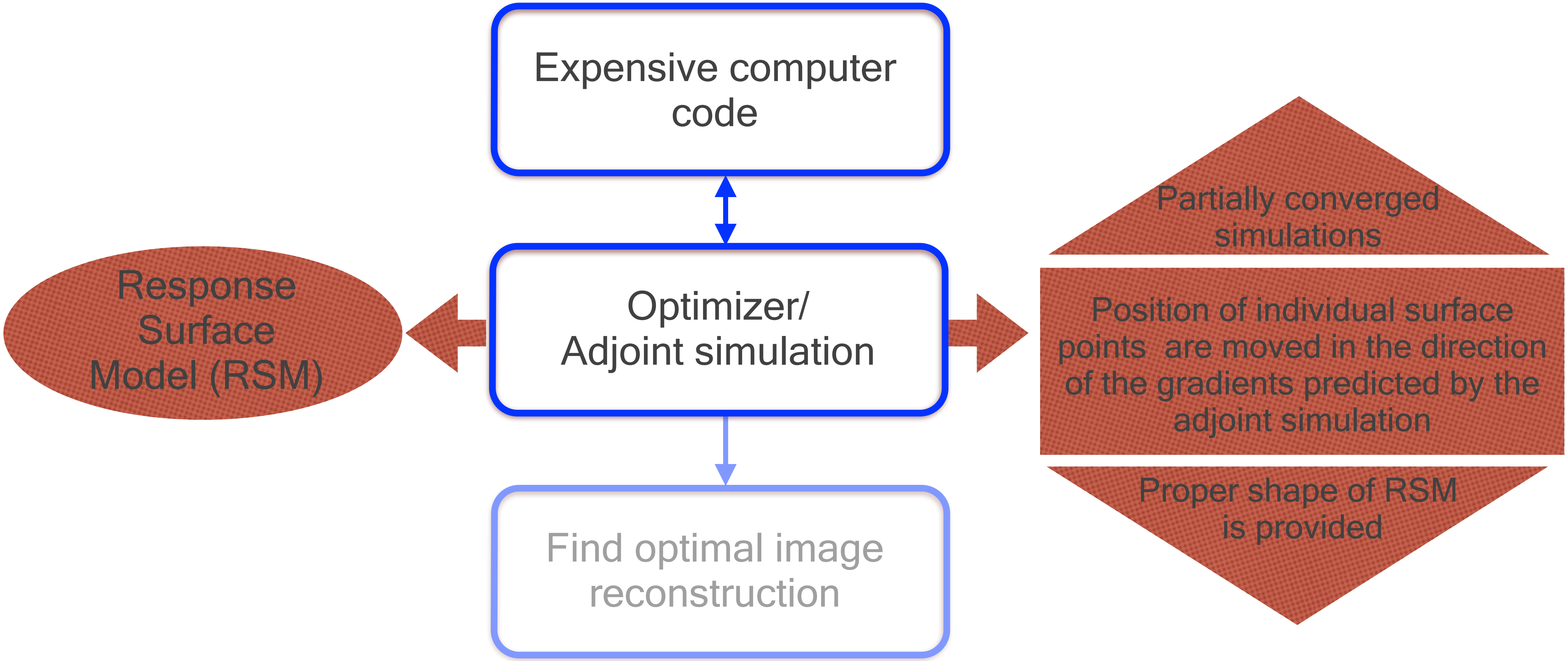
Minimization of (complex) energy function



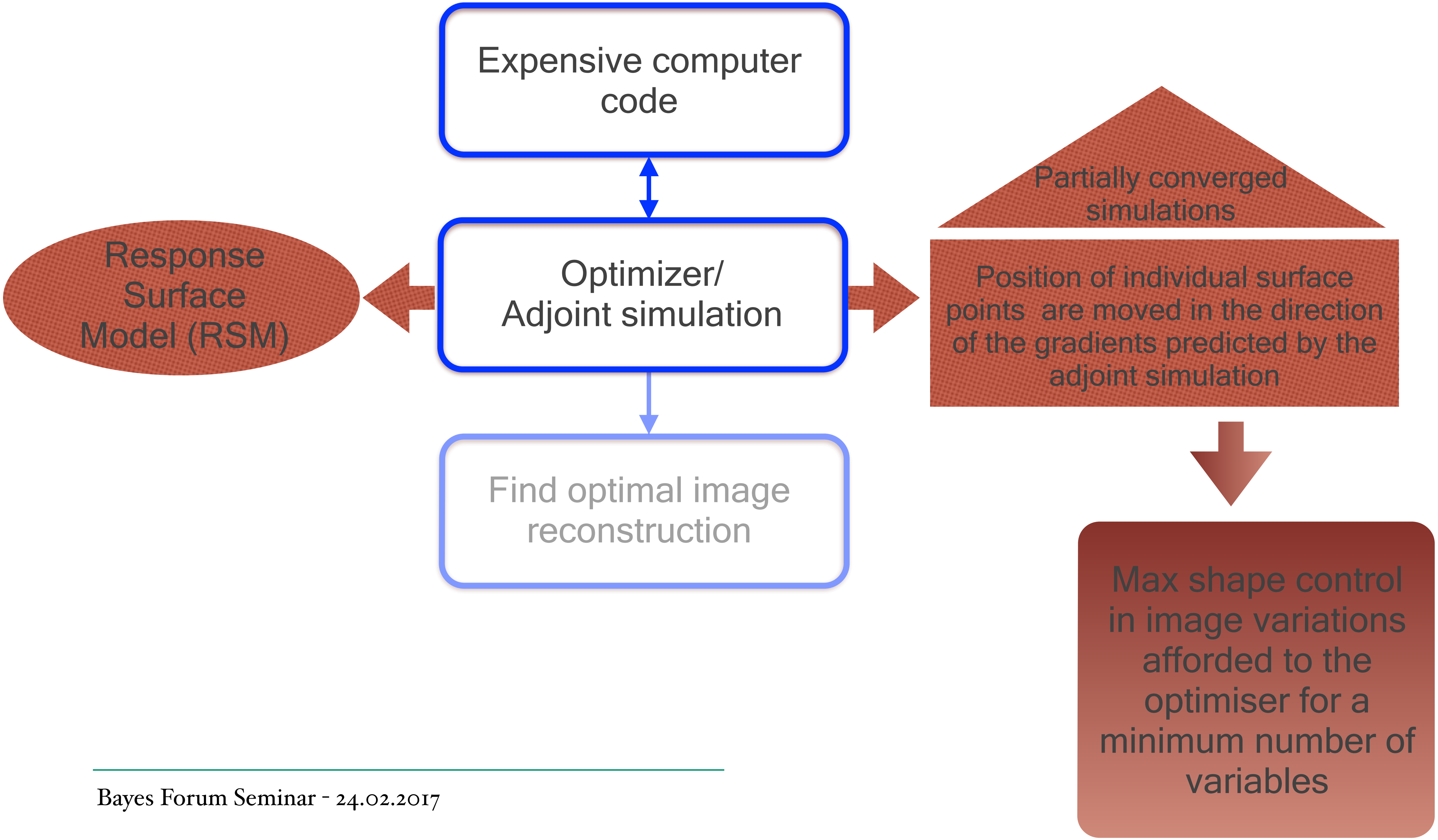
Minimization of (complex) energy function



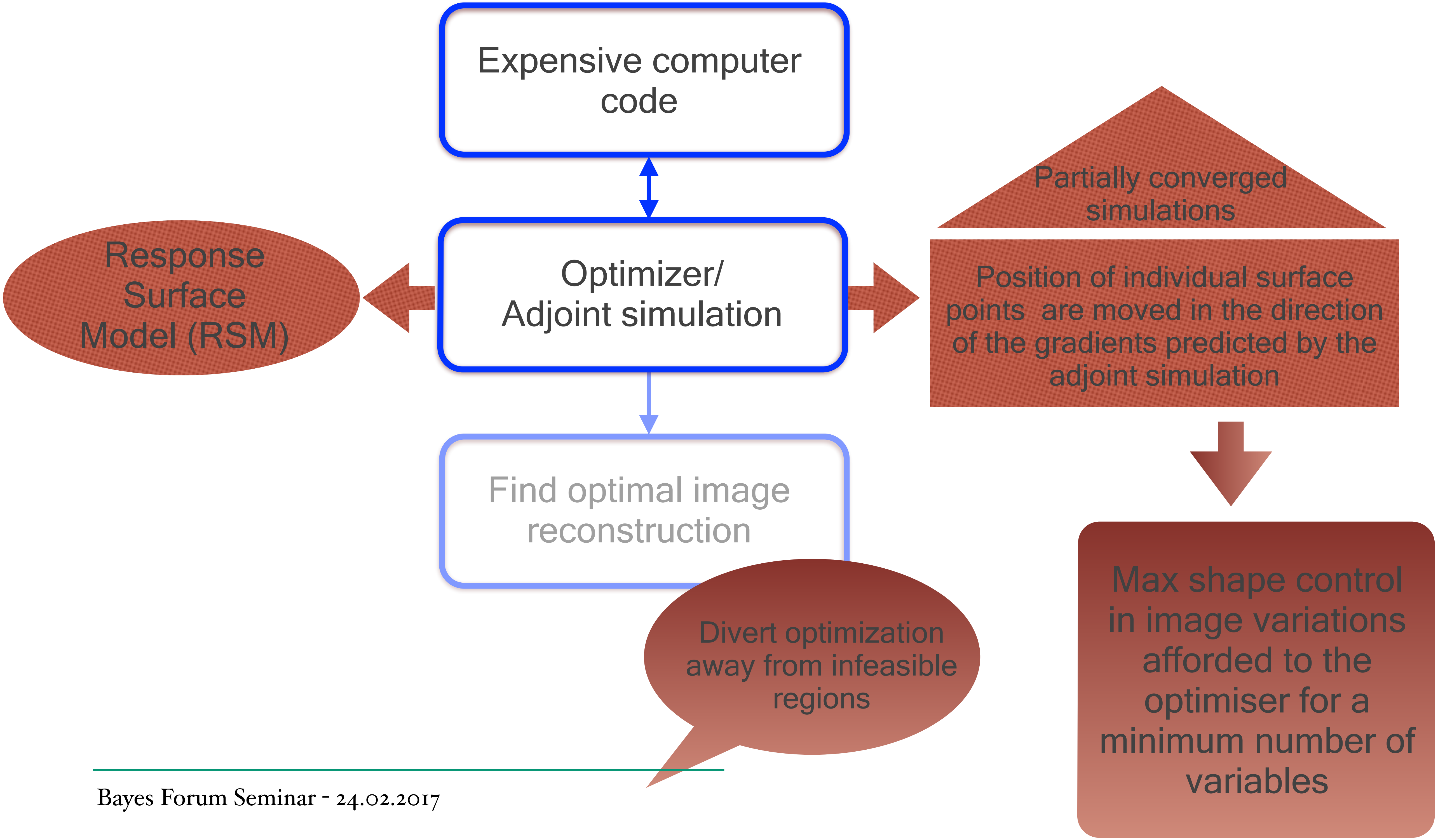
Minimization of (complex) energy function



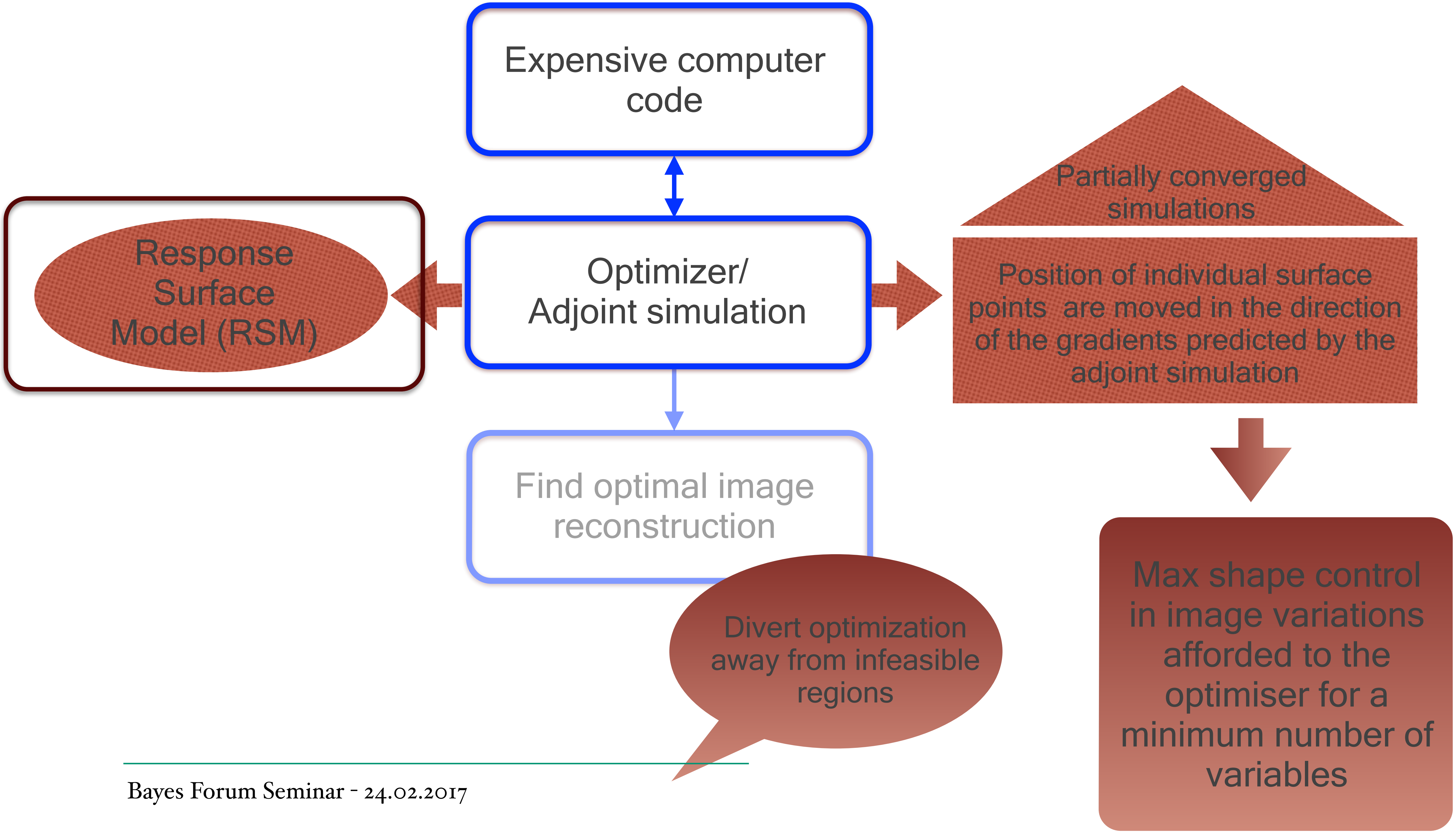
Minimization of (complex) energy function



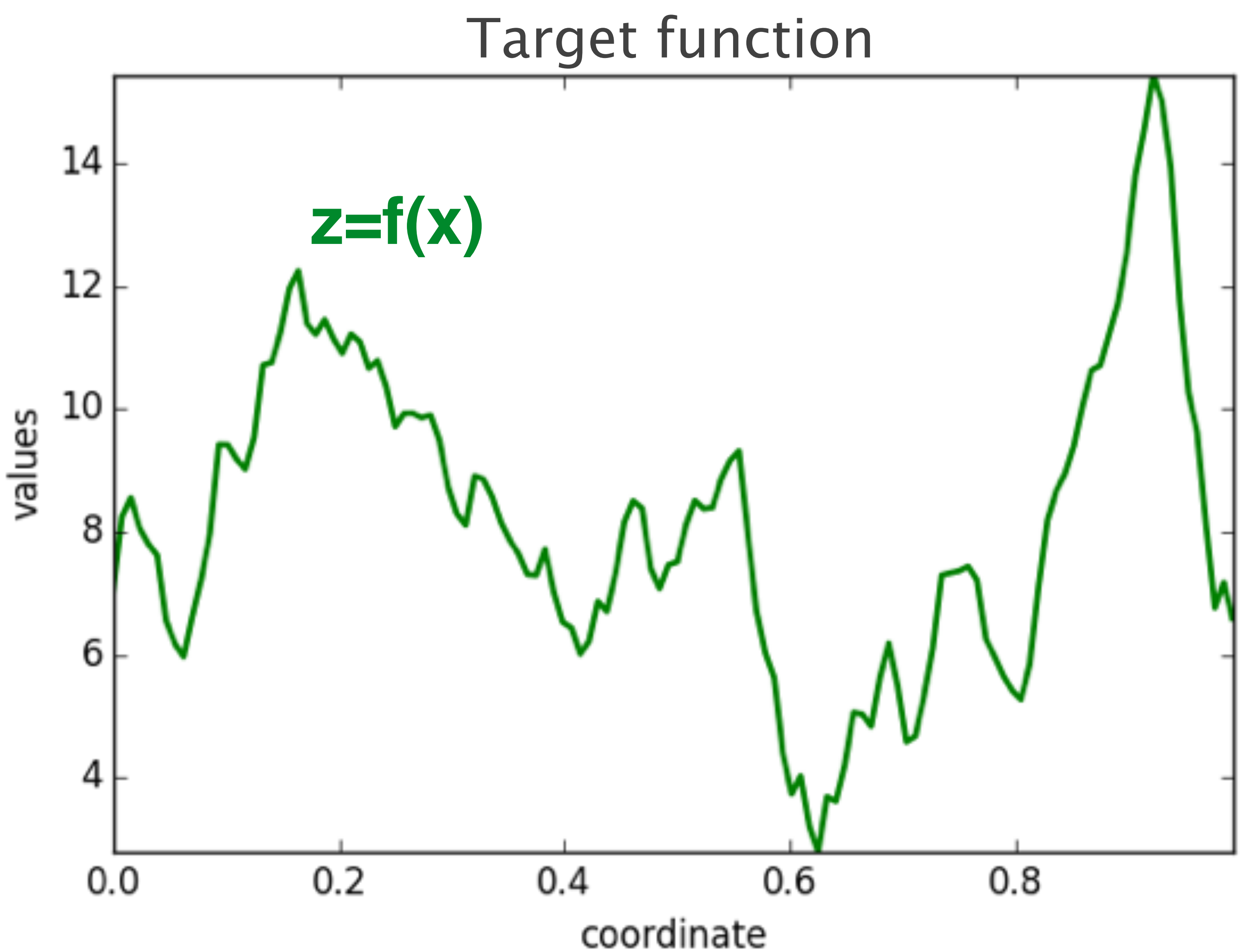
Minimization of (complex) energy function



