



Bayesian Optimisation with Dimension Scheduling Algorithm : Application to Biological Systems

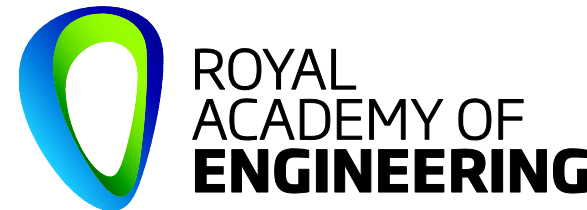
Doniyor Ulmasov ^a , Dr Caroline Baroukh ^b , Dr Benoit Chachuat ^c ,
Dr Marc Peter Deisenroth ^a , and Dr Ruth Misener ^a ,

^a Department of Computing, Imperial College London, UK | ^b INRA, Laboratory of Environmental Biotechnology, France | ^c Department of Chemical Engineering, Imperial College London, UK

Project Partners:



Funding Sources:



EP/M028240/1

Presentation Overview

- Algae Model
- Bayesian Optimisation
- Problem
- Bayesian Optimisation with Dimension Scheduling Algorithm
- Framework
- Questions?

Algae Models

Algae Models

- Learn about reactions of micro-algae metabolism to, e.g., light exposure. Measure lipids, carbohydrates, chlorophyll
- Simulations based on experimental data (Baroukh et al., 2014)
- The objective to minimize the squared error between the simulations and the experimental data

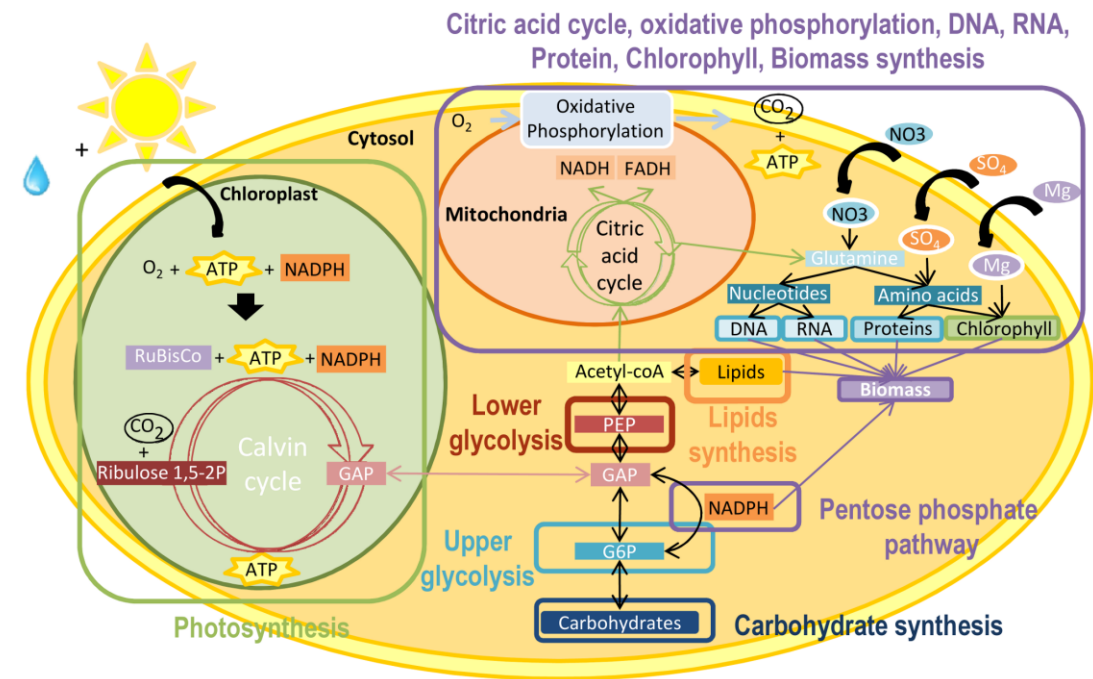


Figure : Simplified central carbon metabolic network of a unicellular photoautotrophic microalgae (Baroukh et al., 2014)

Bayesian Optimisation

Bayesian Optimisation



Legged
Locomotion
(Calandra et al.
(AMAI, 2015)



Pharmaceutical
Processes
(F. Boukouvala
2010)



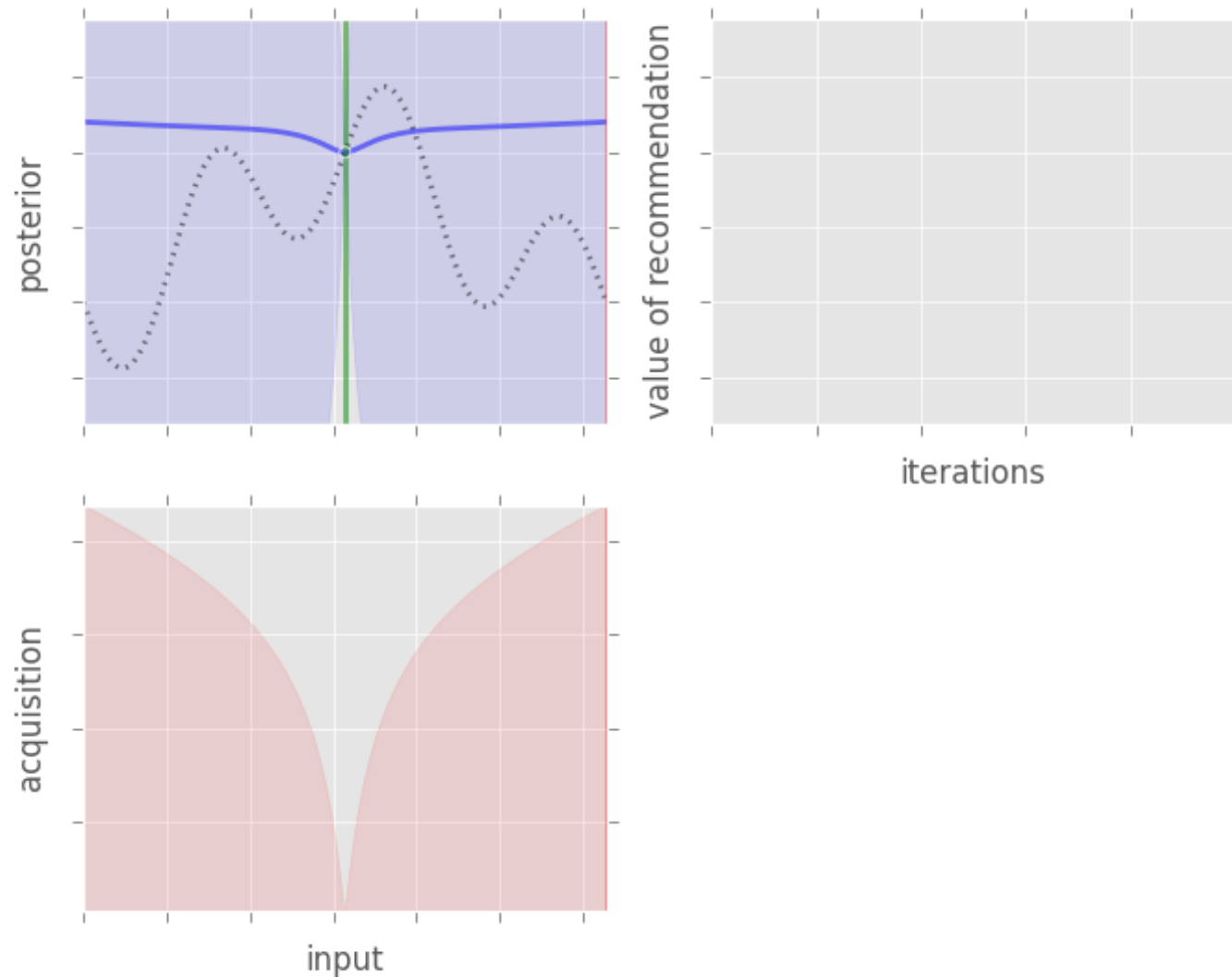
CO2 Capture
Technologies
(M. M. Faruque
2012)



Integrated
gasification
combined cycle
(Y. Lang 2011)

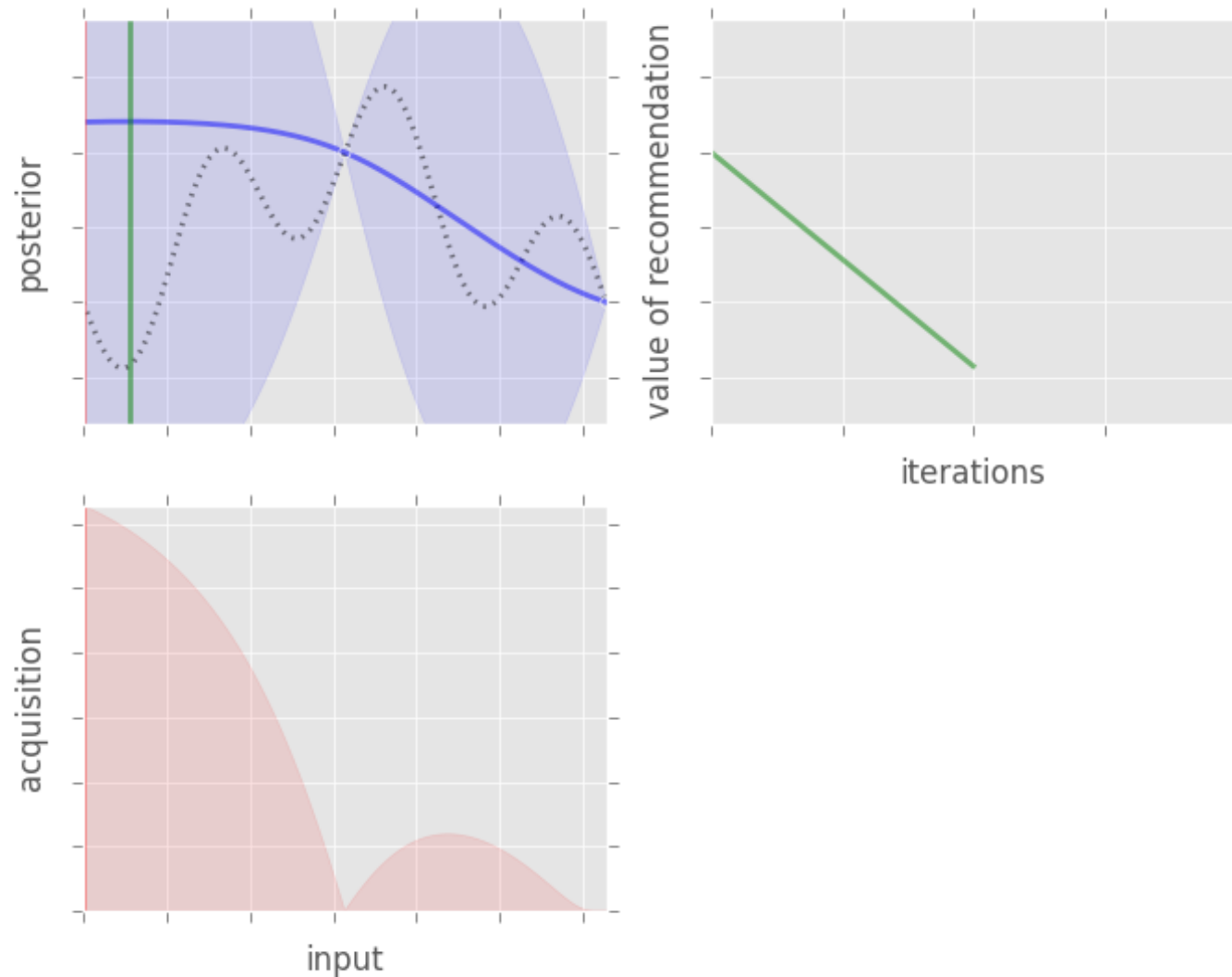
- Bayesian Optimisation is a data-efficient asymptotically complete global black box optimisation technique. That uses all available data, prior and the evidence to calculate the posterior distribution.
- Also known as Kriging

