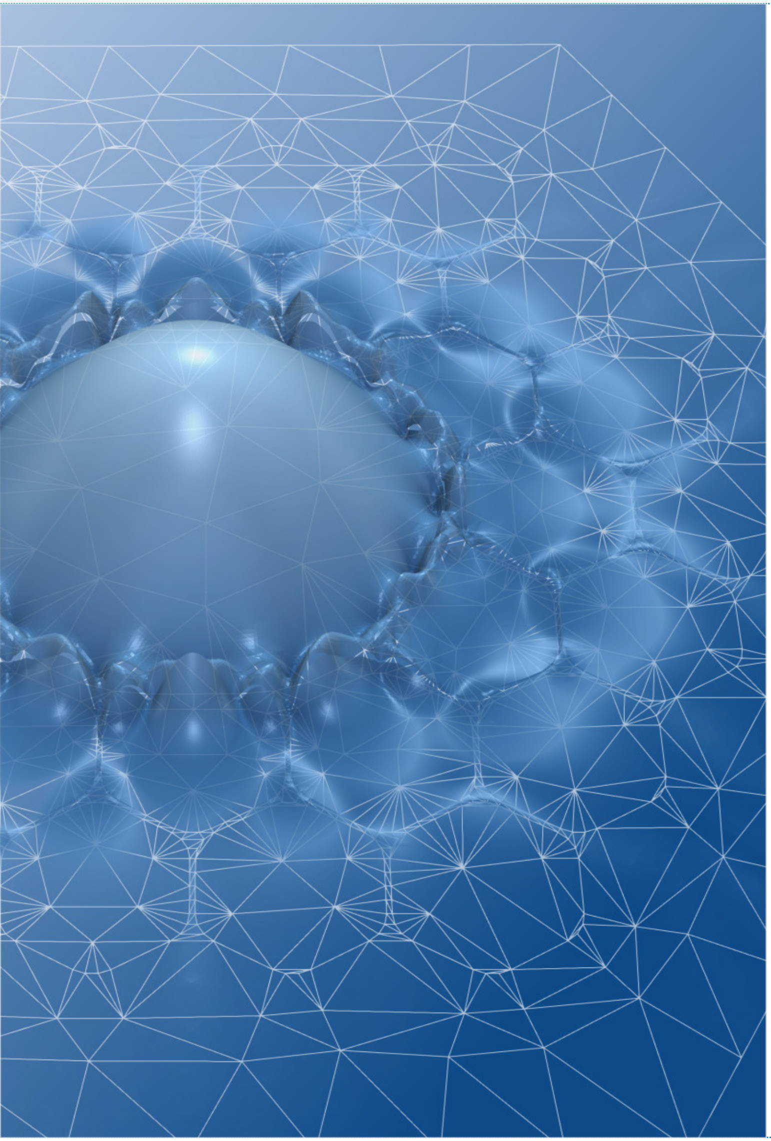


FEM-Simulation und maschinelles Lernen für die Analyse und Optimierung nanooptischer Systeme

Philipp Schneider (CEO)

VIPO - Symposium 2021

09. Juli 2021

JCMwave

JCMwave GmbH

- Founded in 2002 as Spin-Off from Zuse-Institute Berlin (ZIB)
- Simulation software and expertise for photonic applications.
- Benefits from leading edge research in numerical mathematics and computer science (5 running R&D projects)

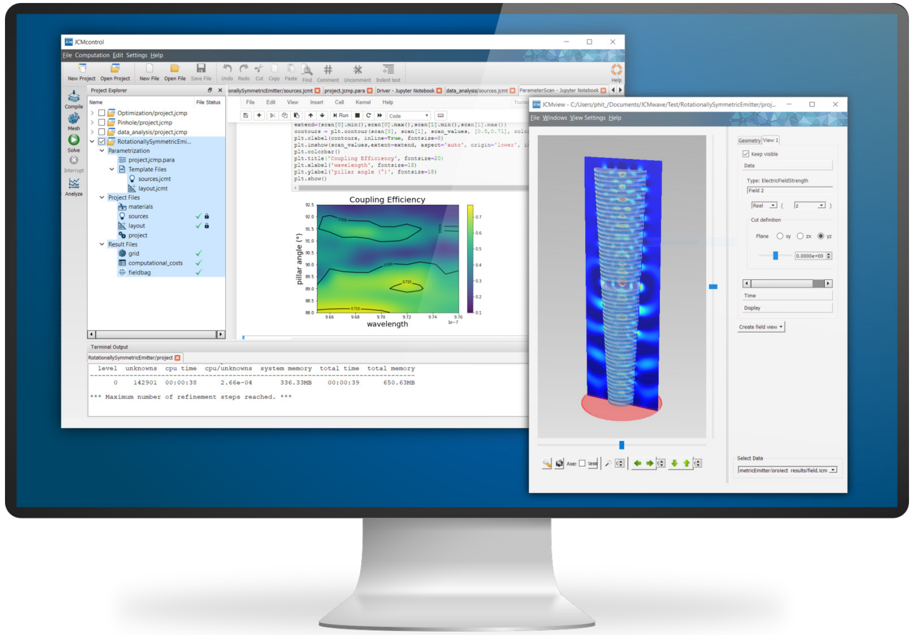
James Clerk Maxwell

JCMwave

JCMsuite - Simulation Suite for Nano-Optics

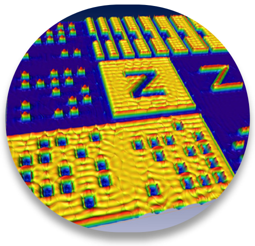
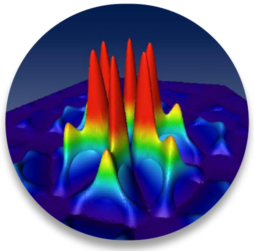
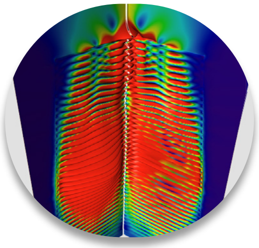
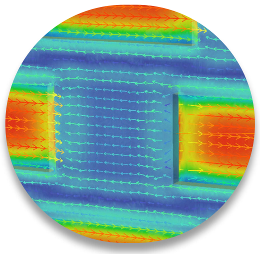
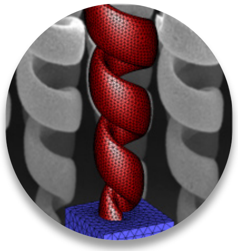
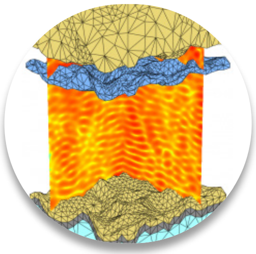
JCMsuite contains all modules for the complete workflow for nano-optics:

- CAD and meshing of complex geometries
- **Efficient and accurate hp-FEM solver**
- Post-processing and visualization
- Interfaces to Matlab® and Python, automatic script generation
- **Analysis and optimization toolkit**

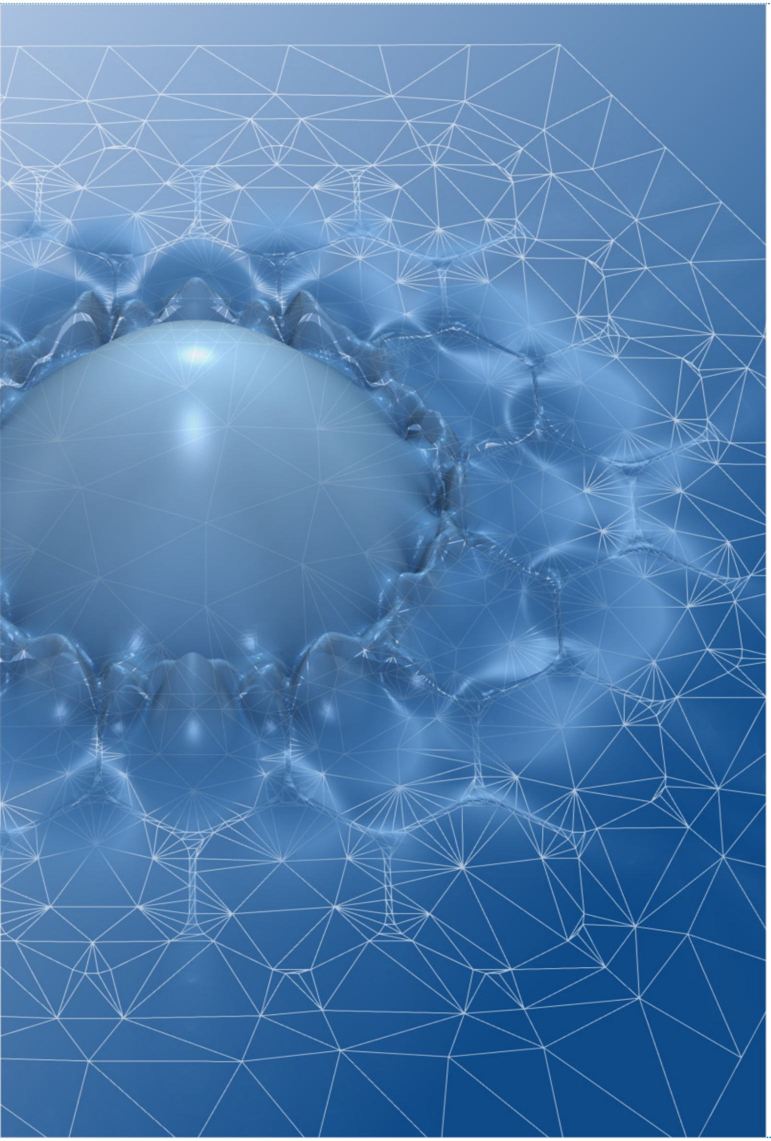


Typical application fields include:

- Computational lithography and metrology
- Waveguides and fibers
- Light sources and sensors
- Nanostructured materials and metasurfaces

JCMwave



Vorteile der Finite Elemente Methode (FEM)