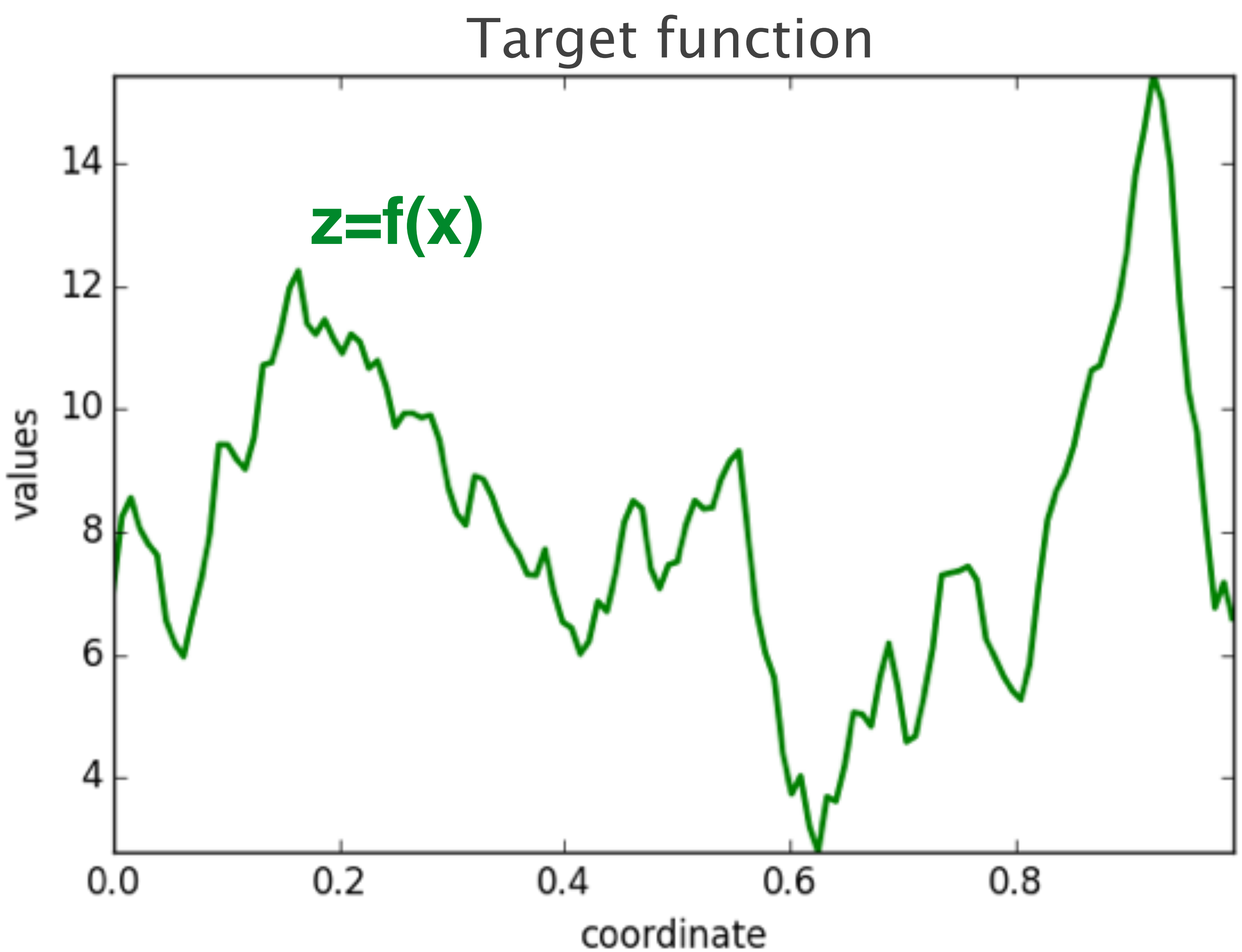
Minimization of (complex) energy function