Objective: $\max f(x)$



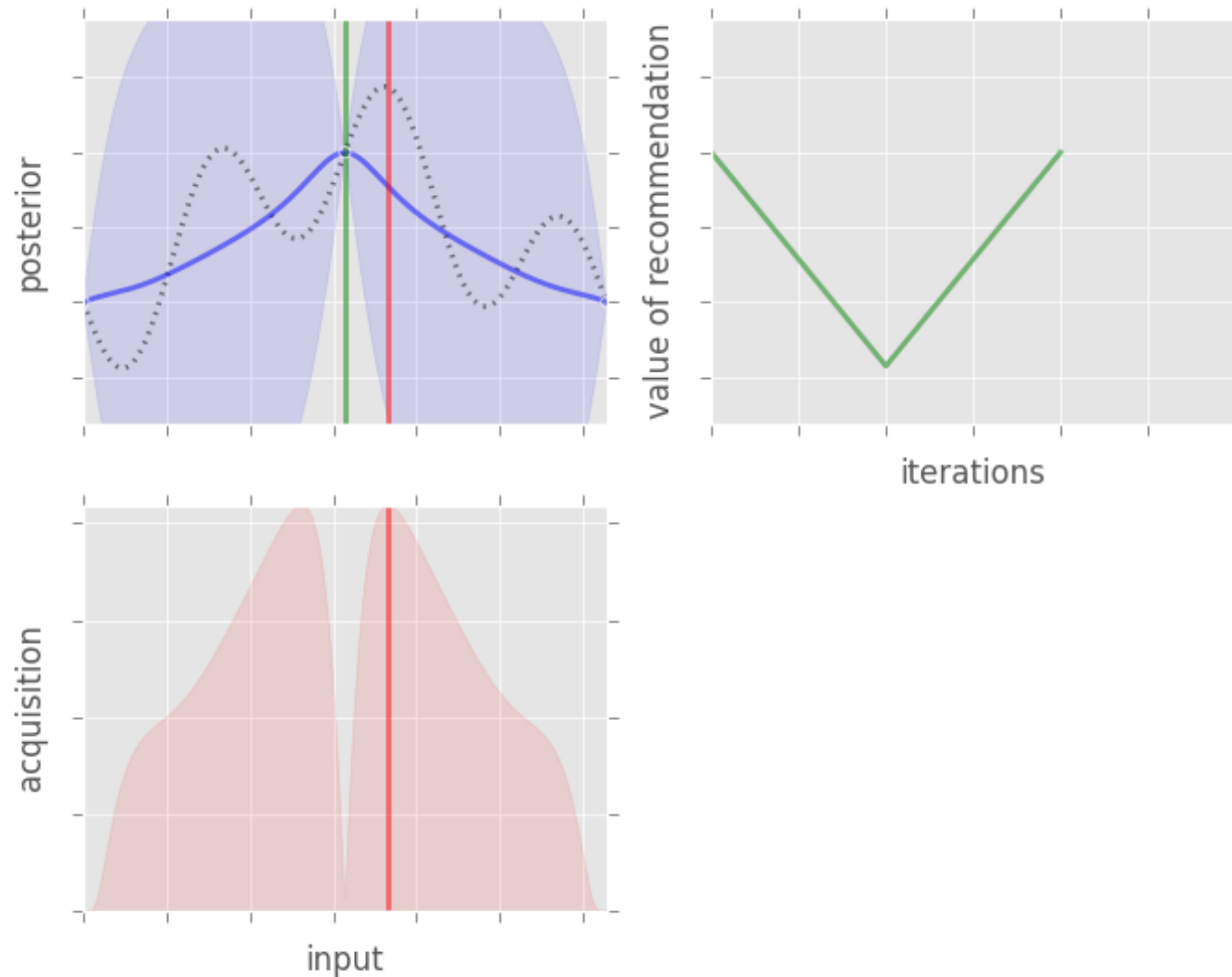
- $f(x) = \cos(x) + \sin(3 \times x)$
- Expected Improvement Policy
- Matérn 5/2 Kernel

Objective: $\max f(x)$



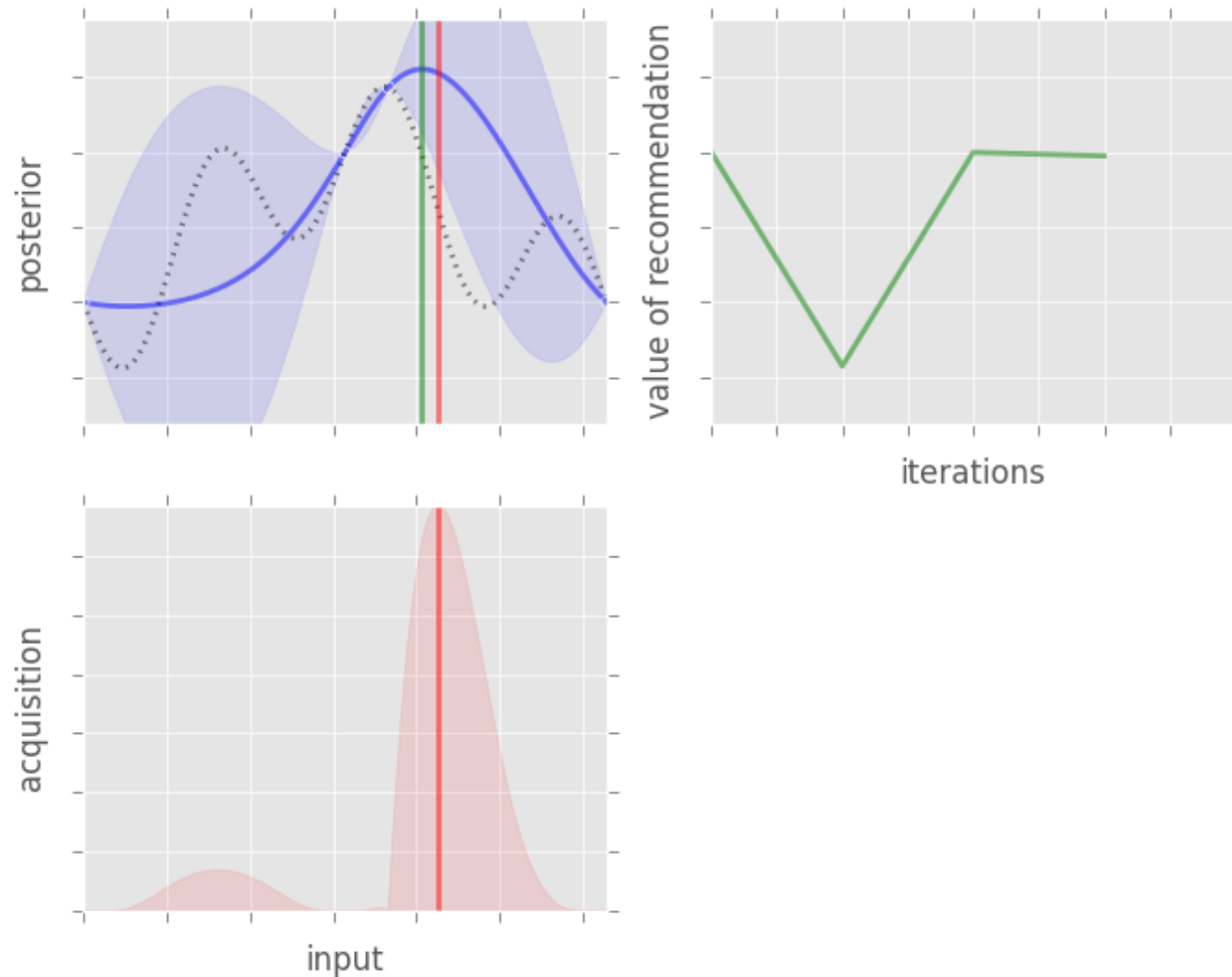
- $f(x) = \cos(x) + \sin(3 \times x)$
- Expected Improvement Policy
- Matérn 5/2 Kernel
- High Variance during start

Objective: $\max f(x)$



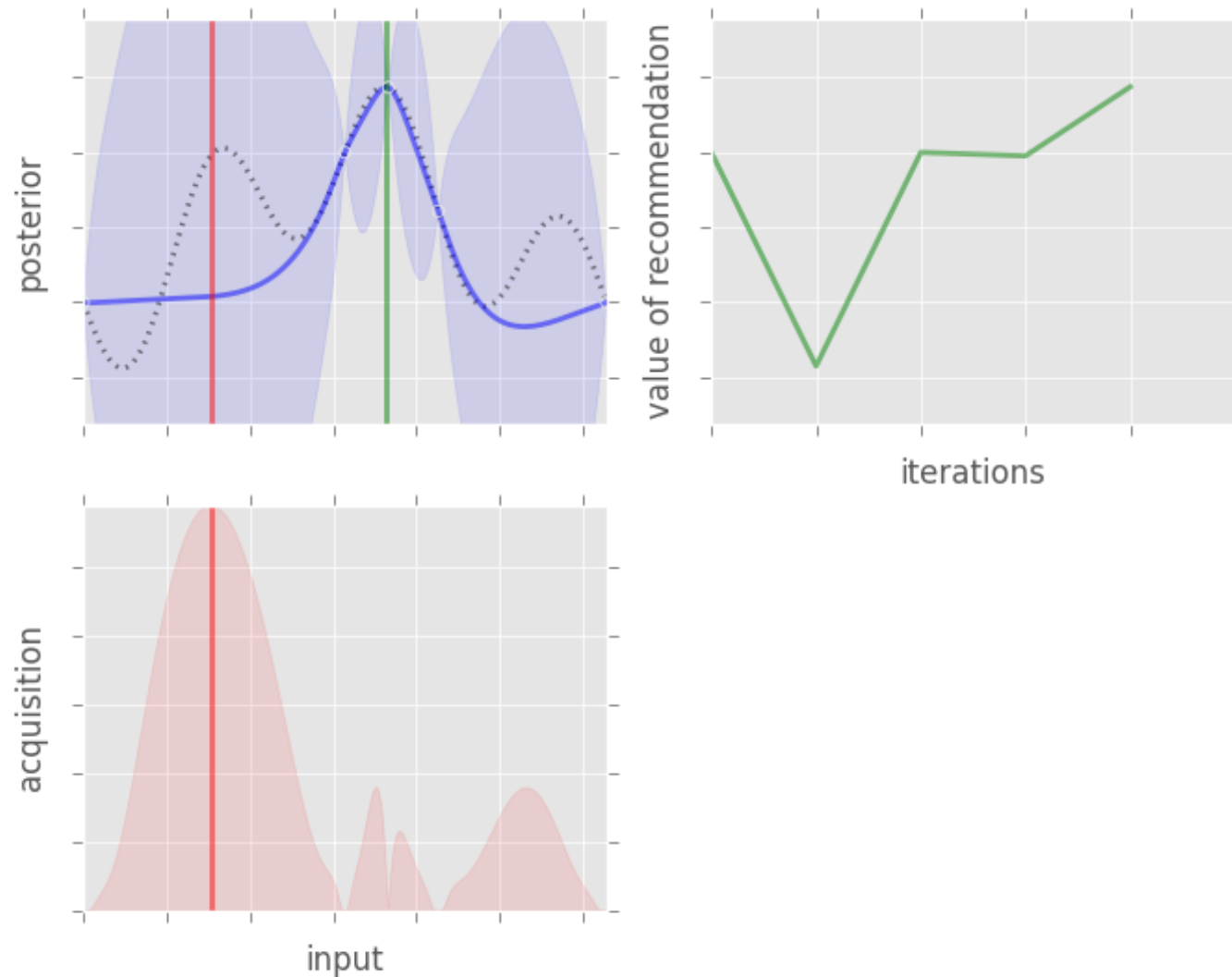
- $f(x) = \cos(x) + \sin(3 \times x)$
- Expected Improvement Policy
- Matérn 5/2 Kernel
- High Variance during start

