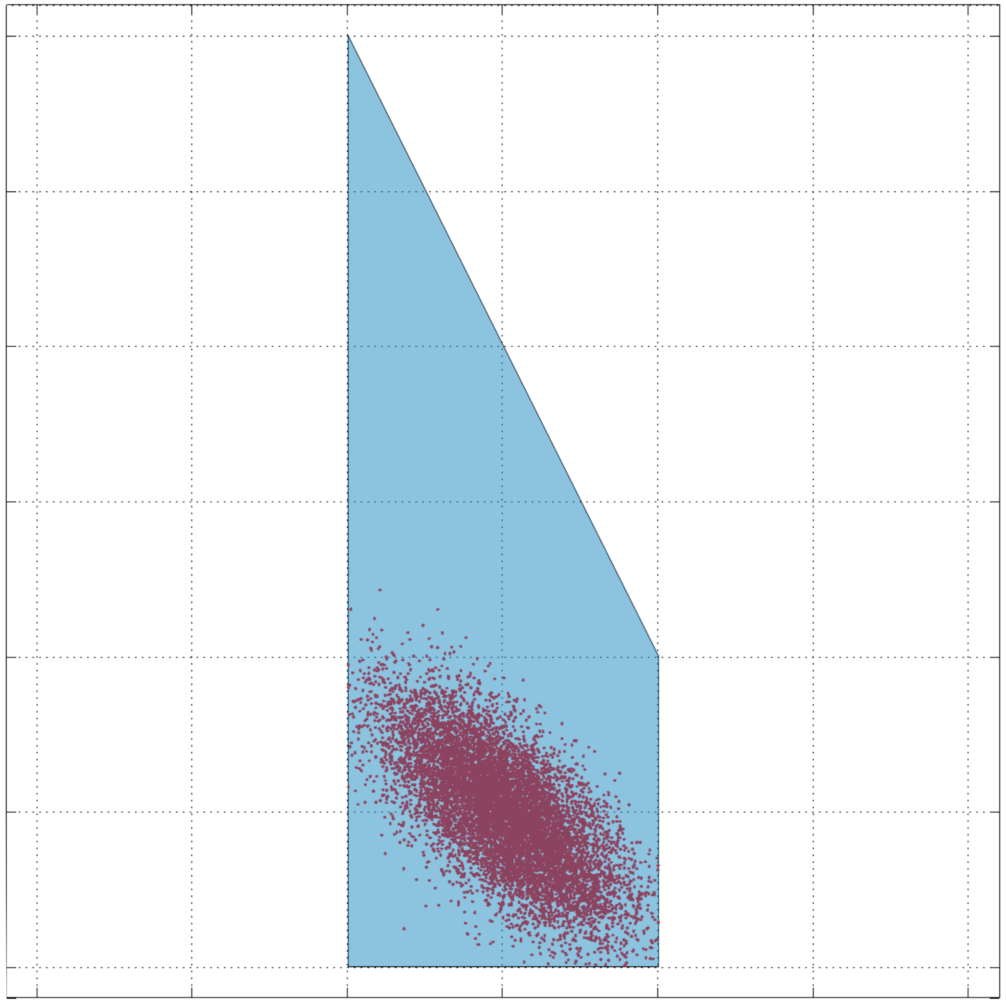
Continuous black-box optimization in linearly constrained domains using Gibbs sampling



Evolve 2011, May 26, Luxembourg

Christian L. Müller, MOSAIC group,
Institute of Theoretical Computer Science, ETH Zürich

Black-box systems

Input
$\mathbf{x} \in \mathbb{R}^n$

Output
$f(\mathbf{x}) \in \mathbb{R}$

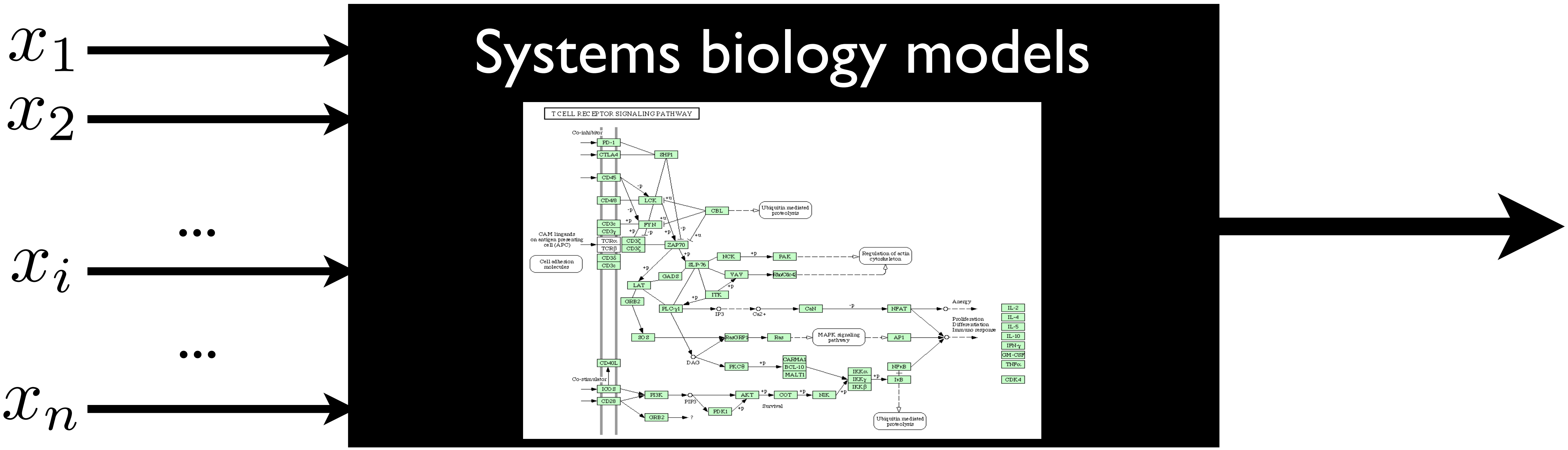
Black-box systems

Input

$\mathbf{x} \in \mathbb{R}^n$

Output

$f(\mathbf{x}) \in \mathbb{R}$



Black-box systems

Input
$\mathbf{x} \in \mathbb{R}^n$

Output
$f(\mathbf{x}) \in \mathbb{R}$

x_1
x_2
...
x_i
...
x_n

Analog circuit design

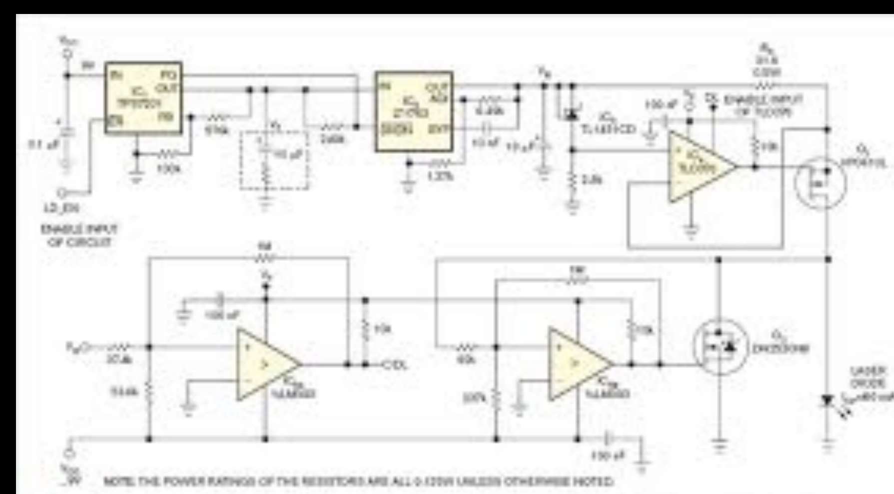
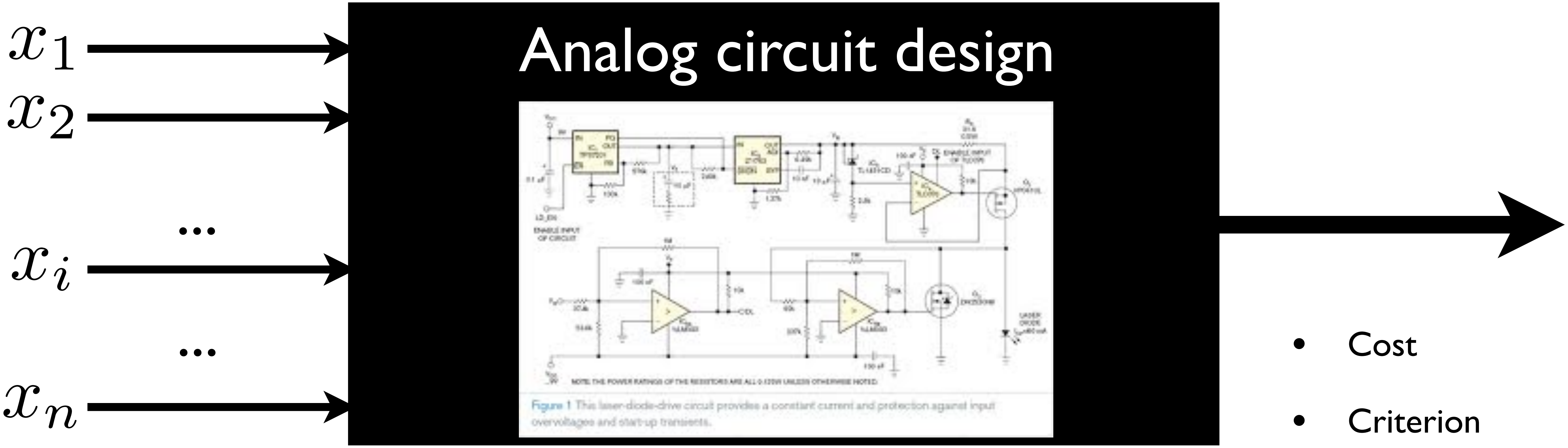


Figure 1 This laser-diode-drive circuit provides a constant current and protection against input overvoltages and start-up transients.