The simulator

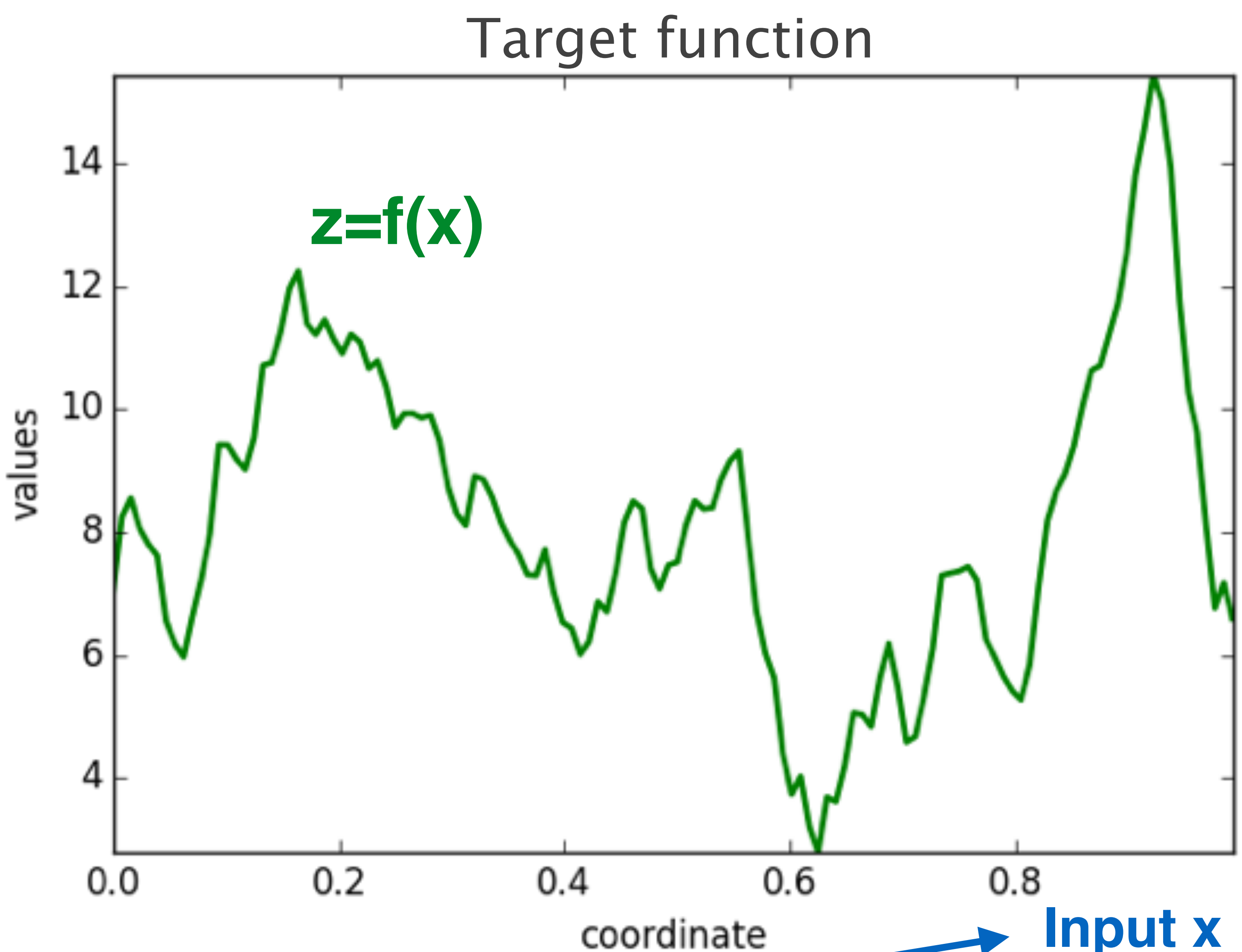


The simulator



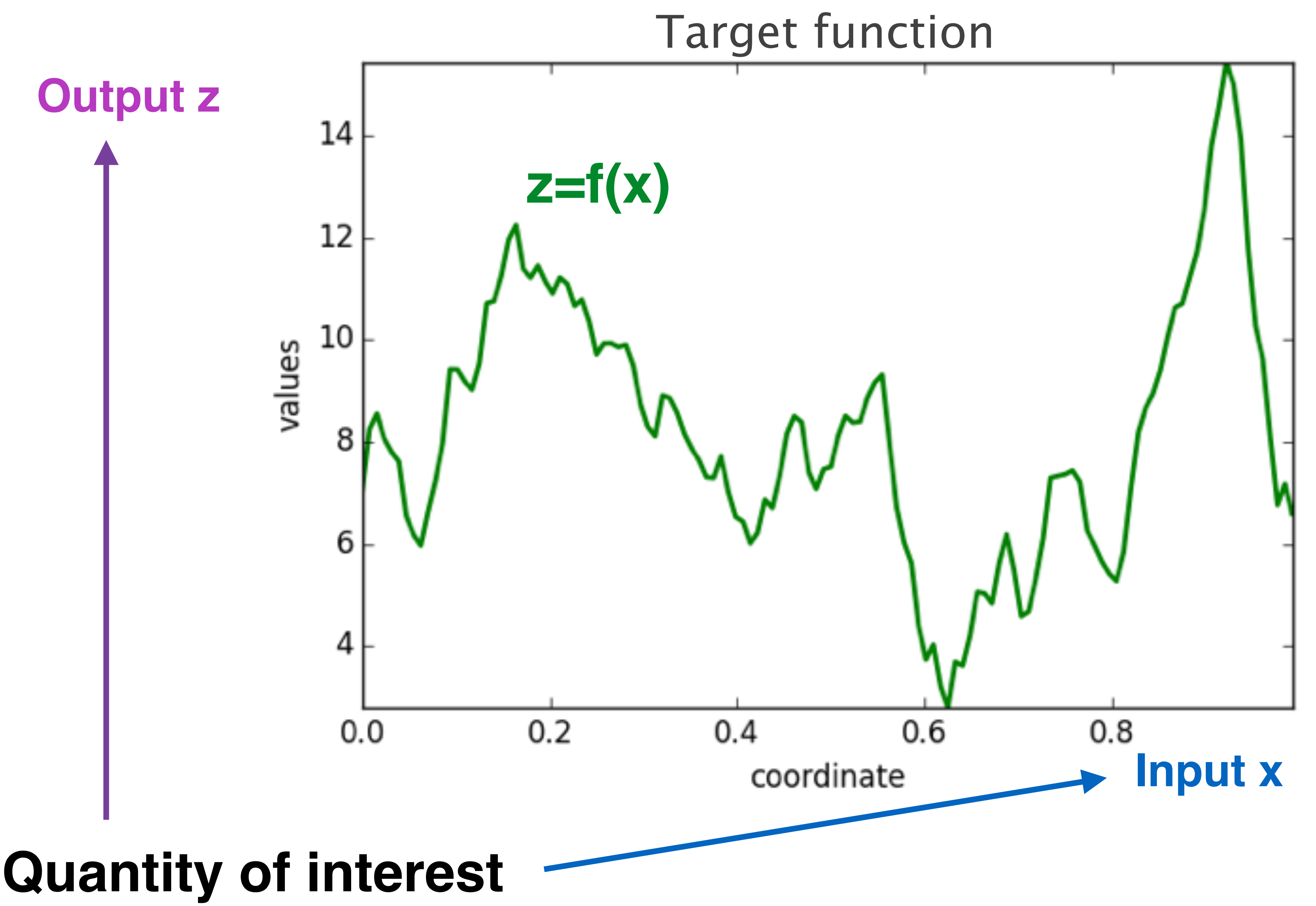
Quantity of interest

The simulator



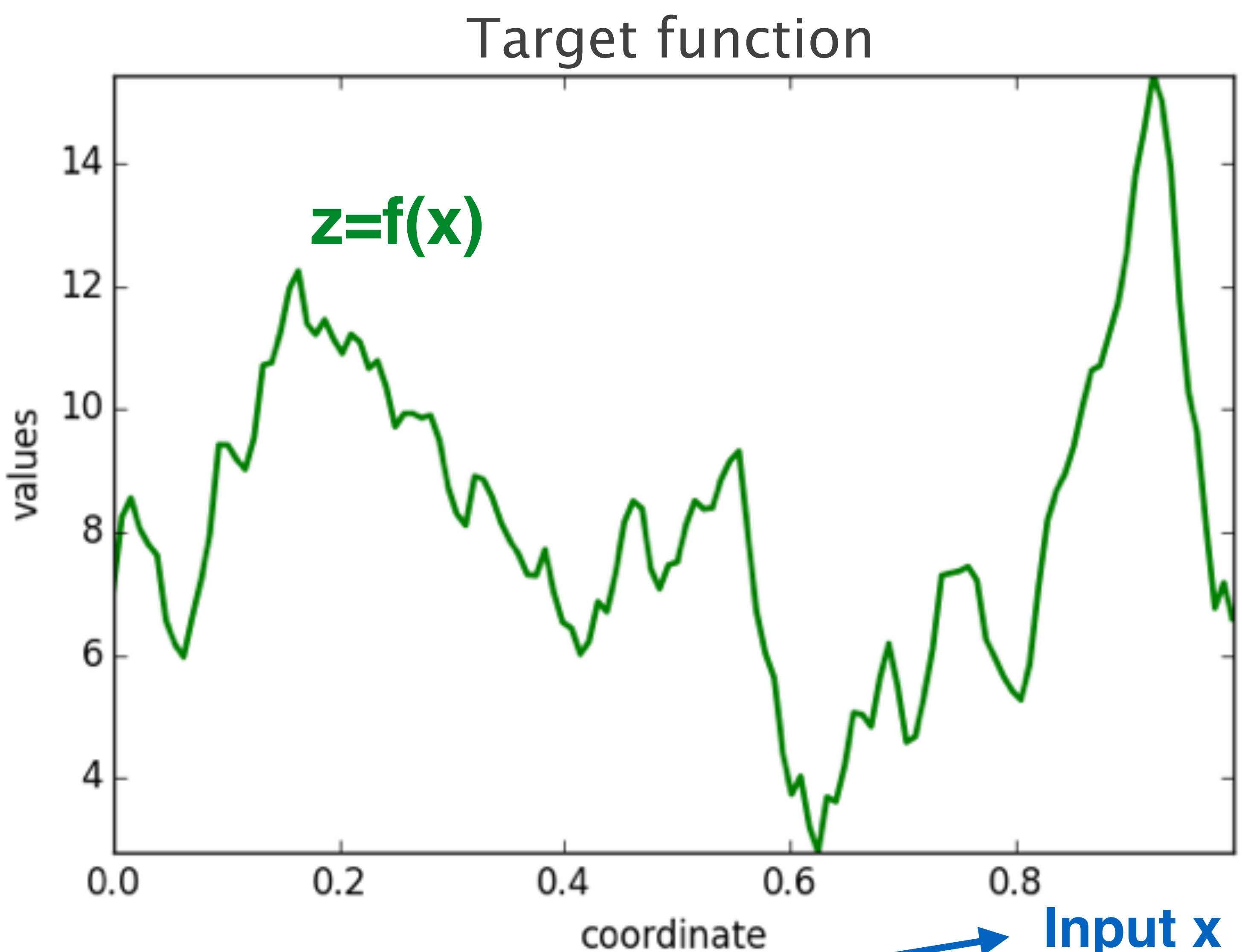
Quantity of interest

The simulator



The simulator

Output z



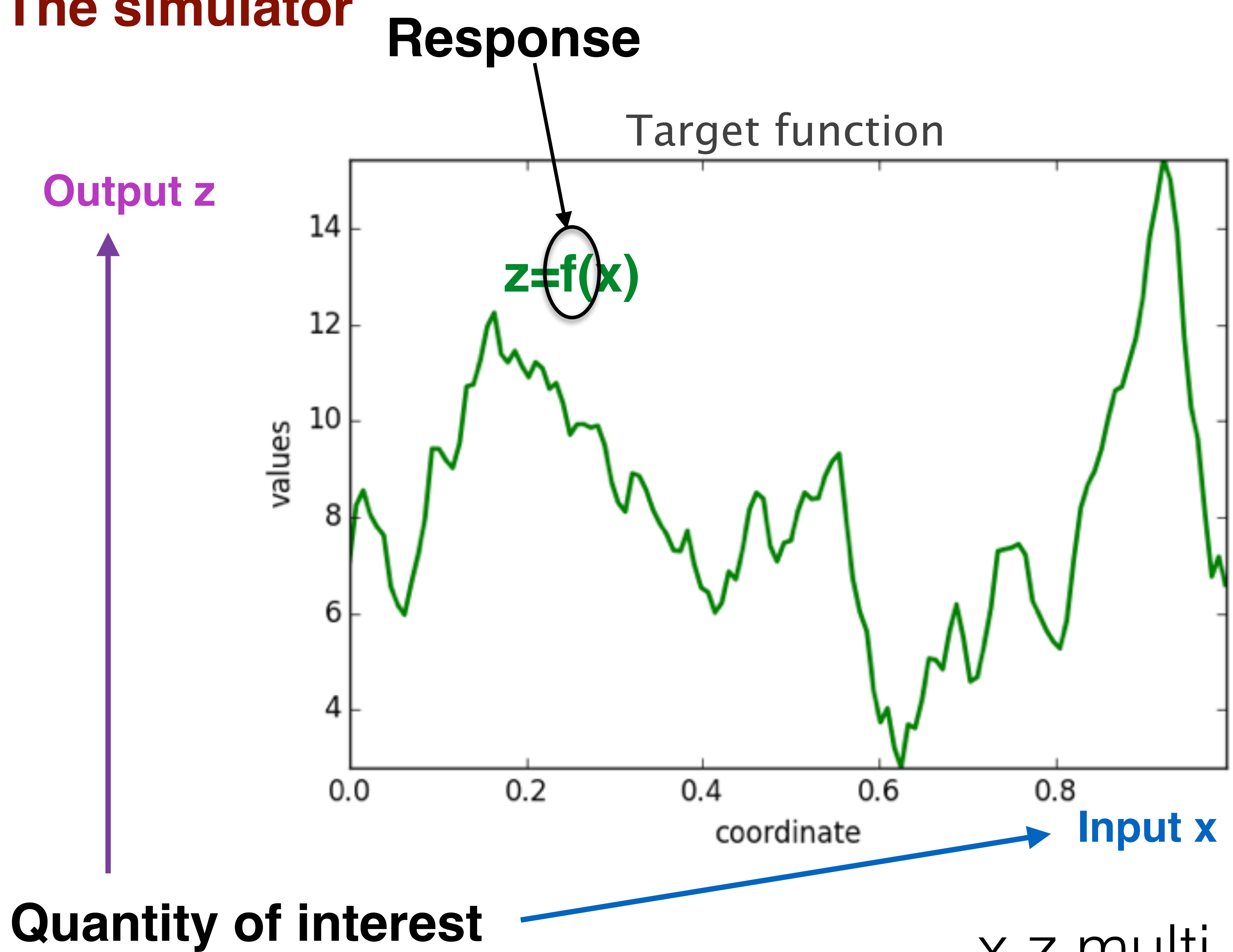
Input x

Quantity of interest

x,z multi-dimensional

Surrogate recipe

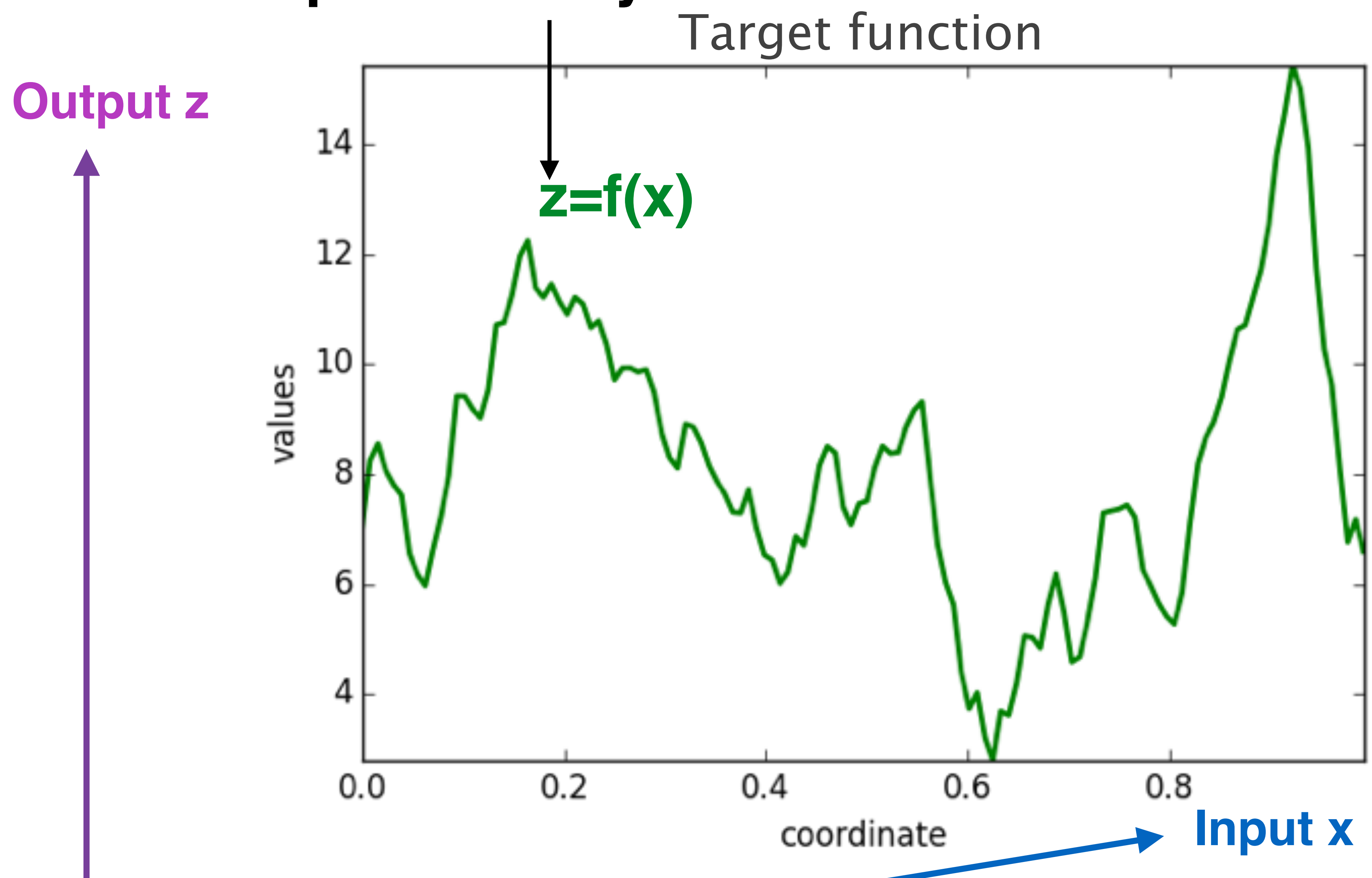
The simulator



x,z multi-dimensional

The simulator

Computationally intensive

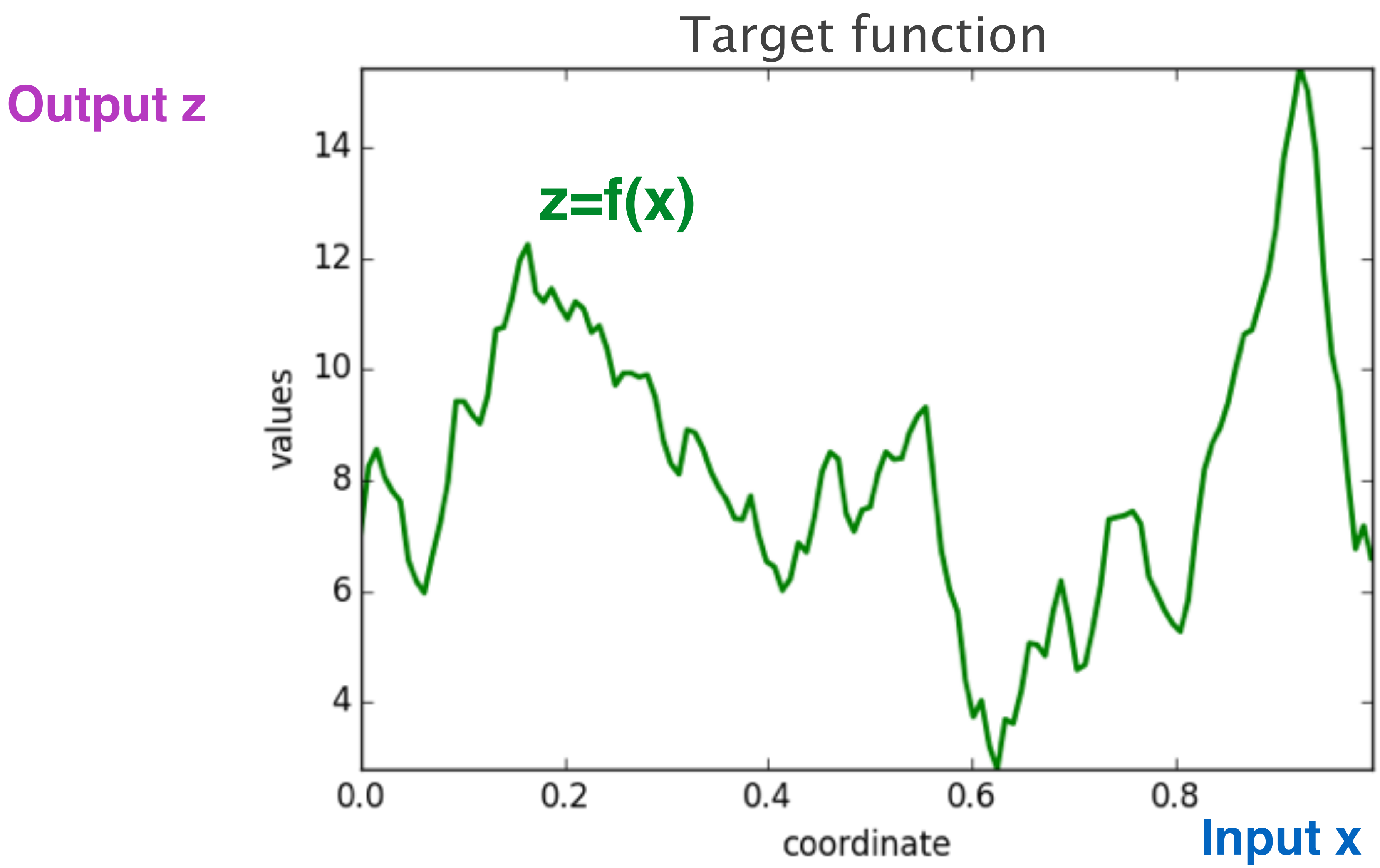


Quantity of interest

x,z multi-dimensional

Surrogate recipe

The aim is to learn about f(x) …

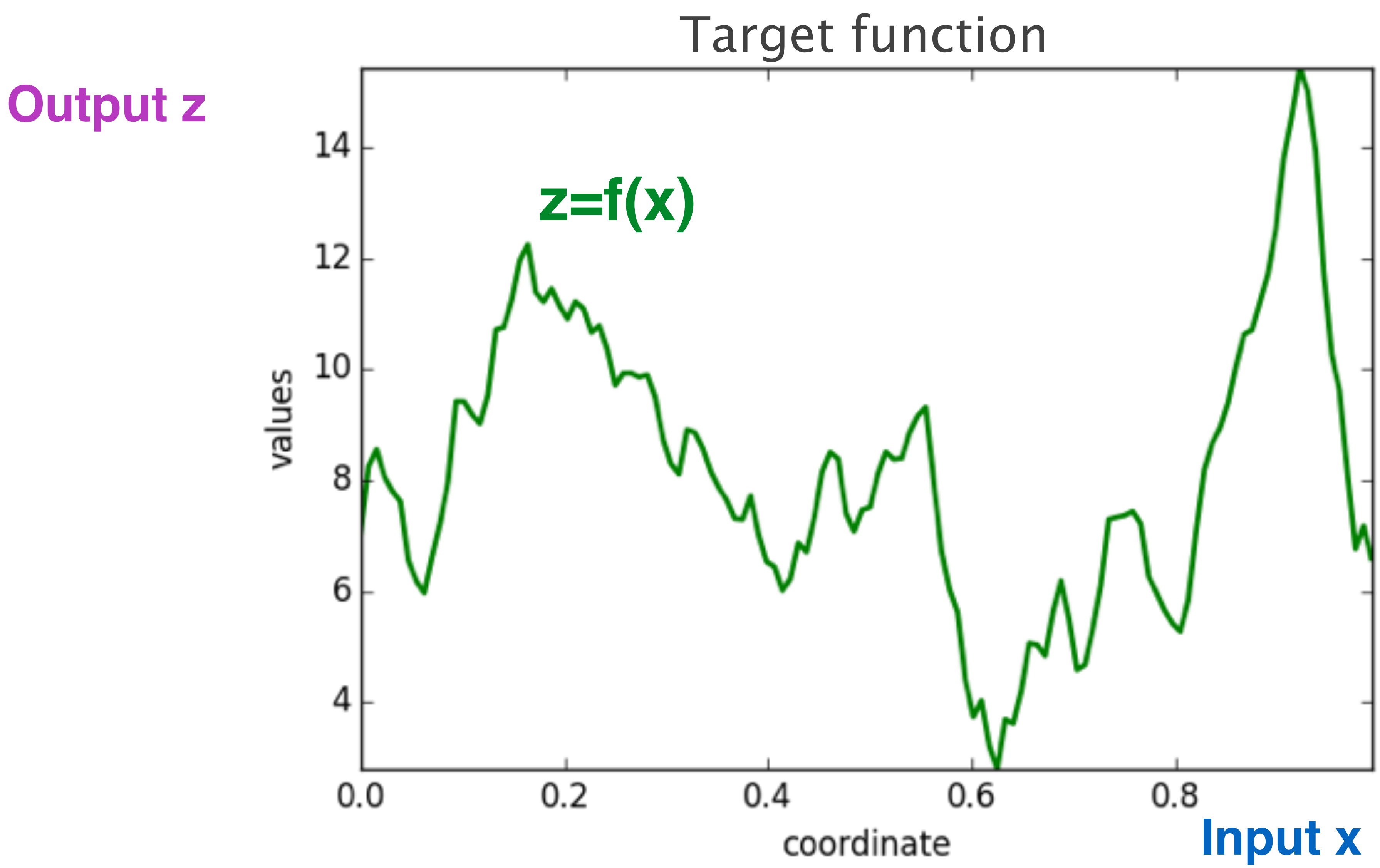


The emulator

x,z multi-dimensional

Surrogate recipe

… assume we can effort only n evaluations of f(x)

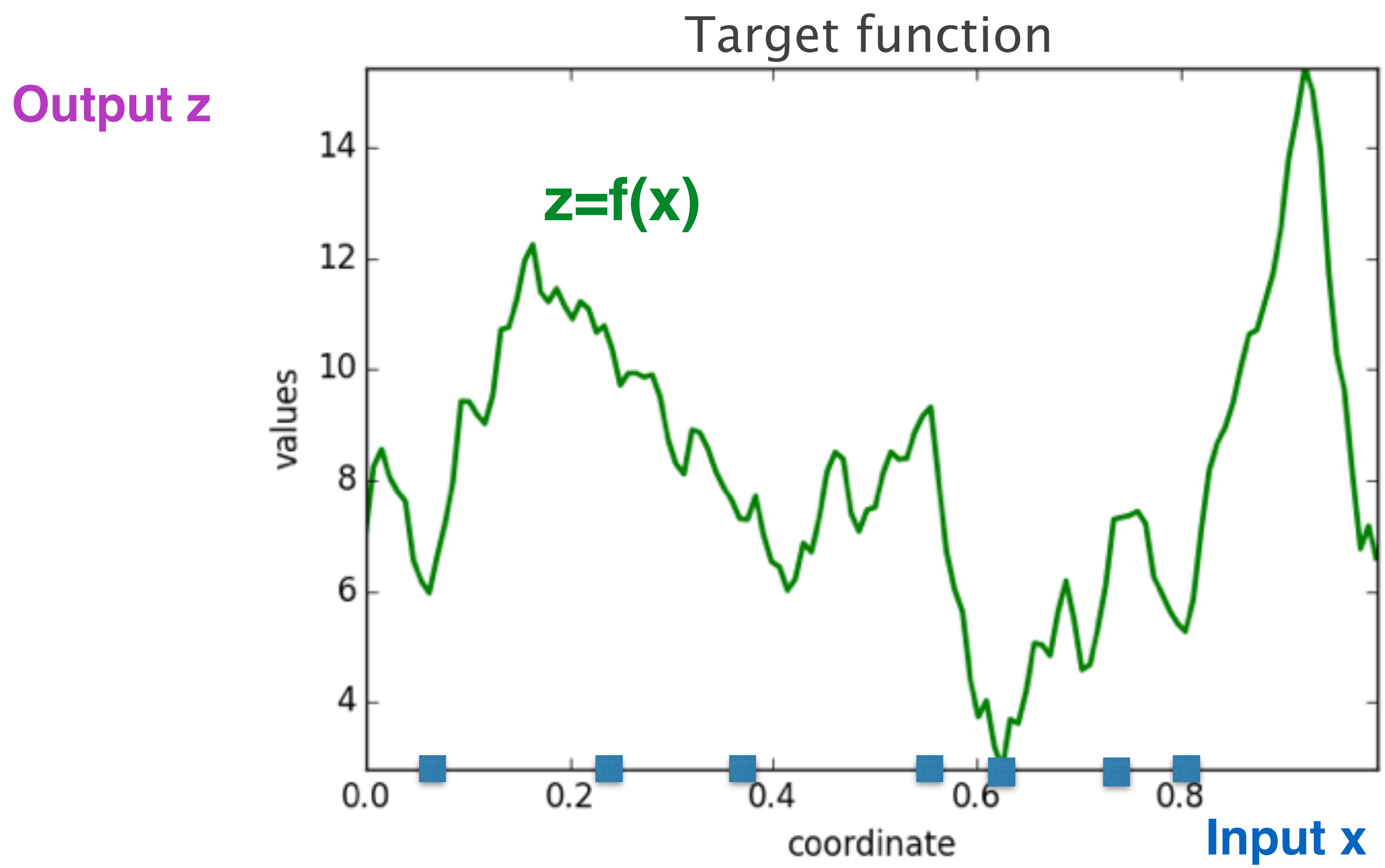


The emulator

x,z multi-dimensional

Surrogate recipe

… assume we can effort only n evaluations of f(x)