Objective: $\max f(x)$



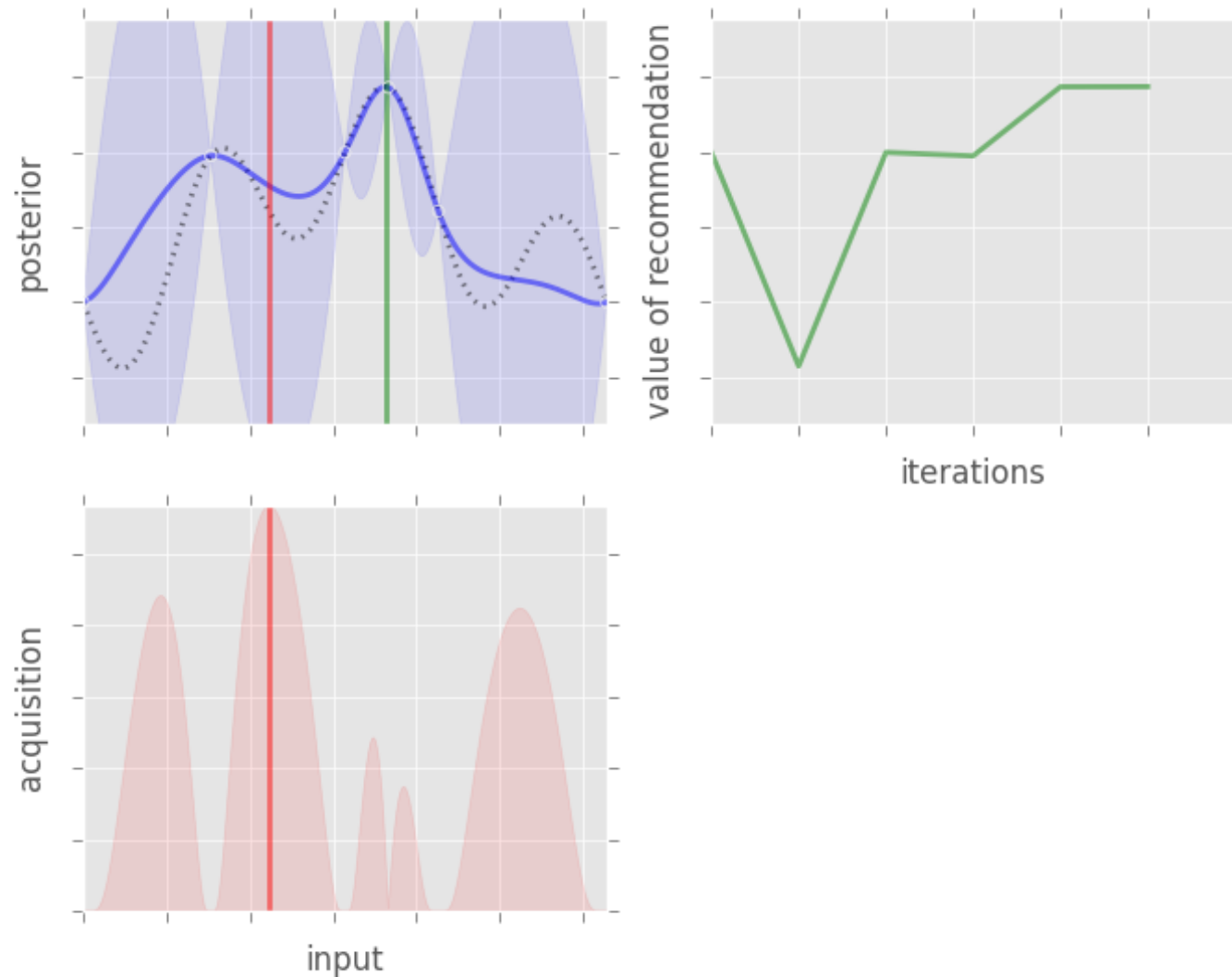
- $f(x) = \cos(x) + \sin(3 \times x)$
- Expected Improvement Policy
- Matérn 5/2 Kernel
- High Variance during start

Objective: $\max f(x)$



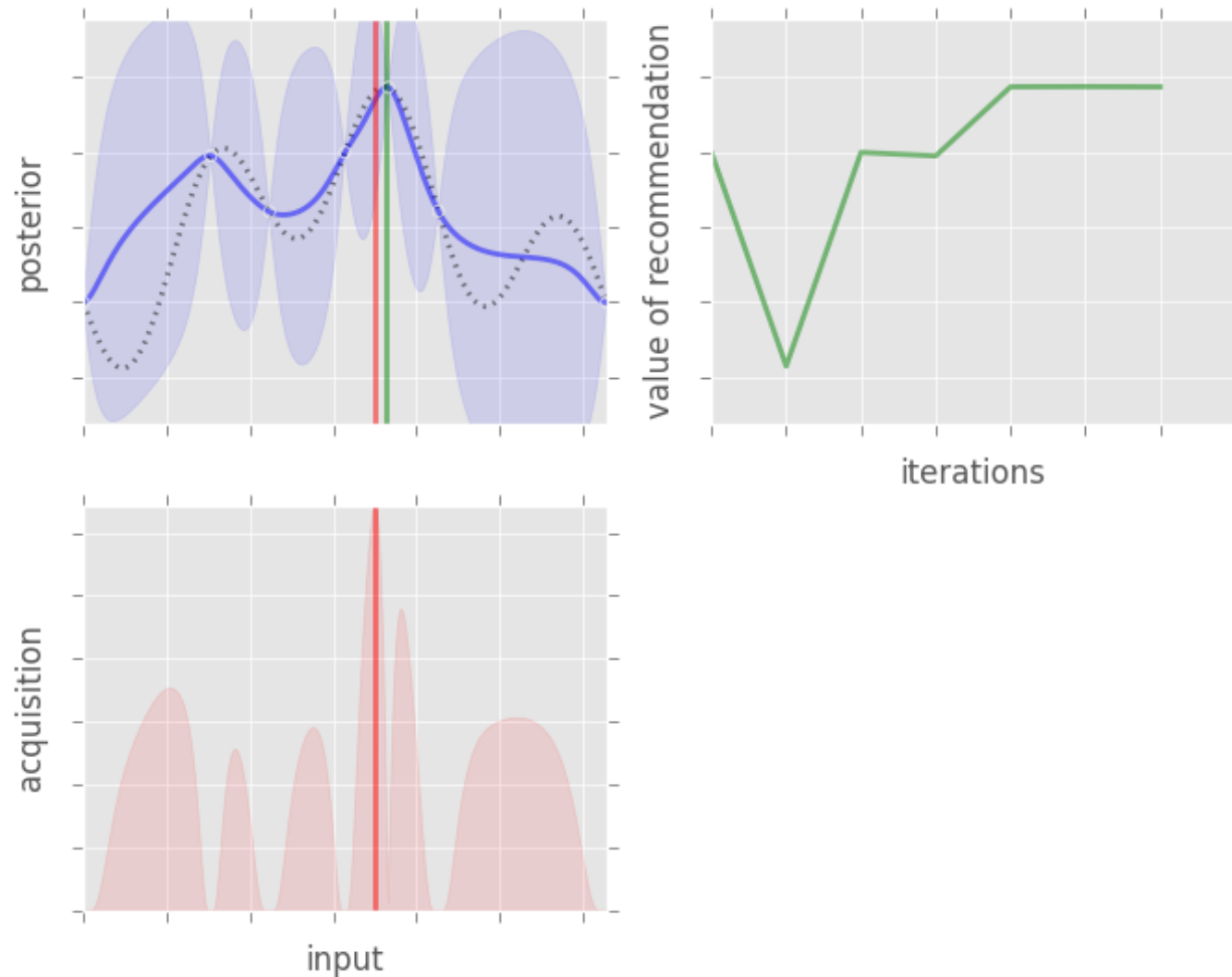
- $f(x) = \cos(x) + \sin(3 \times x)$
- Expected Improvement Policy
- Matérn 5/2 Kernel
- High Variance during start

Objective: $\max f(x)$



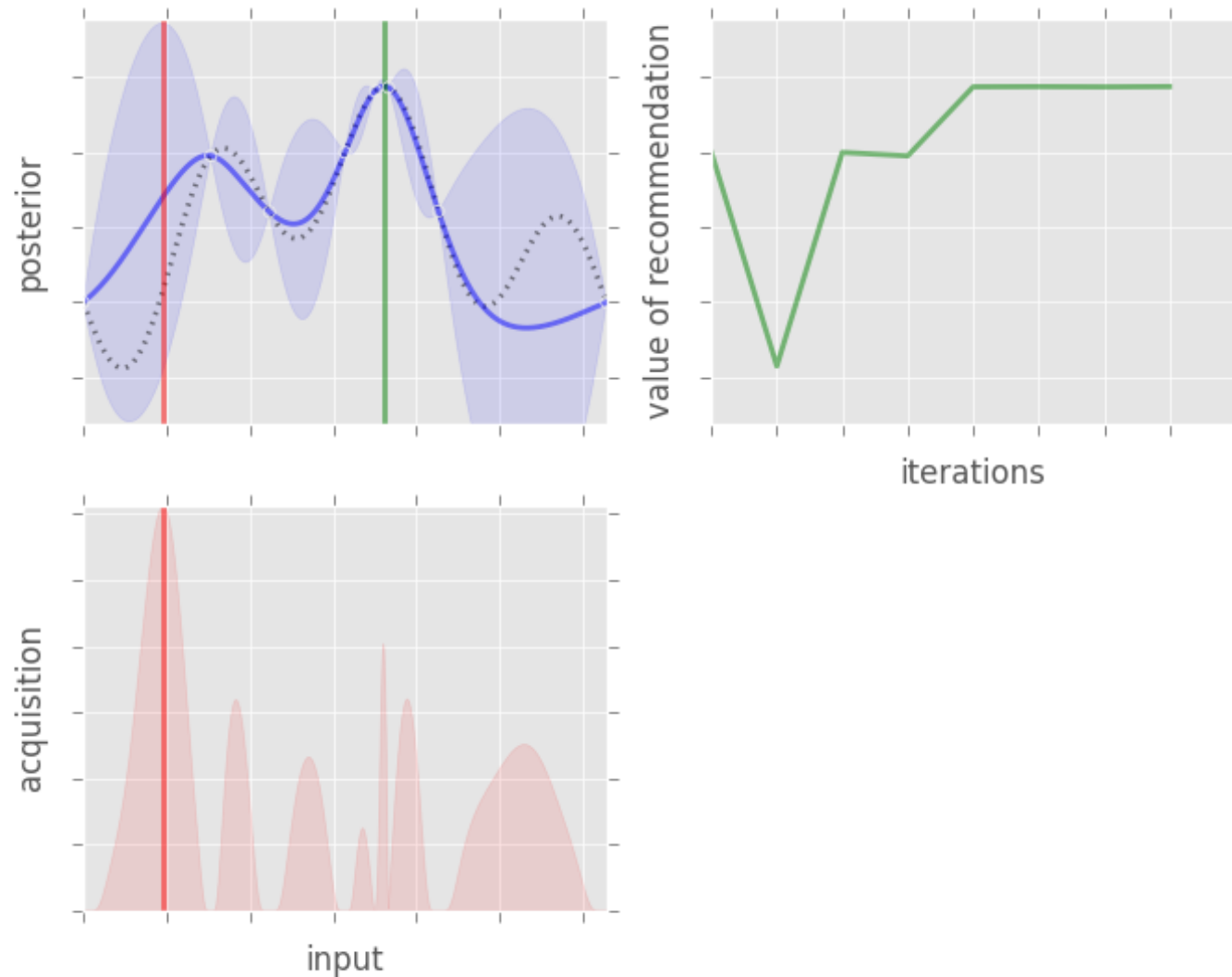
- $f(x) = \cos(x) + \sin(3 \times x)$
- Expected Improvement Policy
- Matérn 5/2 Kernel
- High Variance during start
- Over time the variance is reduced