Black-box systems

Input $\mathbf{x} \in \mathbb{R}^n$

Output $f(\mathbf{x}) \in \mathbb{R}$



- Variables
- Parameters/Factors
- Configuration

- Cost
- Criterion
- Objective
- Energy
- Fitness

Talk Overview

Black-box optimizers using multivariate normal samples

Black-box problems with linear constraints

Gibbs sampling for truncated normal distributions

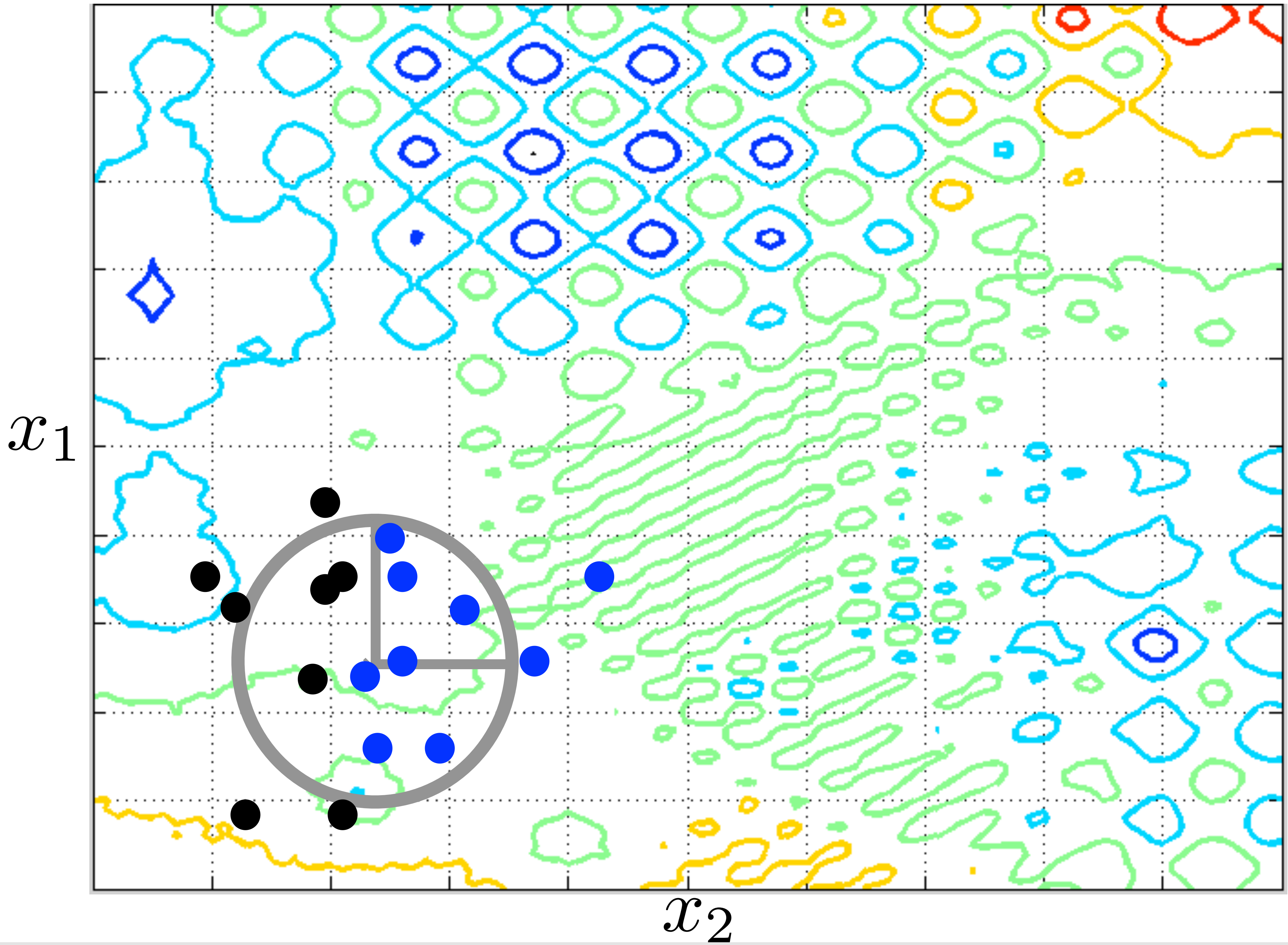
Illustrative Examples

Conclusions and outlook

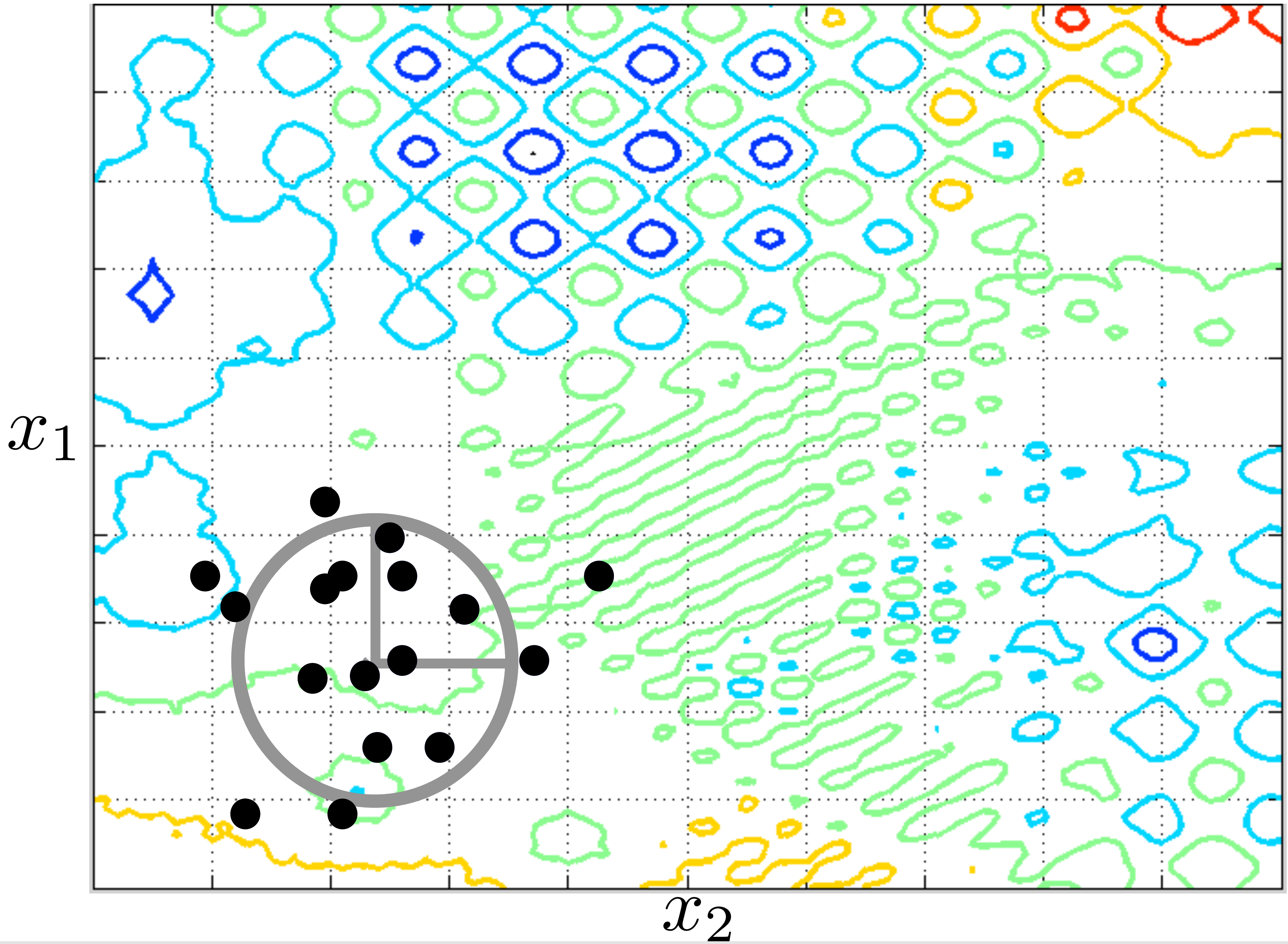
Black-box optimizers using multivariate normal samples

- Evolution Strategies (Rechenberg, Schwefel, and many more)
- **Covariance Matrix Adaptation Evolution Strategy** (CMA-ES) (Hansen et al.)
- **Gaussian Adaptation (GaA)** (Kjellstrom et al. , Mueller et al.)
- Natural Evolution Strategies (NES) (Wierstra et al.)
- Estimation of Distribution Algorithms (EDA’s)
- Cross-Entropy method in continuous domains (Rubinstein et al.)
- ...

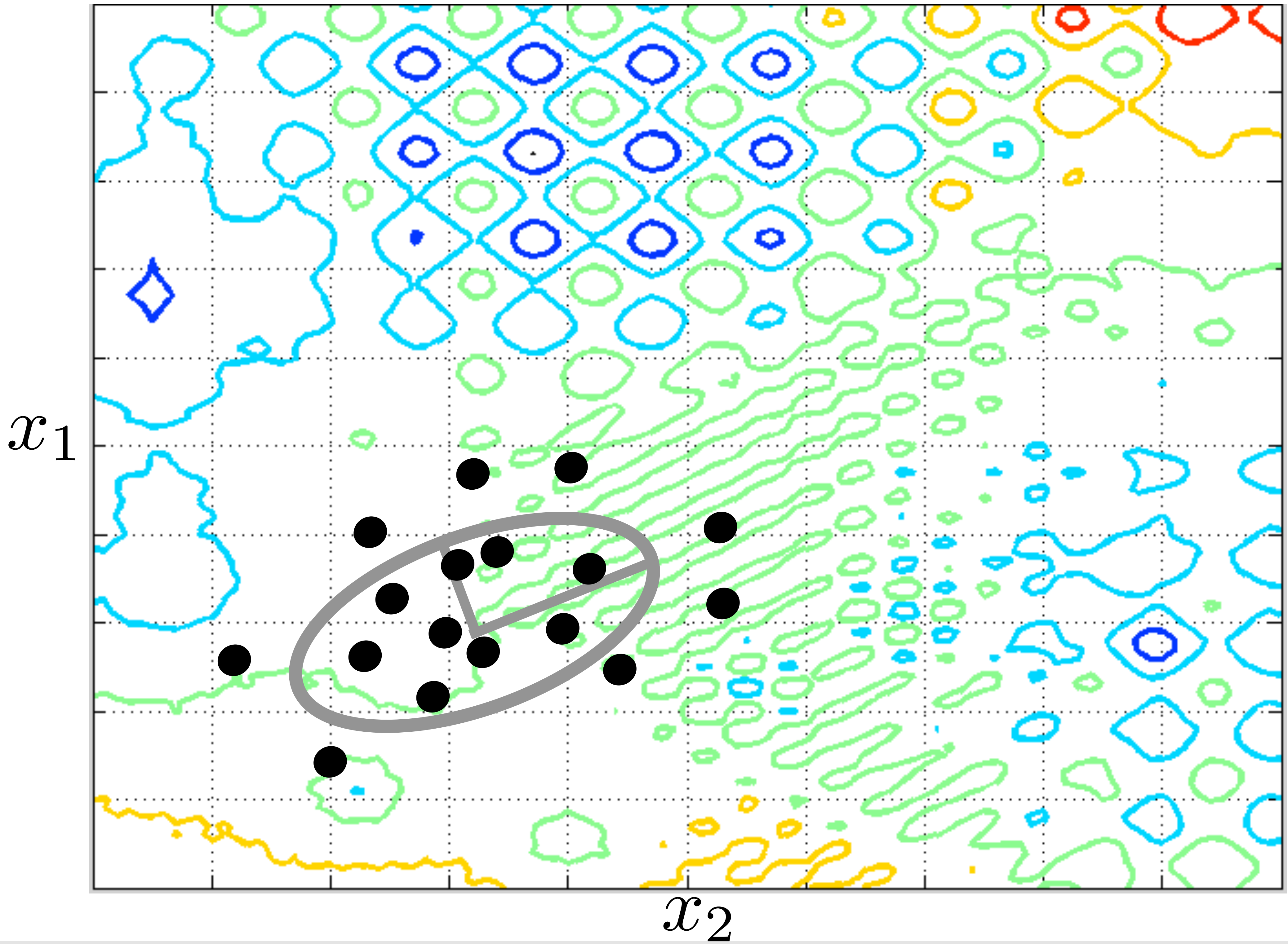
Black-box optimization with CMA-ES



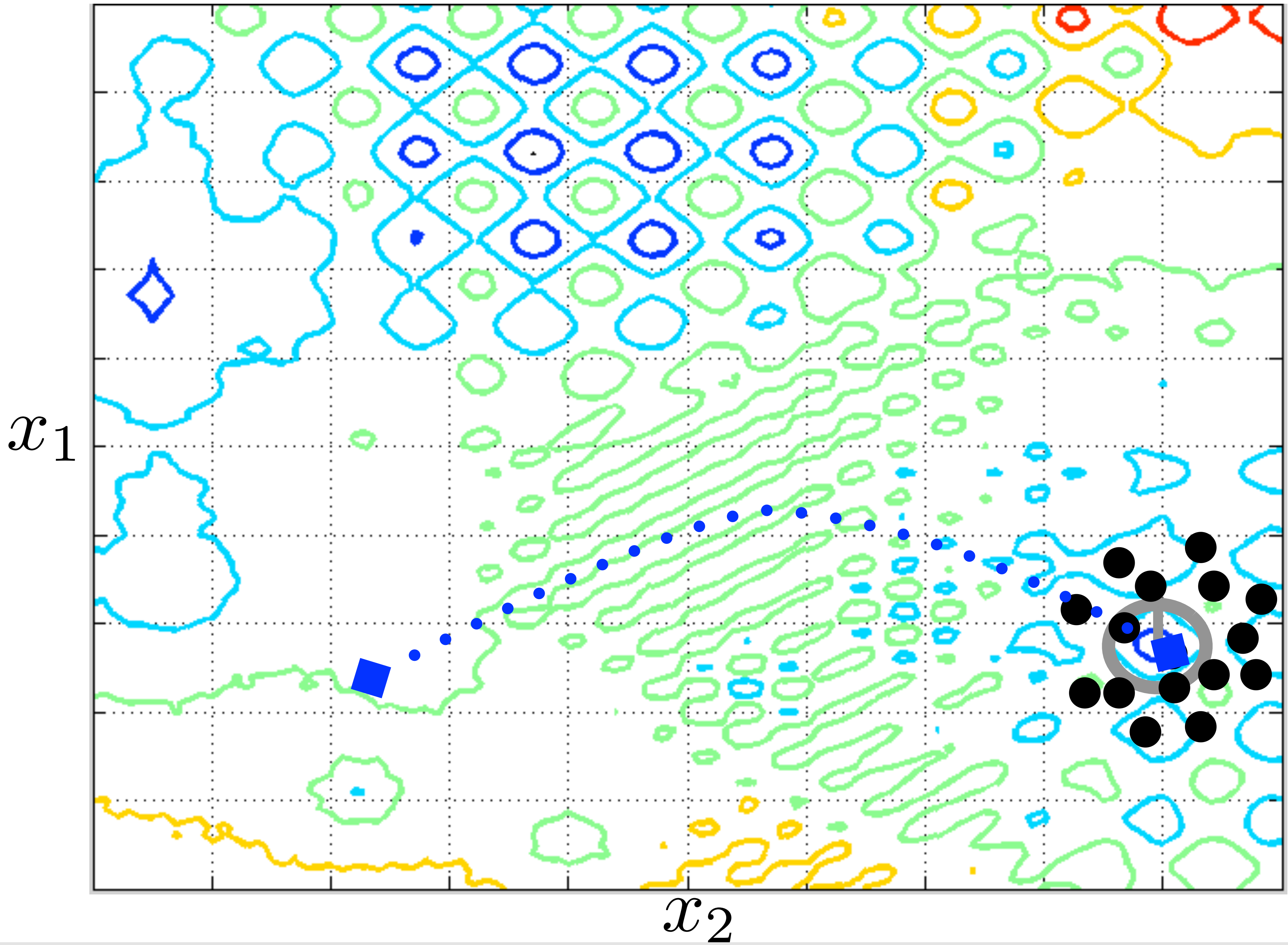
Black-box optimization with CMA-ES



Black-box optimization with CMA-ES



Black-box optimization with CMA-ES



Black-box optimization with GaA

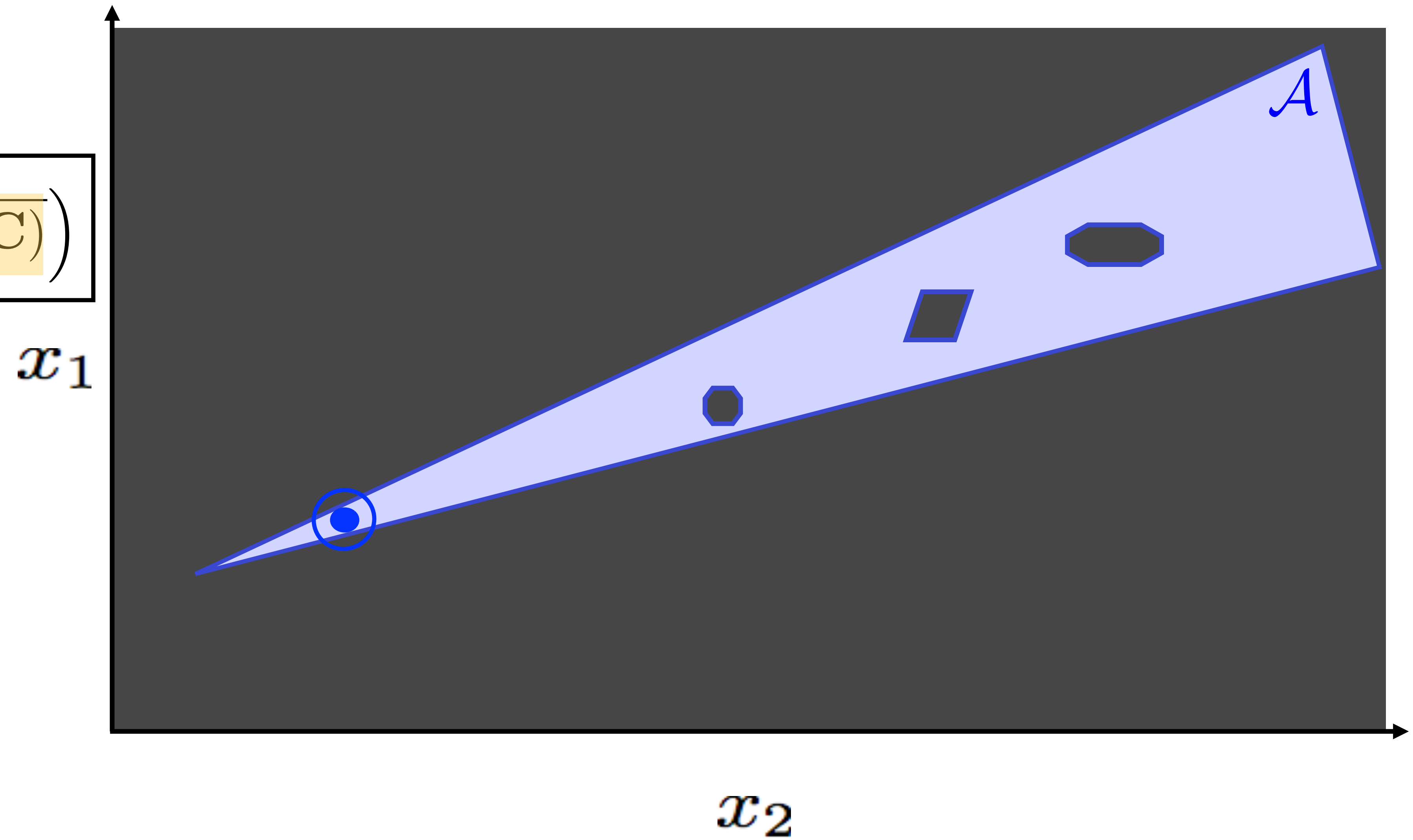
Maximization of the entropy of the Gaussian distribution.