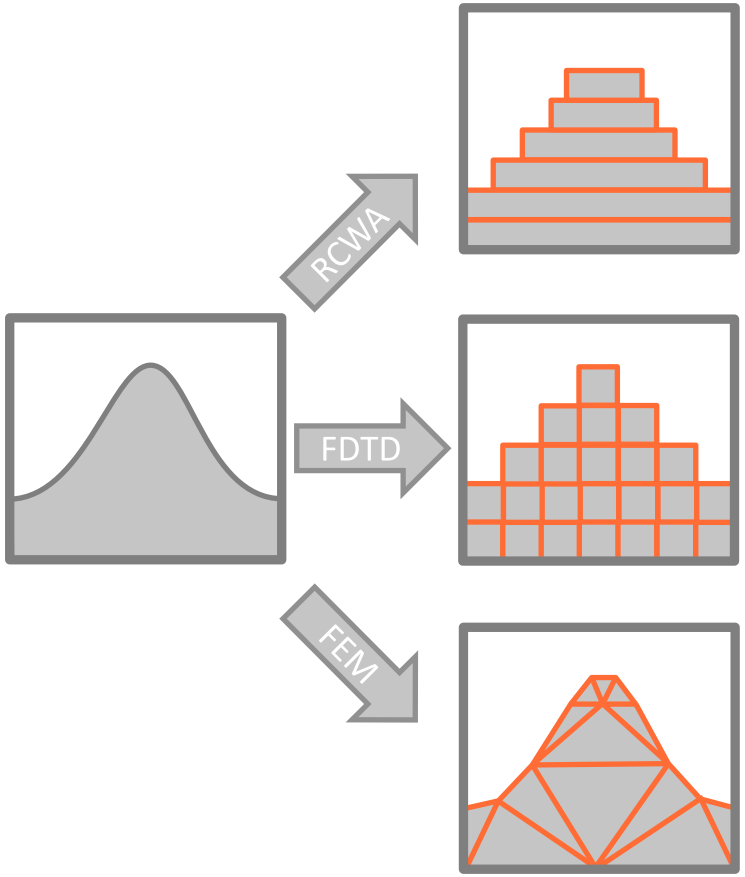
How to solve Maxwell's equations?

Various methods are used to solve Maxwell's equations rigorously, e.g.:

- **RCWA** (rigorous coupled wave analysis): The geometry is discretized into individual **layers**. The diffraction of incident plane waves at the structure is calculated. [Wikipedia]

- **FDTD** (finite difference time-domain method): The geometry is discretized into **uniform patches** (squares, cubes). The equations are solved in a time and space discrete manner. [Wikipedia]

- **FEM** (finite element method): The geometry is discretized into **variable shapes** like triangles, tetrahedrons, prisms.

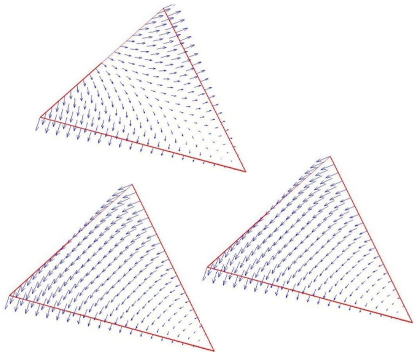


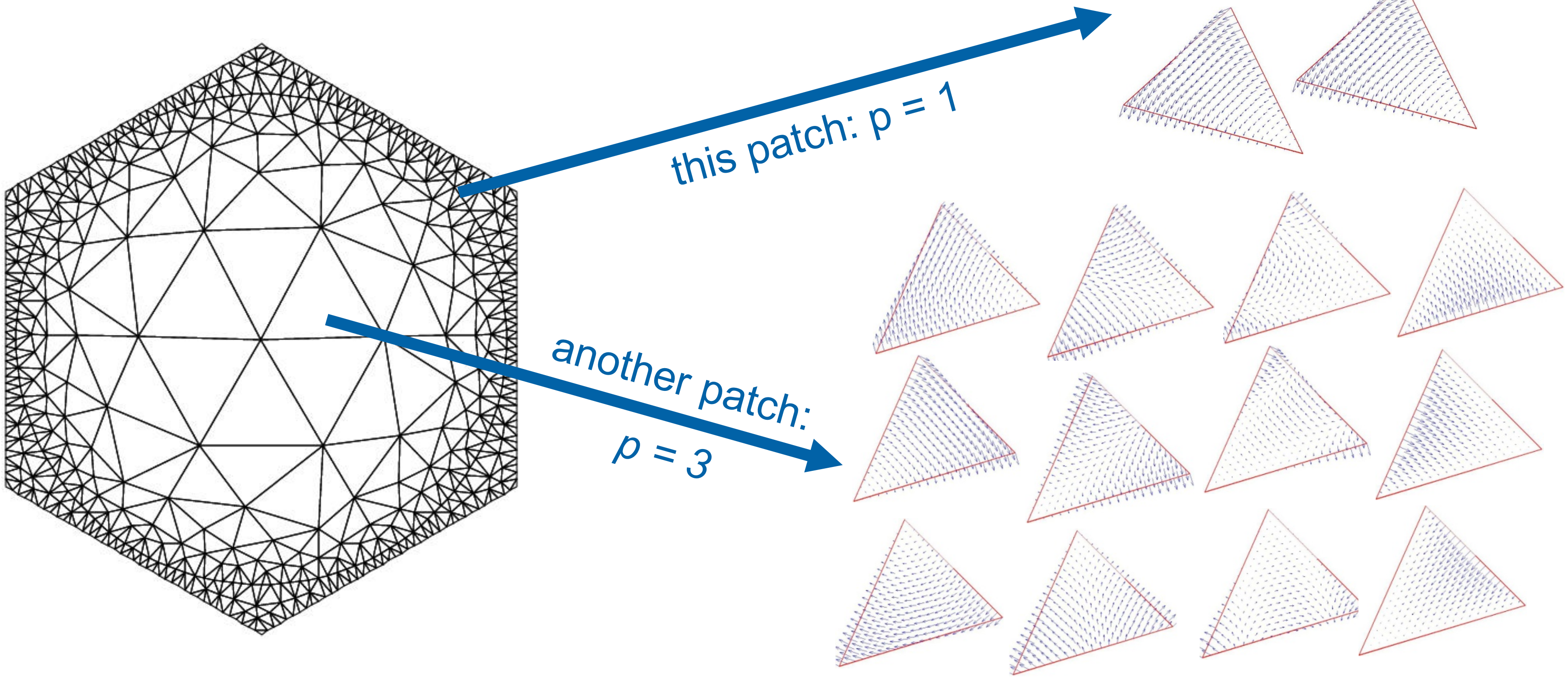
Hp-Finite Element Method

FEM numerical parameters: ***h*** - triangle size, ***p*** - polynomial order

Suitable non-uniform combination of ***p*** and ***h*** refinements

leads to superior convergence [Babuska, 1992]





Comparison of convergence speed

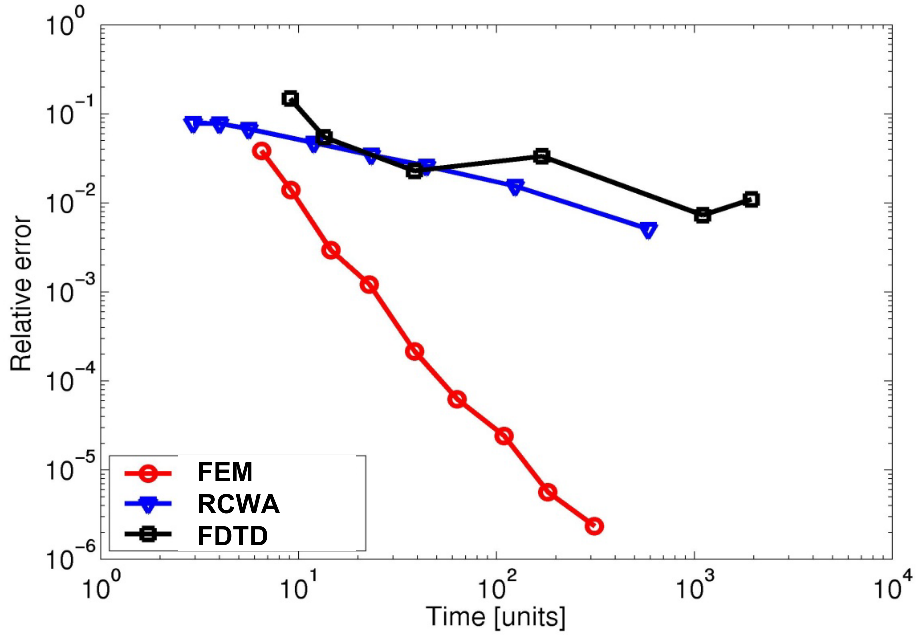
Comparison:

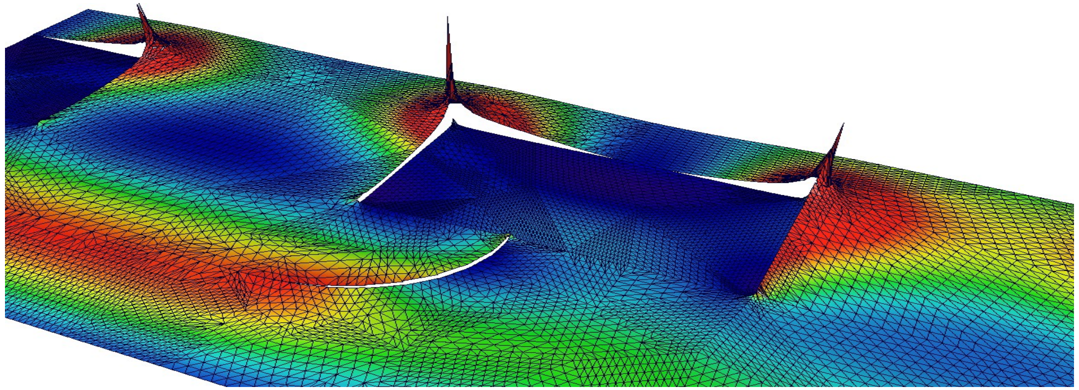
FEM vs. FDTD vs. RCWA

Benchmark Problem:

Rigorous Mask Simulation for Lithography

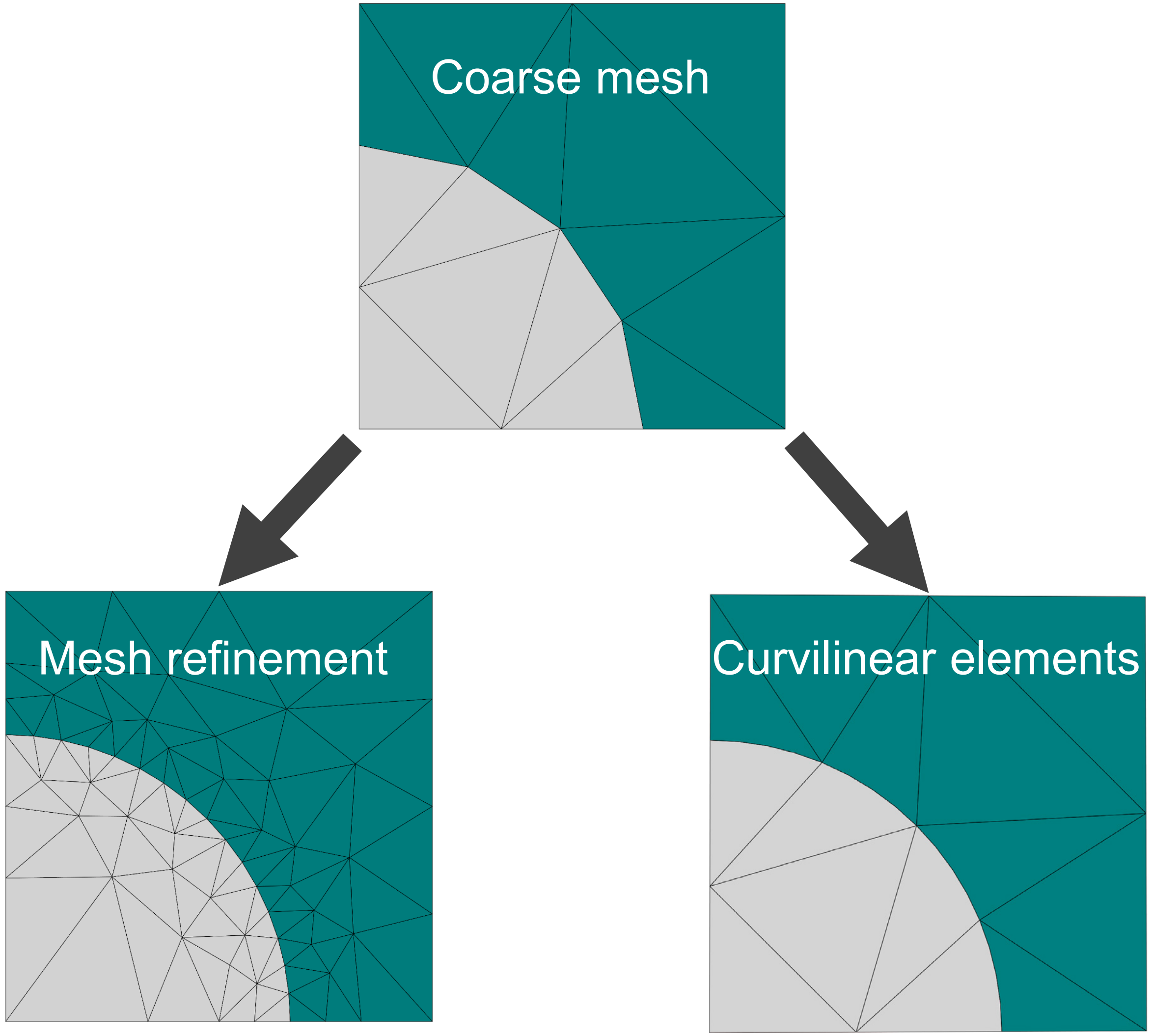
FEM faster and more accurate by orders of magnitude

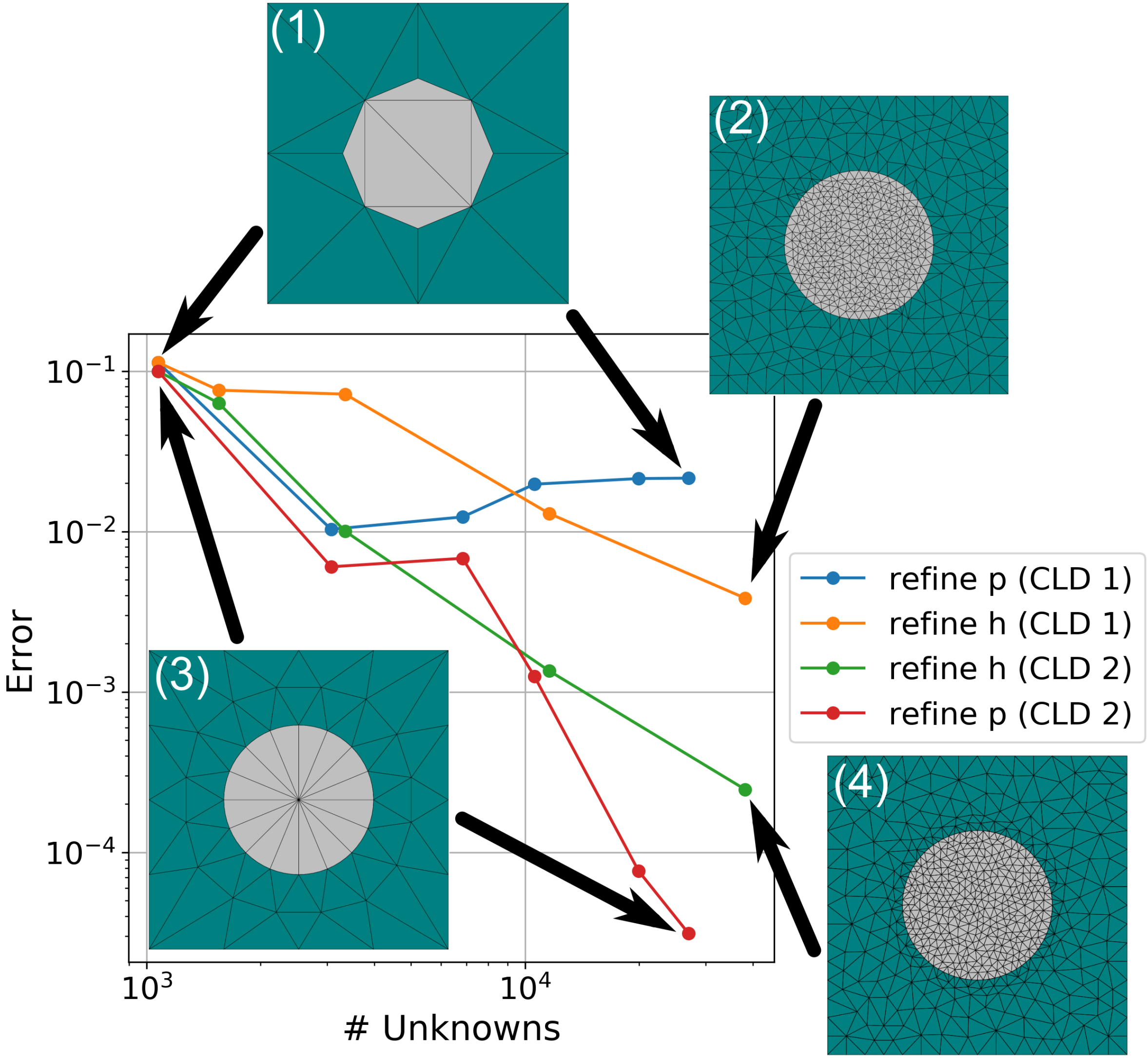


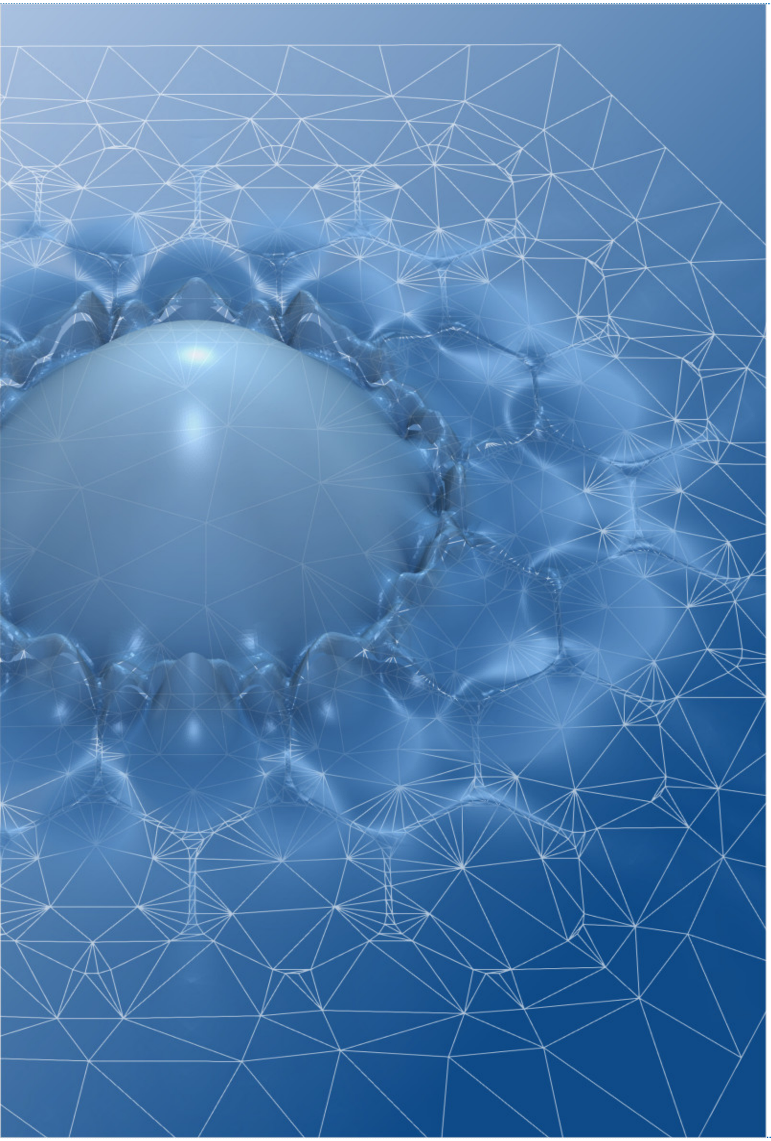


[Benchmark of FEM, Waveguide and FDTD Algorithms for Rigorous Mask Simulation.
Proc. SPIE 5992, 368, 2005.]