Objective: $\max f(x)$



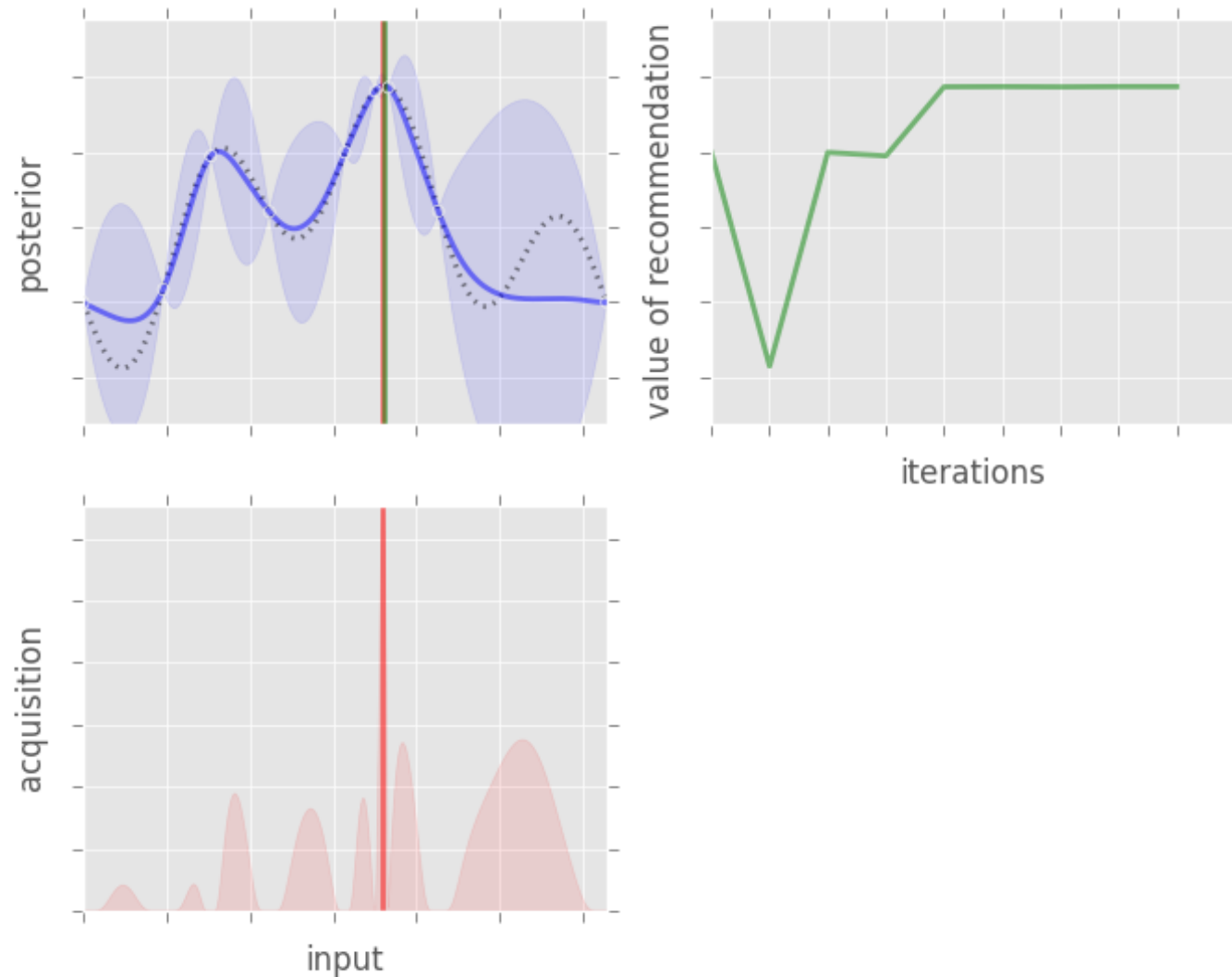
- $f(x) = \cos(x) + \sin(3 \times x)$
- Expected Improvement Policy
- Matérn 5/2 Kernel
- High Variance during start
- Over time the variance is reduced

Objective: $\max f(x)$



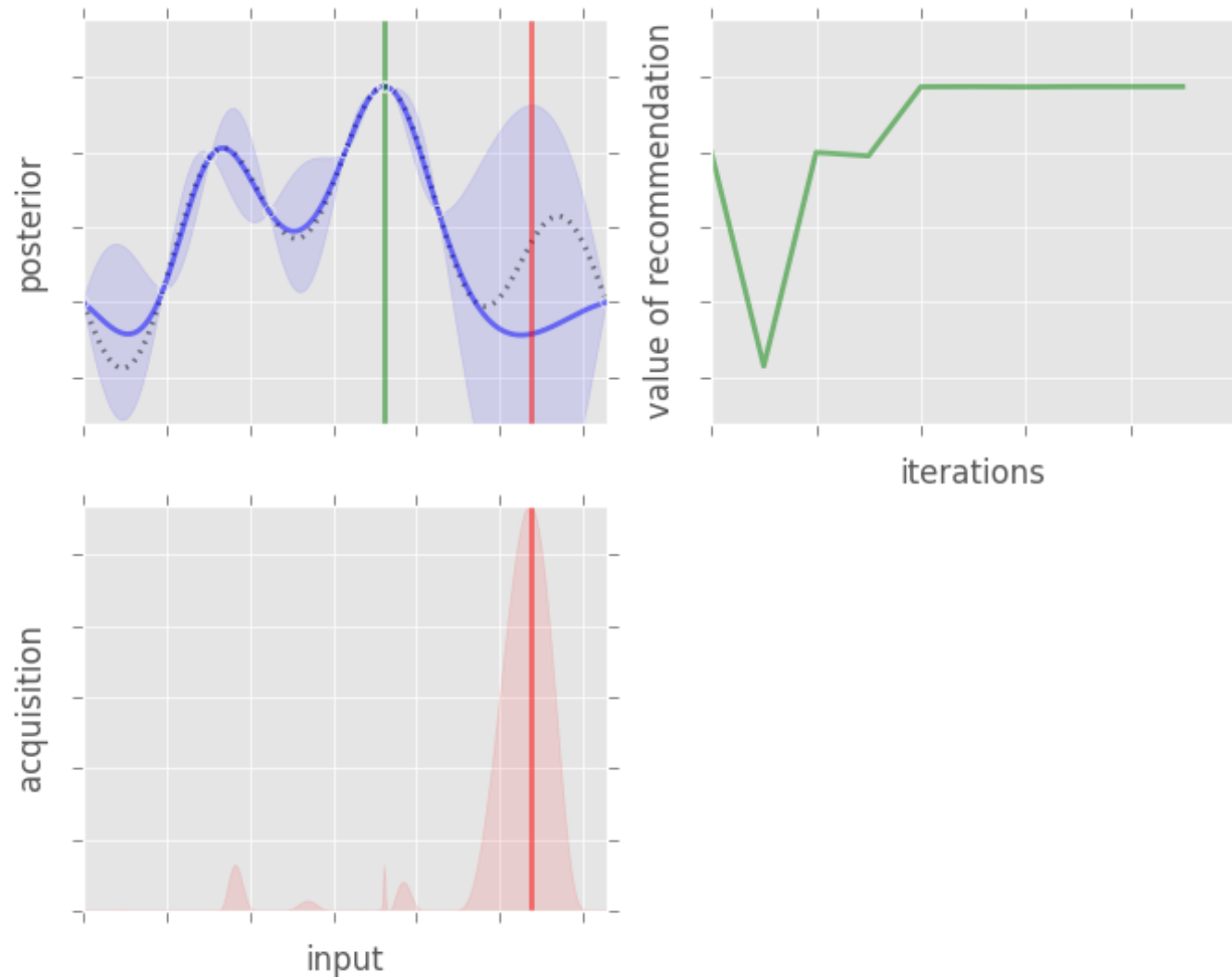
- $f(x) = \cos(x) + \sin(3 \times x)$
- Expected Improvement Policy
- Matérn 5/2 Kernel
- High Variance during start
- Over time the variance is reduced

Objective: $\max f(x)$



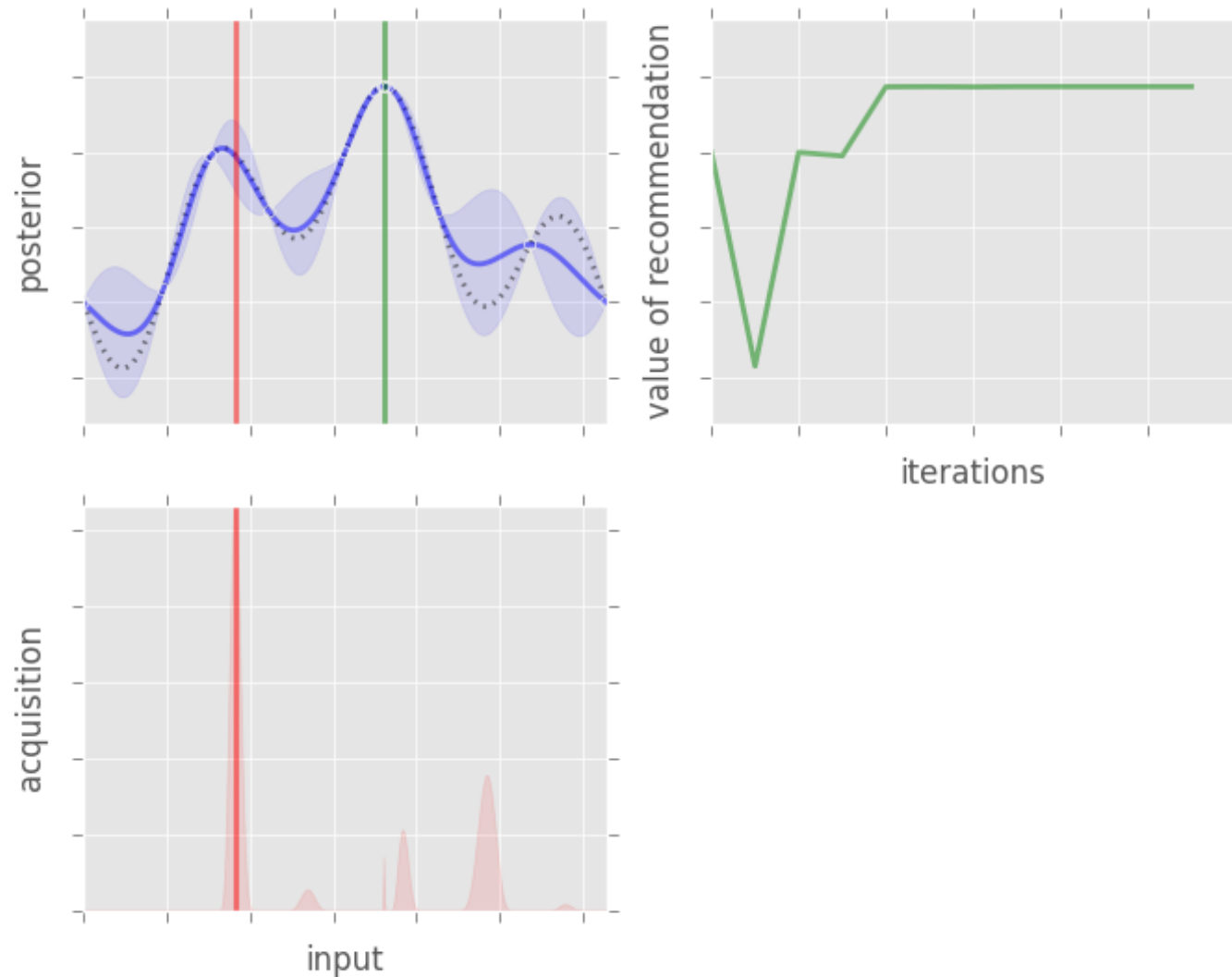
- $f(x) = \cos(x) + \sin(3 \times x)$
- Expected Improvement Policy
- Matérn 5/2 Kernel
- High Variance during start
- Over time the variance is reduced

Objective: $\max f(x)$



- $f(x) = \cos(x) + \sin(3 \times x)$
- Expected Improvement Policy
- Matérn 5/2 Kernel
- High Variance during start
- Over time the variance is reduced

Objective: $\max f(x)$



- $f(x) = \cos(x) + \sin(3 \times x)$
- Expected Improvement Policy
- Matérn 5/2 Kernel
- High Variance during start
- Over time the variance is reduced
- As number of iterations increase we converge to a global optimum