Entropy:

$$\boxed{\mathcal{H}(\mathcal{N}) = \log\left(\sqrt{(2\pi e)^n \det(\mathbf{C})}\right)}$$

Constraint:

$$\boxed{Prob(\mathbf{x} \in \mathcal{A}) = P}$$



Black-box optimization with GaA

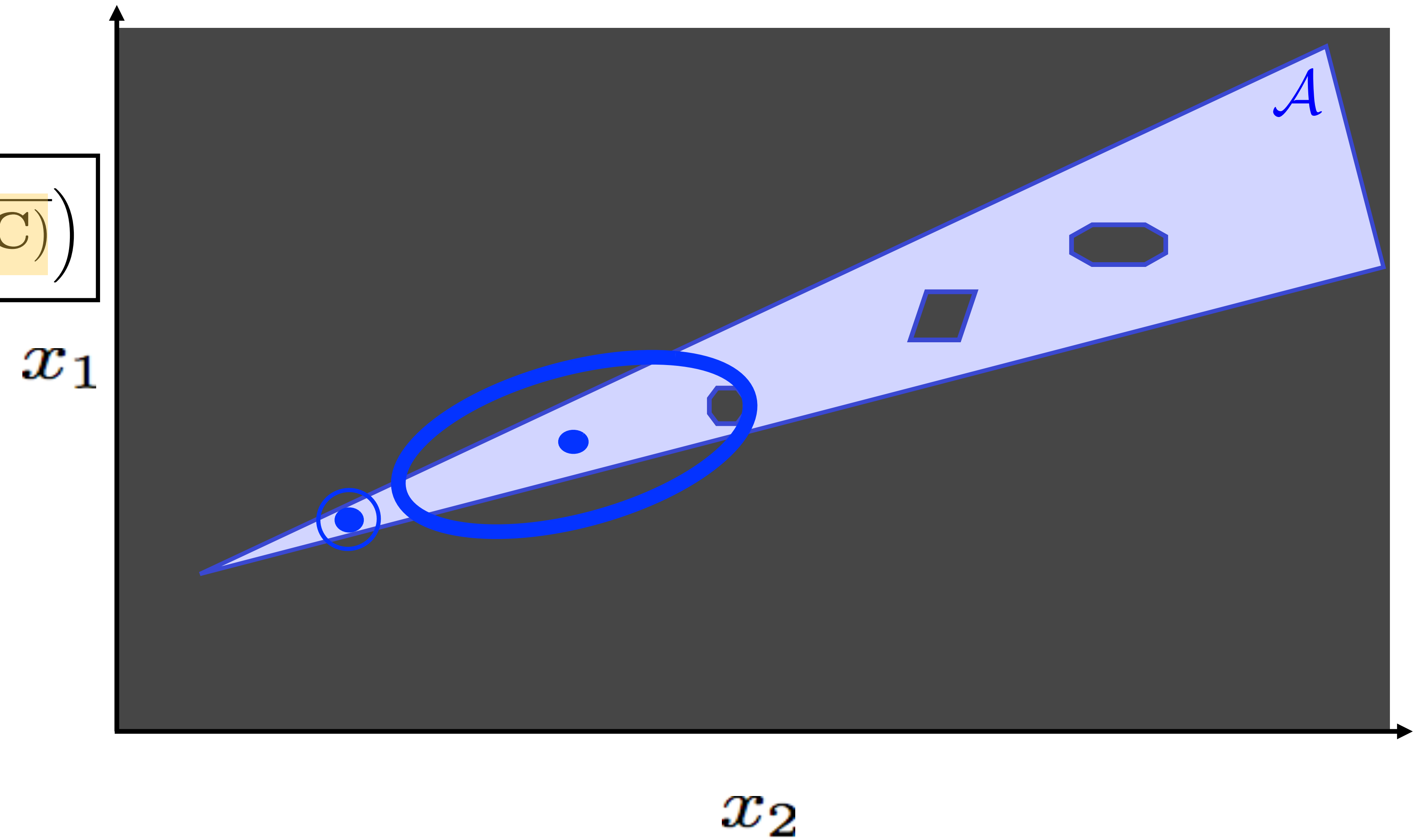
Maximization of the entropy of the Gaussian distribution.

Entropy:

$$\boxed{\mathcal{H}(\mathcal{N}) = \log\left(\sqrt{(2\pi e)^n \det(\mathbf{C})}\right)}$$

Constraint:

$$\boxed{Prob(\mathbf{x} \in \mathcal{A}) = P}$$



Black-box optimization with GaA

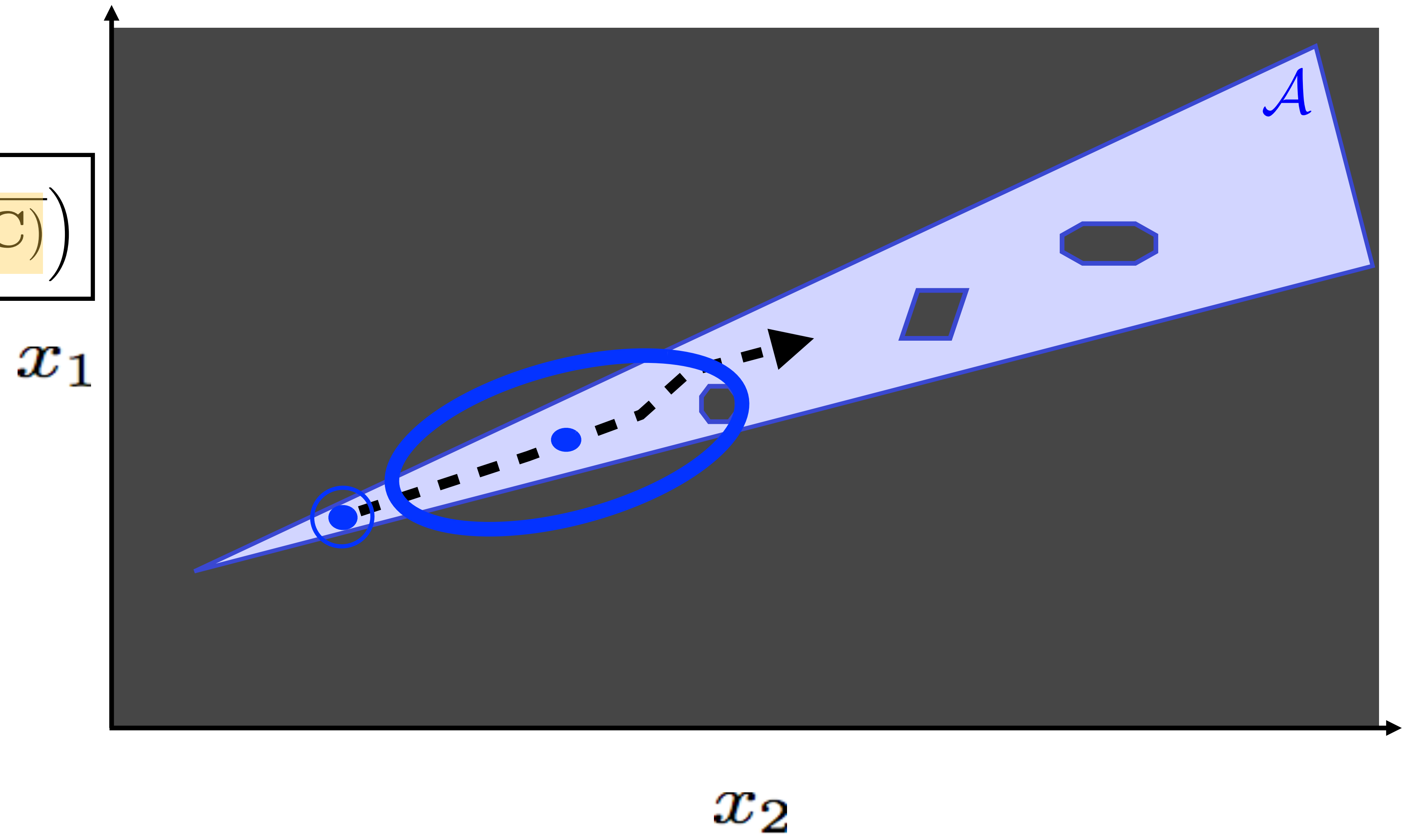
Maximization of the entropy of the Gaussian distribution.

Entropy:

$$\mathcal{H}(\mathcal{N}) = \log\left(\sqrt{(2\pi e)^n \det(\mathbf{C})}\right)$$

Constraint:

$$Prob(\mathbf{x} \in \mathcal{A}) = P$$



Black-box optimization with GaA

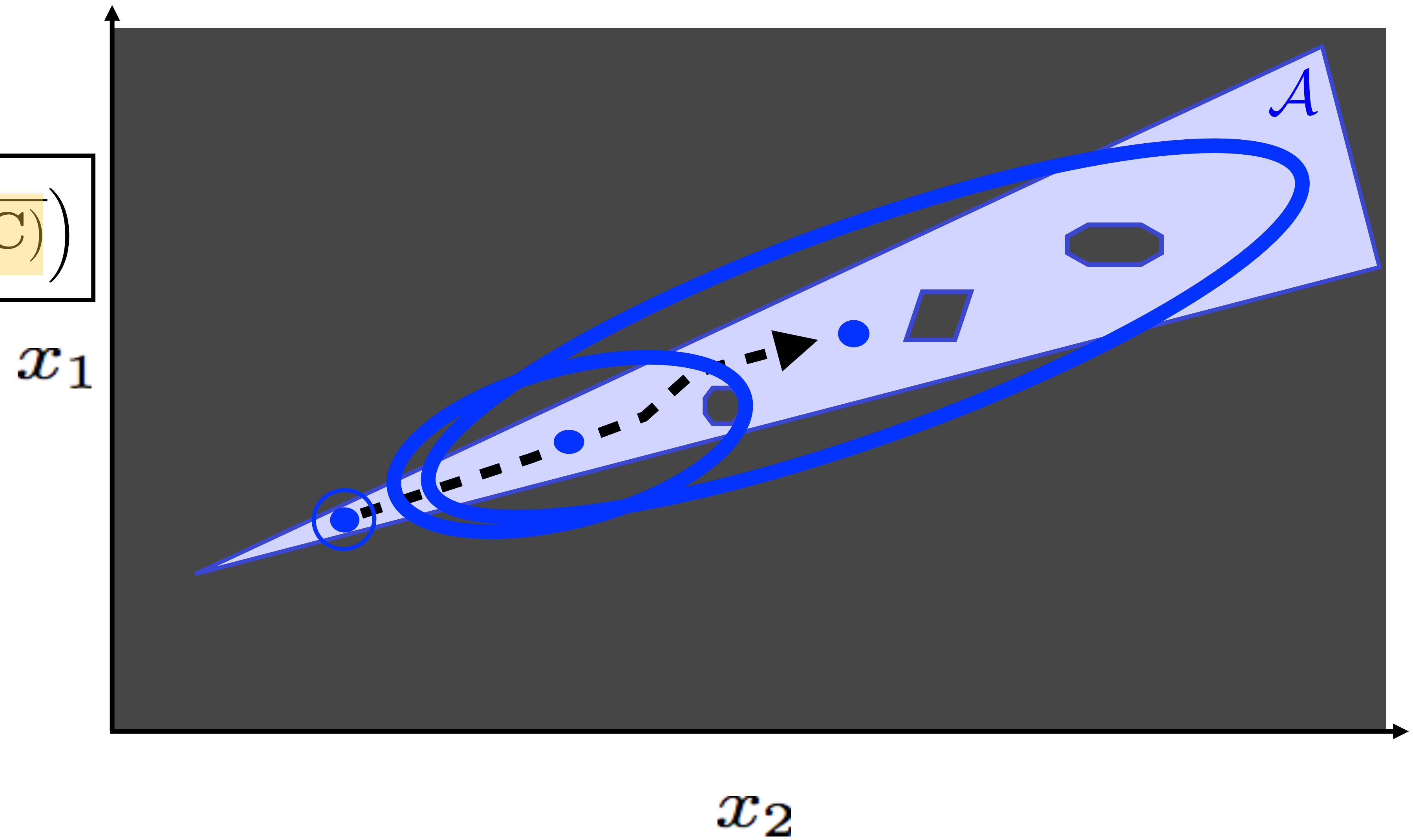
Maximization of the entropy of the Gaussian distribution.

Entropy:

$$\boxed{\mathcal{H}(\mathcal{N}) = \log\left(\sqrt{(2\pi e)^n \det(\mathbf{C})}\right)}$$

Constraint:

$$\boxed{Prob(\mathbf{x} \in \mathcal{A}) = P}$$



Black-box optimization with GaA

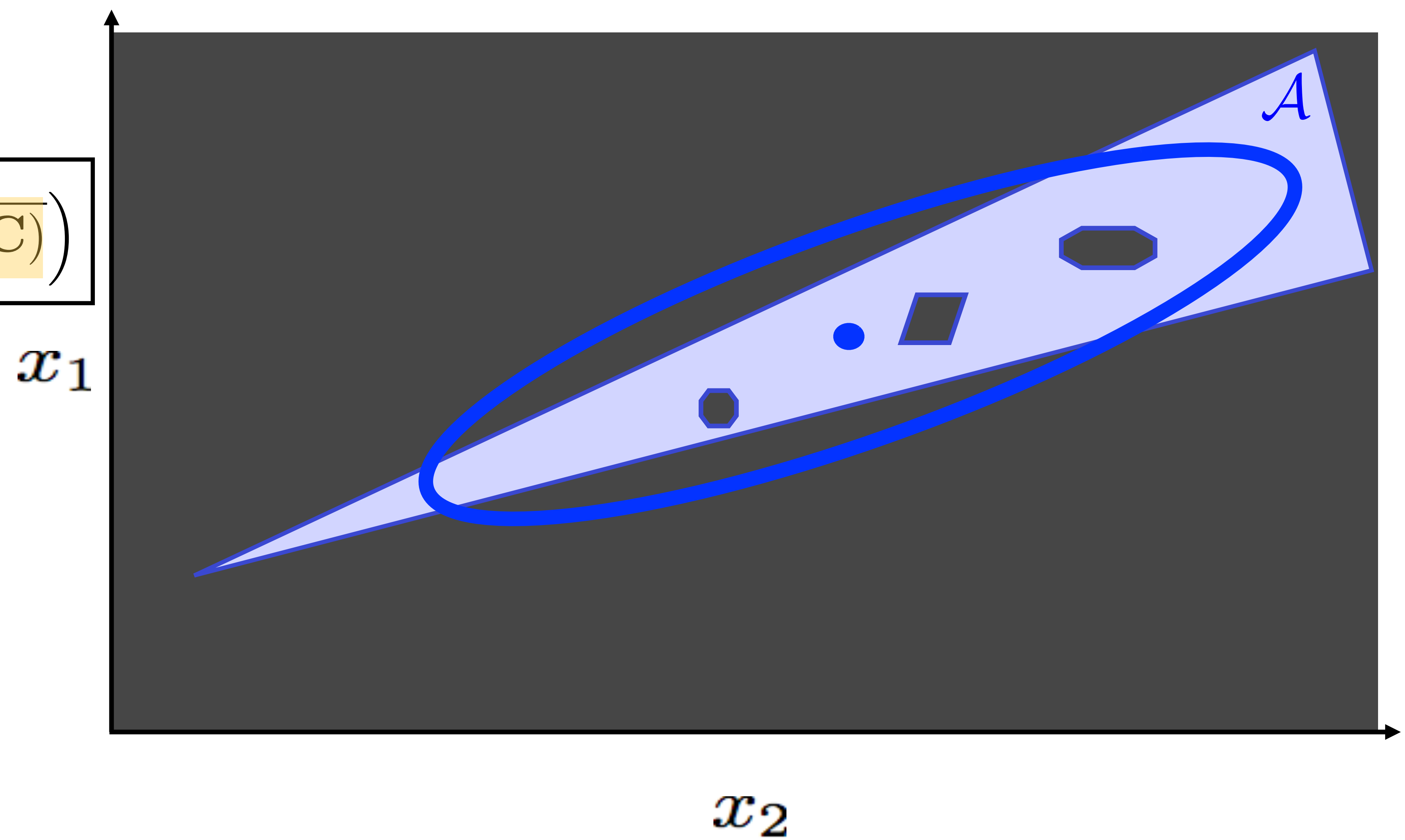
Maximization of the entropy of the Gaussian distribution.