New: Curvilinear elements





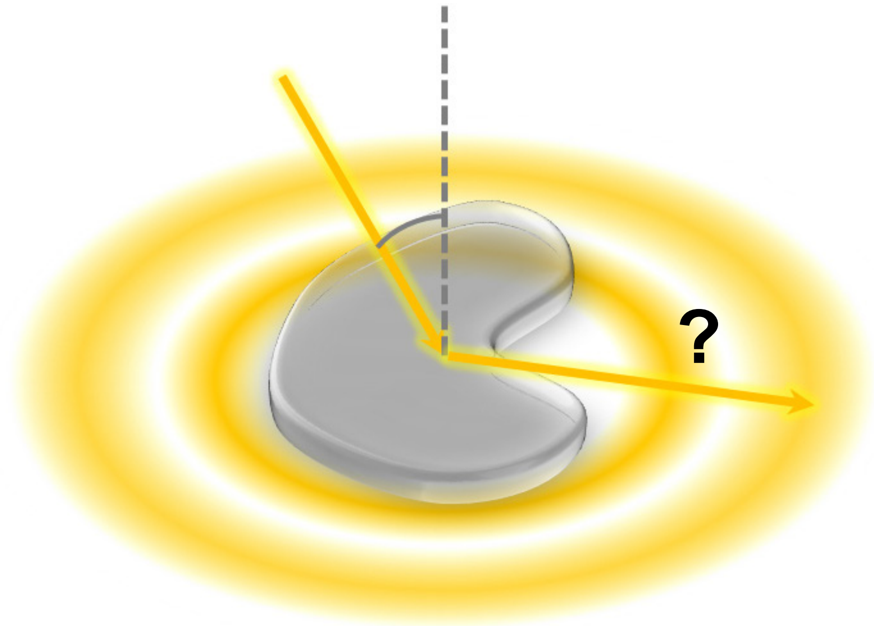


Maschinelles Lernen für die effiziente Analyse und Optimierung

Computational challenges in nano-optics

Optical behavior of small structures (e.g. scattering in certain direction) dominated by diffraction, interference and resonance phenomena

- Solution of Maxwell’s equation required: expensive black-box function
- Multi-modal behavior of objectives



Regression models (small selection)

Increasing predictive power and computational demands

K-nearest neighbors

Linear regression

Support vector machine

Random forest trees

Gaussian process regression (Kriging)

(Deep) neural networks

+ Accurate and data efficient
+ Reliable (provides uncertainties)
+ Interpretable results
– Computationally demanding but not as much as training neural networks

CE Rasmussen, “Gaussian processes in machine learning”. Advanced lectures on machine learning , Springer (2004)

Gaussian process regression

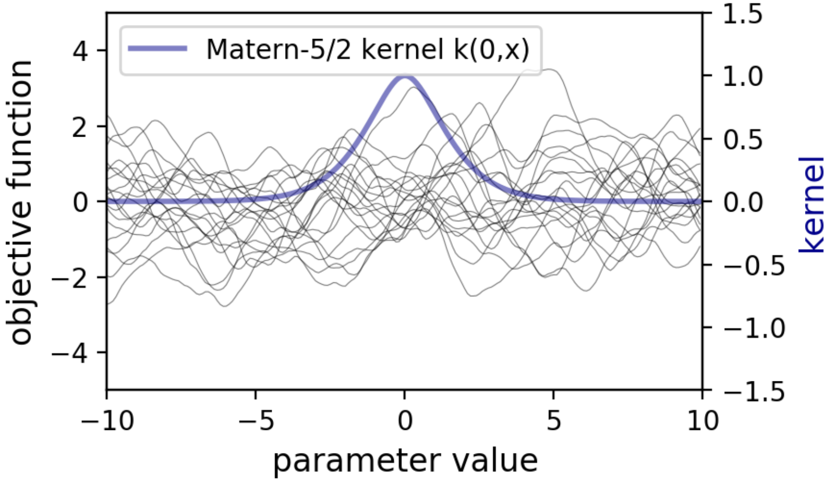
- **Gaussian process (GP)**: distribution of functions in a continuous domain $\mathcal{X} \subset \mathbb{R}^{\mathrm{N}}$
- **Defined by:** mean function $\mu: \mathcal{X} \to \mathbb{R}$ and covariance function (kernel) $k: \mathcal{X} \times \mathcal{X} \to \mathbb{R}$
- **Training data:** M known function values $f(x_1), \dots, f(x_M)$ and corresponding covariance matrix $\mathbf{K} = \left[k(x_i, x_j)\right]_{i,j}$
- **Prediction** for $x^* \in \mathcal{X}$: normal distribution $y \sim \mathcal{N}(\bar{y}, \sigma^2)$ with mean and standard deviation

$$\bar{y} = \mu(x^*) + \sum_{ij} k(x^*, x_i)\, \mathbf{K}_{ij}^{-1}[f(x_j) - \mu(x_j)]$$

$$\sigma^2 = k(x^*, x^*) - \sum_{ij} k(x^*, x_i)\ \mathbf{K}_{ij}^{-1}\, k(x_j, x^*)$$

GP hyperparameters

The mean and covariance function are usually parametrized as



Matern-5/2 function

$$\mu(x) = \mu_0$$

$$k(x, x') = \sigma^2 C_{5/2}(r) = \sigma^2 \left(1 + \sqrt{5}r + \frac{5}{3}r^2\right) \exp\left(-\sqrt{5}r\right)$$

with $r^2 = \sum_i (x_i - x_i')^2 / l_i^2$

Take values of μ_0, σ, l_i are maximized w.r.t. the log-likelihood of the observations:

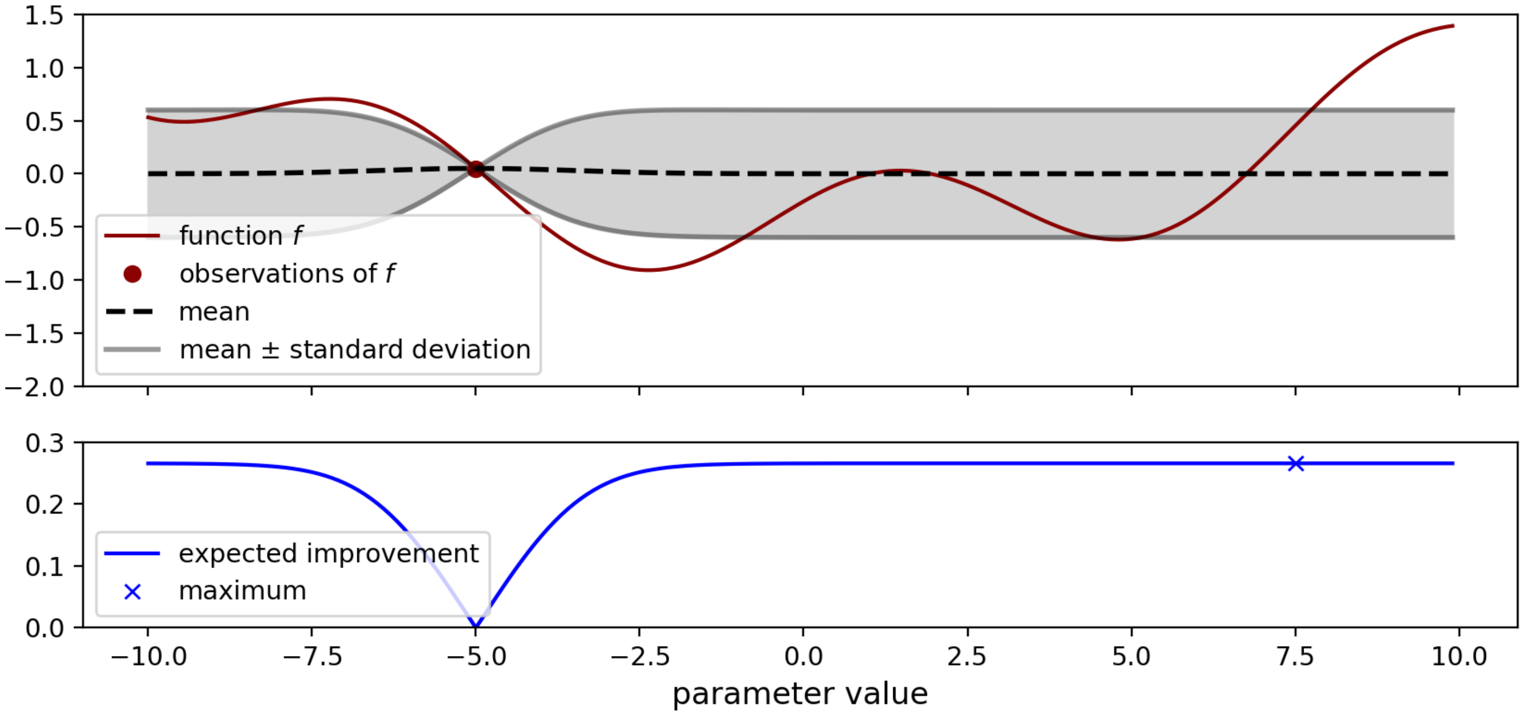
$$\log P(\mathbf{Y}) = -\frac{M}{2}\log(2\pi) - \frac{1}{2}\log(|\mathbf{K}|) - \frac{1}{2}(\mathbf{Y} - \boldsymbol{\mu})^T \mathbf{K}^{-1}(\mathbf{Y} - \boldsymbol{\mu})$$

Bayesian optimization

Problem: Find parameters $x \in \mathcal{X}$ that minimize $f(x)$. For the currently known smallest function value y_{min} we define the improvement

$$I(y) = \begin{cases} 0 & : y \geq y_{min} \\ y_{min} - y & : y < y_{min} \end{cases}$$

Strategy: Sample at points of largest expected improvement $\alpha_{\mathrm{EI}}(x) = \mathbb{E}[I(y)]$ (analytic function derived from normal distribution of y)