Bayesian Optimization Algorithm

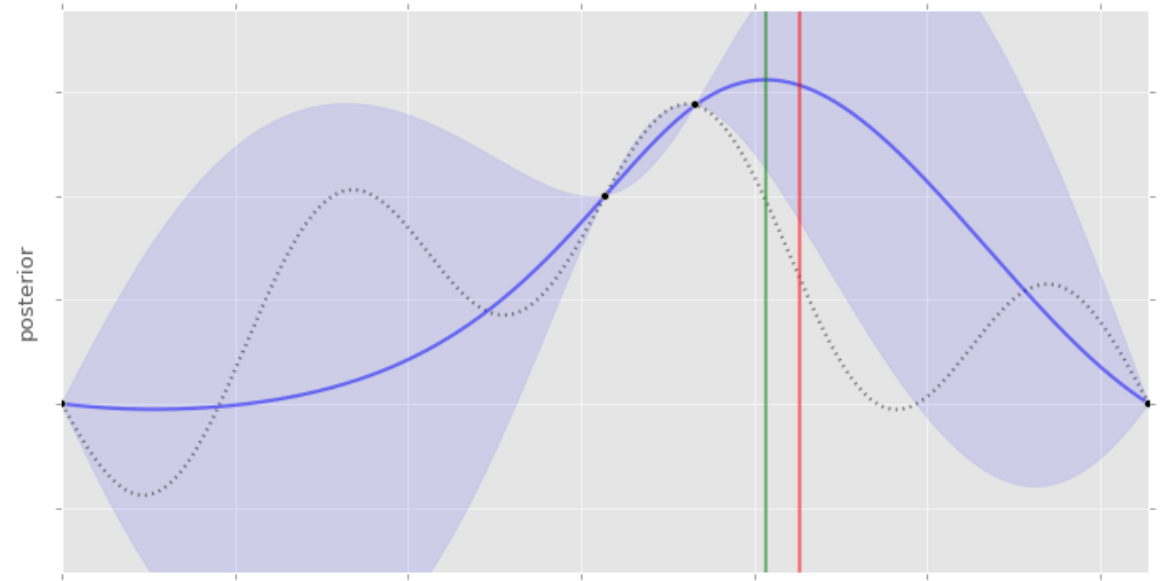
1. Choose several initial points x and evaluate $f(x)$ or $g(x, \omega)$. The points can be randomly generated or used from data samples collected prior.
2. While the stopping criterion is not met:
 - 2a. *Calculate the posterior from the GP on f from the points observed.*
 - 2b. *Use an acquisition function to decide where to evaluate next.*
3. Based on the most recent posterior distribution, decide on the point with the best estimated value or best observed value

The stopping criteria can be arbitrary (e.g. number of iterations or till certain value is reached.)

(E. Brochu 2010)

Gaussian Process

- Gaussian Processes construct a probabilistic model of the function in a form of :
 - $f(x) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$.
- A GP is specified by a mean function, m and a covariance function, k (*kernel*).
- The kernels of a GP incorporate smoothness assumptions made of the data



Problem

Iteration Times of the Algae Model with Bayesian Optimisation

- Iteration time increasing at n^3
- Impractical for our case where the simulation takes under a second
- At 2000th iteration BO spends 1500 seconds to decide the next point of evaluation.

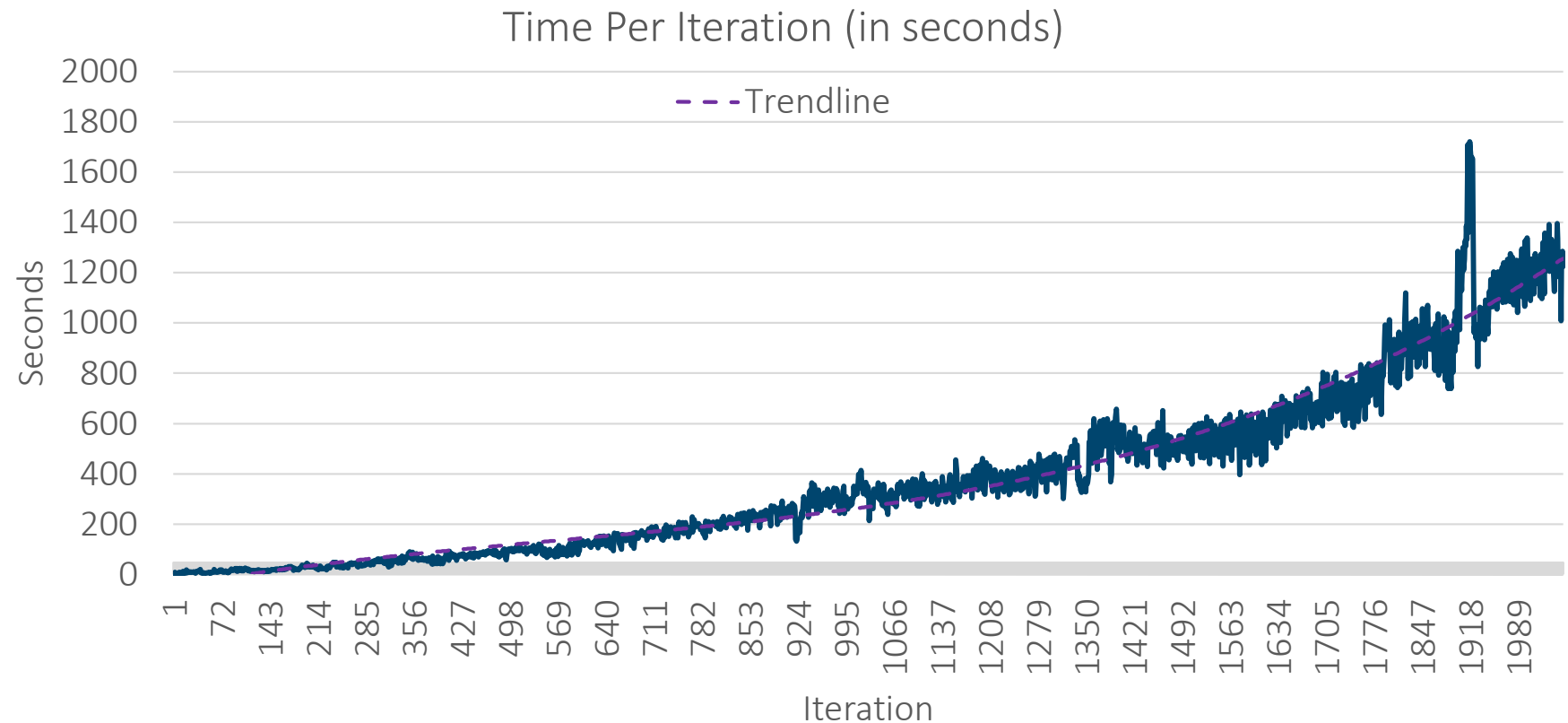


Figure Long run experiment: iteration time in seconds with a fitted n^3 trendline

Objective Value of the Algae Model with Bayesian Optimisation

- Objective Value did not improve further after a certain point.
- There is known optimum value not achieved during 9 day processing period
- Stuck at local minimum

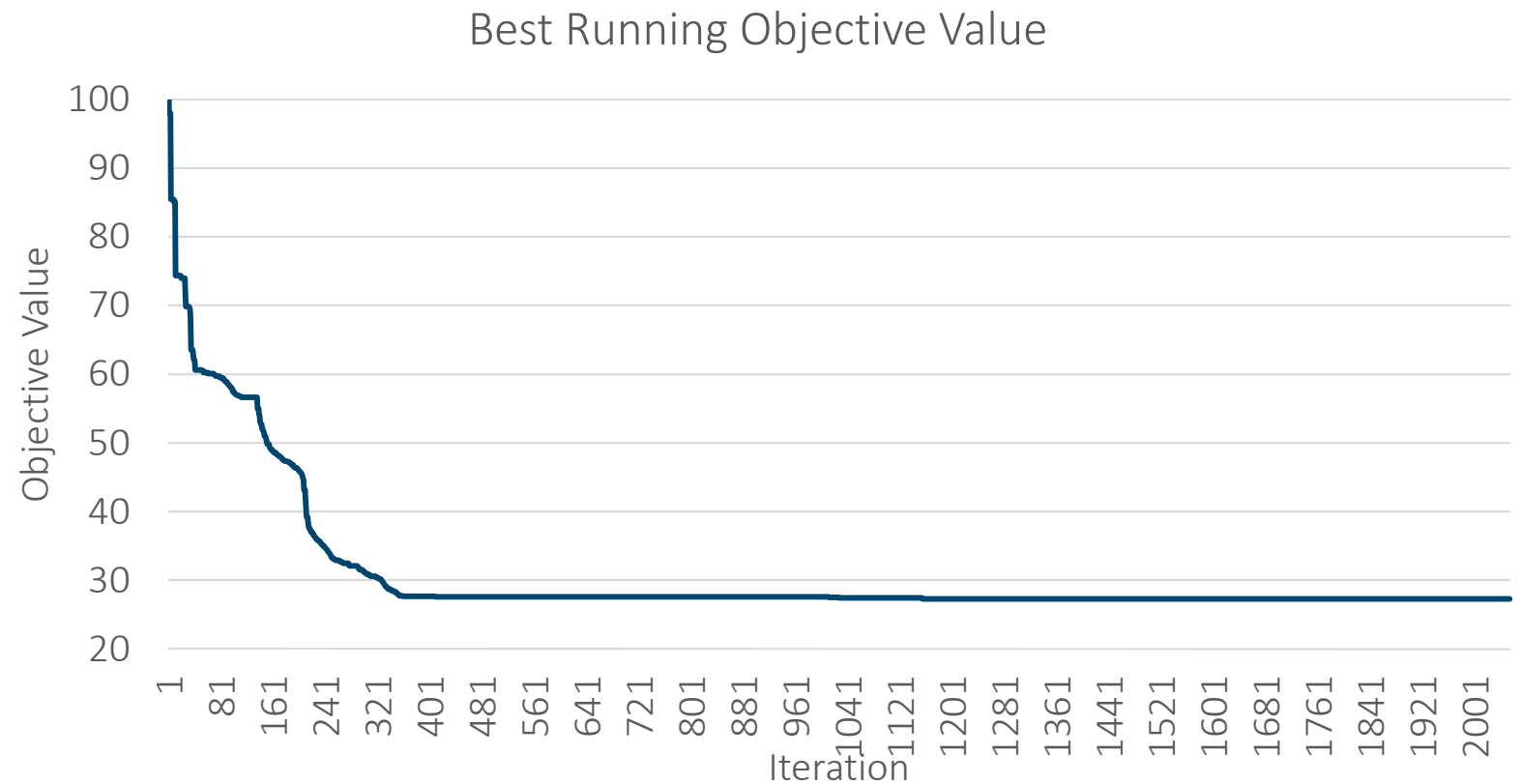
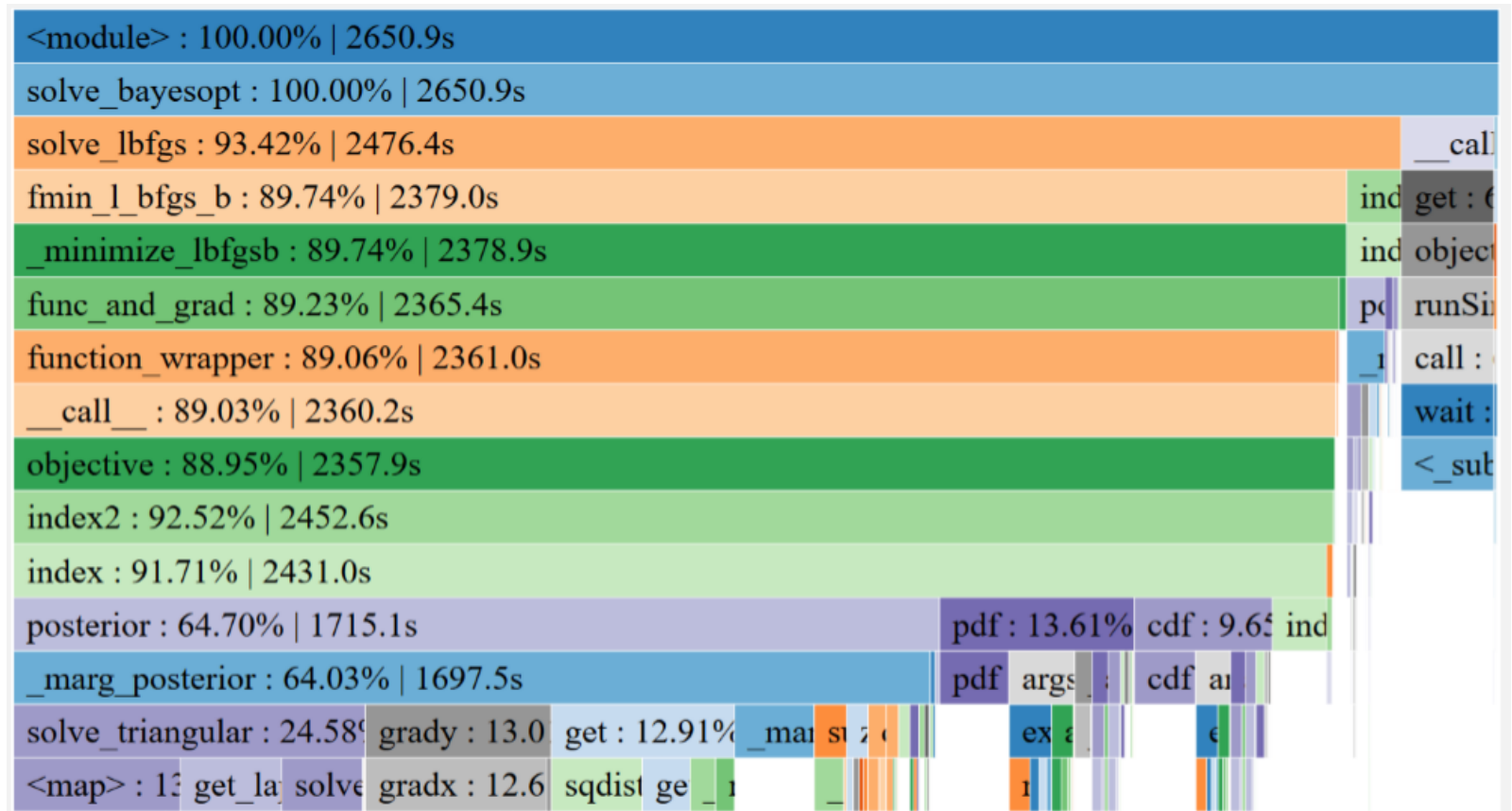