Entropy:

$$\boxed{\mathcal{H}(\mathcal{N}) = \log\left(\sqrt{(2\pi e)^n \det(\mathbf{C})}\right)}$$

Constraint:

$$\boxed{Prob(\mathbf{x} \in \mathcal{A}) = P}$$



Black-box optimization with GaA

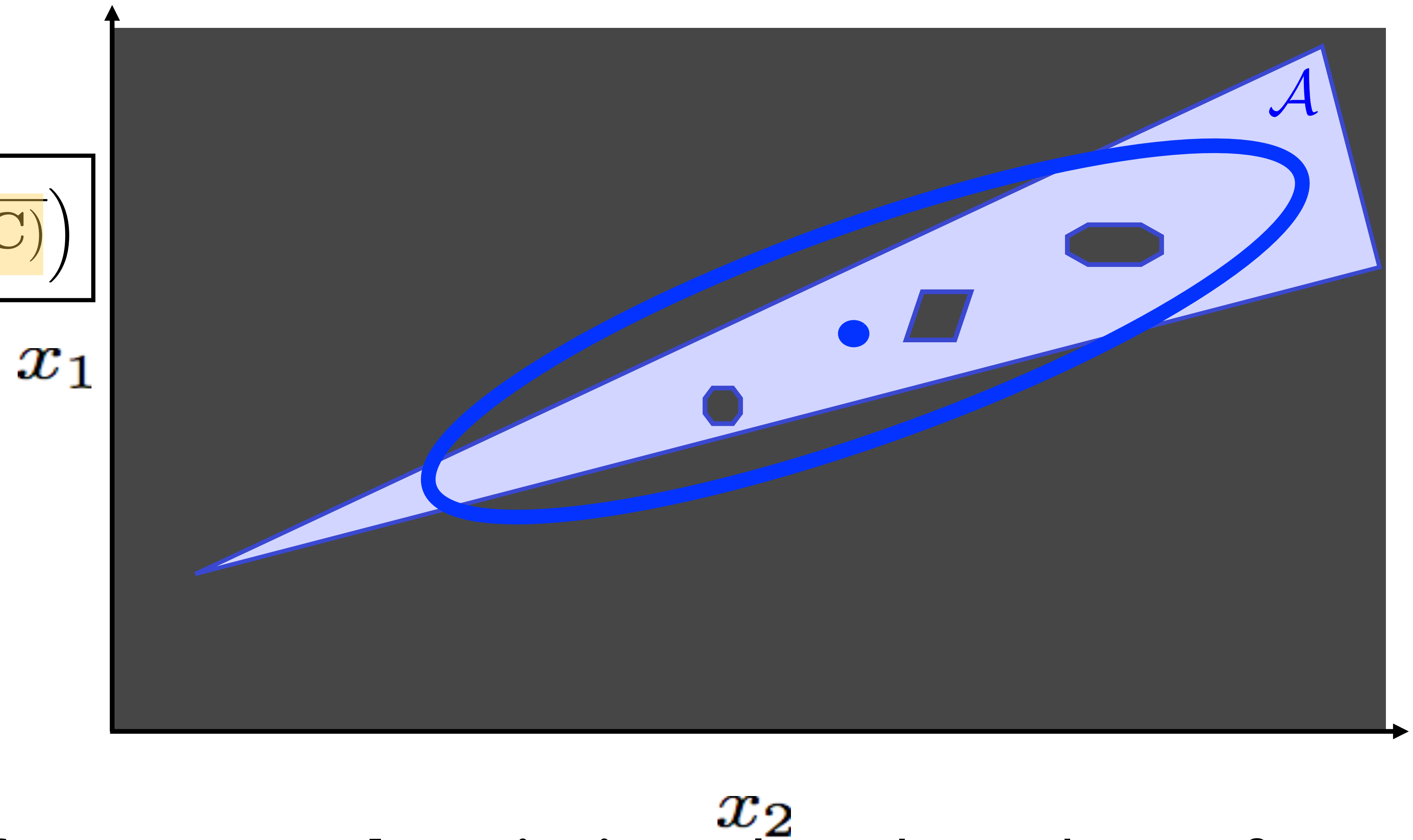
Maximization of the entropy of the Gaussian distribution.

Entropy:

$$\boxed{\mathcal{H}(\mathcal{N}) = \log\left(\sqrt{(2\pi e)^n \det(\mathbf{C})}\right)}$$

Constraint:

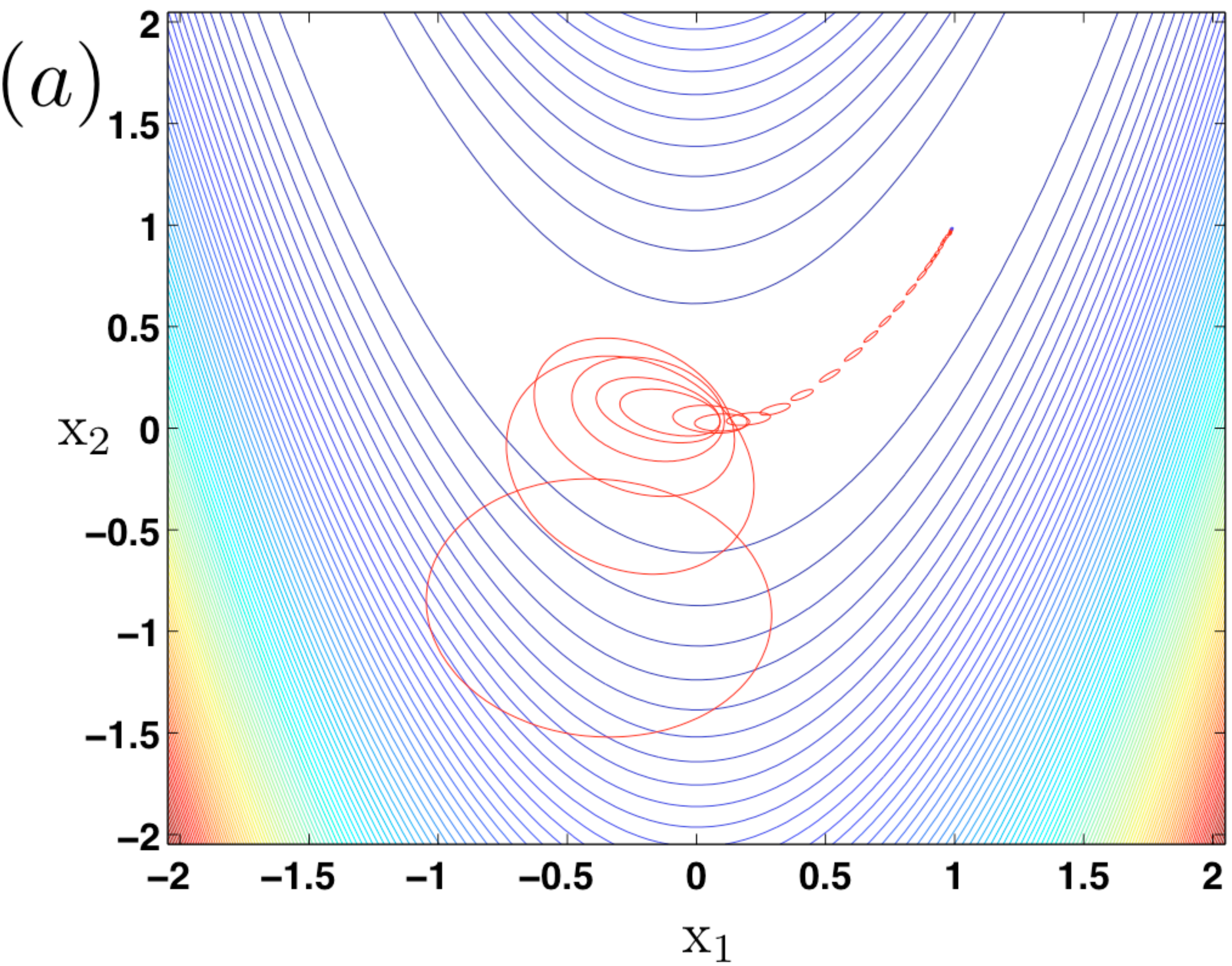
$$\boxed{Prob(\mathbf{x} \in \mathcal{A}) = P}$$



Key idea for a general optimizer: Introduce adaptive fitness thresholds, define convergence criteria and **restart** GaA with slower reduction of thresholds

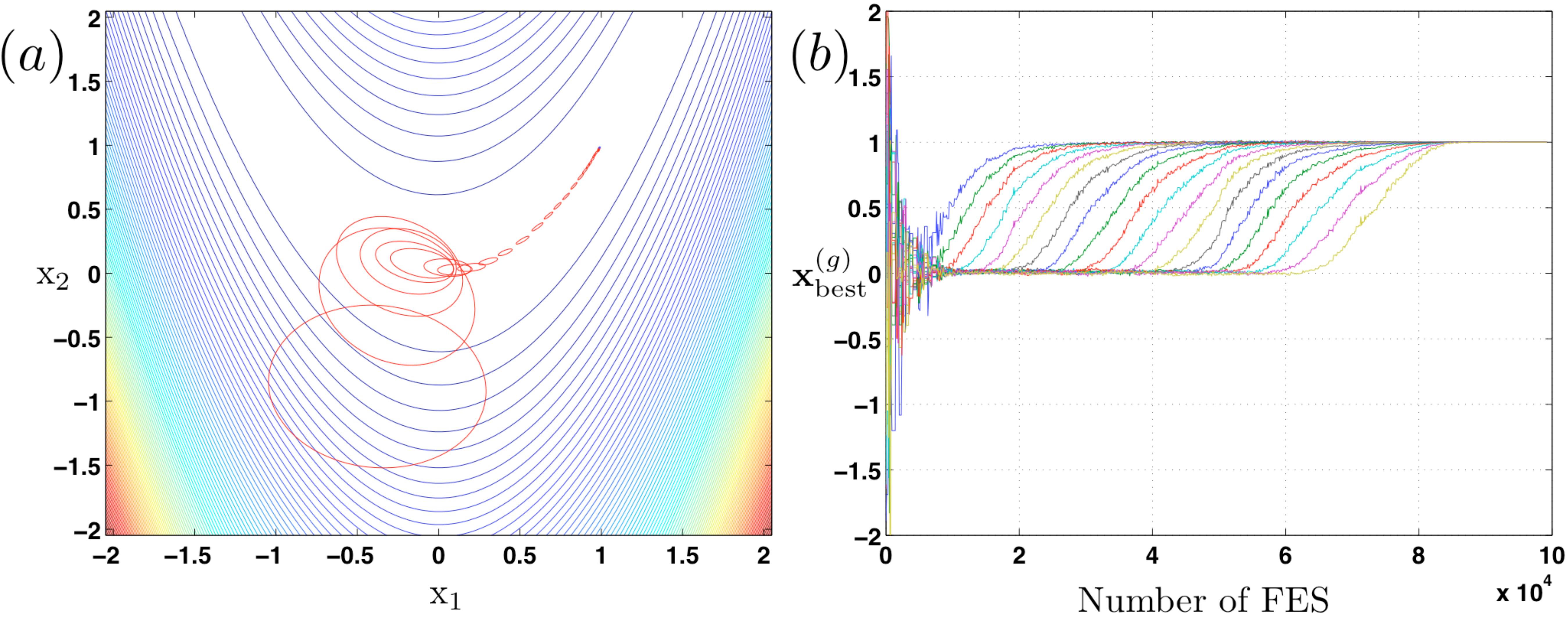
Black-box optimization with GaA

Illustration on Rosenbrock's valley function in n=20



Black-box optimization with GaA

Illustration on Rosenbrock's valley function in n=20



Black-box optimization with linear constraints

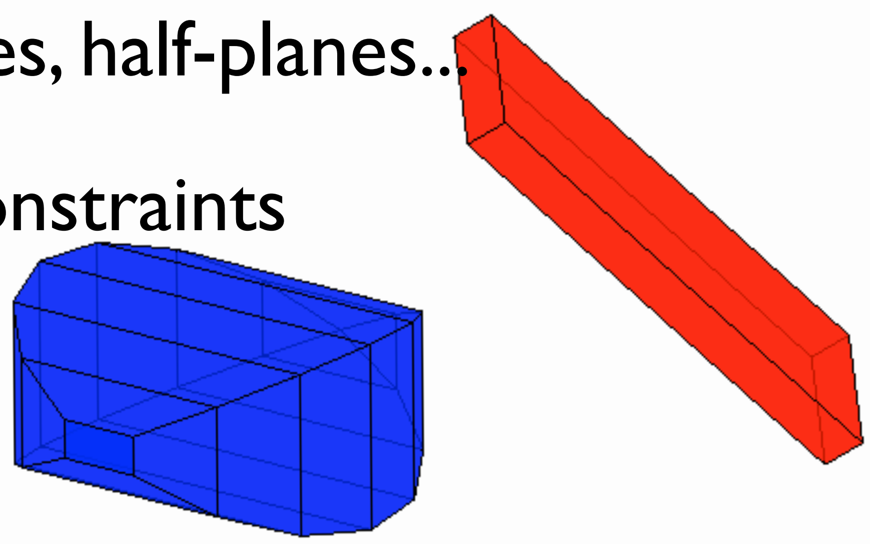
$$\min_{\mathbf{x}\in\mathbb{R}^n} \quad f(\mathbf{x})$$
$$\text{s.t.} \quad \mathbf{A}\mathbf{x} \leq \mathbf{b}\,.$$

- **A** is a mxn matrix and **b** is a m-dimensional vector describing m constraints
- Geometrically, these describe polyhedra, cones, half-planes...
- Box constraints are a special case of linear constraints
- Positive orthant $\mathbf{x}\geq\mathbf{0}$
- Standard simplex