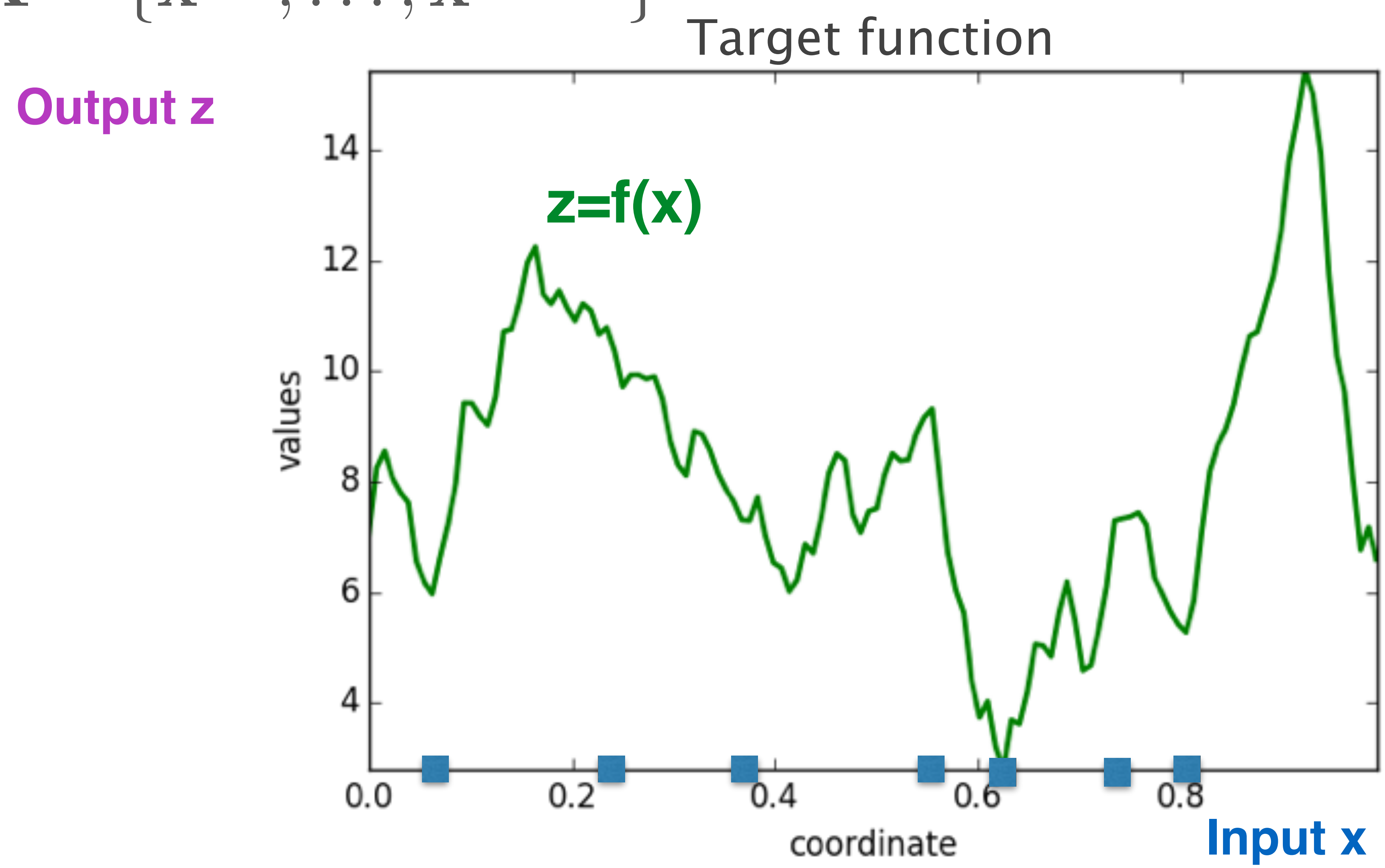
The emulator

x,z multi-dimensional

Surrogate recipe

Build a sampling plan with a set of experimental inputs:

$$\mathbf{X} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n-1)}\}^{\mathbf{T}}$$



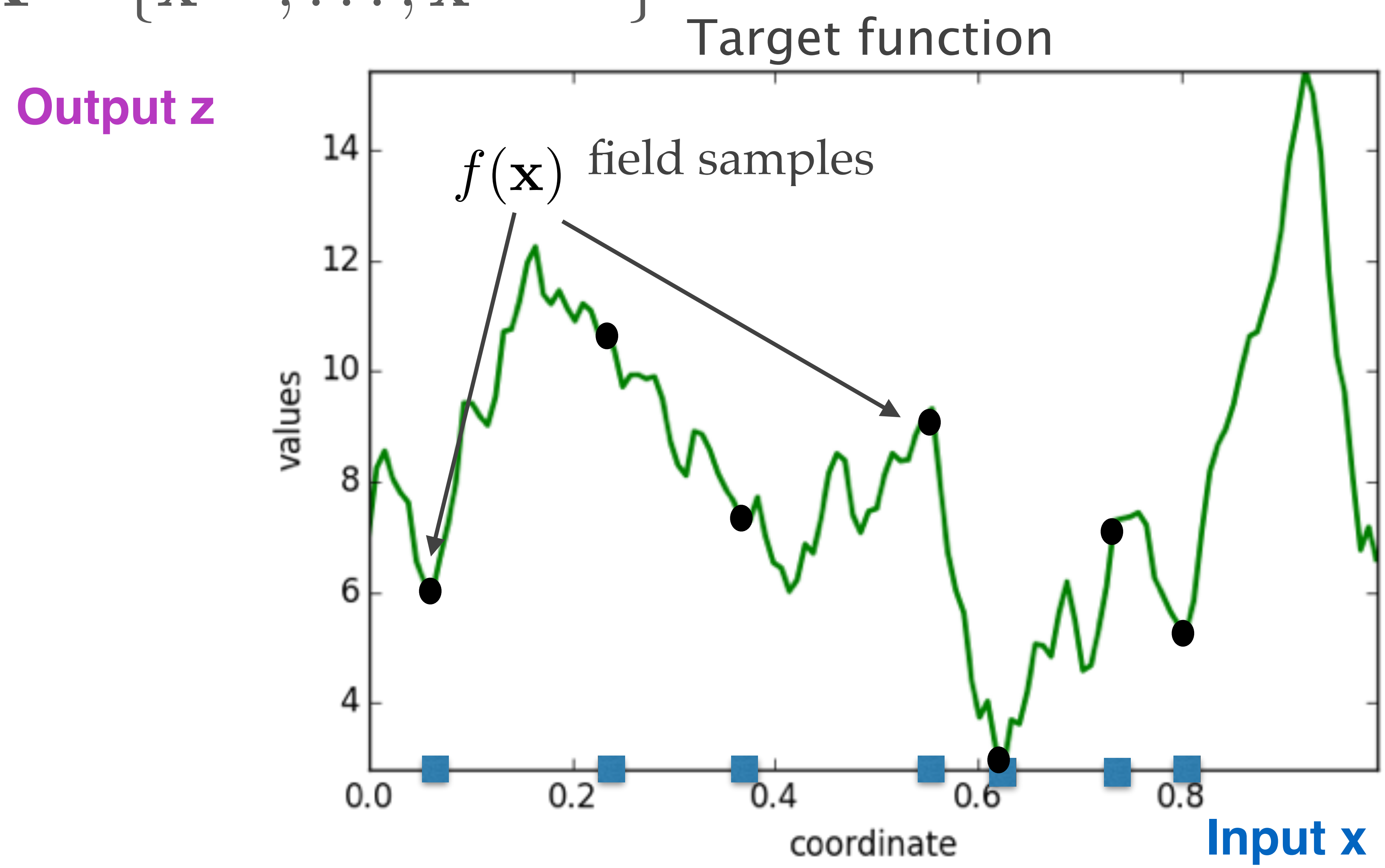
The emulator

x,z multi-dimensional

Surrogate recipe

Build a sampling plan with a set of experimental inputs:

$$\mathbf{X} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n-1)}\}^{\mathbf{T}}$$



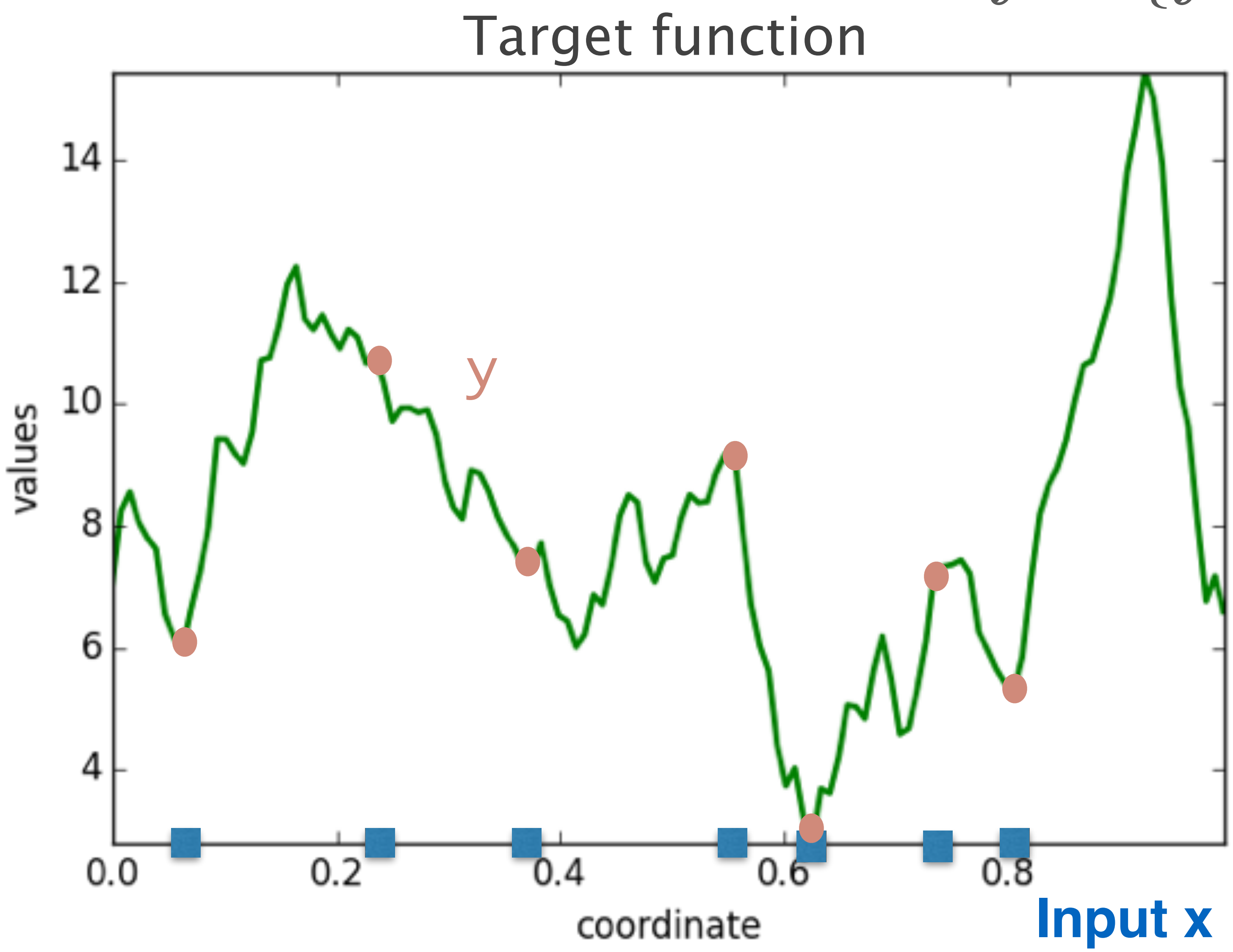
x,z multi-dimensional

The emulator

Surrogate recipe

Use measurement apparatus (numerical solver) to get the observations: i.e. calculate the responses $\mathbf{y} = \{\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(n-1)}\}^{\mathbf{T}}$

Output z



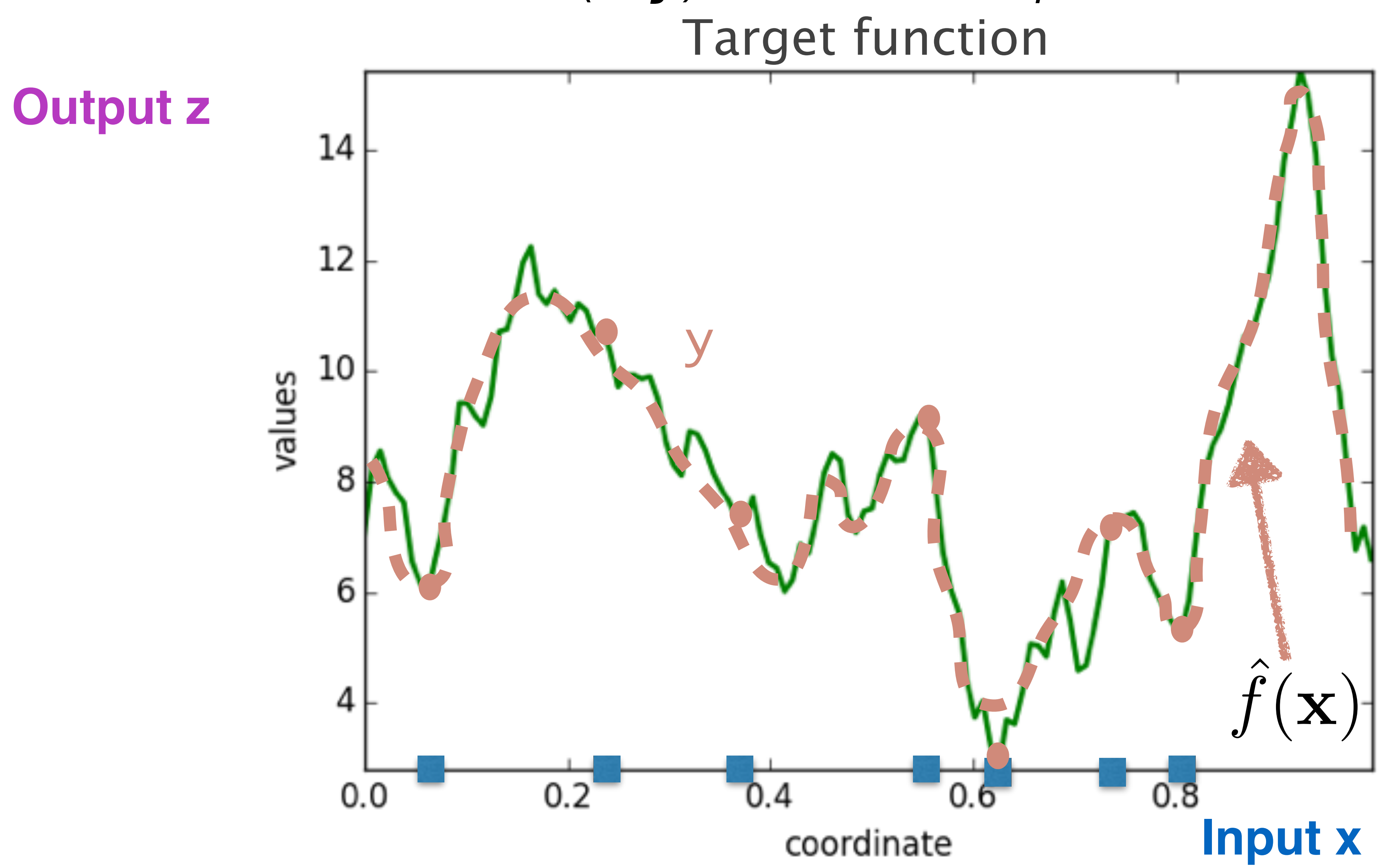
Input x

The emulator

x,z multi-dimensional

Surrogate recipe

Fit a surrogate model $\hat{f}$ to the data:
*from the observations (**x**,**y**) we make a prediction of the response*



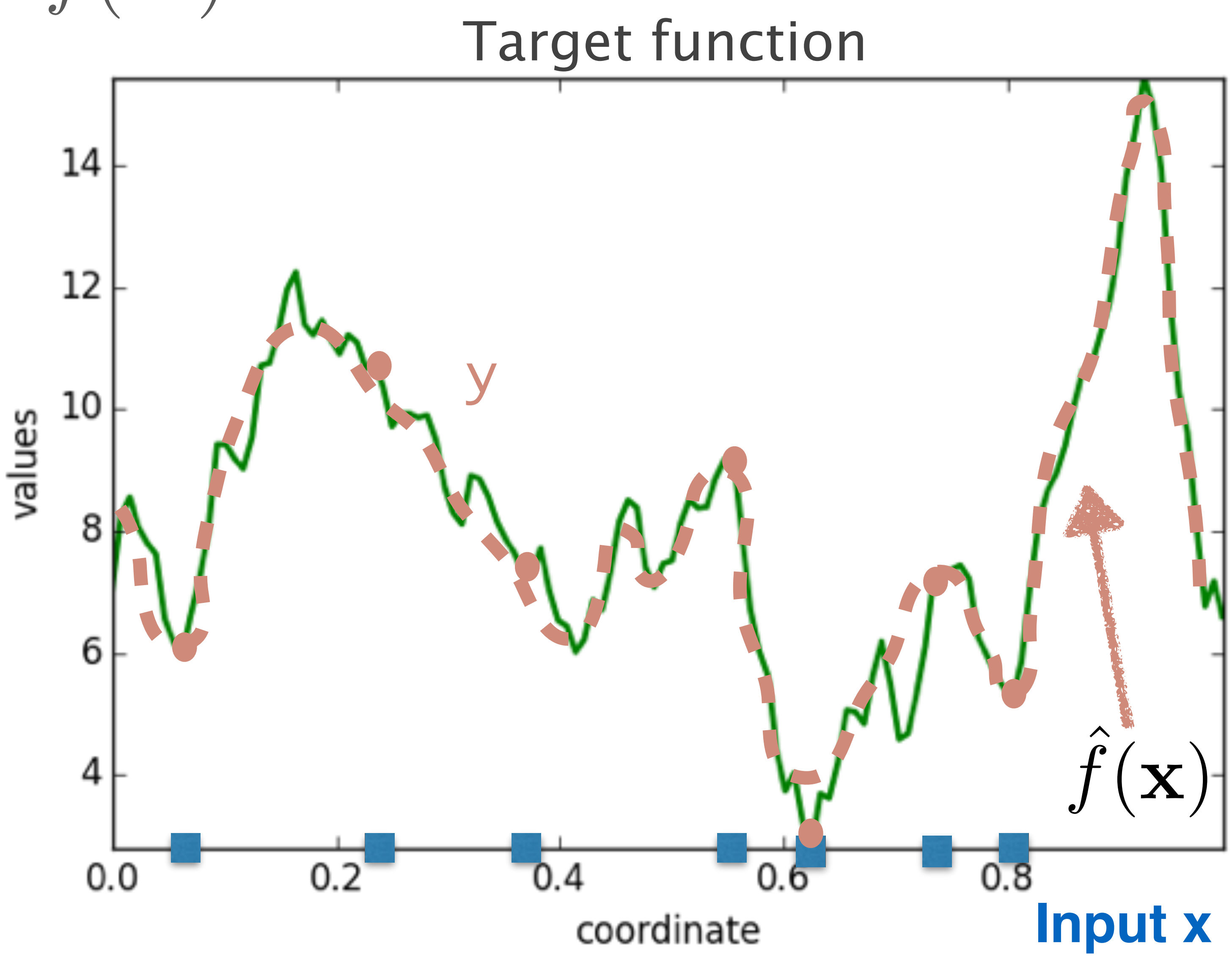
The emulator

x,z multi-dimensional

Surrogate recipe

Assume $\hat{f}$ stands in for f we can find $\mathbf{x}'$ as close as to the true minimum of $\hat{f}(\mathbf{x}')$