Figure Long Run Experiment: Best Running Objective Value Graph

- Majority of the computational resources is used by the GP posterior calculations.
- Each call to the acquisition function requires to calculations from the GP posterior.
- Posterior calculations are $O(n^3)$ operation
- As number of data points increase, the performance degrades due to the $O(n^3)$ operation.



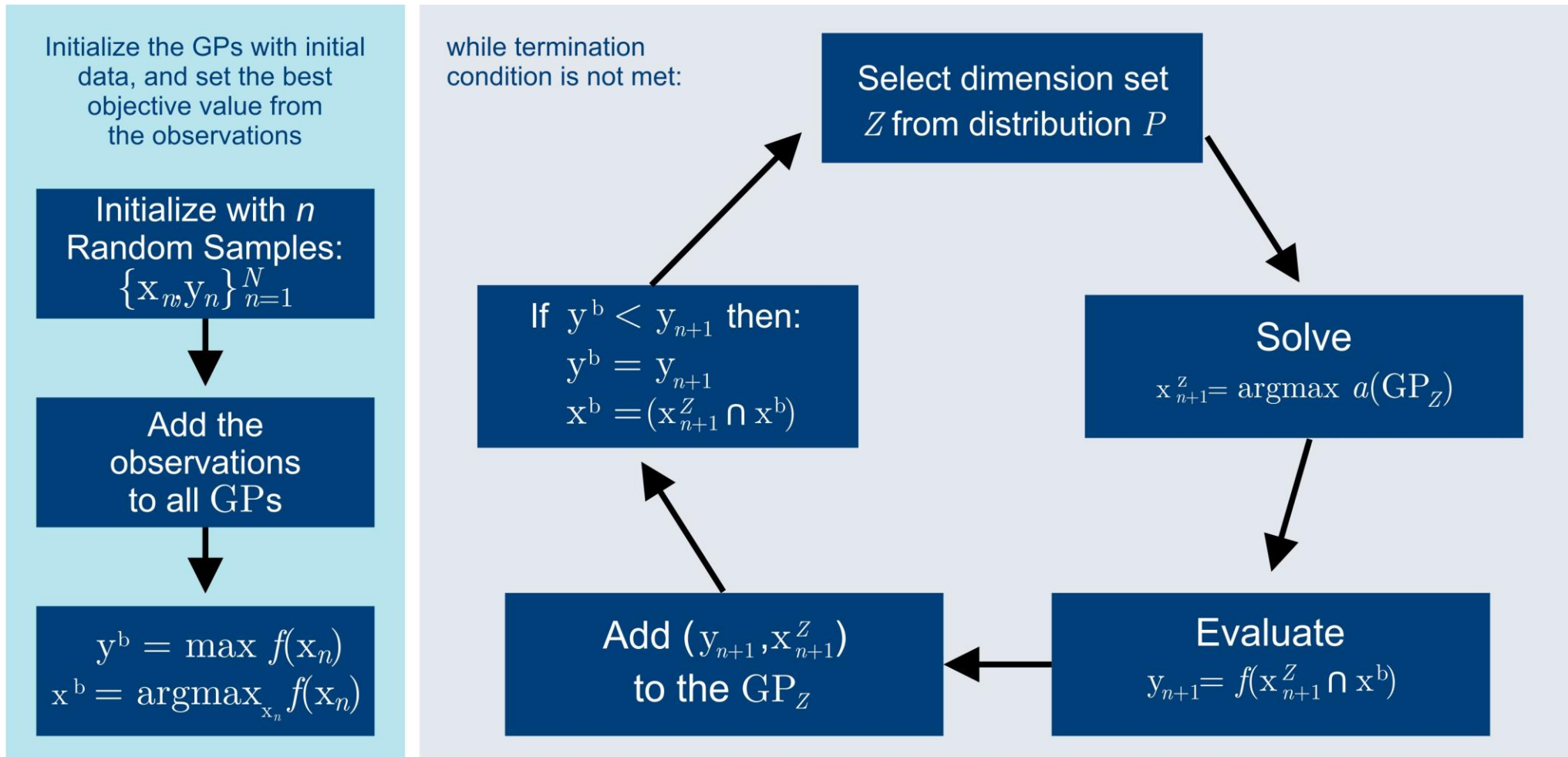
Current and Alternative Solutions

- We attempted to achieve an optimal solution by trying out all the combinations of kernels and policies over the Algae models, none yielded satisfactory results.
- Additive Bayesian Optimisation (Kandasamy et al., 2015) , which breaks the problem into multiple additive functions did not performed well either
- Prior solution involved commercial software suite to optimise with manually added initial points.
- Depending on the initial points, the performance of the software suite varied
- Global Optimisation took up to 24 hours to process.