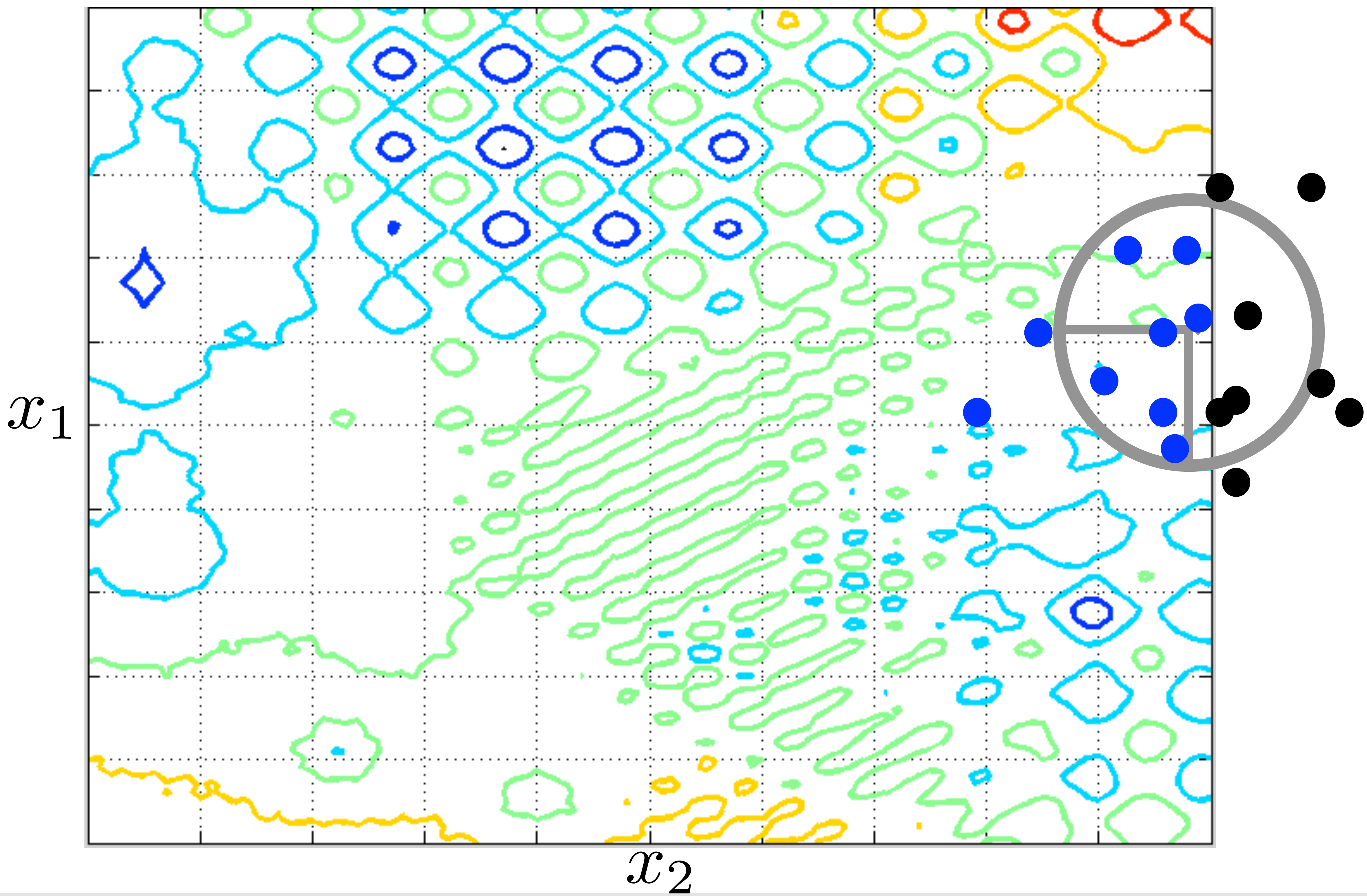
Black-box optimization with linear constraints

$$\min_{\mathbf{x}\in\mathbb{R}^n} \quad f(\mathbf{x})$$
$$\text{s.t.} \quad \mathbf{A}\mathbf{x} \leq \mathbf{b}\,.$$

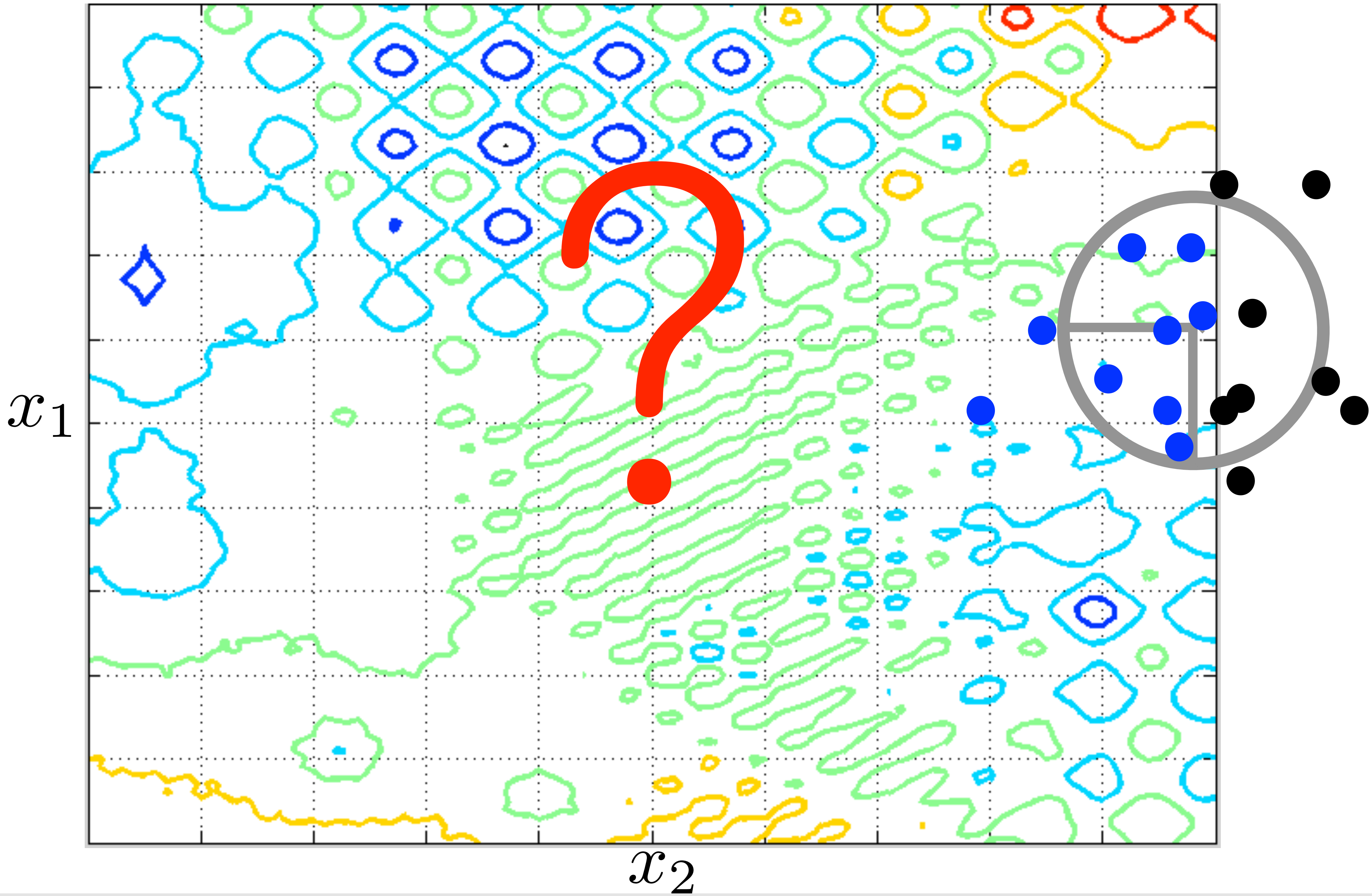
- **A** is a mxn matrix and **b** is a m-dimensional vector describing m constraints
- Geometrically, these describe polyhedra, cones, half-planes...
- Box constraints are a special case of linear constraints
- Positive orthant **x**≥**0**
- Standard simplex



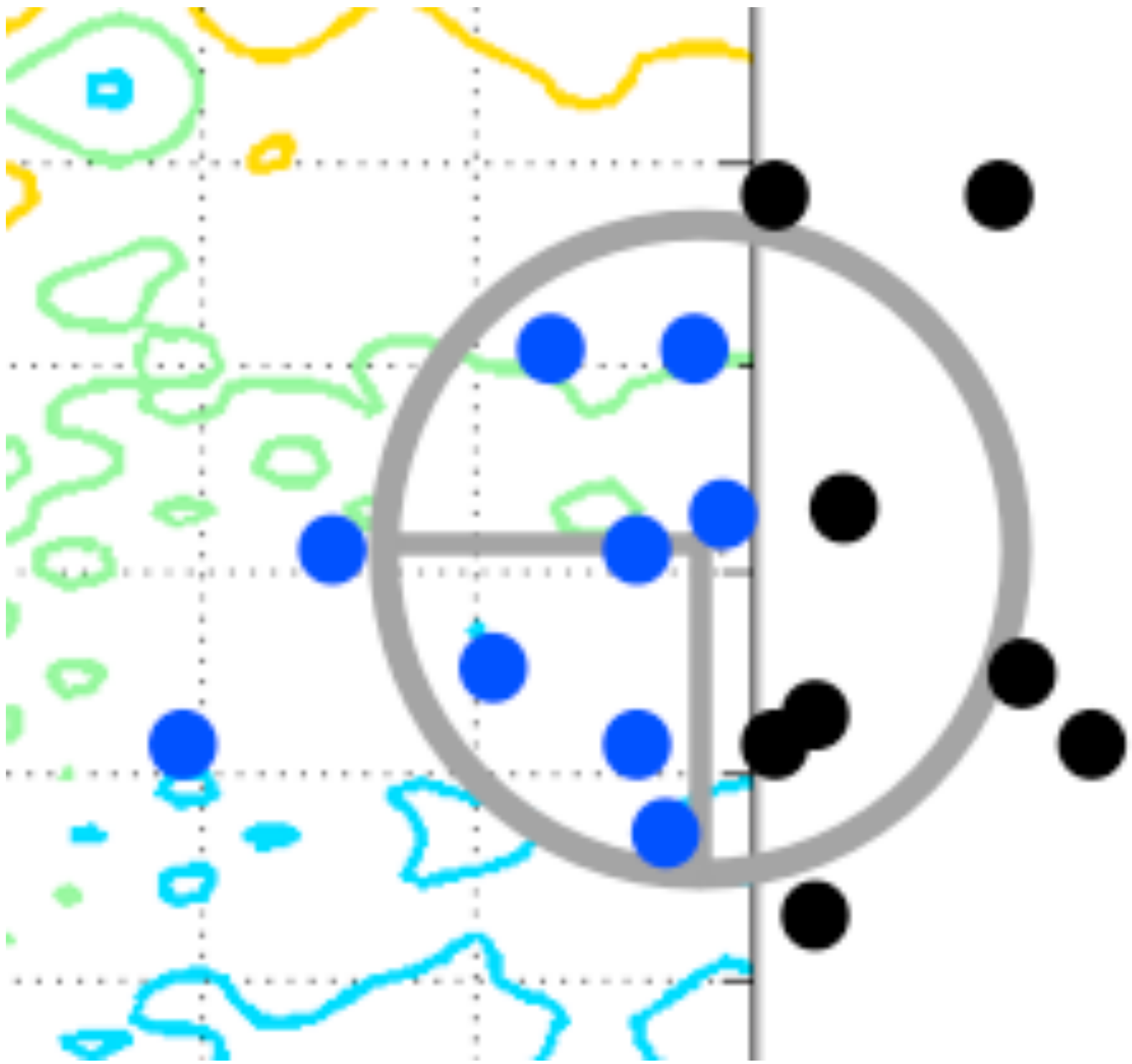
Black-box optimization with linear constraints



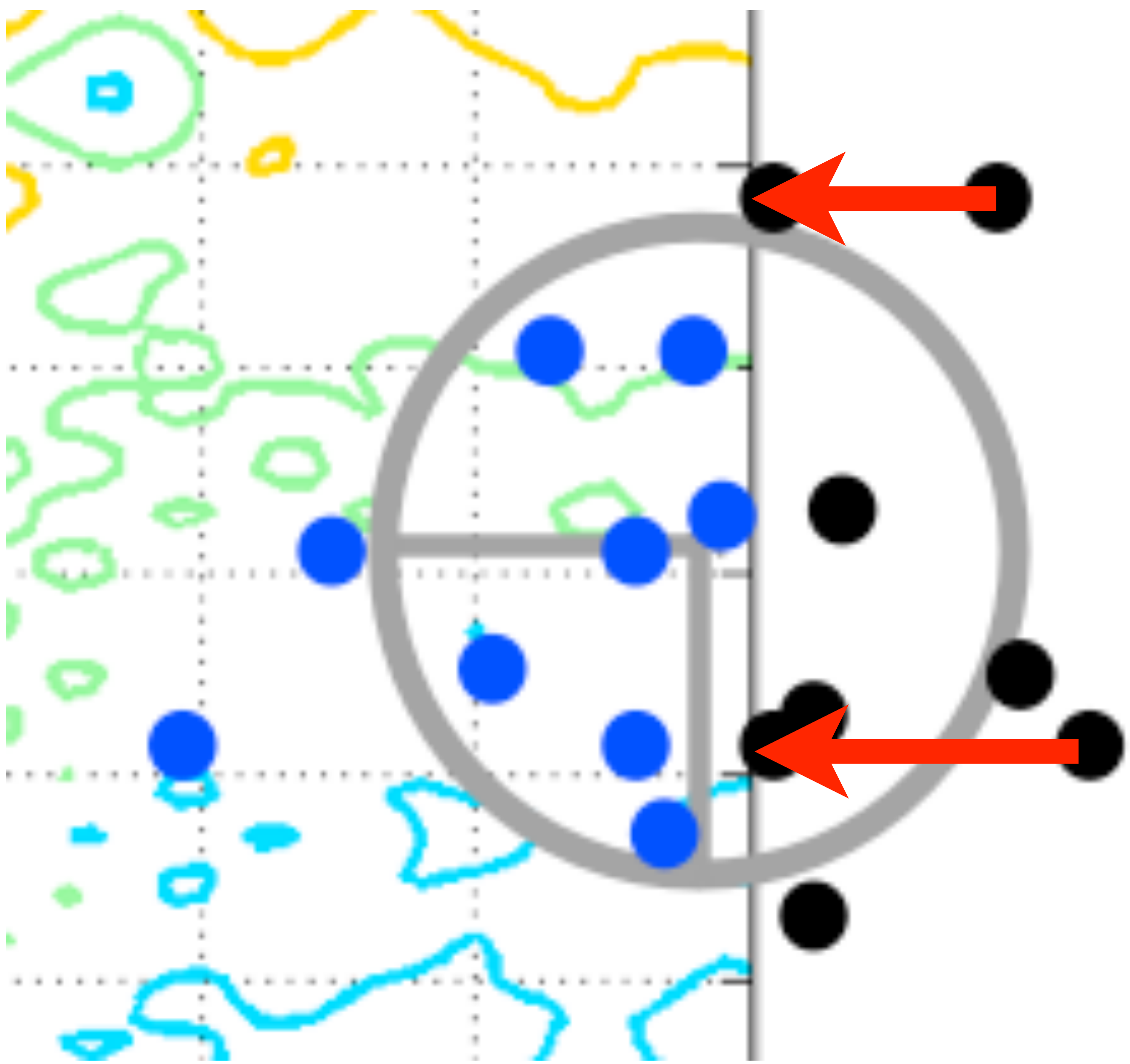
Black-box optimization with linear constraints