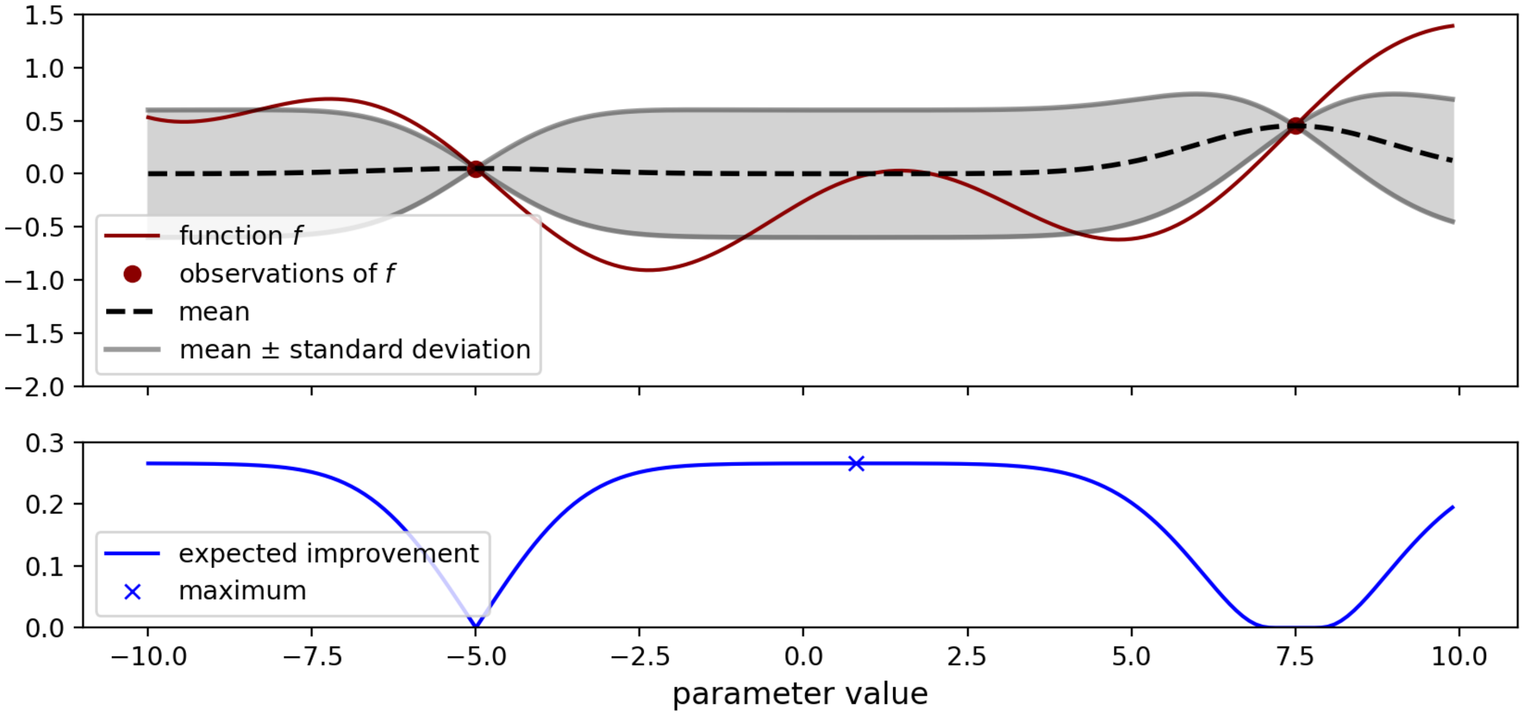
JCMwave

Bayesian optimization

Problem: Find parameters $x \in \mathcal{X}$ that minimize $f(x)$. For the currently known smallest function value y_{min} we define the improvement

$$I(y) = \begin{cases} 0 & : y \geq y_{min} \\ y_{min} - y & : y < y_{min} \end{cases}$$

Strategy: Sample at points of largest expected improvement $\alpha_{\mathrm{EI}}(x) = \mathbb{E}[I(y)]$ (analytic function derived from normal distribution of y)

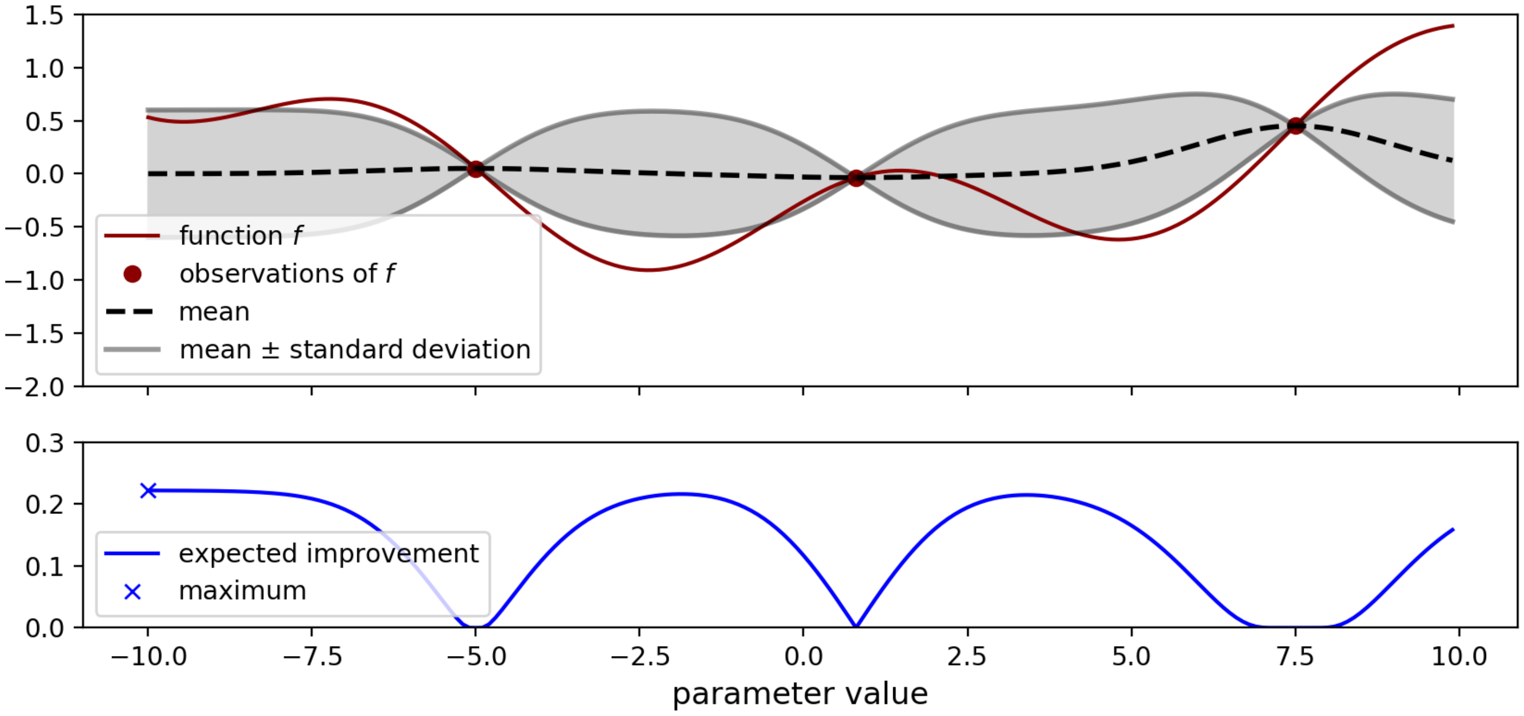


Bayesian optimization

Problem: Find parameters $x \in \mathcal{X}$ that minimize $f(x)$. For the currently known smallest function value y_{min} we define the improvement

$$I(y) = \begin{cases} 0 & : y \geq y_{min} \\ y_{min} - y & : y < y_{min} \end{cases}$$

Strategy: Sample at points of largest expected improvement $\alpha_{\mathrm{EI}}(x) = \mathbb{E}[I(y)]$ (analytic function derived from normal distribution of y)

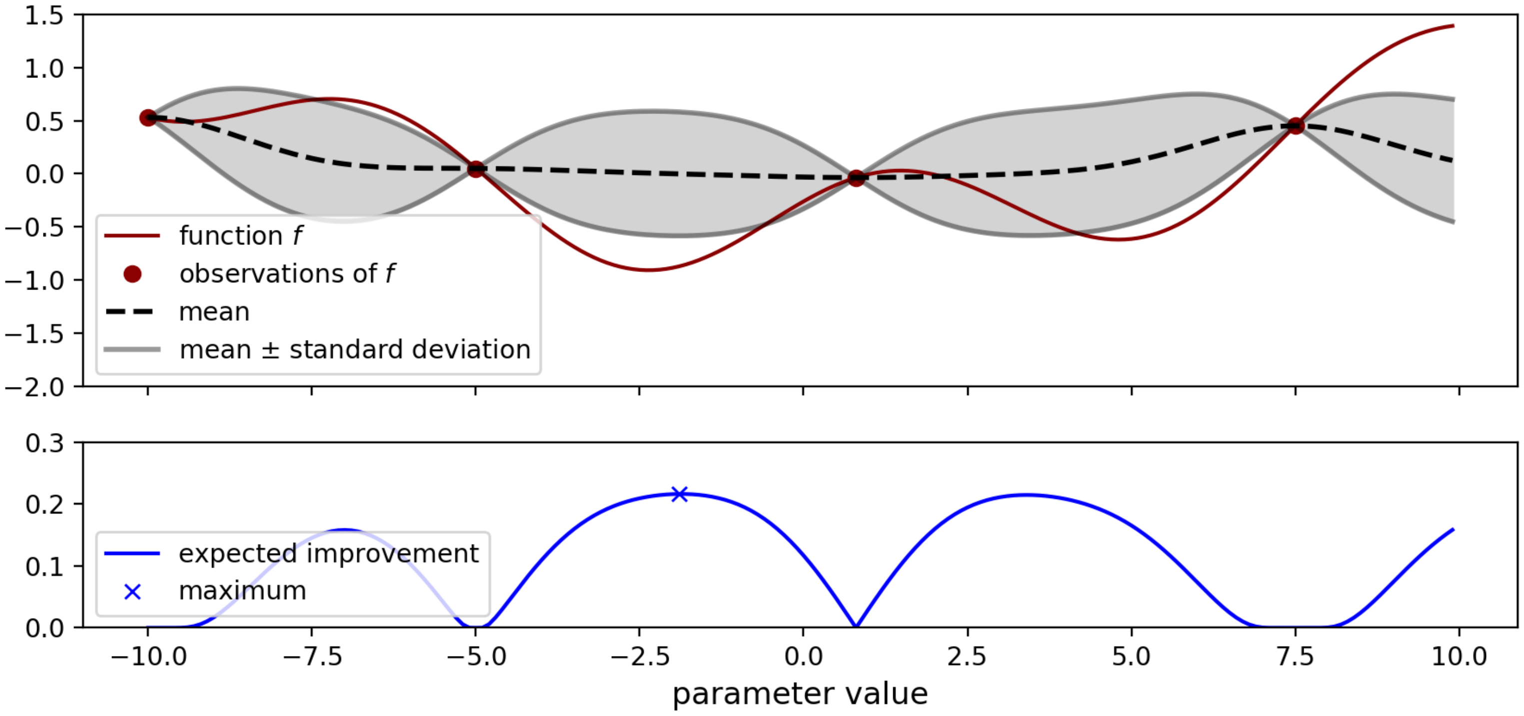


Bayesian optimization

Problem: Find parameters $x \in \mathcal{X}$ that minimize $f(x)$. For the currently known smallest function value y_{min} we define the improvement

$$I(y) = \begin{cases} 0 & : y \geq y_{min} \\ y_{min} - y & : y < y_{min} \end{cases}$$