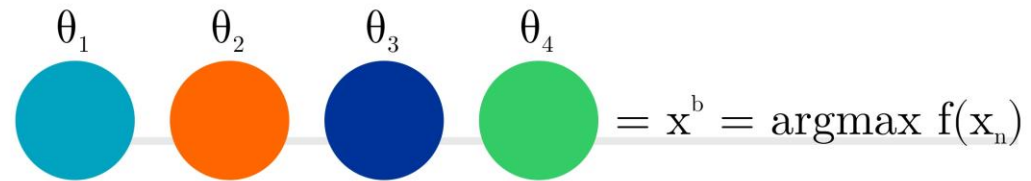
Dimension Scheduling Algorithm

The Algorithm

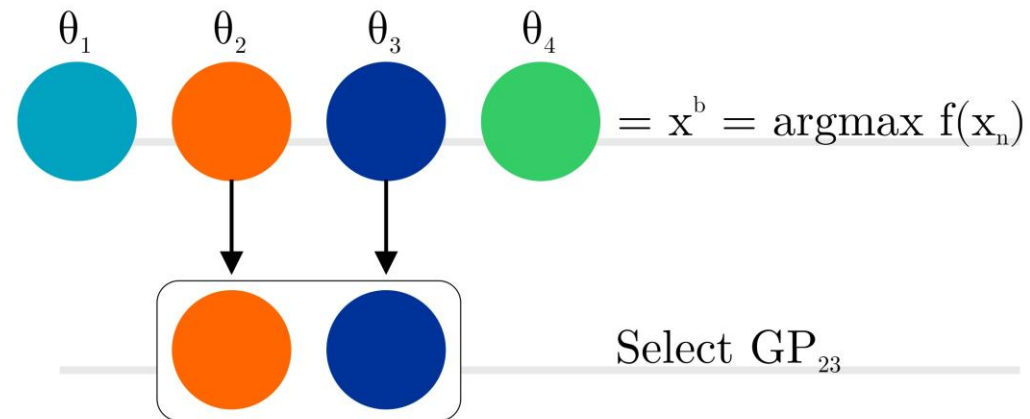
Dimension Scheduling Algorithm



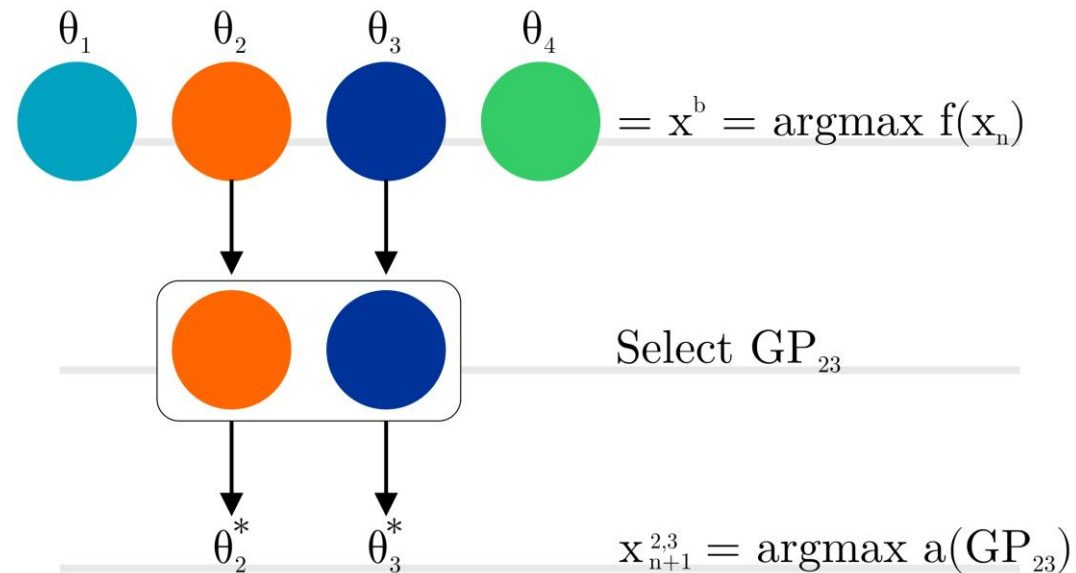
DSA Illustration



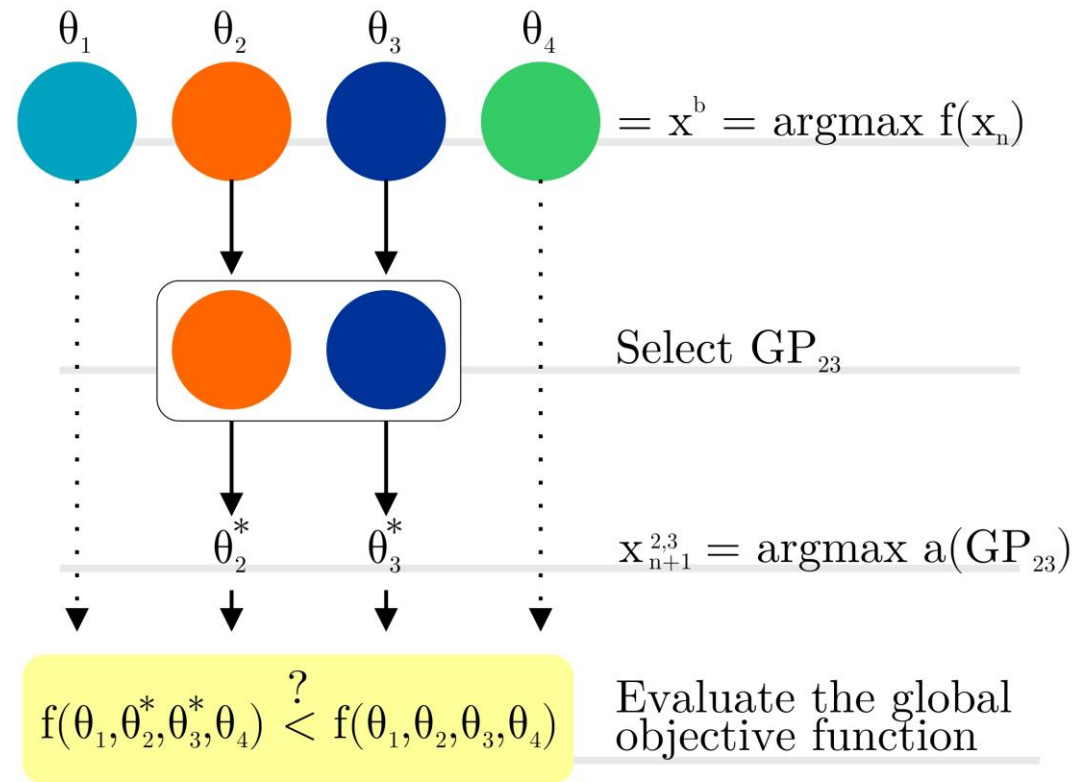
DSA Illustration



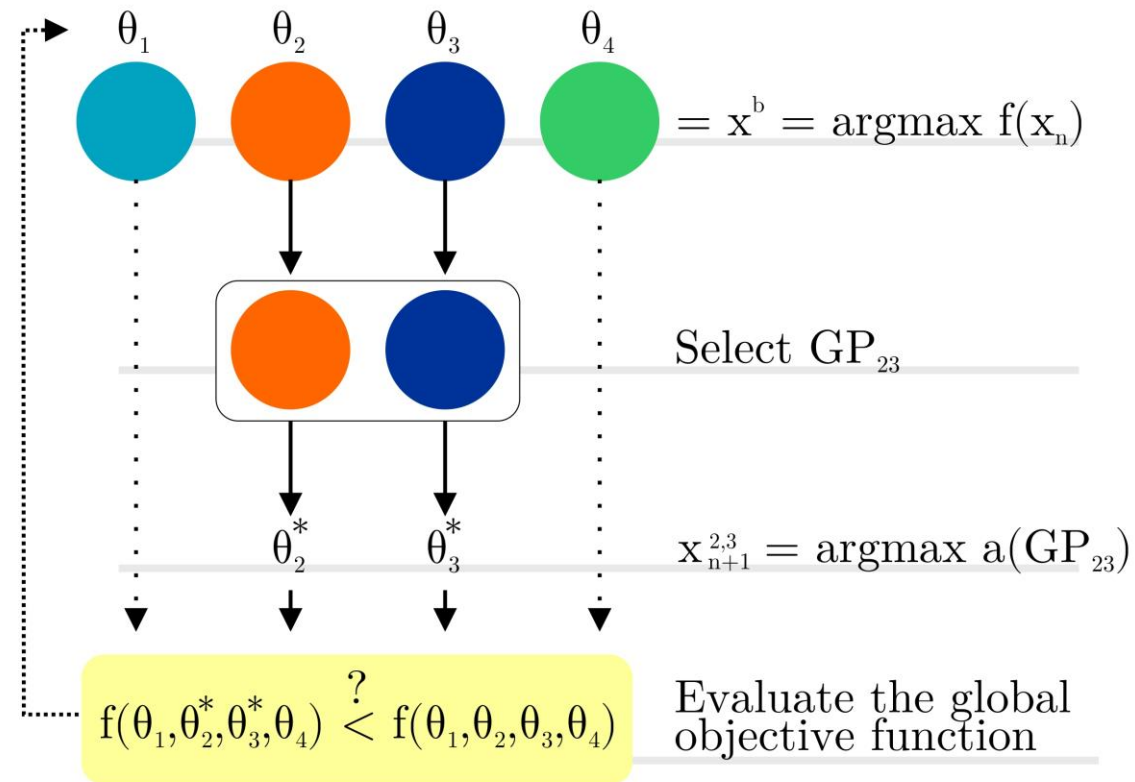
DSA Illustration



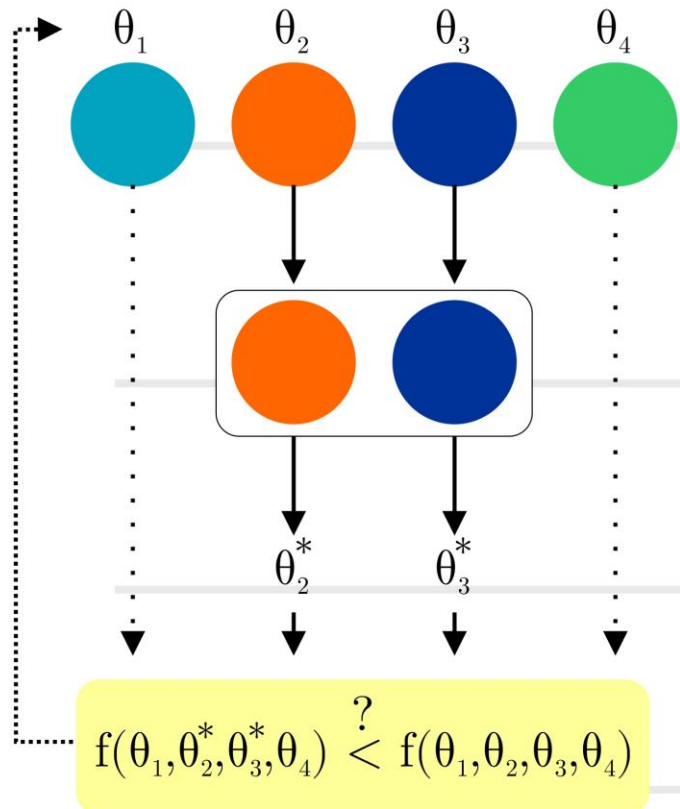
DSA Illustration



DSA Illustration



DSA Properties



- The training data is distributed across $\frac{d!}{k!(d-k)!}$ GPs
- Each GP contains the data corresponding to its dimension, plus the initialisation data.
- Each new iteration select a new subset of the dimensions from a probability distribution, which reflects the importance of the corresponding parameter.
- Thus we reduce the computation times by keeping less data in each GP

Comparison with Traditional BO

Experiment Setup:

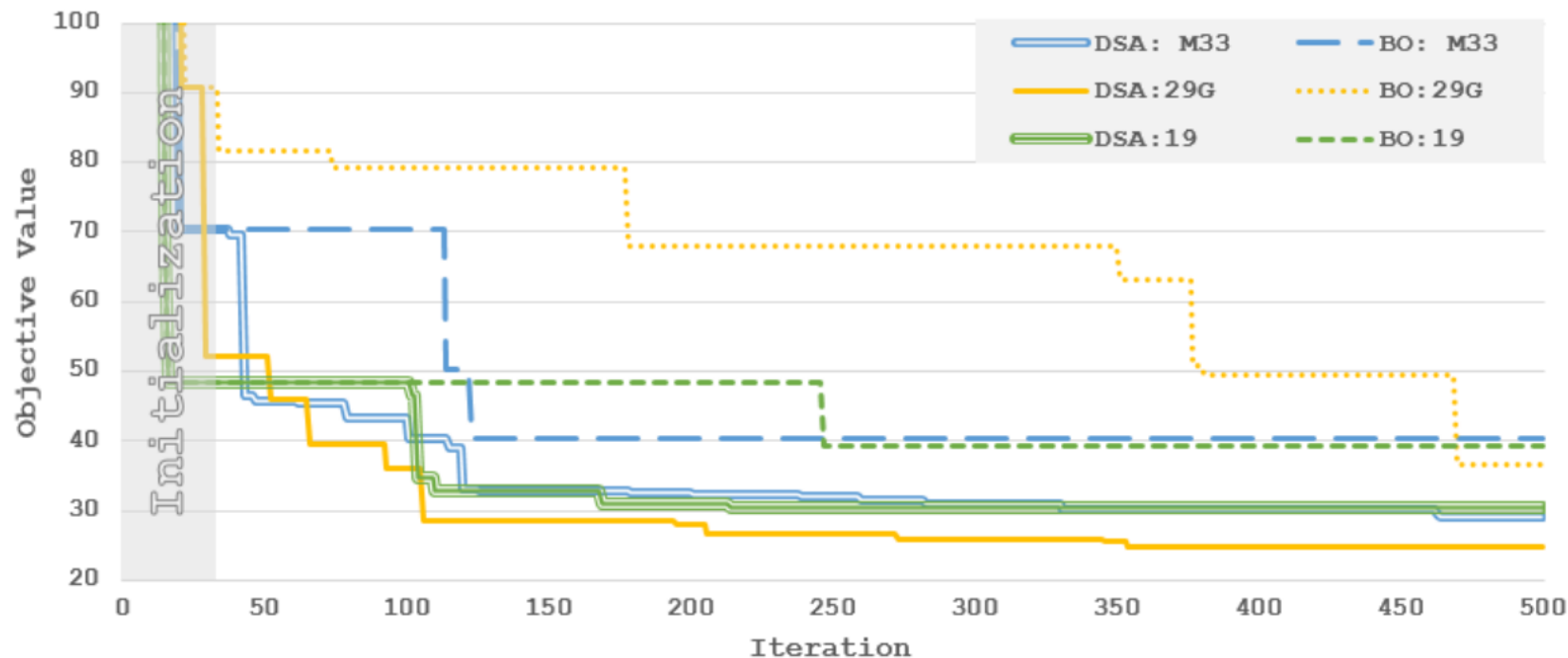
- 10 different variants of Algae yield models.
- 3 with 10 dimensions, 3 with 11 dimensions, 4 with 12 dimensions
- Online probability distribution for the DSA
- Squared Exponential kernel
- Expected Improvement policy
- Only 500 iterations
- One set of initial sampled data per experiment
- DSA built on top of pybo, Bayesian Optimisation library.
 - *Therefore same underlying code for the kernels, acquisitions functions and the Gaussian Processes.*

Results: Objective value

- Best achieved objective in four experiments using standard BO and DSA. Each experiment ran for 500 iterations
- The lowest objective value achieved lets us compare the different Algae models

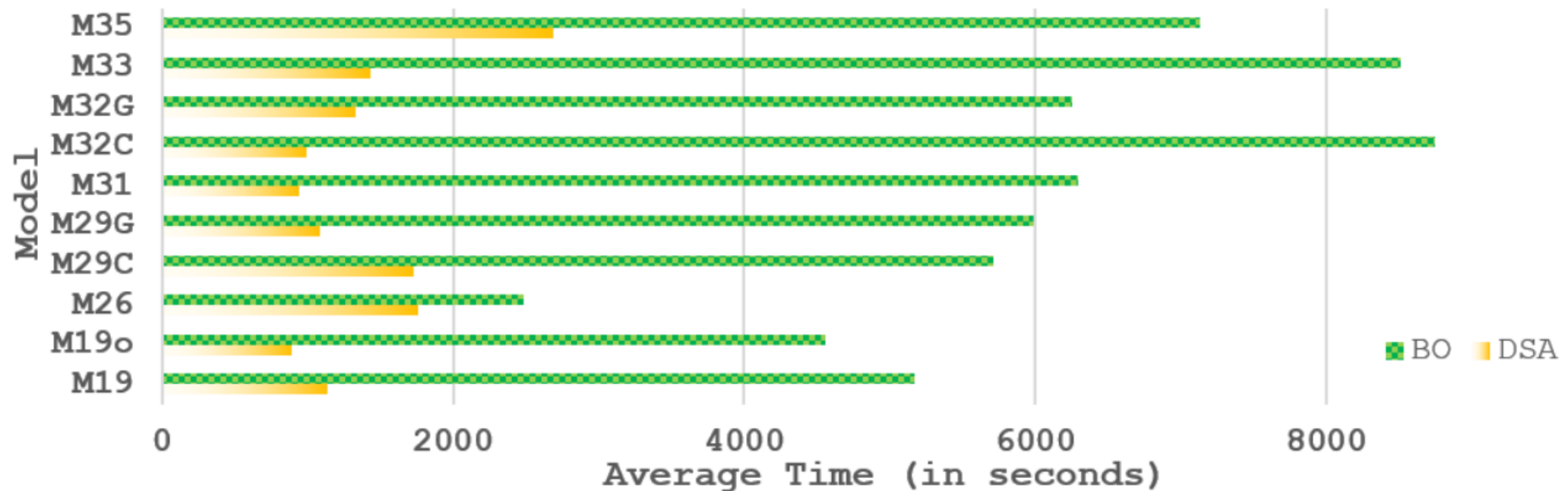
Model	d	BO:1	DSA:1	BO:2	DSA:2	BO:3	DSA:3	BO:4	DSA:4
M19	10	54.62	32.32	52.66	47.20	39.35	30.46	41.70	24.48
M19o	10	77.78	66.09	114.78	354.81	78.64	133.36	80.05	69.54
M26	11	87.91	35.86	32.56	26.66	55.40	31.53	57.58	26.54
M29G	11	46.32	59.77	72.19	35.16	86.61	77.40	73.62	64.59
M29C	11	36.50	24.78	44.71	37.25	41.47	34.59	52.74	33.77
M31	11	39.24	24.37	60.39	29.75	49.52	26.26	44.18	48.53
M32C	12	43.33	33.14	52.19	31.15	50.62	28.02	51.13	26.18
M32G	12	42.51	27.12	47.55	26.78	52.38	31.45	53.21	23.54
M33	12	44.07	24.82	38.78	24.64	40.26	29.09	46.68	20.61
M35	12	50.16	25.86	57.72	22.86	57.06	28.97	57.00	40.11

Results: Objective value



- Best achieved objective in four experiments using standard BO and DSA. Each experiment ran for 500 iterations

Results: Running Times



- Average computation time of DSA and traditional BO for all models. DSA results in a substantial computational speed-up.