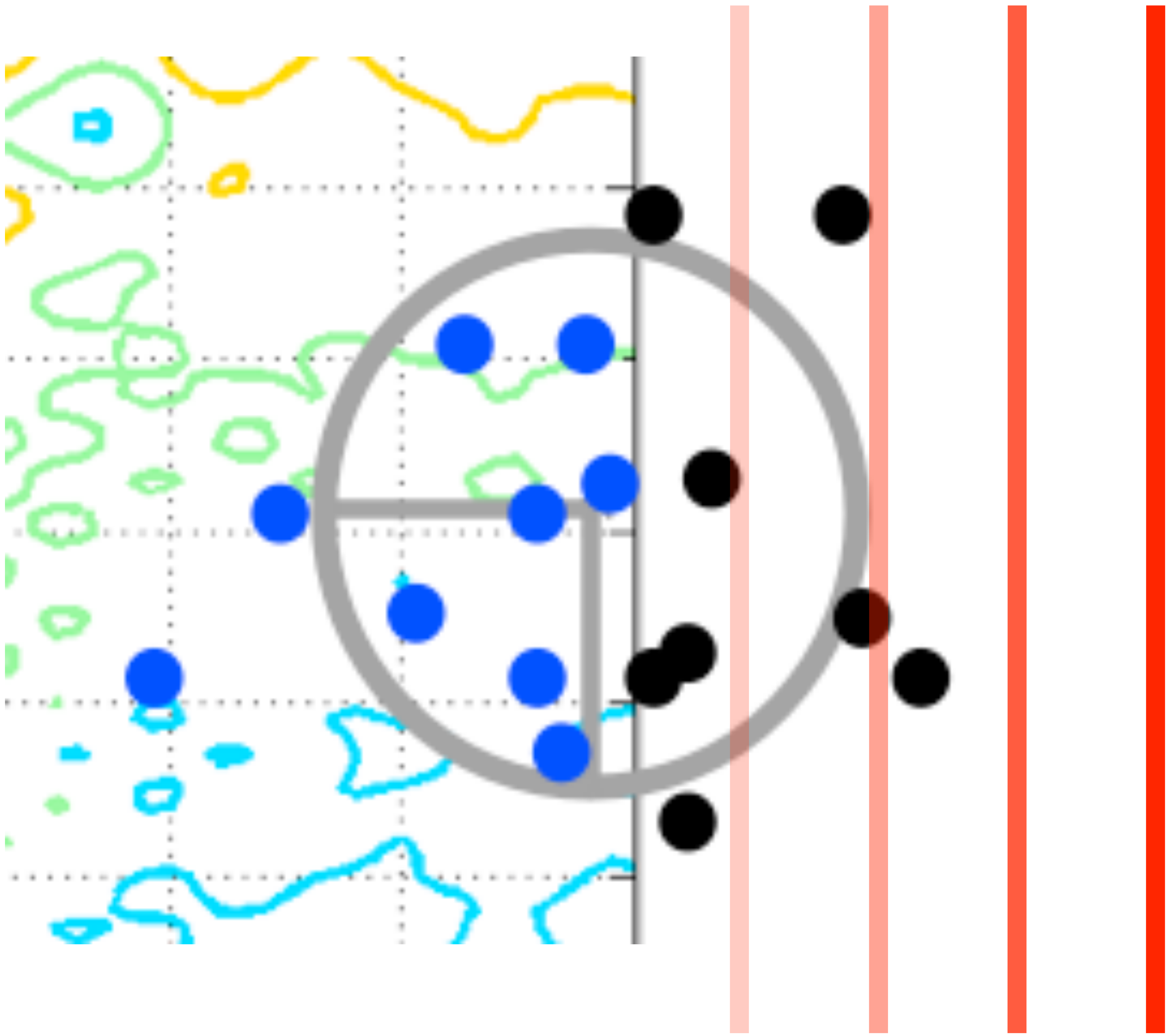
Black-box optimization with linear constraints



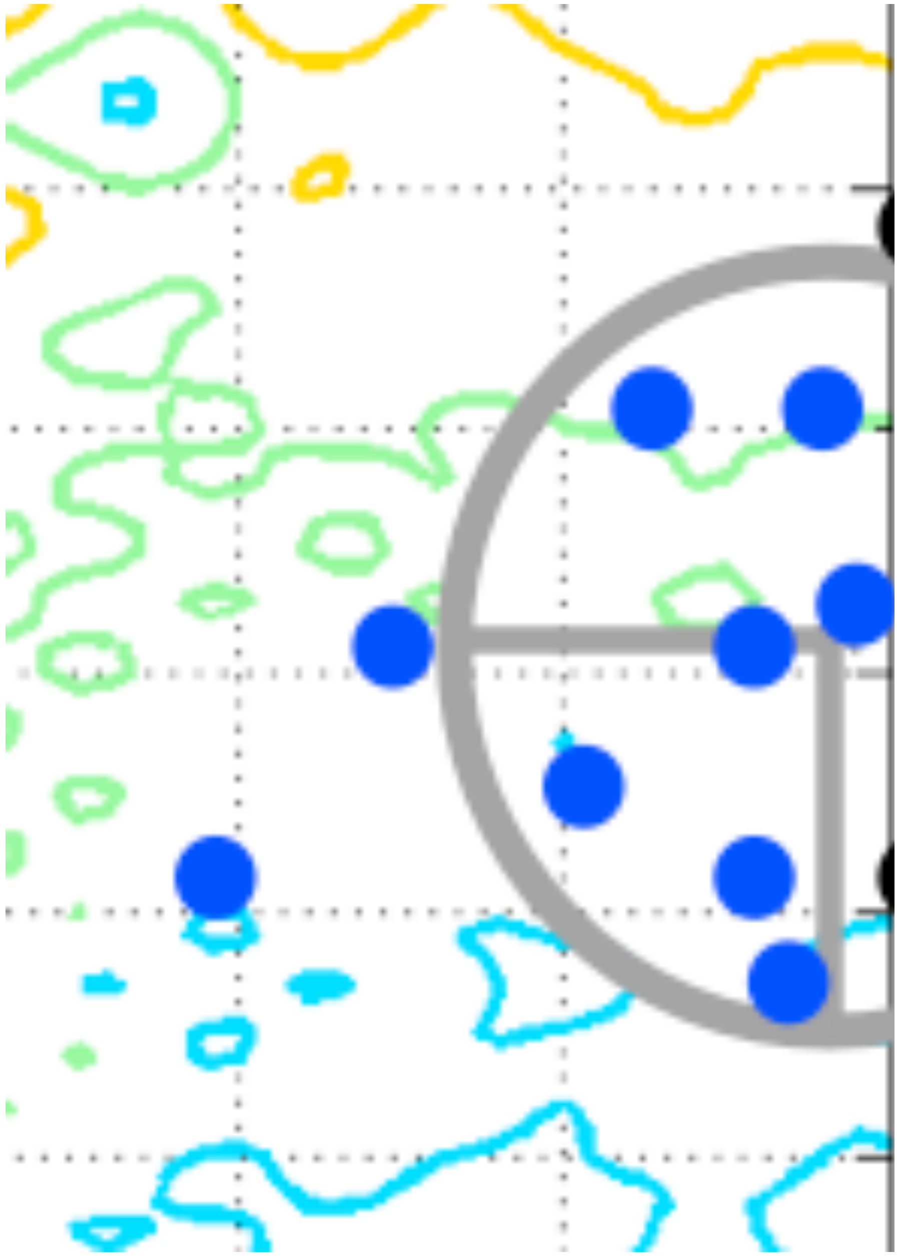
Black-box optimization with linear constraints



Black-box optimization with linear constraints



Black-box optimization with linear constraints



Black-box optimization with truncated normals

- Simplest idea: Rejection sampling

Black-box optimization with truncated normals

- Simplest idea: Rejection sampling

→ Only works if acceptance ratio is high. In high dimensions this is often impossible, hence the sampling becomes exponentially slow

Black-box optimization with truncated normals

- Simplest idea: Rejection sampling

→ Only works if acceptance ratio is high. In high dimensions this is often impossible, hence the sampling becomes exponentially slow

- Alternative idea: Gibbs sampling (Geman and Geman, 1984)

→ Only works if all conditional distributions of a multivariate distributions are known! This is true for truncated normal distributions, T-distributions, etc.

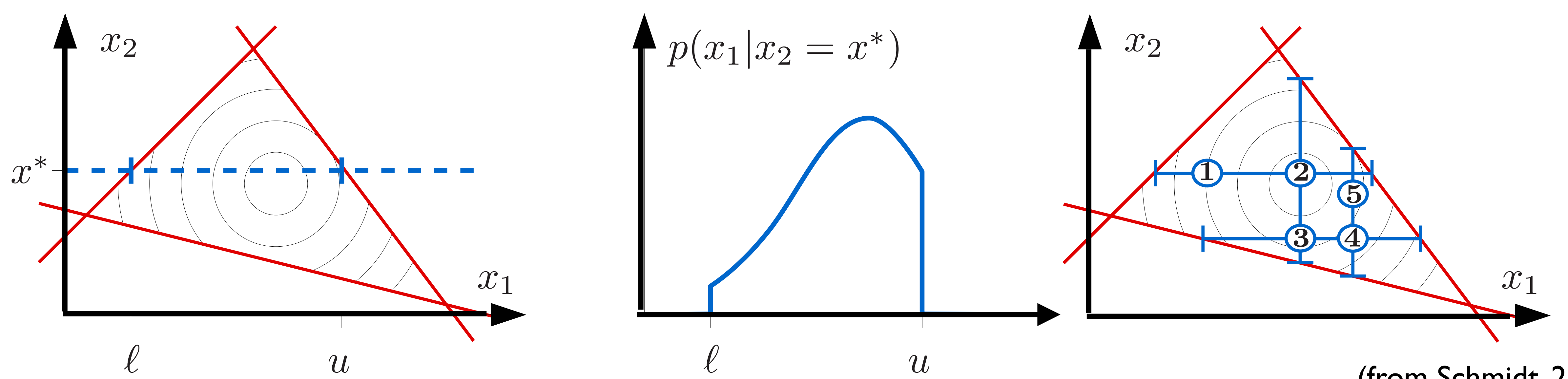
Sampling cost is polynomial in the dimension

Gibbs sampling for truncated multivariate normals

- First sampler for multivariate truncated normals introduced by Geweke, 1991 only for box constraints
- Generalization for arbitrary linear constraints by Rodriguez-Yam et al, 2004, but not recognized and no implementation is available

Gibbs sampling for truncated multivariate normals

- First sampler for multivariate truncated normals introduced by Geweke, 1991 only for box constraints
- Generalization for arbitrary linear constraints by Rodriguez-Yam et al, 2004, but not recognized and no implementation is available



(from Schmidt, 2009)

Swiss Institute of Bioinformatics

Gibbs sampling for truncated multivariate normals

$$\mathbf{p}(\mathbf{x}) \propto \mathcal{N}_\Omega(\mathbf{m}, \mathbf{C}) = \begin{cases} \mathcal{N}(\mathbf{m}, \mathbf{C}) & \text{if } \ \mathbf{Ax} \leq \mathbf{b} \\ 0 & \text{otherwise,} \end{cases}$$

Let $\mathbf{x} \sim \mathcal{N}_\Omega(\mathbf{m}, \mathbf{C})$ be an n-dimensional multivariate normal vector. We first decompose $\mathbf{C} = \sigma^2\Sigma$ with σ a scalar. Let $\mathbf{T} \in \mathbb{R}^{n\times n}$ be a matrix of full rank such that $\mathbf{T}\Sigma\mathbf{T}^T = \mathbf{I}$ where $\mathbf{I}$ denotes the n-dimensional identity matrix. $\mathbf{T}$ can be found by Cholesky or eigenvalue decomposition of Σ because the rescaled covariance matrix Σ is positive definite. Let $\mathbf{z} = \mathbf{T}\,\mathbf{x}$ and $\mathbf{c} = \mathbf{T}\,\mathbf{m}$.

Gibbs sampling for truncated multivariate normals

$$\mathbf{p}(\mathbf{z}) \propto \mathcal{N}_{\mathbf{S}}(\mathbf{c}, \sigma^2\mathbf{I}) = \begin{cases} \mathcal{N}(\mathbf{c}, \sigma^2\mathbf{I}) & \text{if} \quad \mathbf{D}\mathbf{z} \leq \mathbf{b} \\ 0 & \text{otherwise}, \end{cases}$$

where $\mathbf{D} = \mathbf{A}\,\mathbf{T}^{-1}$ and $\mathbf{S} = \{\mathbf{z} \in \mathbb{R}^n : \mathbf{D}\mathbf{z} \leq \mathbf{b}\}$ and the original $\mathbf{x}$ can be recovered by $\mathbf{x} = \mathbf{T}^{-1}\mathbf{z}$. This reformulation drastically simplifies the sampling distribution but not the constraints.

SIB Swiss Institute of Bioinformatics

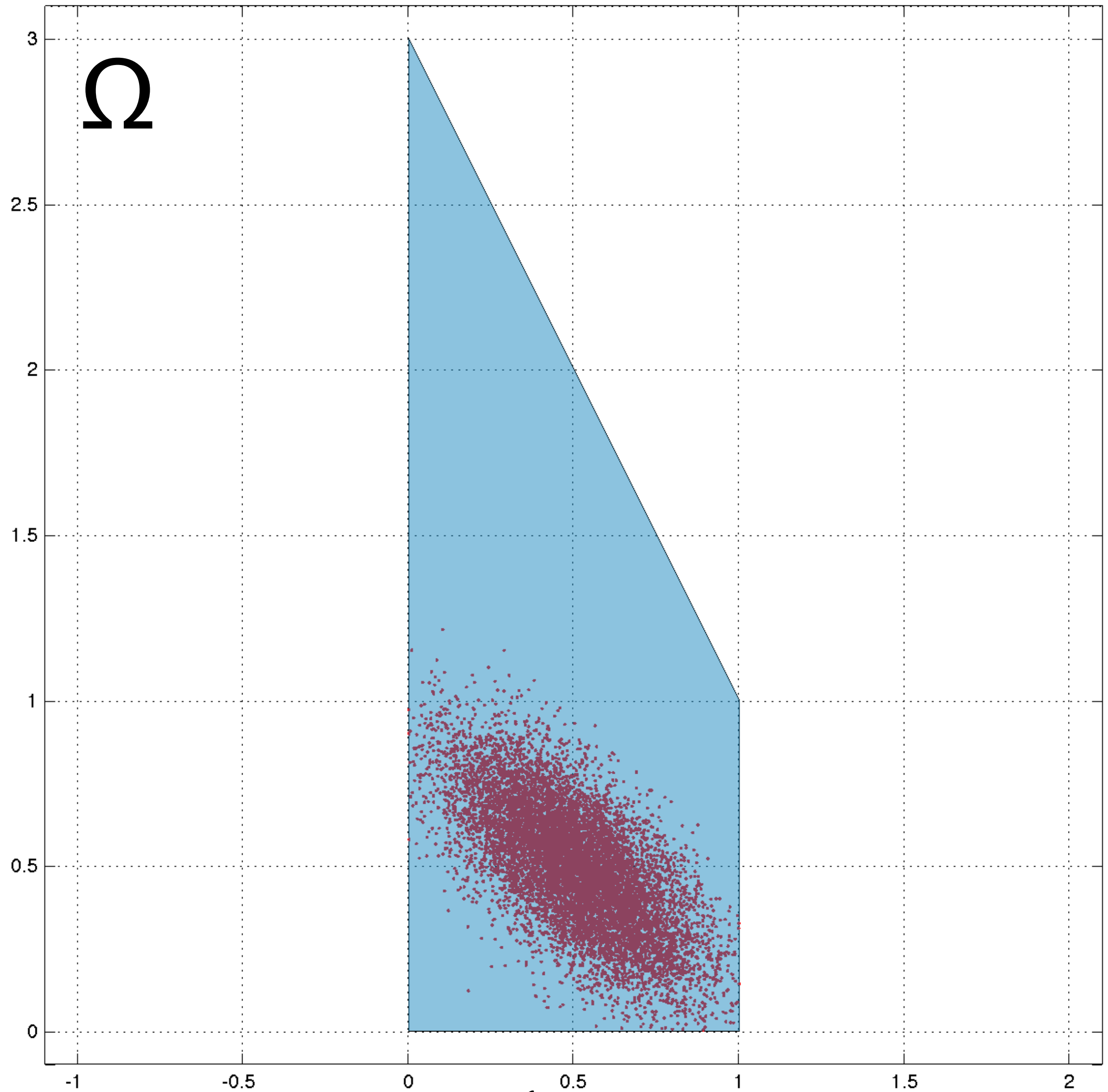
Gibbs sampling for truncated multivariate normals