Output z



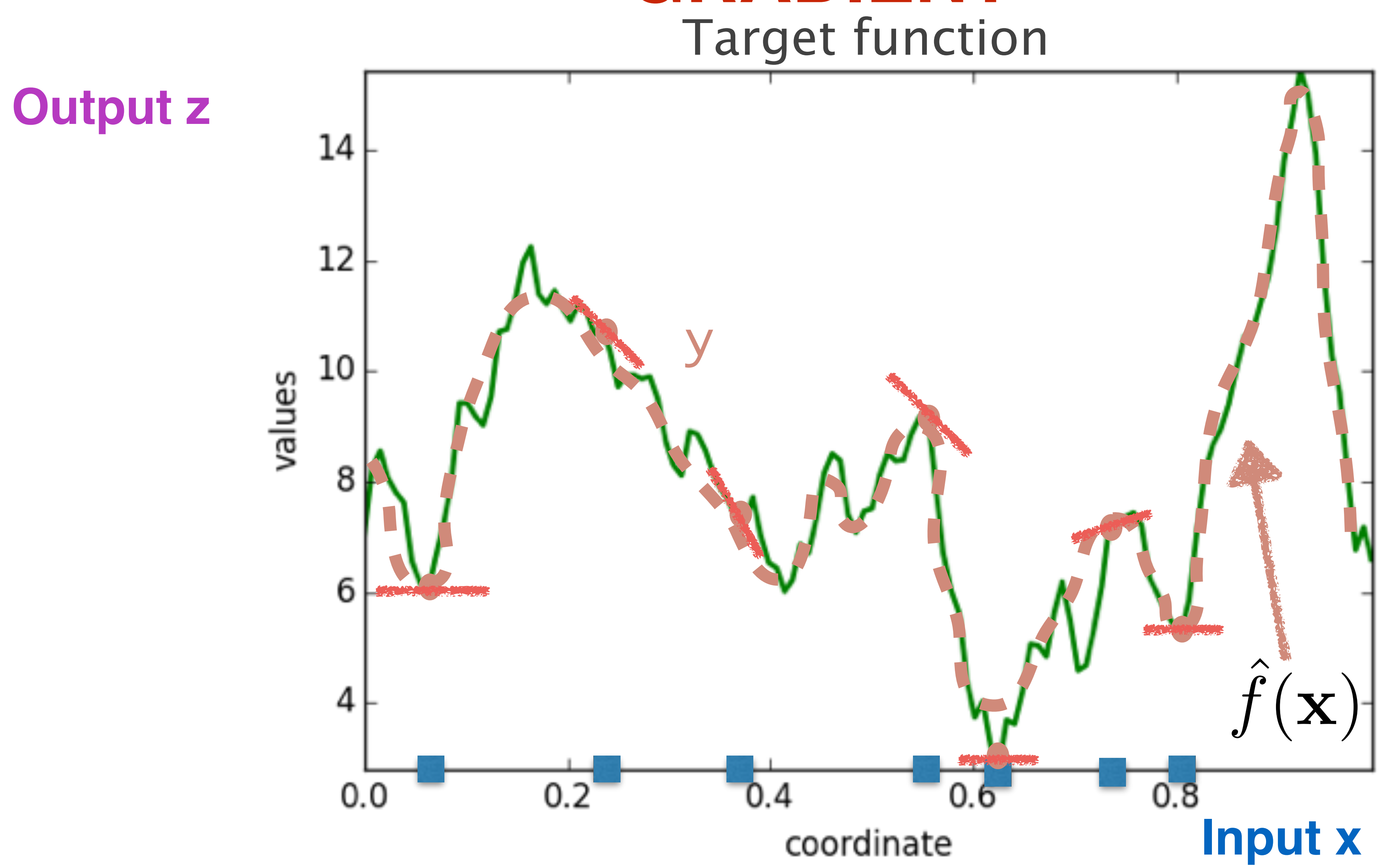
Input x

The emulator

x,z multi-dimensional

Surrogate recipe enhanced

Improve inference by including observations of the **GRADIENT**

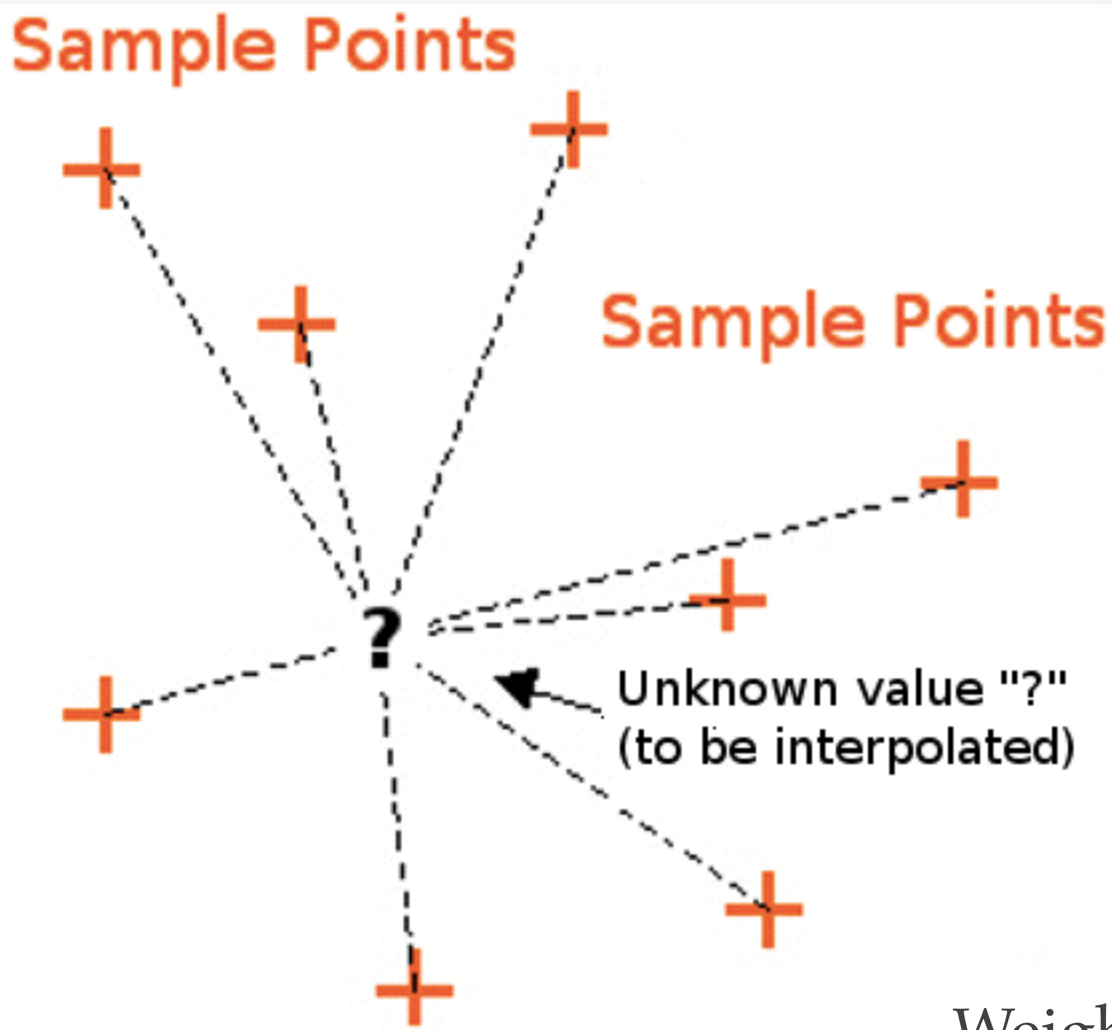


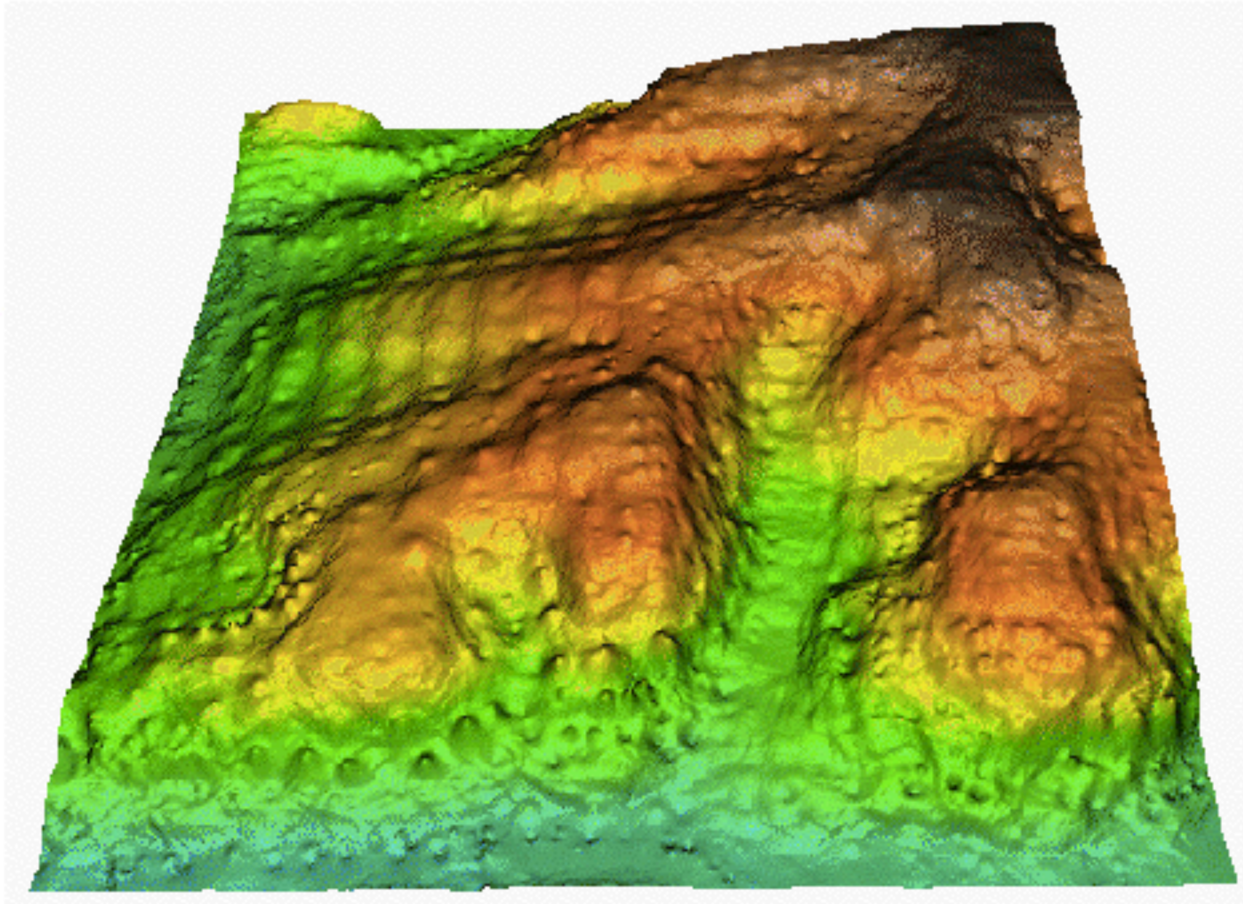
reduce computational cost of higher-dimensional optimisation problems

The emulator

x,z multi-dimensional

KRIGING WEIGHTING SCHEME



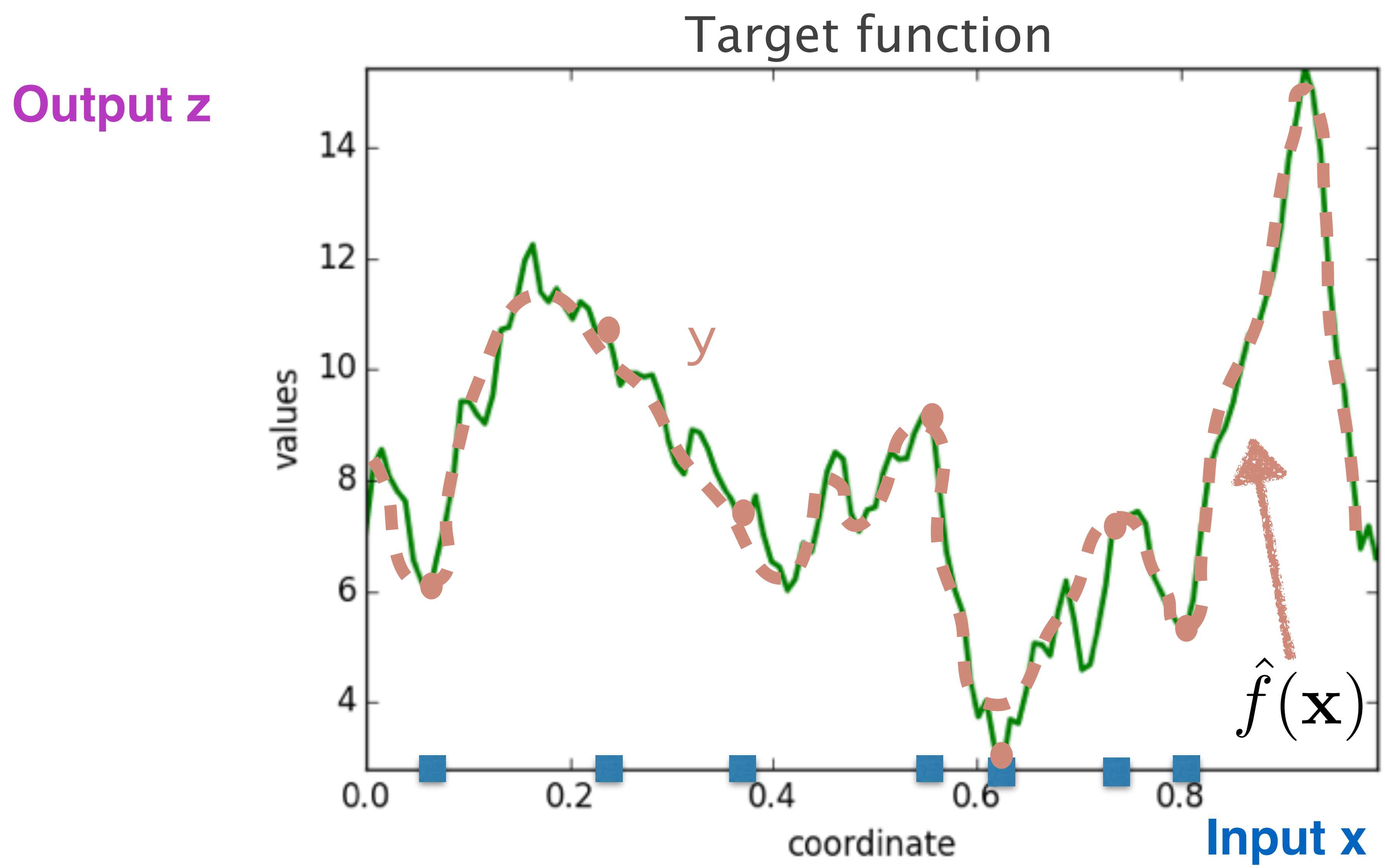


Weights are assigned to sample points according to a data driven weighting function: sample points closer to the new sample point to be predicted receive more weight

The emulator

Surrogate recipe

Validate $\hat{f}(\mathbf{x}')$ against the true expensive f : computation of $f(\mathbf{x}')$



The emulator

Surrogate challenges

1. Multimodal and multidimensional landscape

2. Surrogate function has to emulate well the real function, at least in the location of the optima

Kriging surrogate - Gaussian process

- Empirical modelling in high dimensional data:
  - y(**x**) assumed to underlie the data $\{\mathbf{x}^{(n)}, t_n\}$: adaption of model to data corresponds to inference of the function given the data
- Generalisation of a (multivariate) Gaussian distribution to a function space of infinite dimension
- Specified by mean and covariance functions $y(x) \sim GP\big(\mu(x), C(x, x')\big)$
  - mean is a function of **x** (the zero function)
  - cov is a function C(**x**,**x'**), expected covariance between the values of the function y at the points **x** and **x'**

$$P(y(x)|\mu(x), A) = \frac{1}{Z}\exp\Big[-\frac{1}{2}\big(\mathrm{y(x)} - \mu(\mathrm{x})\big)^{\mathrm{T}}\mathrm{A}\big(\mathrm{y(x)} - \mu(\mathrm{x})\big)\Big]$$

- y(**x**) lives in the infinite-dimensional space of all continuous functions of **x**

Kriging surrogate

- Gaussian process regression, Wiener-Kolmogorov prediction
- Kriging is employed within Bayesian formalism
- Assumptions:
  - prior distribution and covariance of unknown function f(x)
  - observed data (**x**) are normally distributed

    -> responses y(**x**) Gaussian process

$$P(y(\mathbf{x})|\mathbf{t}_i, \mathbf{X}_i, \mathbf{I}) = \frac{P(\mathbf{t}_i|y(\mathbf{x}), \mathbf{X}_i, \mathbf{I})P(y(\mathbf{x}))}{P(\mathbf{t}_i)}$$

Kriging surrogate

- Gaussian process regression, Wiener-Kolmogorov prediction
- Kriging is employed within Bayesian formalism
- Assumptions:
  - prior distribution and covariance of unknown function f(x)
  - observed data (**x**) are normally distributed

    -> responses y(**x**) Gaussian process

Posterior pdf of the emulated signal

$$P(y(\mathbf{x})|\mathbf{t}_i, \mathbf{X}_i, \mathbf{I}) = \frac{P(\mathbf{t}_i|y(\mathbf{x}), \mathbf{X}_i, \mathbf{I})P(y(\mathbf{x}))}{P(\mathbf{t}_i)}$$

Kriging surrogate

- Gaussian process regression, Wiener-Kolmogorov prediction
- Kriging is employed within Bayesian formalism
- Assumptions:
  - prior distribution and covariance of unknown function f(x)
  - observed data (**x**) are normally distributed