Further Work and Limitations

- Current state of the algorithm does not attempt to reduce the total regret accumulated through out the function evaluations.
- The algorithm dimension selection is based on randomness, as a result there is no 100% guarantee on the performance of the algorithm in terms of the objective value. Further work is required to define a deterministic dimension selection algorithm.
- Global Gaussian Process potentially can be used to determine if the solution generated by the solver on reduced dimension should be evaluated or not to reduce the total regret.
- Parallelised and Distributed implementation of the Algorithm
 - Combined with Distributed Gaussian Processes for further Performance Boost (M. P. Deisenroth 2015)

The Framework

Bayesian Optimisation Framework GUI

Available at github.com/udoniyor/projectb

- Project B.(ayes) is a Graphical and Command Line interface on top a of Bayesian Optimisation framework pybo by Mathew Hoffman (2014).
- The framework allows any command line simulation to be optimised via BO or DSA techniques.
- The interface provides:
 - *Graphical user interface to configure BO/DSA hyper-parameters*
 - *Visual feedback of the optimisation process*
 - *Easy sharing file format*
 - *Start and Pause and Continue of the optimisation process*

Framework: Graphical User interface

Available at github.com/udoniyor/projectb

The graphical user interface is divided into three phases.

Phase 1: Selection - user sets all the parameters for the model to be optimised.

- Users can set the parameters from a file (as specified prior)
- Provides the user with details on requirements, and performs validation as user enters the parameters
- Can set multiple parameter values

Phase 2: Observation –user can view current state of the optimisation

- Provides with a overview of the current state of the optimisation
- Shows information on current policy, kernel, iteration being optimised
- Can continue from previous file

Phase 3: Evaluation – provides with a complete overview of the optimization results

- User may query the GPs created during the optimisation process for the mean and variance of selected points

Framework: github.com/udoniyor/projectb

ProjectB

ProjectB: Selection

Import Settings

Export Settings

Advanced Settings

?

Model Parameters

Commandline command for the model:

`i.e. python C://Users/Desktop/model/final4.py`

Select your model I/O:

Select Input File

Select Input File

Select Output File

Select Output File

Set model bounds:

Set or View Bounds

From File

Set the Bayesian Optimization Parameters

Policies:

☒ Expected Improvement

☐ Probability Improvement

☐ Upper Confidence Bound

☐ Thompson

Kernels:

☒ Squared Exponential

☐ Matern 5

☐ Matern 3

☐ Matern 1

Iterations:

150

Objective:

☐ Minimize

☒ Maximize

Model Output Directory (Optional):

Output Directory

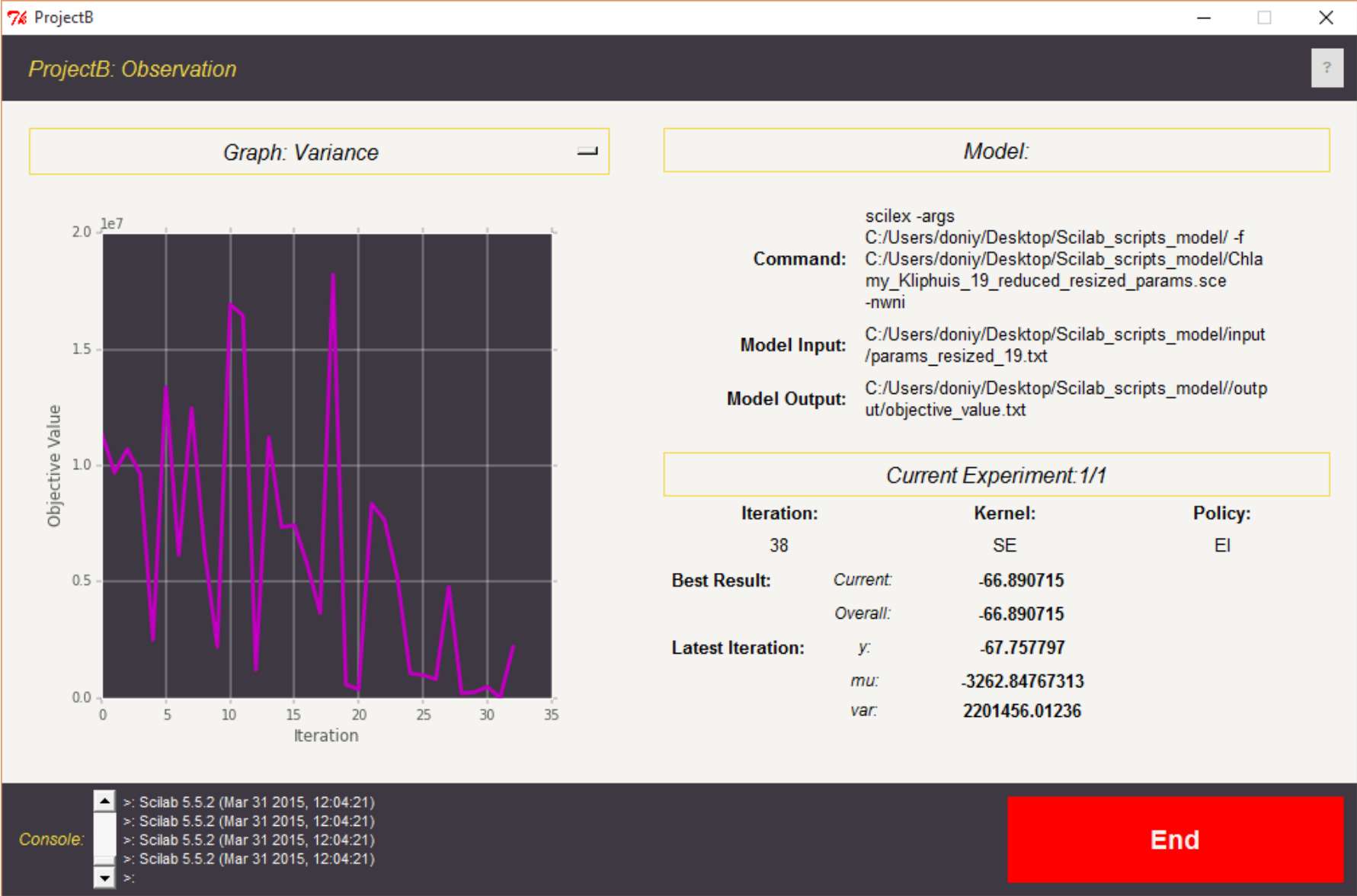
Select Directory

Console:

Welcome to Project Bi

Optimize

Framework: github.com/udoniyor/projectb



Framework: github.com/udoniyor/projectb

ProjectB

ProjectB: Evaluation

?

Model

Command:

scilex -args C:/Users/doniy/Desktop/Scilab_scripts_model/ -f C:/Users/doniy/Desktop/Scilab_scripts_model/Chlamy_Kliphuis_19_reduced_resized_params.sce -nwni

Model Input:

C:/Users/doniy/Desktop/Scilab_scripts_model/input/params_resized_19.txt

Model Output:

C:/Users/doniy/Desktop/Scilab_scripts_model/output/objective_value.txt

Results

#	Kernel:	Policy:	Iterations:	Best Result:	Time (s):	
1	SE	EI	45	-66.09366	129.47	Query

Console:

>: Saving the data...

>:

>: Finished experiment

>: Ended Bayesian Optimization

>: Now you can interact with the model(s)

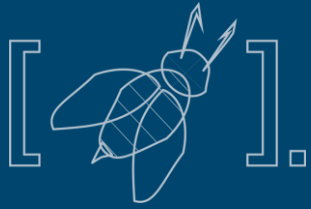
Save Console Logs

Output Directory

Exit

References:

- [1] Boukouvala, F., Muzzio, F. J. and Ierapetritou, M. G. (2010), Predictive modeling of pharmaceutical processes with missing and noisy data. *AIChE J.*, 56: 2860–2872. doi:10.1002/aic.12203
- [2] Calandra et al. (LION, 2014): Bayesian Gait Optimization for Bipedal Locomotion
- [3] Calandra et al. (AMAI, 2015): Bayesian Optimization for Learning Gaits under Uncertainty
- [4] C. Baroukh, R. M. noz Tamayo, J.-P. Steyer, and O. Bernard. DRUM: A New Framework for Metabolic Modeling under Non-Balanced Growth. Application to the Carbon Metabolism of Unicellular Microalgae. *PLOS One*, August 2014.
- [5] E. BROCHU, V. M. CORA, AND N. DE FREITAS, A tutorial on Bayesian optimization of expensive cost functions, (2010).
- [6] K. Kandasamy, J. Schneider, and B. Póczos. High Dimensional Bayesian Optimisation and Bandits via Additive Models. In *Proceedings of the International Conference on Machine Learning*, 2015.
- [7] M. P. Deisenroth, “Distributed Gaussian Processes” vol. 37, 2015
- [8] M. W. Hoffman and R. Shariari, “Modular mechanism for Bayesian optimization” *NIPS Work. Bayesian Optim*, pp. 1-5, 2014
- [9] M. M. Faruque Hasan, Richard C. Baliban, Josephine A. Elia, and Christodoulos A. Floudas , Modeling, Simulation, and Optimization of Postcombustion CO₂ Capture for Variable Feed Concentration and Flow Rate. 2. Pressure Swing Adsorption and Vacuum Swing Adsorption Processes , *Industrial & Engineering Chemistry Research* 2012 51 (48), 15665-15682
- [10] R. Calandra, J. Peters, A. Seyfarth, and M. P. Deisenroth. An Experimental Evaluation of Bayesian Optimization on Bipedal Locomotion. In *Proceedings of the International Conference on Robotics and Automation*, 2014.
- [11] Yidong Lang, Stephen E. Zitney, Lorenz T. Biegler, Optimization of IGCC processes with reduced order CFD models, *Computers & Chemical Engineering*, Volume 35, Issue 9, 14 September 2011, Pages 1705-1717, ISSN 0098-1354, <http://dx.doi.org/10.1016/j.compchemeng.2011.01.018>. (<http://www.sciencedirect.com/science/article/pii/S0098135411000287>)



Bayesian Optimisation with Dimension Scheduling Algorithm

Doniyor Ulmasov ^a, Dr Caroline Baroukh ^b, Dr Benoit Chachuat ^c,
Dr Marc Peter Deisenroth ^a, and Dr Ruth Misener ^a,

Questions?

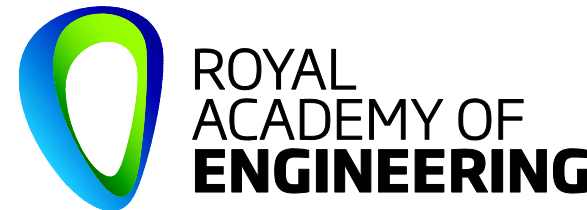
Full Thesis: <https://www.Doniyor.com/>

Framework: <https://github.com/udoniyor/projectb>

Project Partners:



Funding Sources:



EP/M028240/1