Let $\mathbf{z} = [\mathbf{z}_1, \ldots, \mathbf{z}_j, \ldots, \mathbf{z}_n]^T$ and $\mathbf{z}_{-j} = [z_1, \ldots, z_{j-1}, z_{j+1}, \ldots, z_n]^T$. A Gibbs sampler for the multivariate distribution thus generates in each sweep components z_j of $\mathbf{z}$ according to

$$\mathbf{p}(z_j|\mathbf{z}_{-j}) = \mathcal{N}_{S_j}(c_j, \sigma^2)\,,$$

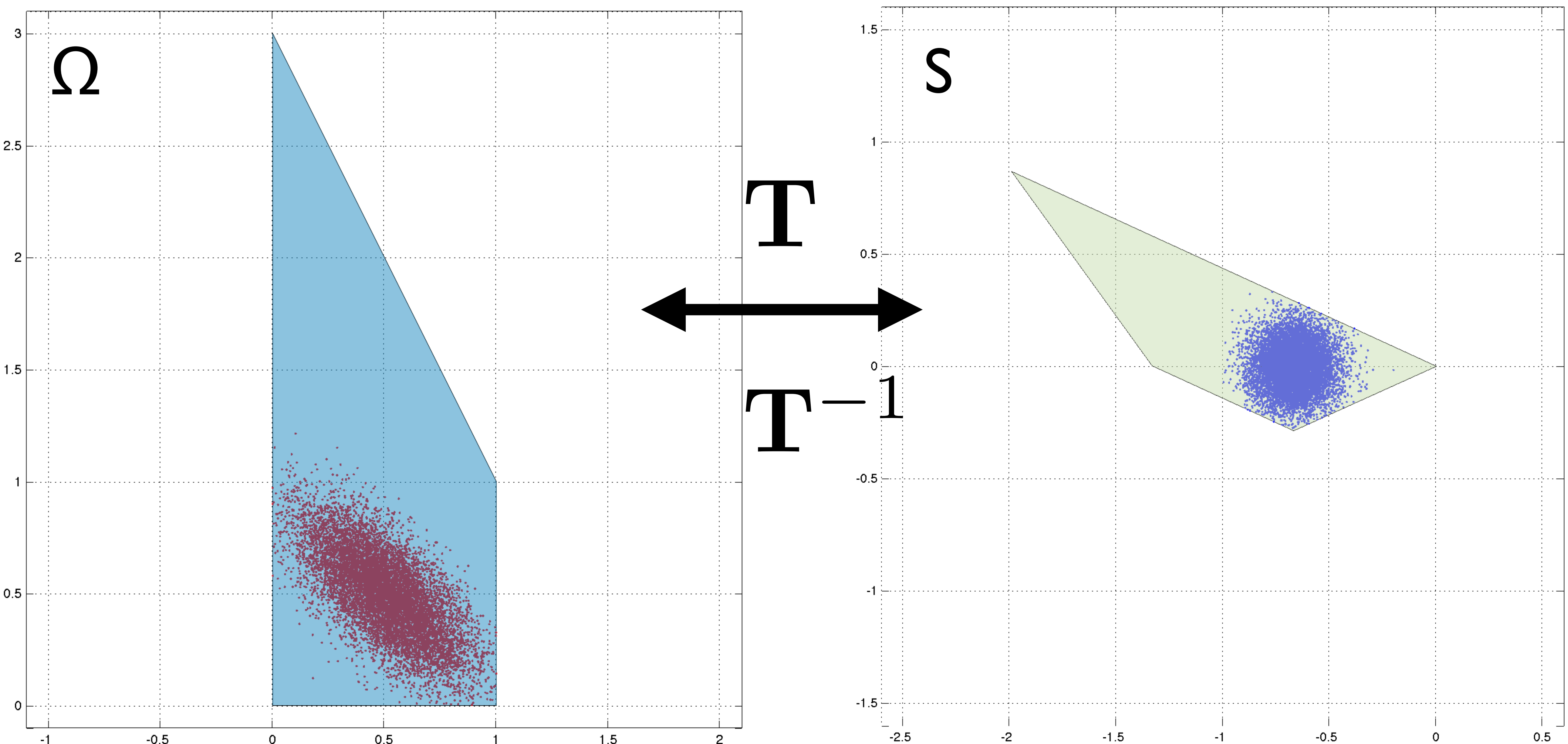
where $S_j = \{z_j \in \mathbb{R}, \mathbf{z} \in \mathbb{R}^n : \mathbf{D}\mathbf{z} \leq \mathbf{b}\}$. Let $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_j, \ldots, , \mathbf{d}_n]$ with $\mathbf{d}_j \in \mathbb{R}^m$ and $\mathbf{D}_{-j}$ the matrix $\mathbf{D}$ without the j^{th} column $\mathbf{d}_j$. S_j can then be computed from the set of at most m linear inequalities $S_j = \{z_j \in \mathbb{R} : \mathbf{d}_j z_j \leq \mathbf{b} - \mathbf{D}_{-j}\mathbf{z}_{-j}\}$ the solution of which forms a one-dimensional convex set, i.e. either an (left/right) open interval or a closed interval.

Gibbs sampling for truncated multivariate normals



$$\mathbf{p}(\mathbf{x}) \propto \mathcal{N}_{\Omega}(\mathbf{m}, \mathbf{C}) = \begin{cases} \mathcal{N}(\mathbf{m}, \mathbf{C}) & \text{if } \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ 0 & \text{otherwise,} \end{cases}$$

Gibbs sampling for truncated multivariate normals

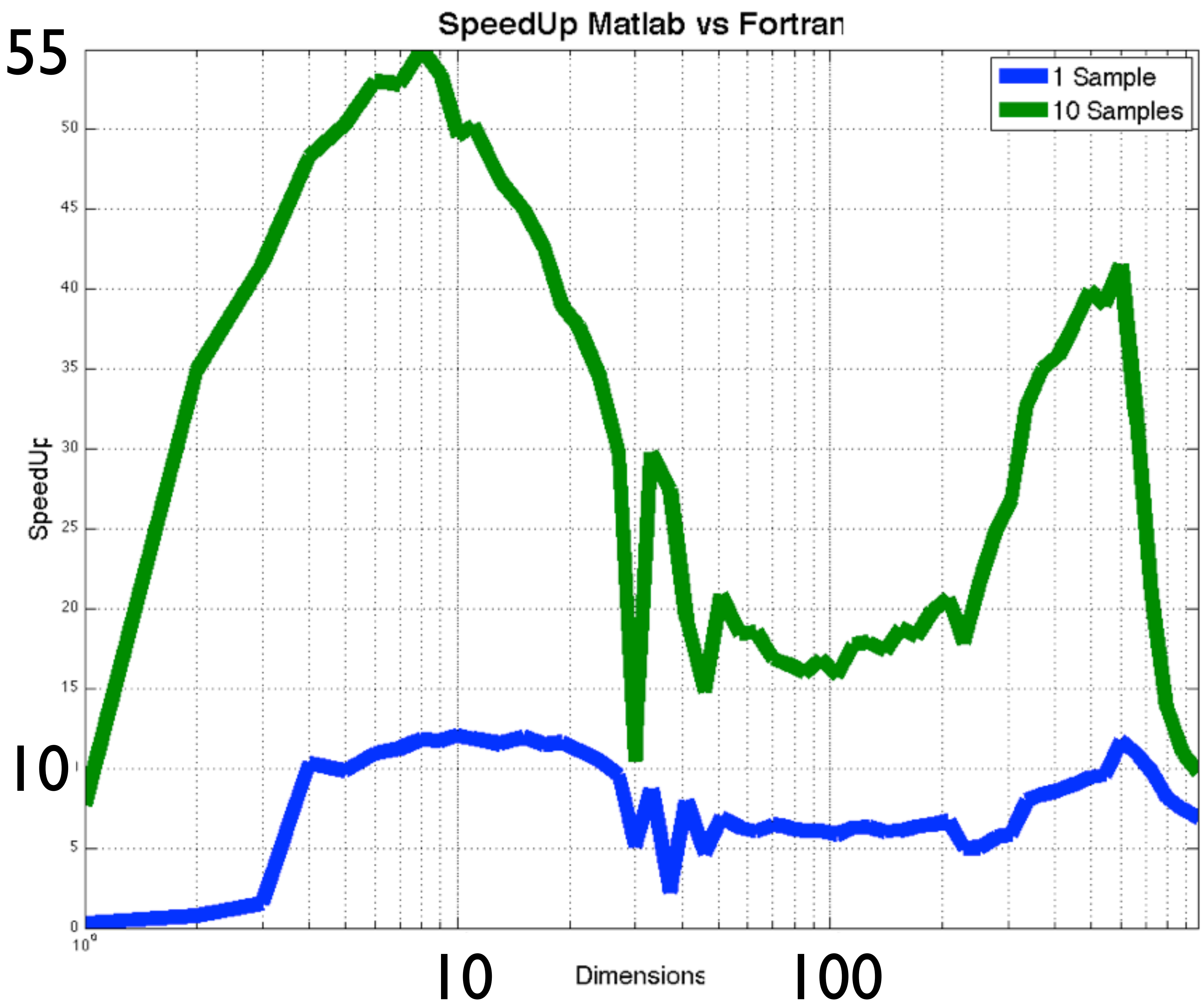


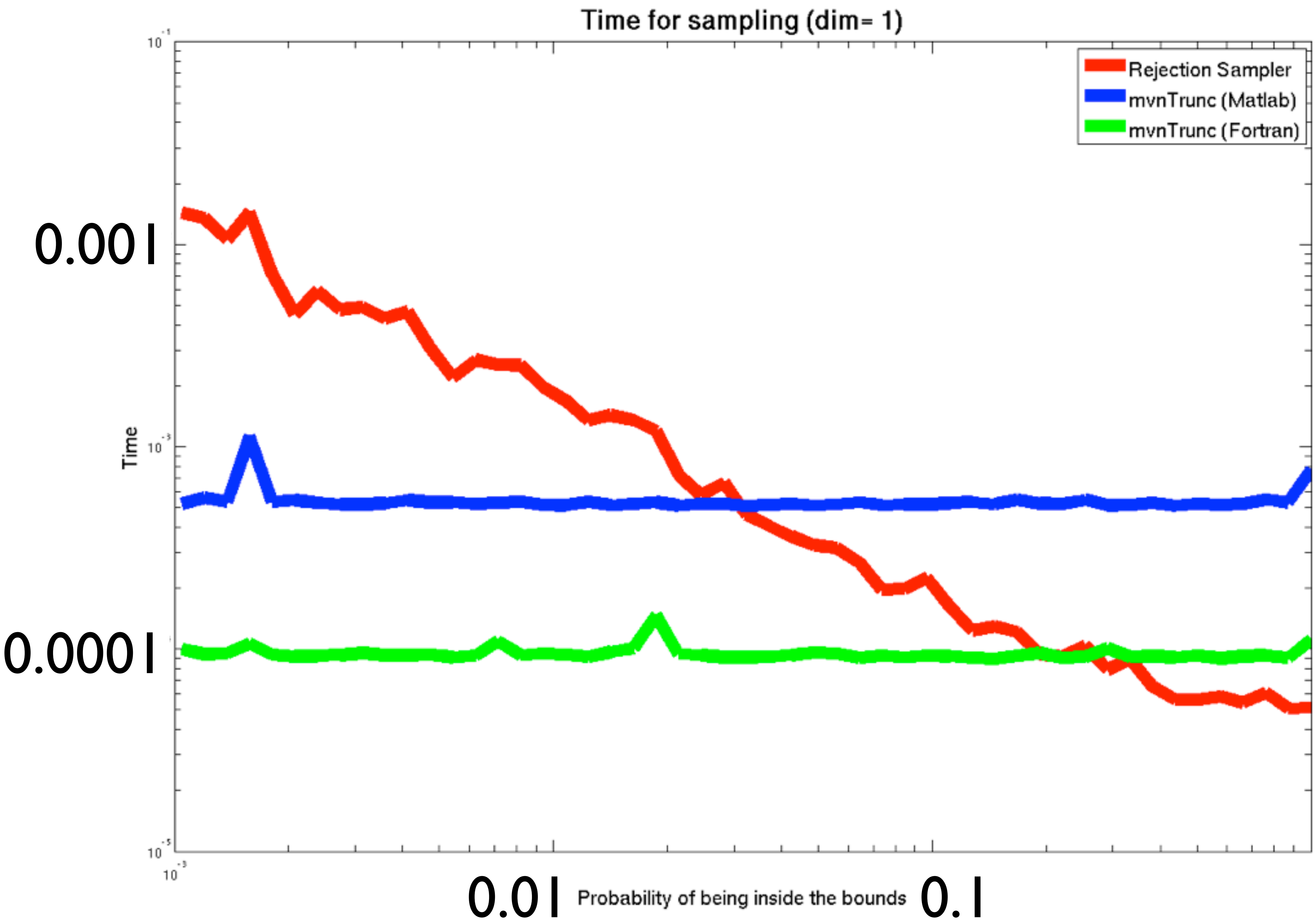
$$\mathbf{p}(\mathbf{x}) \propto \mathcal{N}_\Omega(\mathbf{m},\mathbf{C}) = \begin{cases} \mathcal{N}(\mathbf{m},\mathbf{C}) & \text{if } \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ 0 & \text{otherwise,} \end{cases}$$