    -> responses y(**x**) Gaussian process

$$P(y(\mathbf{x})|\mathbf{t}_i, \mathbf{X}_i, \mathbf{I}) = \frac{P(\mathbf{t}_i|y(\mathbf{x}), \mathbf{X}_i, \mathbf{I})P(y(\mathbf{x}))}{P(\mathbf{t}_i)}$$

$$\mathbf{t}_n = \{t_i\}_{i=1}^{n}$$

Kriging surrogate

- Gaussian process regression, Wiener-Kolmogorov prediction
- Kriging is employed within Bayesian formalism
- Assumptions:
  - prior distribution and covariance of unknown function f(x)
  - observed data (**x**) are normally distributed

    -> responses y(**x**) Gaussian process

$$P(y(\mathbf{x})|\mathbf{t}_i, \mathbf{X}_i, \mathbf{I}) = \frac{P(\mathbf{t}_i|y(\mathbf{x}), \mathbf{X}_i, \mathbf{I})P(y(\mathbf{x}))}{P(\mathbf{t}_i)}$$

$$\mathbf{t}_n = \{t_i\}_{i=1}^{n}$$

target values

Kriging surrogate

- Gaussian process regression, Wiener-Kolmogorov prediction
- Kriging is employed within Bayesian formalism
- Assumptions:
  - prior distribution and covariance of unknown function f(x)
  - observed data (**x**) are normally distributed

    -> responses y(**x**) Gaussian process

$$P(y(\mathbf{x})|\mathbf{t}_i, \mathbf{X}_i, \mathbf{I}) = \frac{P(\mathbf{t}_i|y(\mathbf{x}), \mathbf{X}_i, \mathbf{I})P(y(\mathbf{x}))}{P(\mathbf{t}_i)} \equiv \frac{e^{-H(y(\mathbf{x}),\mathbf{t}_i)}}{Z_{t_i}}$$

Minimization challenge

$$d = Rs + n$$

Minimization challenge

$$d = Rs + n$$

$$\mathcal{G}(s, S) = \frac{1}{\sqrt{|2\pi S|}} \exp(-\frac{1}{2}\mathrm{s}^{\dagger}\mathrm{S}^{-1}\mathrm{s})$$

Minimization challenge

$$d = Rs + n$$

- $$\mathcal{G}(s,S) = \frac{1}{\sqrt{|2\pi S|}} \exp(-\frac{1}{2} \mathrm{s}^{\dagger} \mathrm{S}^{-1} \mathrm{s})$$
- $$H(d|s) = -\ln P(d|s) = \frac{1}{2}(d - Rs)^{\dagger} N^{-1} (d - Rs)$$

Minimization challenge

$$d = Rs + n$$

- $\mathcal{G}(s, S) = \frac{1}{\sqrt{|2\pi S|}} \exp(-\frac{1}{2} \mathrm{s}^{\dagger} \mathrm{S}^{-1} \mathrm{s})$

- $H(d|s) = -\ln P(d|s) = \frac{1}{2}(d - Rs)^{\dagger} N^{-1} (d - Rs)$

- $H(d, s) = -\mathrm{lnP}(\mathrm{d}, \mathrm{s}) \;\widehat{=}\; \frac{1}{2} \mathrm{s}^{\dagger} \mathrm{S}^{-1} \mathrm{s} + \mathrm{H}(\mathrm{d}|\mathrm{s})$

Minimization challenge

$$d = Rs + n$$

- $$\mathcal{G}(s, S) = \frac{1}{\sqrt{|2\pi S|}} \exp(-\frac{1}{2}\mathrm{s}^{\dagger}\mathrm{S}^{-1}\mathrm{s})$$
- $$H(d|s) = -\ln P(d|s) = \frac{1}{2}(d - Rs)^{\dagger} N^{-1} (d - Rs)$$
- $$H(d, s) = -\mathrm{lnP(d, s)} \widehat{=} \frac{1}{2}\mathrm{s}^{\dagger}\mathrm{S}^{-1}\mathrm{s} + \mathrm{H(d|s)}$$

$$m = arg\, \underset{\mathrm{s}}{min}\, H(d, s)$$

H and dH/ds are calculated at several positions to slide towards the minimum -> computationally expensive

Surrogate minimization

$$d = Rs + n$$

$$x = S^{-1/2} s$$

- $H(x) = H(d, s = S^{1/2} x) \widehat{=} \frac{1}{2} x^\dagger x + \underbrace{H(d|s = S^{\frac{1}{2}} x)}_{E(x)}$

Surrogate minimization

$$d = Rs + n$$

$$x = S^{-1/2}s$$

- $H(x) = H(d, s = S^{1/2}x) \widehat{=} \frac{1}{2}x^{\dagger}x + \underbrace{H(d|s = S^{\frac{1}{2}}x)}_{E(x)}$

Surrogate minimization

$$d = Rs + n$$

$$x = S^{-1/2} s$$

- $H(x) = H(d, s = S^{1/2} x) \widehat{=} \frac{1}{2} x^\dagger x + \underbrace{H(d|s = S^{\frac{1}{2}} x)}$

  $\boldsymbol{E(x)} \approx F_{I_n}(x)$

- $H(x) \widehat{=} \frac{1}{2} x^\dagger x + F_{I_n}(x) = H_{I_n}(x)$

Surrogate minimization

$$d = Rs + n$$

$$x = S^{-1/2}s$$

- $H(x) = H(d, s = S^{1/2}x) \widehat{=} \frac{1}{2}x^\dagger x + \underbrace{H(d|s = S^{\frac{1}{2}}x)}$
  $\boldsymbol{E}(\boldsymbol{x}) \approx F_{I_n}(x)$
- $H(x) \widehat{=} \frac{1}{2}x^\dagger x + F_{I_n}(x) = H_{I_n}(x)$

Surrogate minimization

$$d = Rs + n$$

$$x = S^{-1/2} s$$

- $H(x) = H(d, s = S^{1/2} x) \widehat{=} \frac{1}{2} x^\dagger x + \underbrace{H(d|s = S^{\frac{1}{2}} x)}_{\boldsymbol{E}(\boldsymbol{x}) \approx F_{I_n}(x)}$
- $H(x) \widehat{=} \frac{1}{2} x^\dagger x + F_{I_n}(x) = H_{I_n}(x)$

$$I_n = (x_i, E_i = E(x_i), v_i = -\partial_x E(x)|_{x_i})_{i=1}^{n}$$

Surrogate minimization

$$d = Rs + n$$

$$x = S^{-1/2} s$$

- $H(x) = H(d, s = S^{1/2} x) \widehat{=} \frac{1}{2} x^\dagger x + \underbrace{H(d|s = S^{\frac{1}{2}} x)}_{\boldsymbol{E}(\boldsymbol{x}) \approx F_{I_n}(x)}$

- $H(x) \widehat{=} \frac{1}{2} x^\dagger x + F_{I_n}(x) = H_{I_n}(x)$

$$I_n = (x_i, E_i = E(x_i), v_i = -\partial_x E(x)|_{x_i})_{i=1}^n$$

$$F_{I_n}(x) = \sum_{i=1}^{n} w_i(x) \begin{cases} E_i e^{\frac{-v_i^\dagger (x - x_i)}{E_i}} & \text{if } v_i^\dagger (x - x_i) > 0 \\ E_i - v_i^\dagger (x - x_i) & \text{if } v_i^\dagger (x - x_i) \leq 0 \end{cases}$$

Surrogate minimization

$$d = Rs + n$$

$$x = S^{-1/2} s$$

- $H(x) = H(d, s = S^{1/2} x) \widehat{=} \frac{1}{2} x^\dagger x + \underbrace{H(d|s = S^{\frac{1}{2}} x)}_{\boldsymbol{E(x)} \approx F_{I_n}(x)}$

- $H(x) \widehat{=} \frac{1}{2} x^\dagger x + F_{I_n}(x) = H_{I_n}(x)$

$$I_n = (x_i, E_i = E(x_i), v_i = -\partial_x E(x)|_{x_i})_{i=1}^n$$

$$F_{I_n}(x) = \sum_{i=1}^{n} w_i(x) \begin{cases} E_i e^{\frac{-v_i^\dagger (x - x_i)}{E_i}} & \text{if } v_i^\dagger (x - x_i) > 0 \\ E_i - v_i^\dagger (x - x_i) & \text{if } v_i^\dagger (x - x_i) \leq 0 \end{cases}$$

Surrogate minimization

$$d = Rs + n$$

$$x = S^{-1/2} s$$

- $H(x) = H(d, s = S^{1/2}x) \widehat{=} \frac{1}{2} x^\dagger x + \underbrace{H(d|s = S^{\frac{1}{2}}x)}_{\boldsymbol{E}(\boldsymbol{x}) \approx F_{I_n}(x)}$

- $H(x) \widehat{=} \frac{1}{2} x^\dagger x + F_{I_n}(x) = H_{I_n}(x)$

$$I_n = (x_i, E_i = E(x_i), v_i = -\partial_x E(x)|_{x_i})_{i=1}^n$$

$$F_{I_n}(x) = \sum_{i=1}^{n} w_i(x) \begin{cases} E_i e^{\frac{-v_i^\dagger (x - x_i)}{E_i}} & \text{if } v_i^\dagger (x - x_i) > 0 \\ E_i - v_i^\dagger (x - x_i) & \text{if } v_i^\dagger (x - x_i) \leq 0 \end{cases}$$

$$H(d|s) = \frac{1}{2} (d - Rs)^\dagger N^{-1} (d - Rs)$$

(13)

Surrogate minimization

$$d = Rs + n$$

$$x = S^{-1/2} s$$

- $H(x) = H(d, s = S^{1/2}x) \widehat{=} \frac{1}{2} x^\dagger x + \underbrace{H(d|s = S^{\frac{1}{2}}x)}_{E(x) \approx F_{I_n}(x)}$

- $H(x) \widehat{=} \frac{1}{2} x^\dagger x + F_{I_n}(x) = H_{I_n}(x)$

$$I_n = (x_i, E_i = E(x_i), v_i = -\partial_x E(x)|_{x_i})_{i=1}^n$$

$$F_{I_n}(x) = \sum_{i=1}^n w_i(x) \begin{cases} E_i e^{\frac{-v_i^\dagger (x - x_i)}{E_i}} & \text{if } v_i^\dagger (x - x_i) > 0 \\ E_i - v_i^\dagger (x - x_i) & \text{if } v_i^\dagger (x - x_i) \leq 0 \end{cases}$$

$$w_i(x) = \frac{(|x - x_i|^2 + \epsilon^2)^{-\frac{\alpha}{2}}}{\sum_{j=1}^n (|x - x_j|^2 + \epsilon^2)^{-\frac{\alpha}{2}}}$$

Surrogate minimization

$$d = Rs + n$$

$$x = S^{-1/2} s$$

- $H(x) = H(d, s = S^{1/2} x) \widehat{=} \frac{1}{2} x^\dagger x + \underbrace{H(d|s = S^{\frac{1}{2}} x)}_{E(x)}$

- $H(x) \widehat{=} \frac{1}{2} x^\dagger x + F_{I_n}(x) = H_{I_n}(x)$

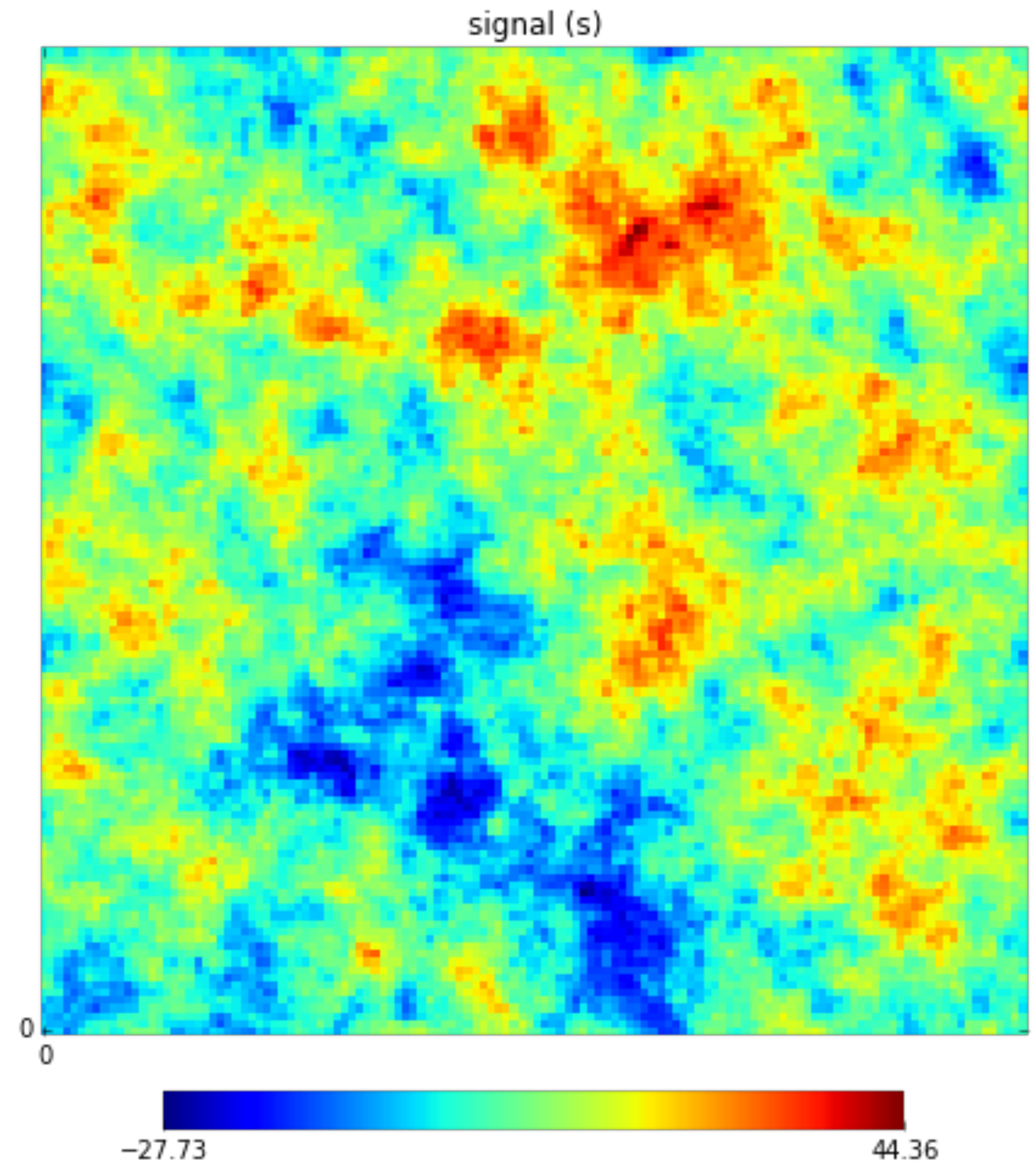
- $\partial_x H_{I_n}(x) = x - \sum_{i=1}^{n} w_i(x) \begin{cases} \alpha A E_i e^{\frac{-v_i^\dagger (x - x_i)}{E_i}} + v_i e^{-\frac{v_i^\dagger (x - x_i)}{E_i}} & \text{if } v_i^\dagger (x - x_i) > 0 \\ \alpha A \big[E_i - v_i^\dagger (x - x_i) \big] + v_i & \text{if } v_i^\dagger (x - x_i) \leq 0 \end{cases}$

where $A = [\frac{x - x_i}{(|x - x_i|^2 + \epsilon^2)^\alpha} - \sum_j w_j(x) \frac{x - x_j}{(|x - x_j|^2 + \epsilon^2)^\alpha}]$

Application

Simulated sky signal employing NIFTY package

1.6E+4 dimensions



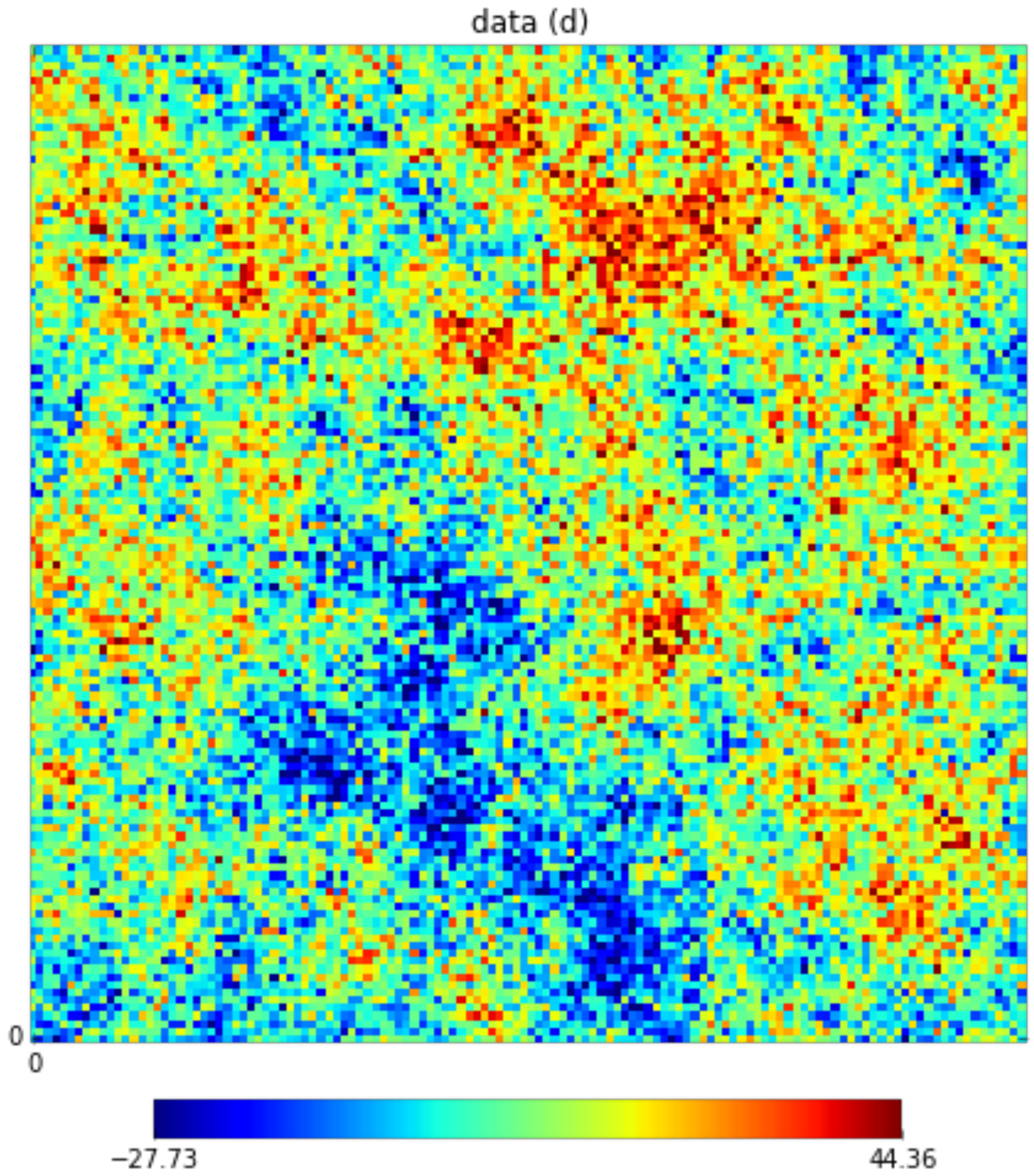
NIFTY= Numerical Information Field Theory

(M.Selig, T. Enßlin et al.)