Strategy: Sample at points of largest expected improvement $\alpha_{\mathrm{EI}}(x) = \mathbb{E}[I(y)]$ (analytic function derived from normal distribution of y)

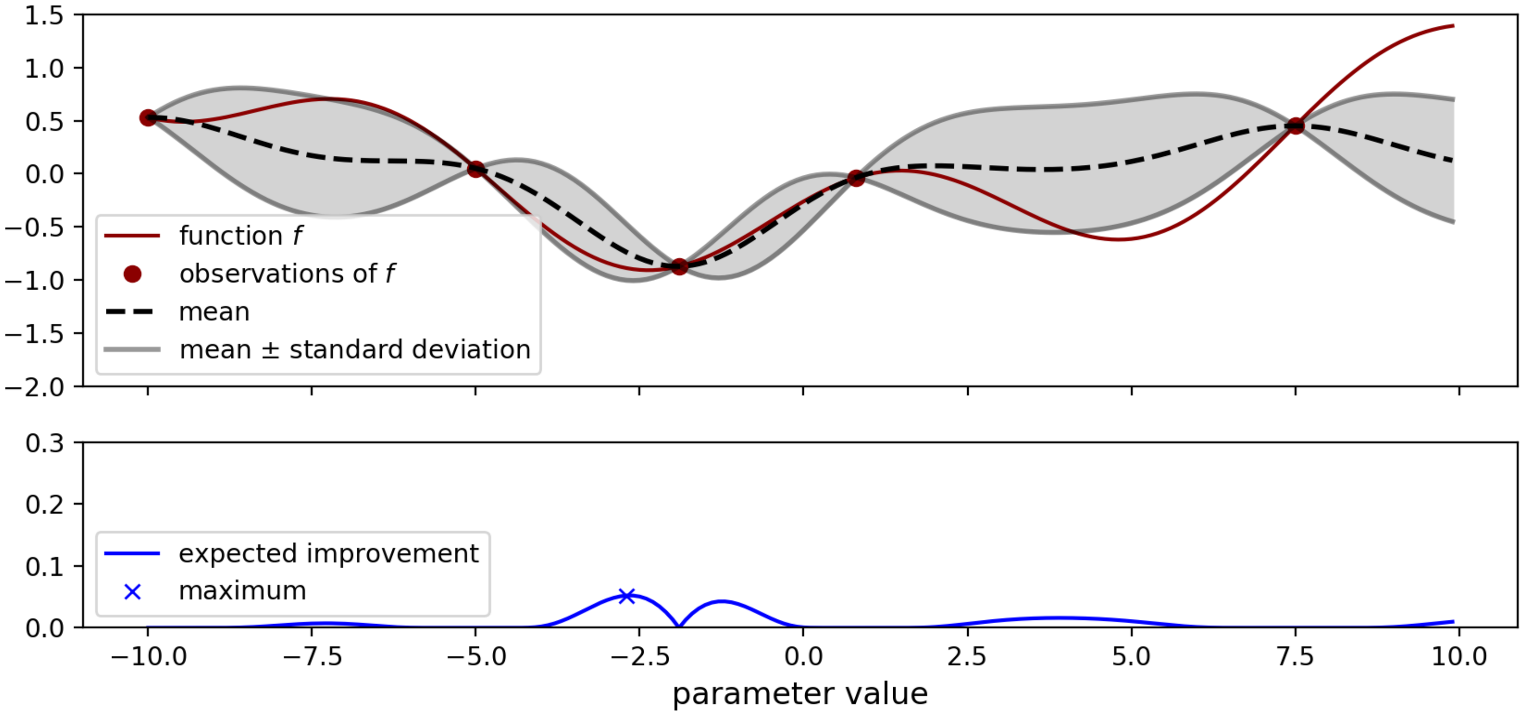


Bayesian optimization

Problem: Find parameters $x \in \mathcal{X}$ that minimize $f(x)$. For the currently known smallest function value y_{min} we define the improvement

$$I(y) = \begin{cases} 0 & : y \geq y_{min} \\ y_{min} - y & : y < y_{min} \end{cases}$$

Strategy: Sample at points of largest expected improvement $\alpha_{\mathrm{EI}}(x) = \mathbb{E}[I(y)]$ (analytic function derived from normal distribution of y)



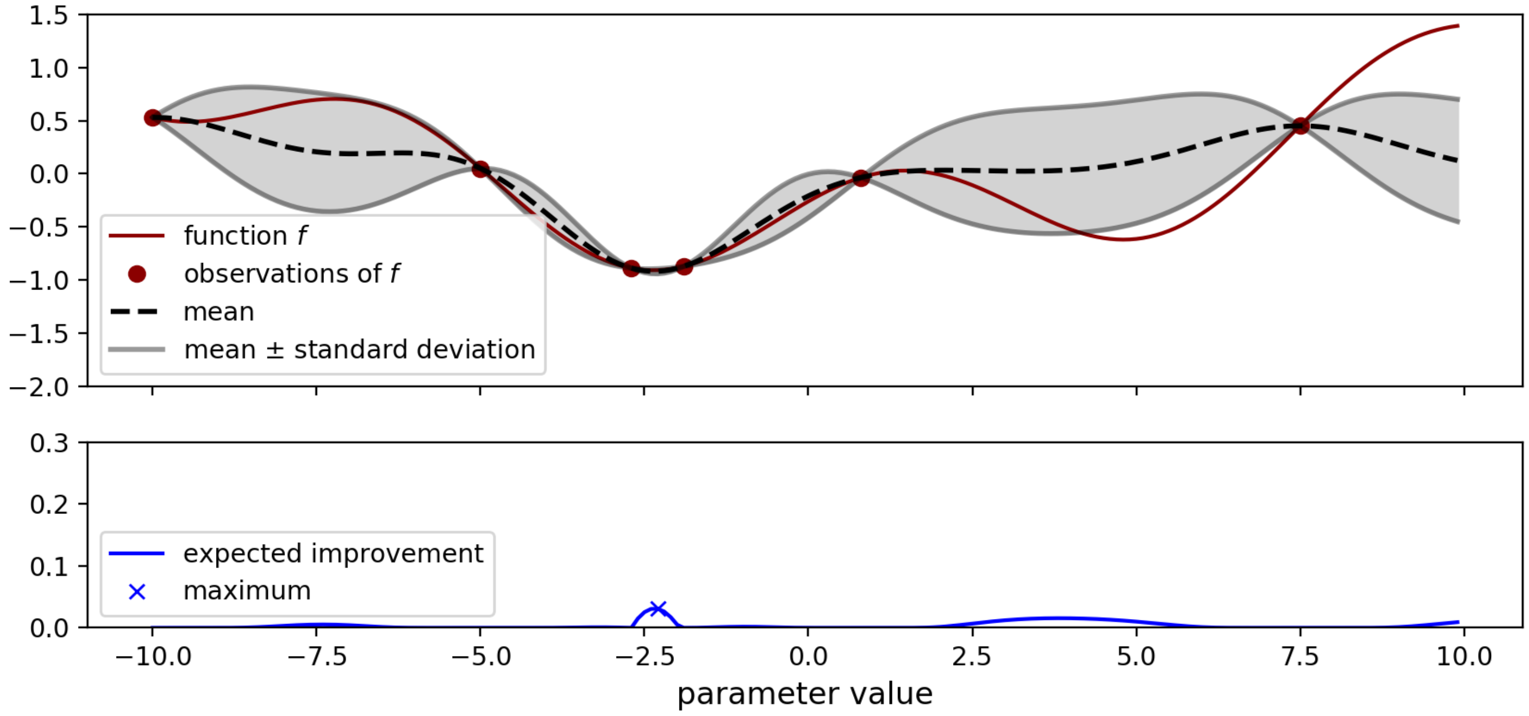
JCMwave

Bayesian optimization

Problem: Find parameters $x \in \mathcal{X}$ that minimize $f(x)$. For the currently known smallest function value y_{min} we define the improvement

$$I(y) = \begin{cases} 0 & : y \geq y_{min} \\ y_{min} - y & : y < y_{min} \end{cases}$$

Strategy: Sample at points of largest expected improvement $\alpha_{\mathrm{EI}}(x) = \mathbb{E}[I(y)]$ (analytic function derived from normal distribution of y)



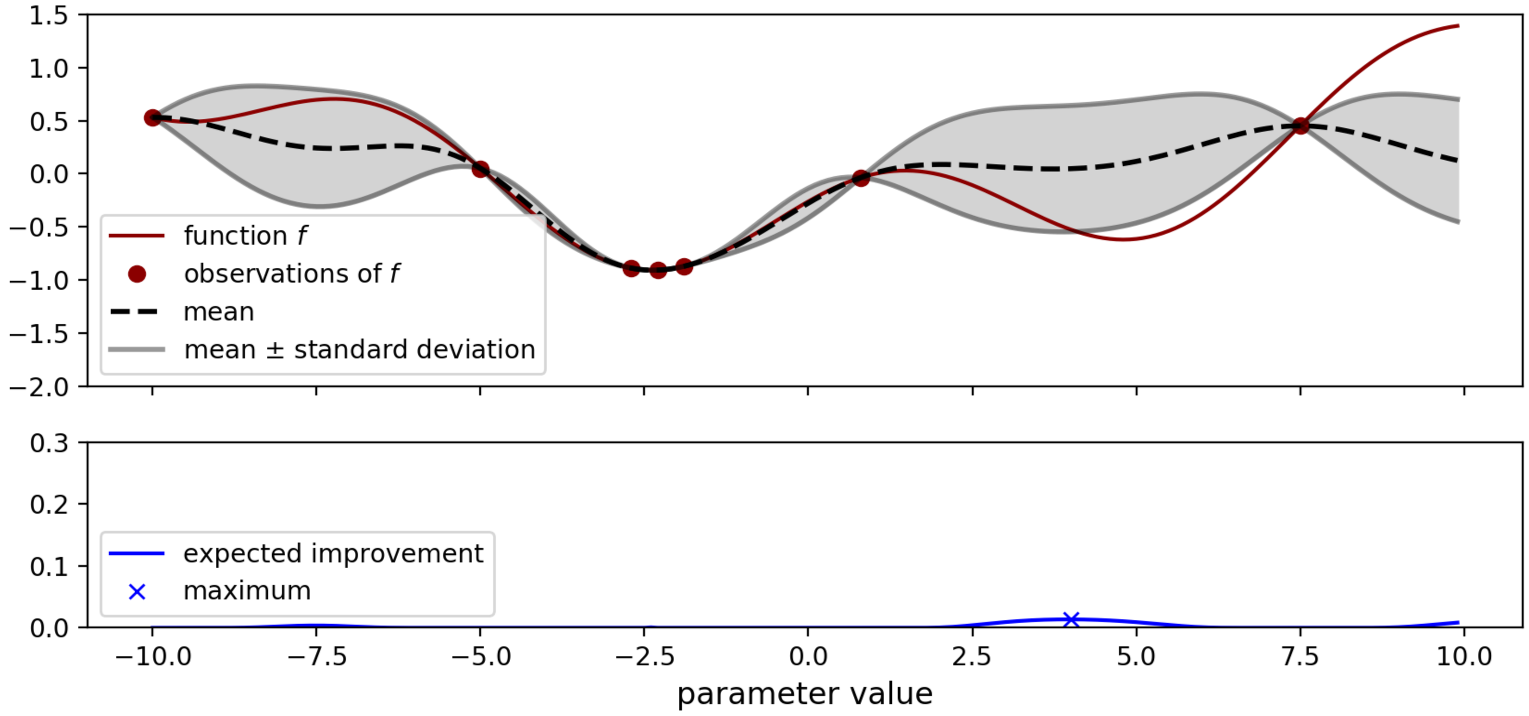
JCMwave

Bayesian optimization

Problem: Find parameters $x \in \mathcal{X}$ that minimize $f(x)$. For the currently known smallest function value y_{min} we define the improvement

$$I(y) = \begin{cases} 0 & : y \geq y_{min} \\ y_{min} - y & : y < y_{min} \end{cases}$$

Strategy: Sample at points of largest expected improvement $\alpha_{\mathrm{EI}}(x) = \mathbb{E}[I(y)]$ (analytic function derived from normal distribution of y)

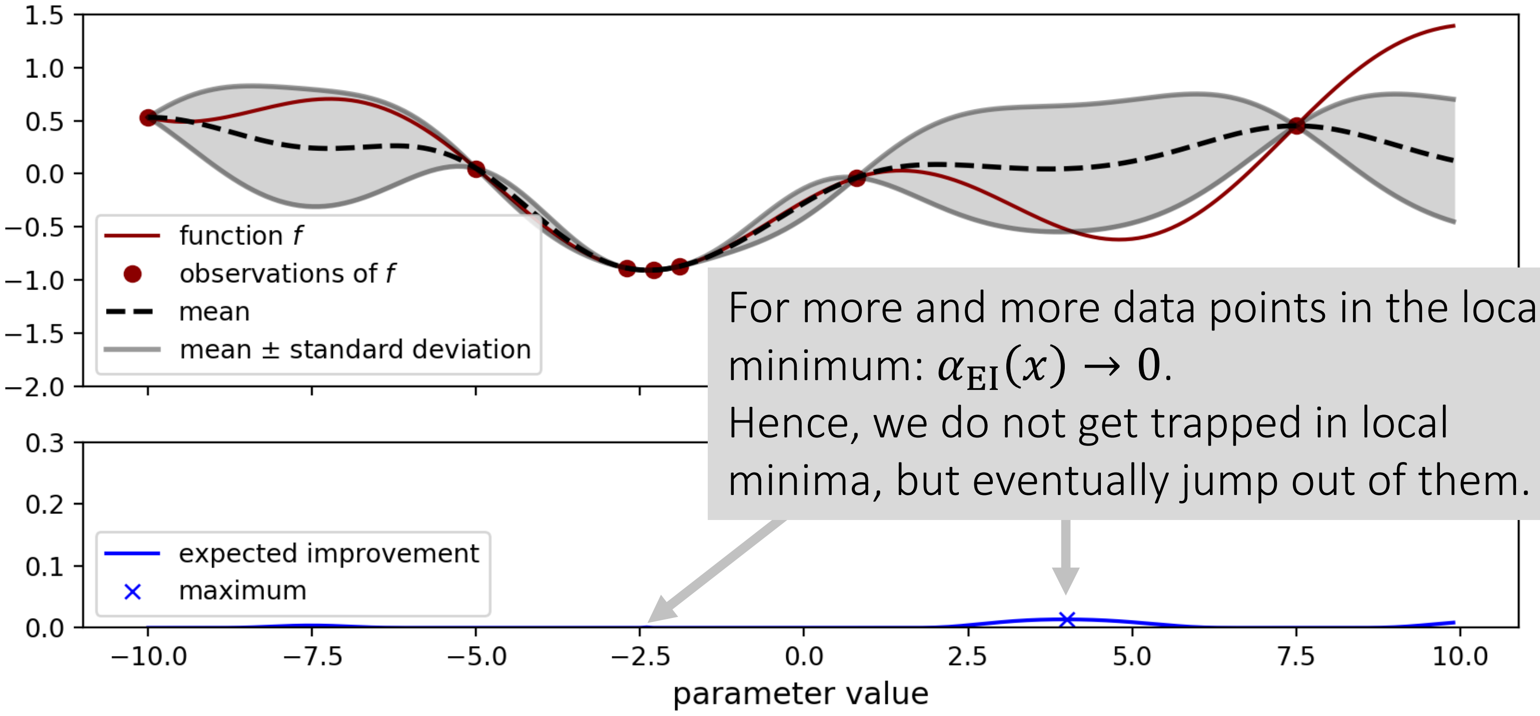


Bayesian optimization

Problem: Find parameters $x \in \mathcal{X}$ that minimize $f(x)$. For the currently known smallest function value y_{min} we define the improvement

$$I(y) = \begin{cases} 0 & : y \geq y_{min} \\ y_{min} - y & : y < y_{min} \end{cases}$$

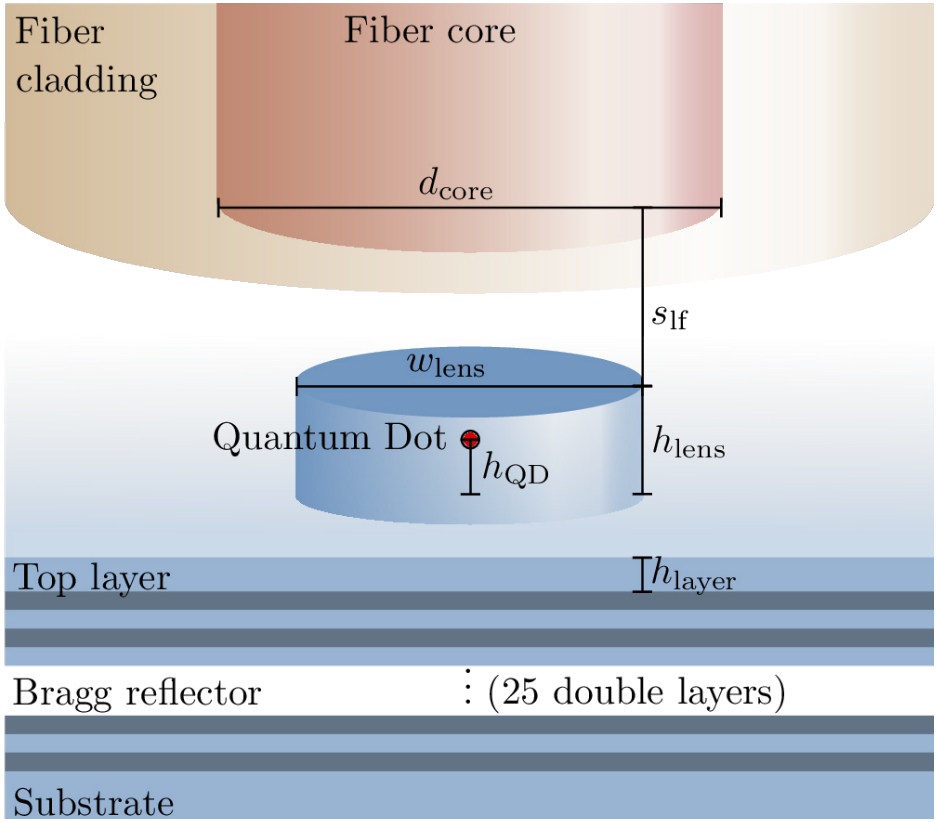
Strategy: Sample at points of largest expected improvement $\alpha_{\mathrm{EI}}(x) = \mathbb{E}[I(y)]$ (analytic function derived from normal distribution of y)



For more and more data points in the local minimum: $\alpha_{\mathrm{EI}}(x) \to 0$.
Hence, we do not get trapped in local minima, but eventually jump out of them.

JCMwave

Example: Single photon source



[Optics Express **26**, 8479 (2018)]

System

- A **quantum-dot emitter** embedded into a **mesa** structure (GaAs, blue),
- a **Bragg reflector** made of alternating layers of GaAs (blue) and AlGaAs (gray),
- an **optical fiber** with homogeneous fiber core (orange) and fiber cladding (yellow).

Optimization task

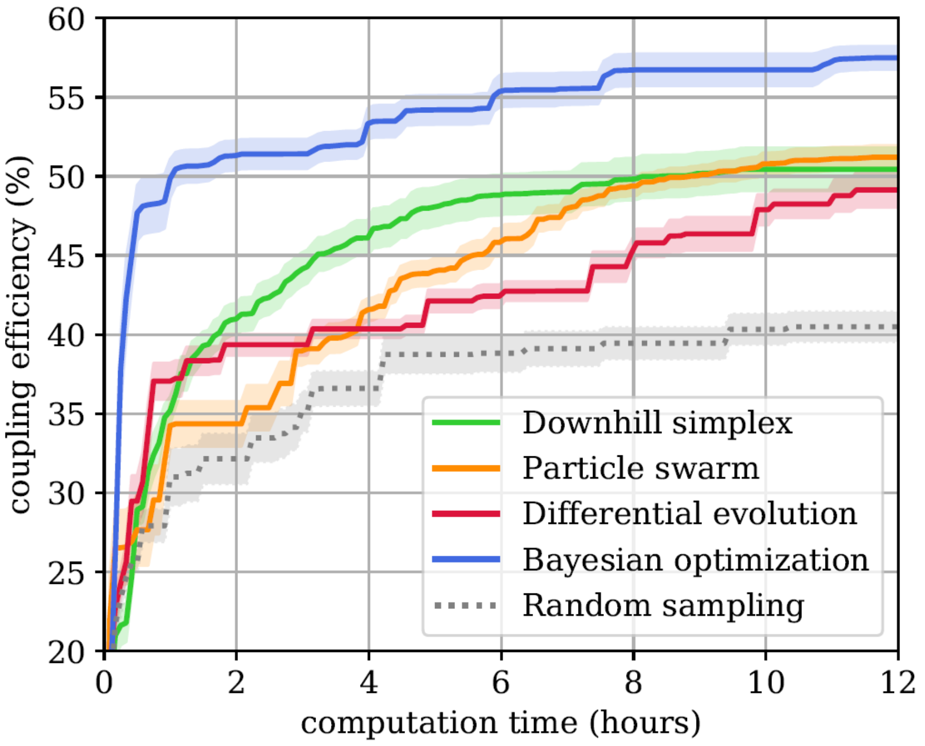
Maximization of the **coupling efficiency** of the light emitted by the quantum dot into the fundamental modes of the optical fiber.