Application

Simulated dataset employing NIFTY package

$$d = R(s) + n$$



NIFTY= Numerical Information Field Theory

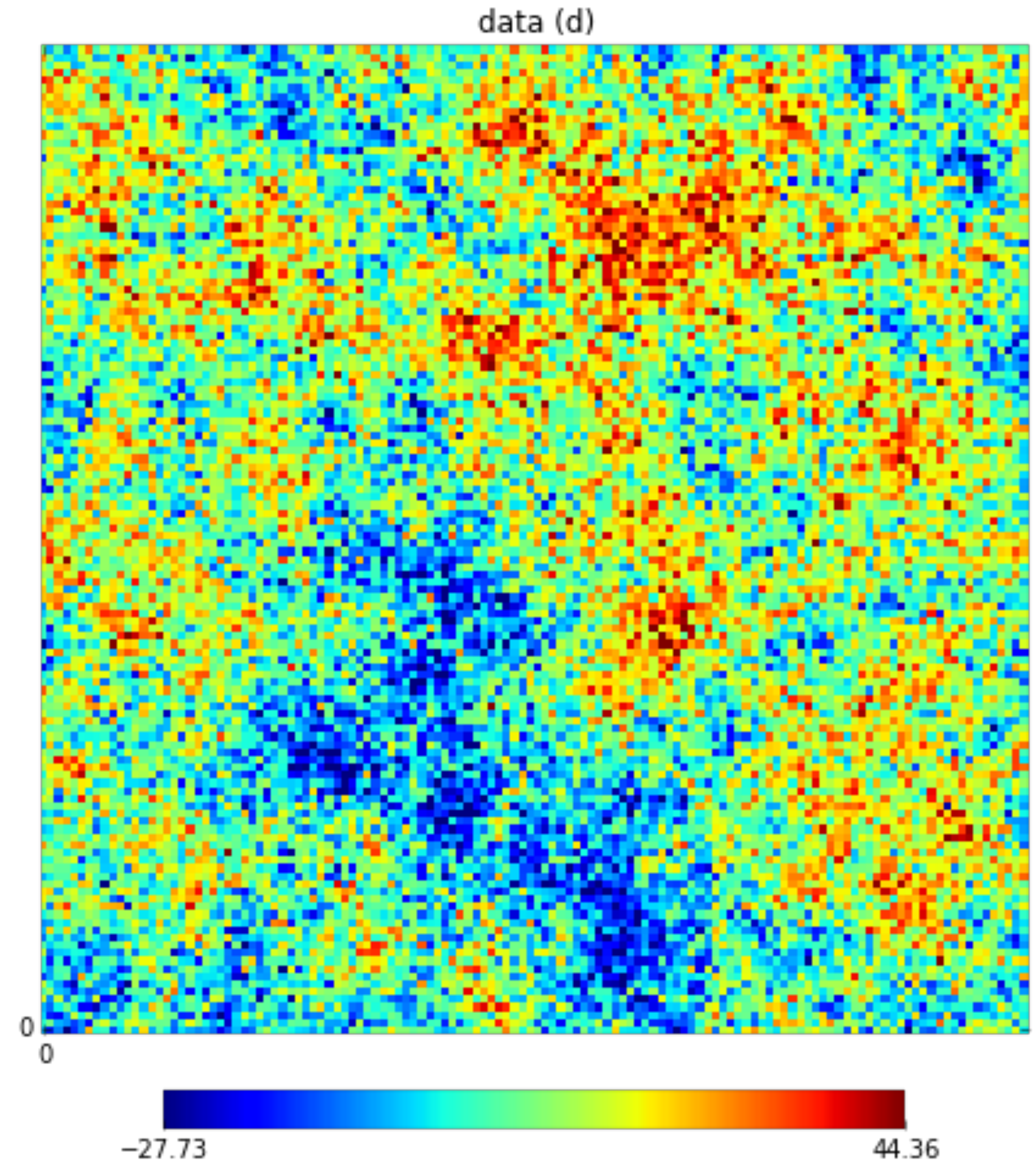
(M.Selig, T. Enßlin et al.)

Application

Simulated dataset employing NIFTY package

$d = R(s) + n$

find optimal solution for s given d



NIFTY= Numerical Information Field Theory

(M.Selig, T. Enßlin et al.)

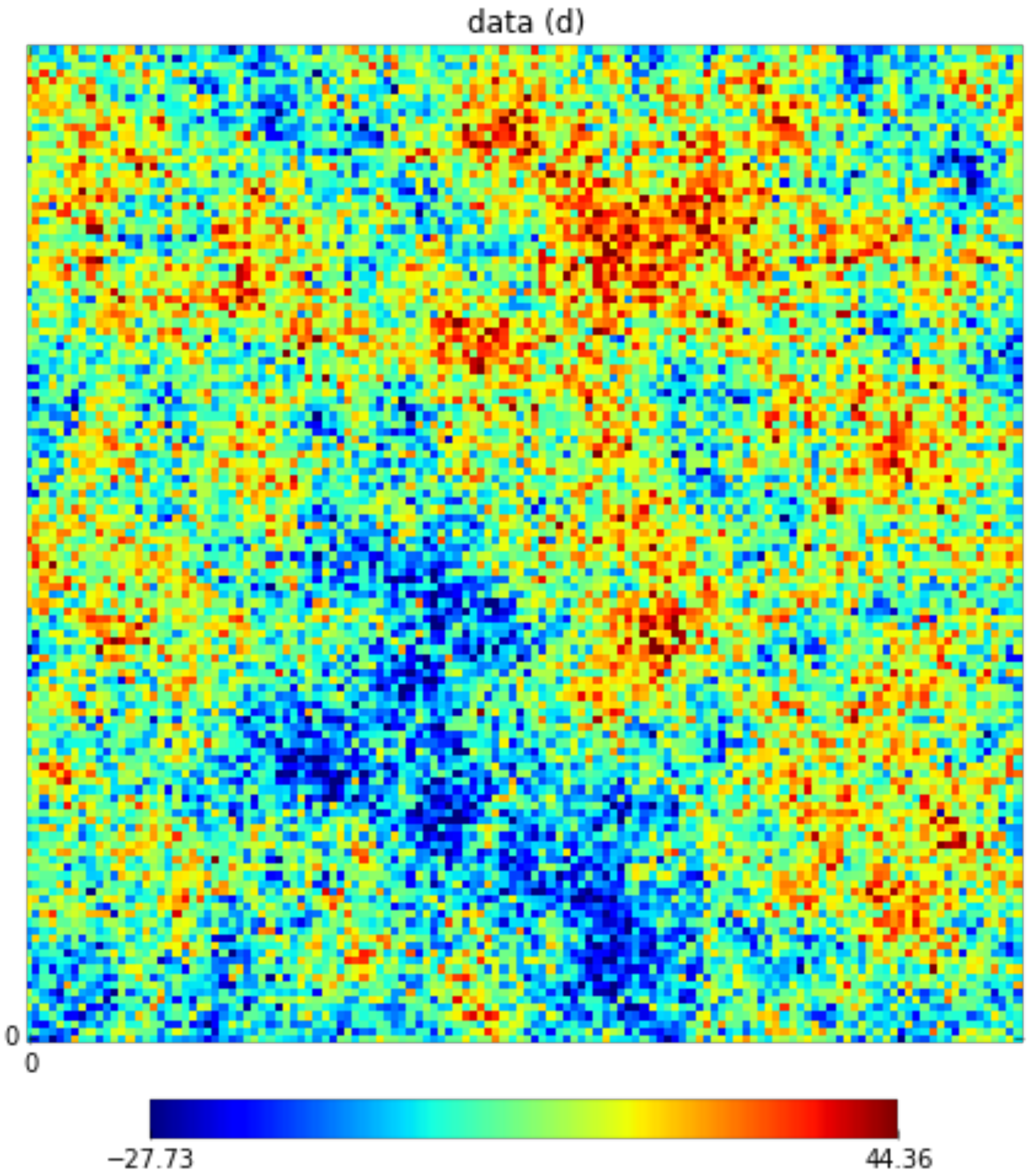
Application

Simulated dataset employing NIFTY package

$$d = R(s) + n$$

find optimal solution for s given d

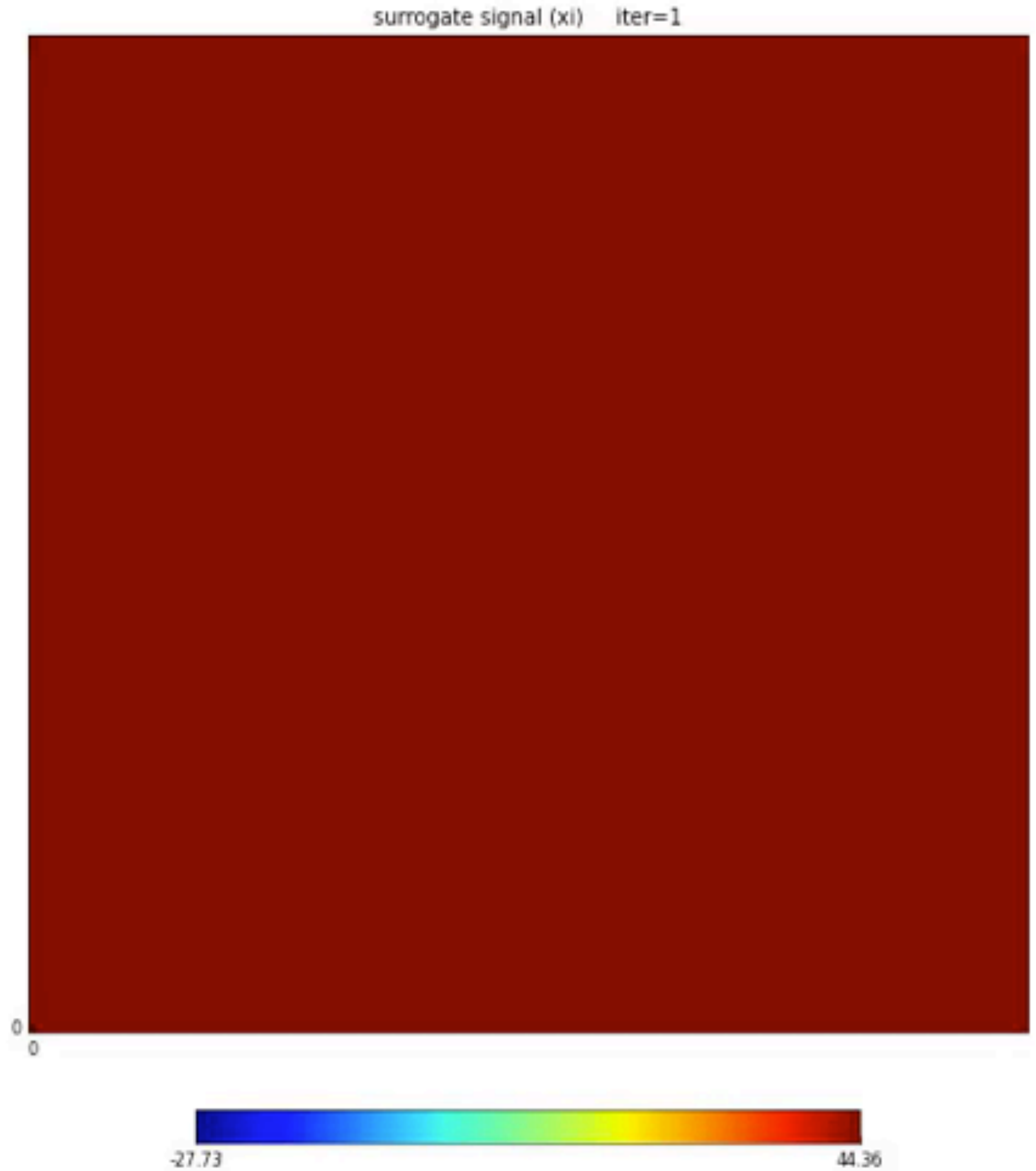
assumption: s is random field following some statistics and being constrained by the data

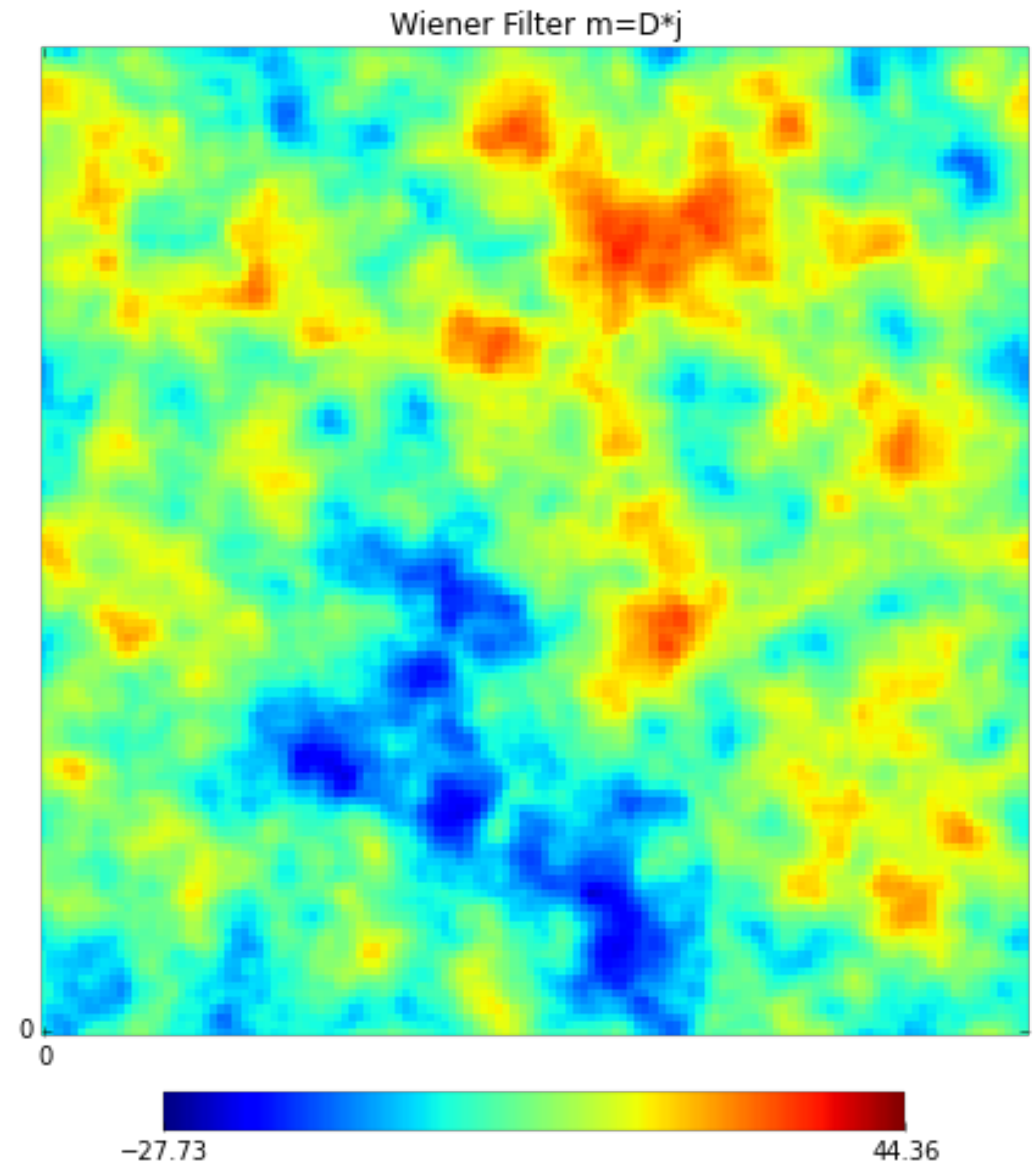


NIFTY= Numerical Information Field Theory

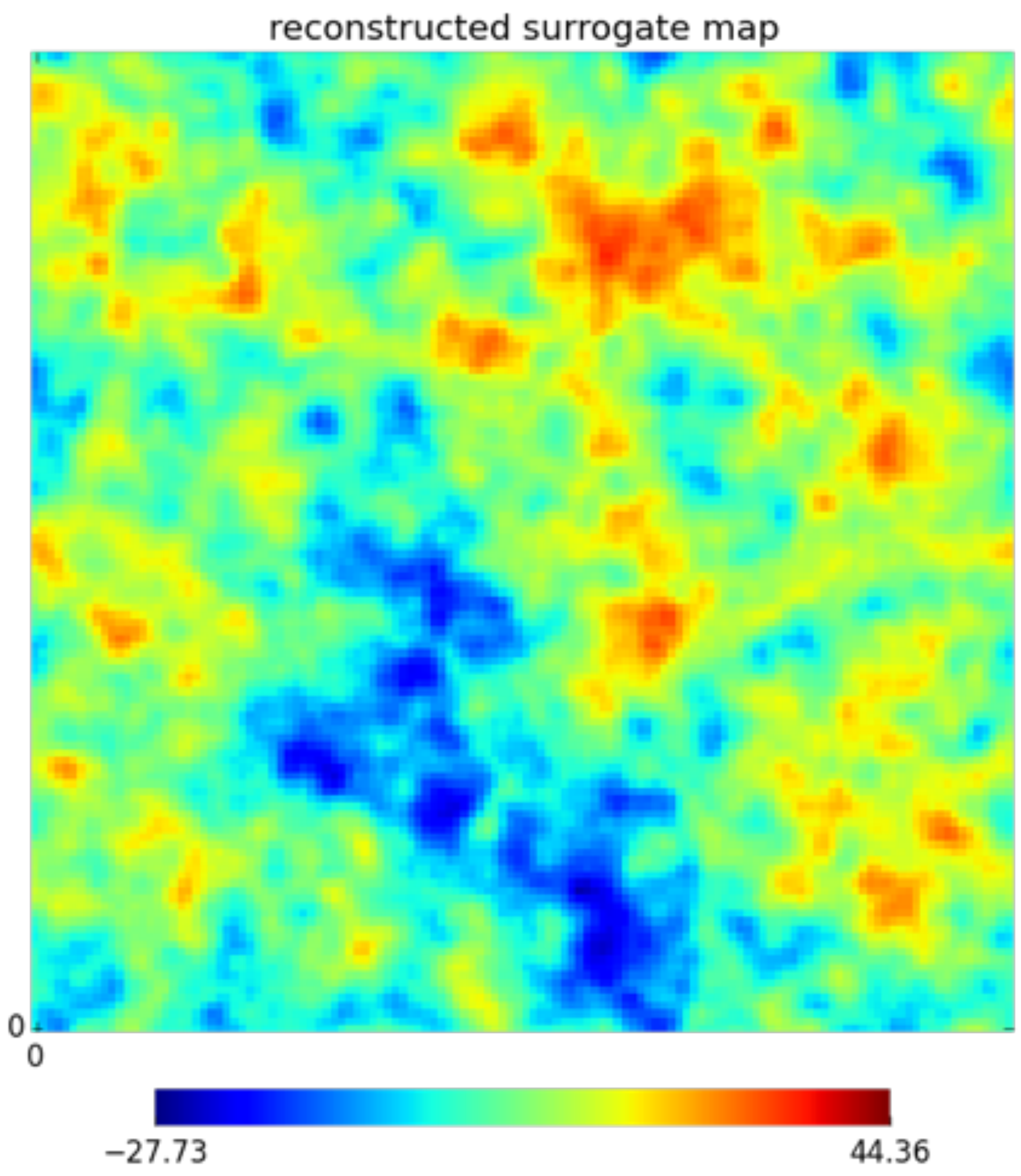
(M.Selig, T. Enßlin et al.)