JCMwave

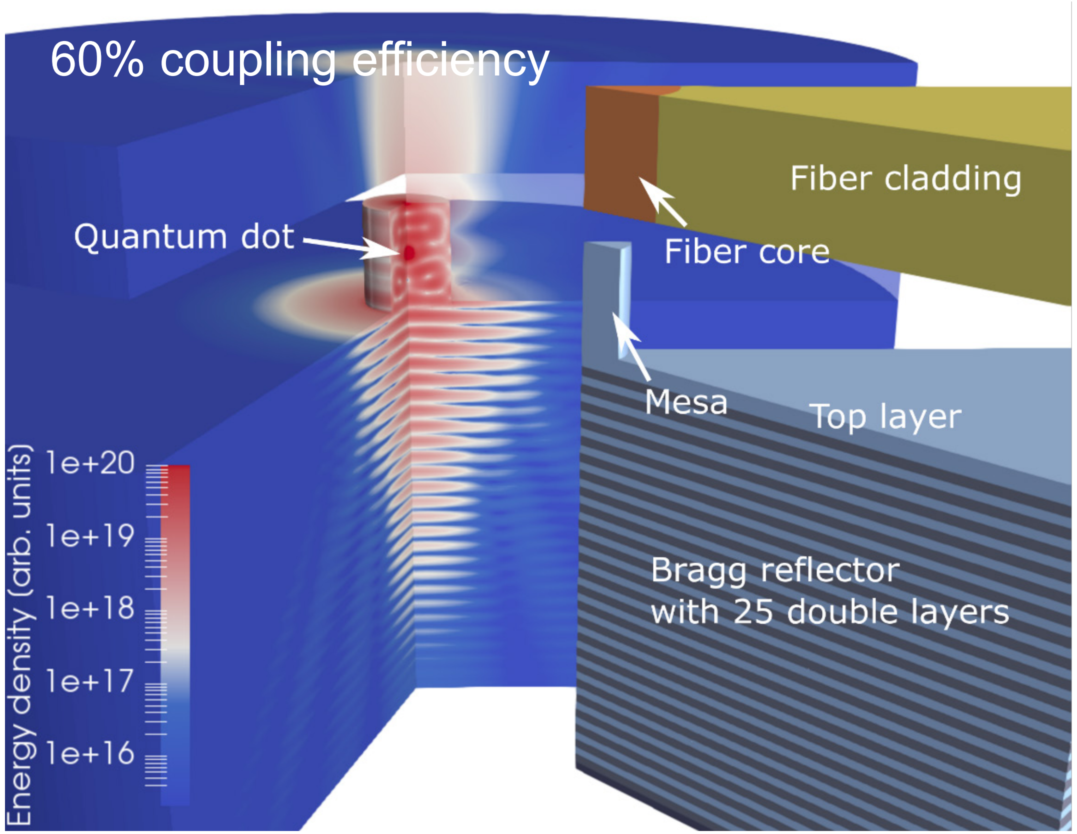
Single-photon source

Benchmark on 6-core Intel Xeon @ 3.2 GHz with 4 parallel evaluations of objective:

➔ Bayesian optimization more efficient by 1 order of magnitude

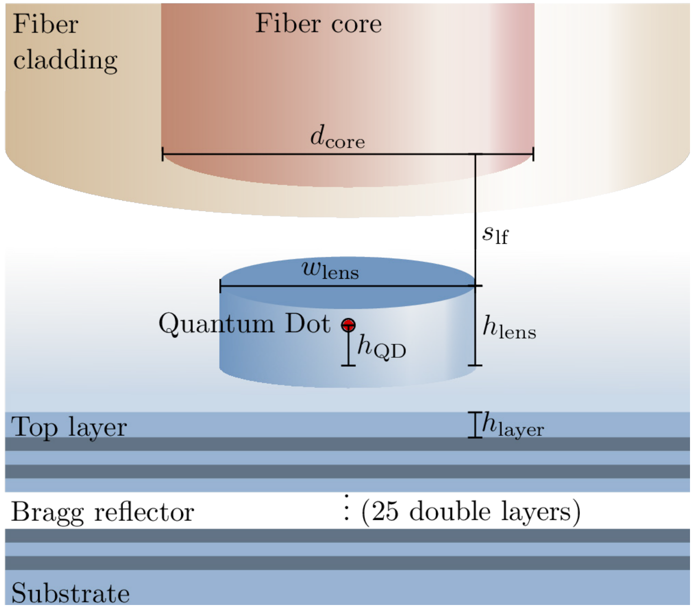


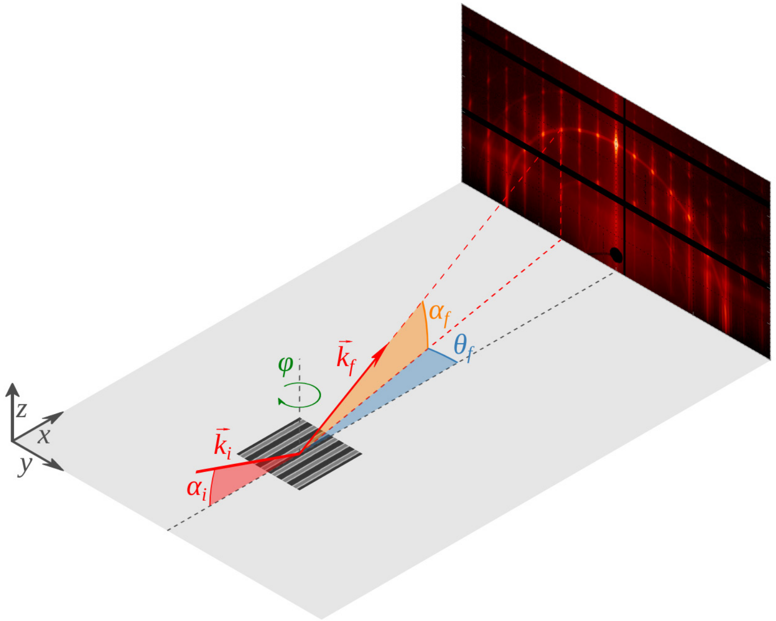
More Benchmarks: ACS Photonics **6,** 2726 (2019)



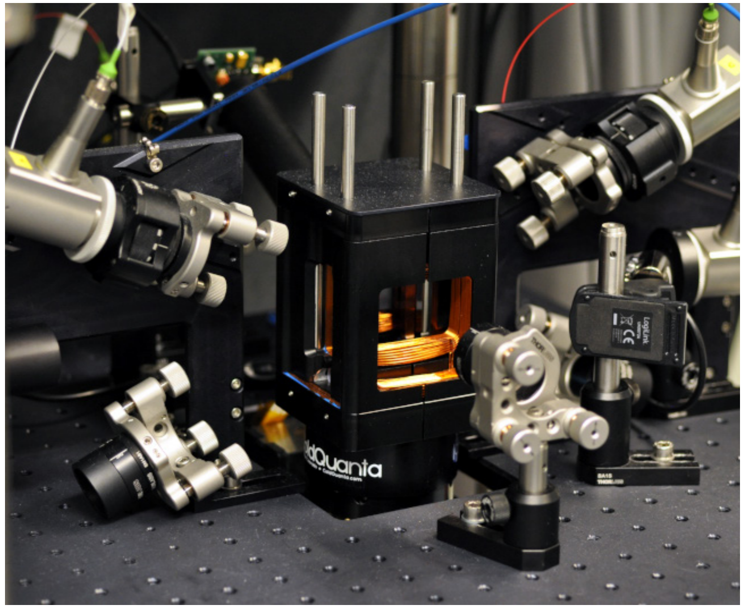
Applications of Bayesian Optimization

Computer experiments (FEM)		Real experiments
Shape/material optimization	Parameter reconstruction based on measurements	Optimization of experimental setups (cold atom traps)





M. Pflüger et al., J. Micro/Nanolith. MEMS MOEMS **19**, 014001 (2020)

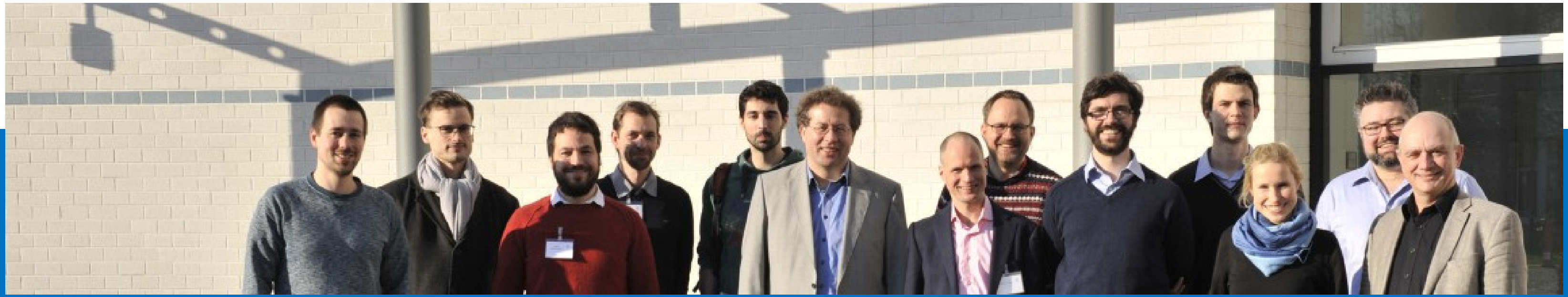


Exp. Setup of HU Berlin, Joint Lab Integrated Quantum Sensors

Vielen Dank für die Aufmerksamkeit!

JCMwave

ZIB
ZUSE INSTITUTE BERLIN



Förderung:

OPTIMAL-QT

DLR / BMWi - 50WM2067

QUPUS

ZIM / BMWi - ZF4450901

Gefördert durch:

Bundesministerium
für Wirtschaft
und Energie

aufgrund eines Beschlusses
des Deutschen Bundestages