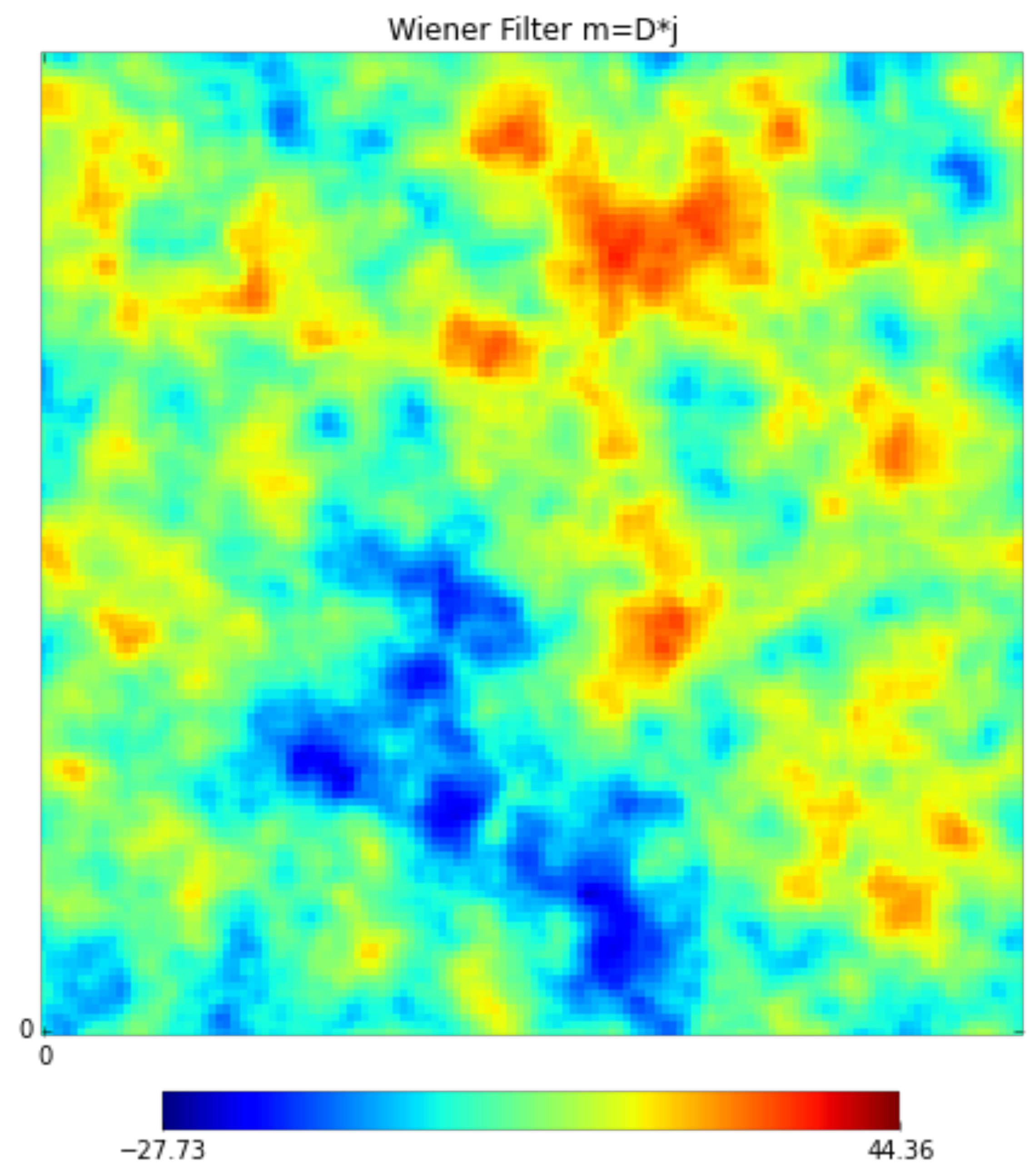
Application -Results



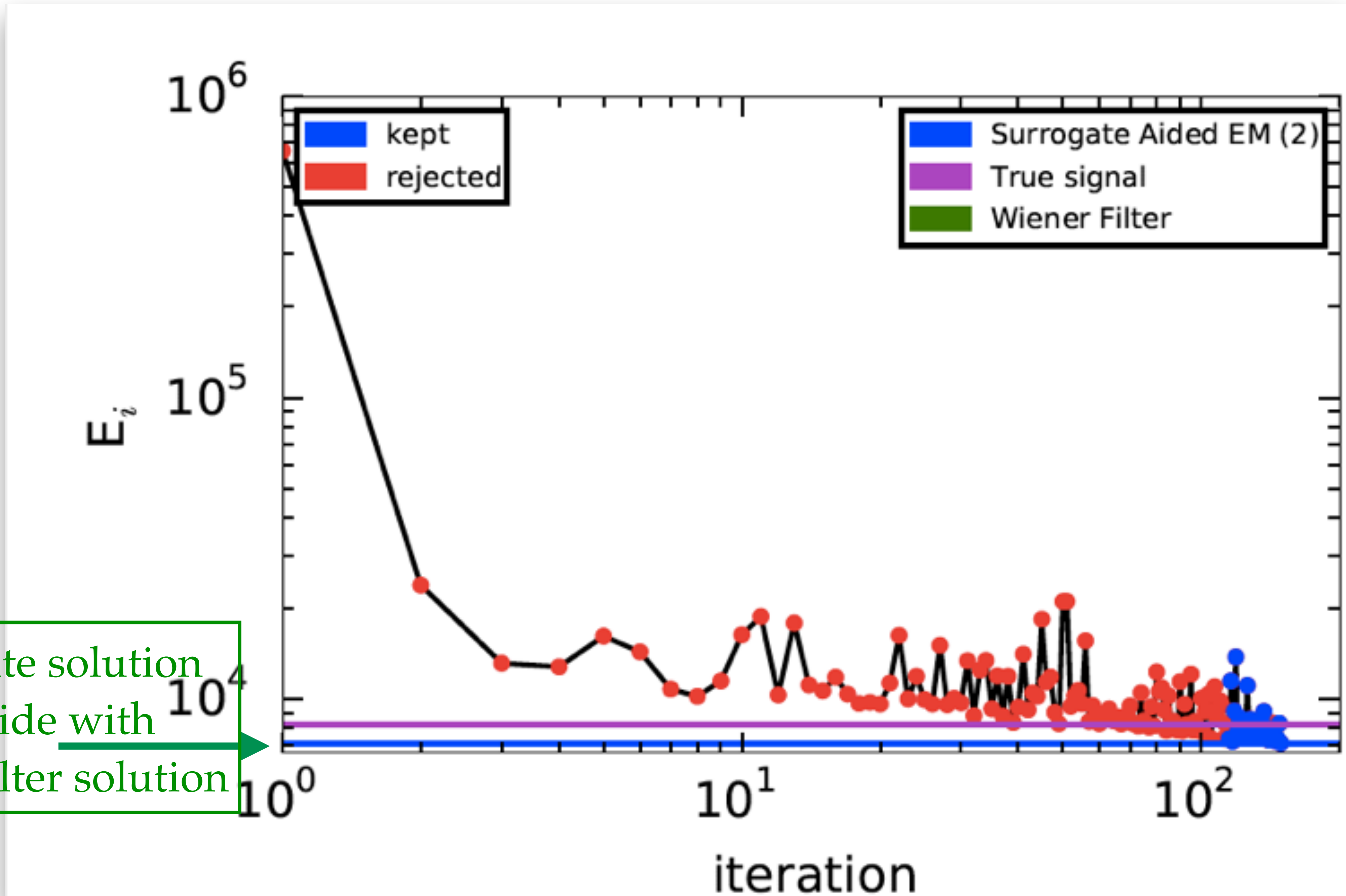


Application -Results



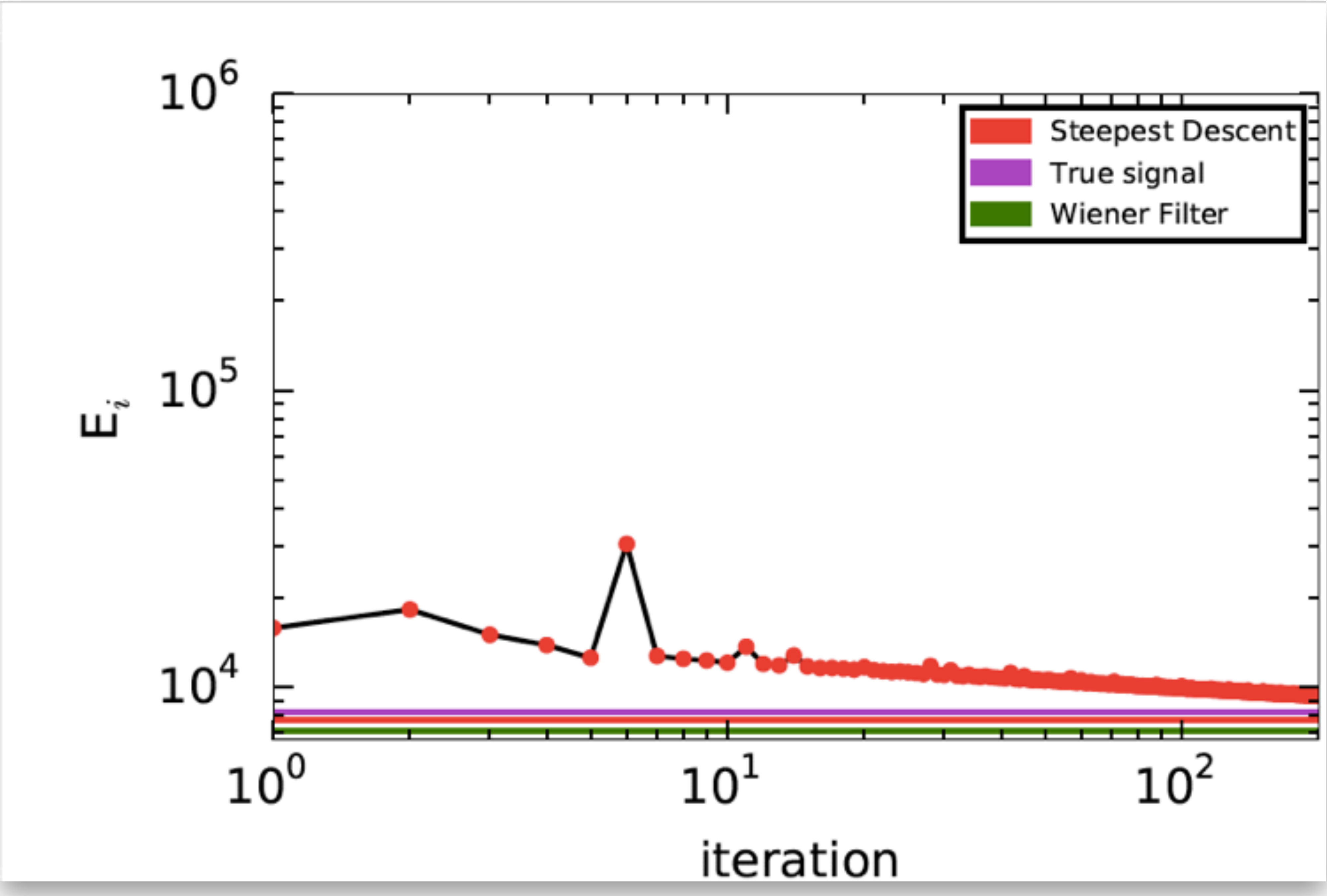


Application -Results

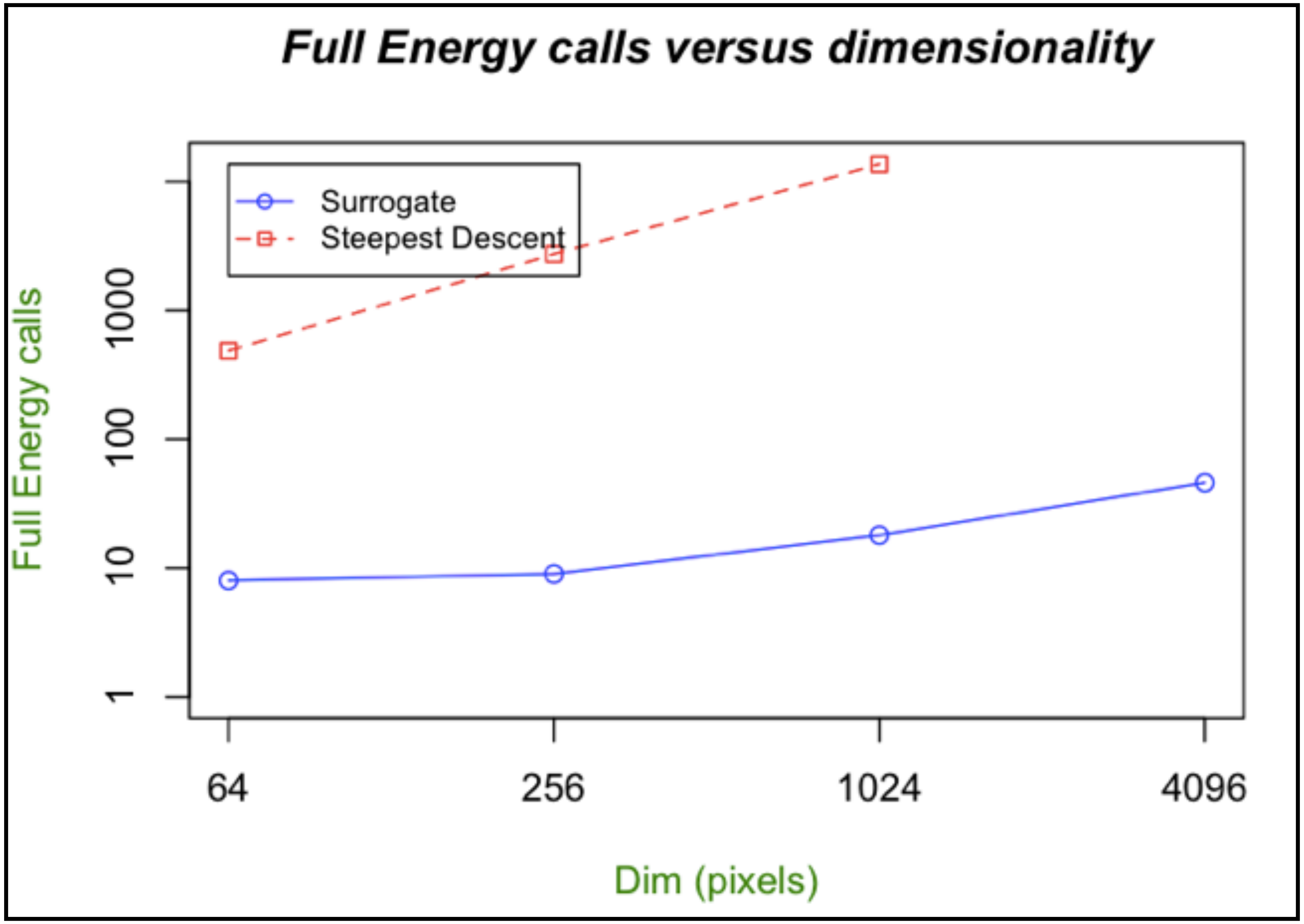


Surrogate solution coincide with Wiener filter solution

Application -Results



Application -Results



Conclusion & Summary

- Speed up of computer run of minimisation of Complex Energy function is desired
- A *Kriging Surrogate sampler* is developed to emulate the behaviour of the Complex Energy Function
  - powerful way to perform Bayesian inference about functions in high-dimensional space
  - non-intrusive approach, but still an approximation
  - reduces number of function evaluations
  - includes sensitivity analysis
- Application in high dimensions of Kriging Surrogate within NIFTY framework is shown
  - can speed up to a **factor of ~100** other optimisation schemes