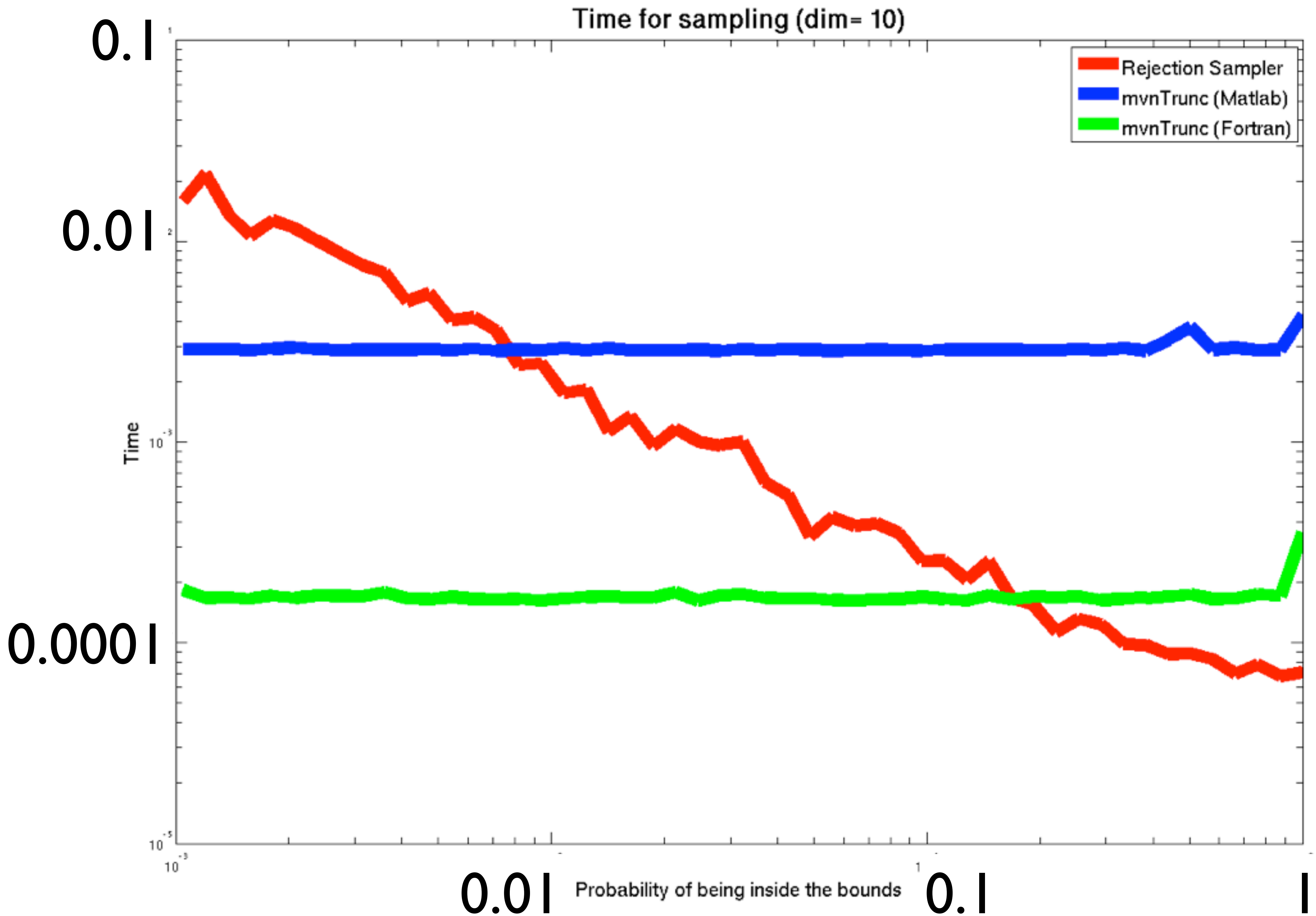
$$\mathbf{p}(\mathbf{z}) \propto \mathcal{N}_\mathbf{S}(\mathbf{c},\sigma^2\mathbf{I}) = \begin{cases} \mathcal{N}(\mathbf{c},\sigma^2\mathbf{I}) & \text{if } \mathbf{D}\mathbf{z} \leq \mathbf{b} \\ 0 & \text{otherwise,} \end{cases}$$

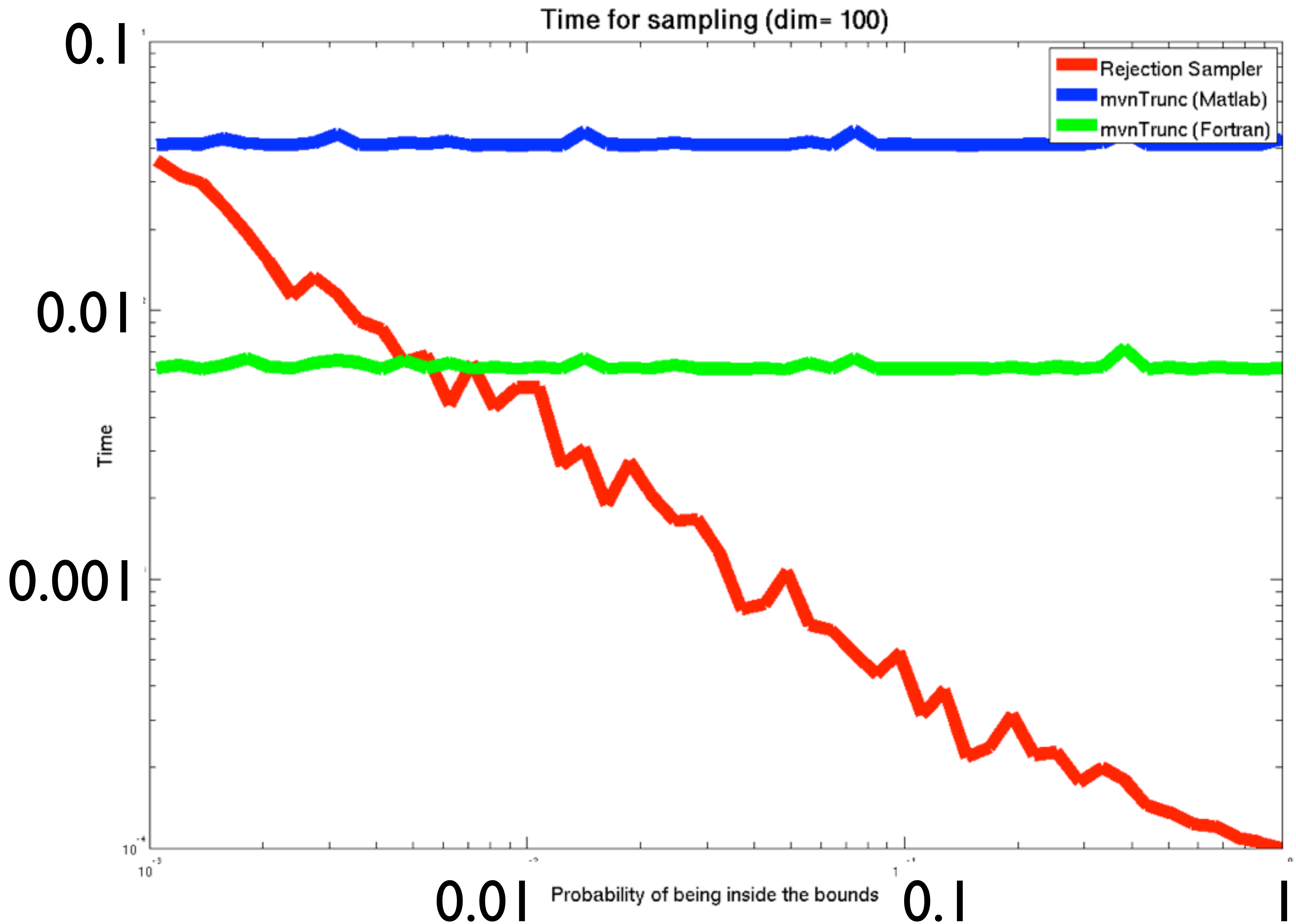
Gibbs sampling for truncated multivariate normals: Efficient Implementation

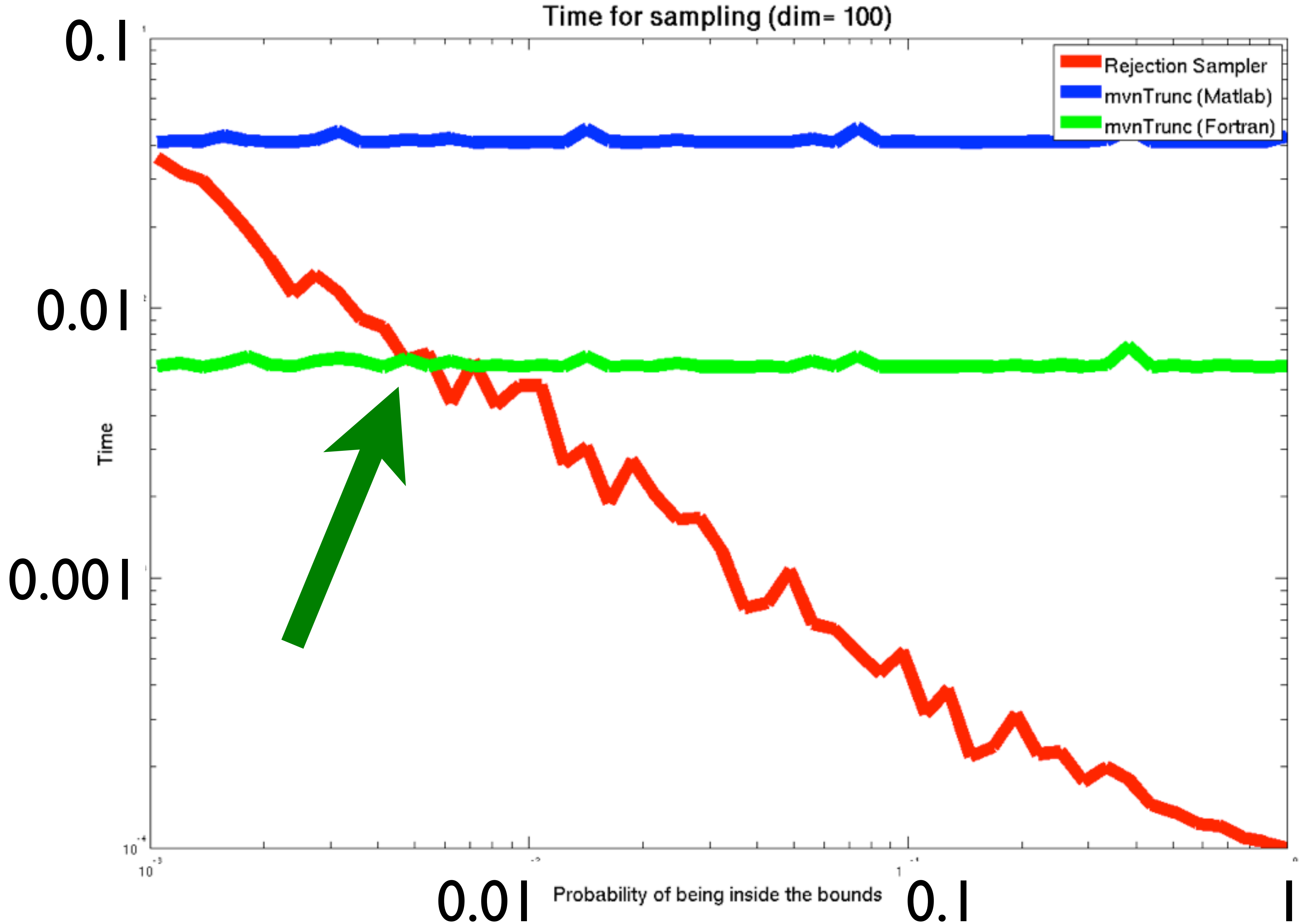
- MATLAB implementation for rapid prototyping
- FORTRAN90 implementation with Intel MKL functions and LAPACK/BLAS routines with mex interface for MATLAB
- Different modes of operations available dependent on whether the eigendecomposition of the covariance matrix is already known (as in CMA-ES and GaA)
- FORTRAN90 code could also be used for usage in R

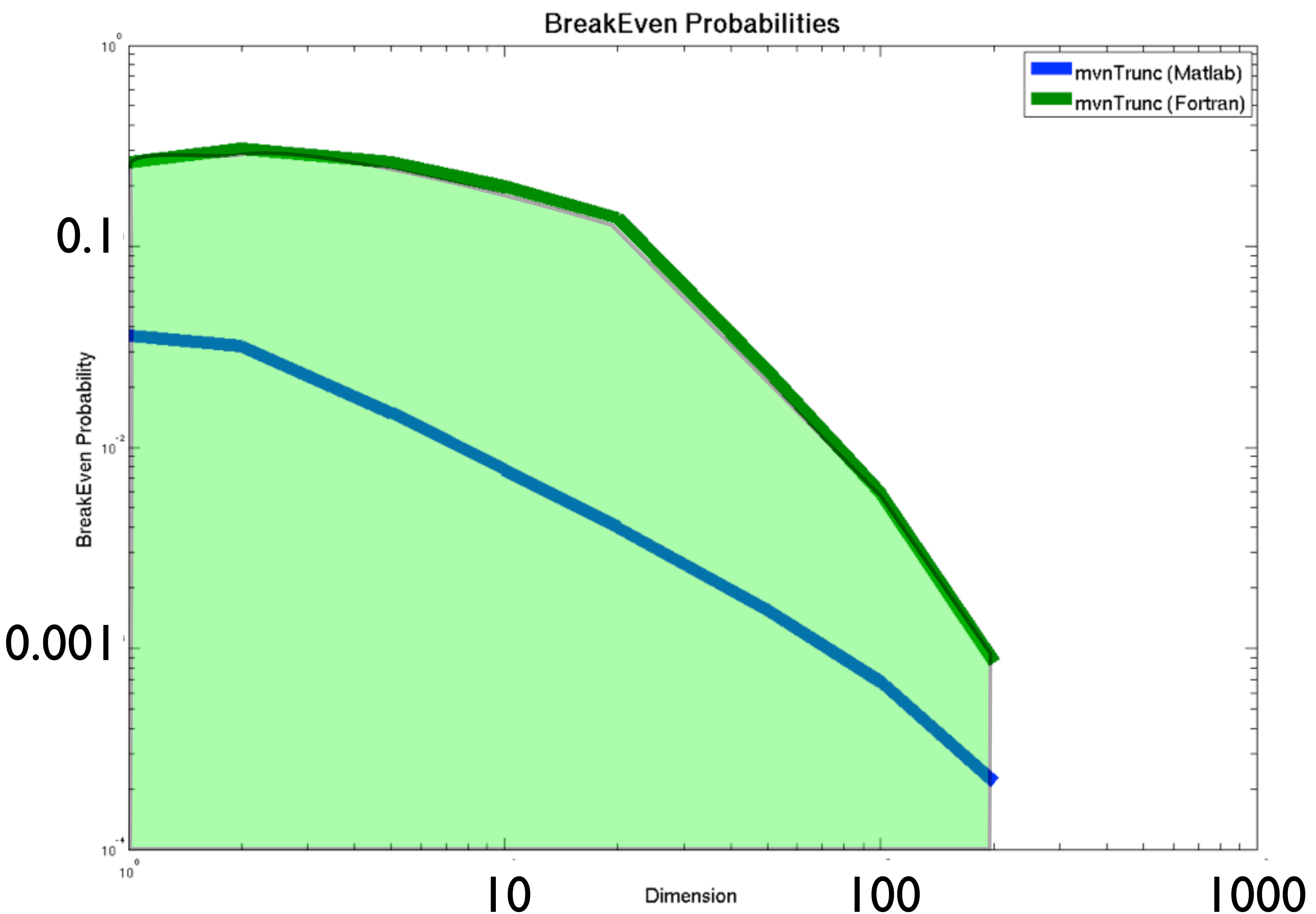

SpeedUp Matlab vs Fortran
1 Sample
10 Samples
SpeedUp
55
50
45
40
35
30
25
20
15
10
5
0
10
100
Dimensions


Time for sampling (dim= 1)
Rejection Sampler
mvnTrunc (Matlab)
mvnTrunc (Fortran)
Time
0.001
0.0001
0.01
Probability of being inside the bounds
0.1
1


Time for sampling (dim= 10)
Rejection Sampler
mvnTrunc (Matlab)
mvnTrunc (Fortran)
Time
Probability of being inside the bounds
0.1
0.01
0.0001
0.01
0.1
1


Time for sampling (dim= 100)
Rejection Sampler
mvnTrunc (Matlab)
mvnTrunc (Fortran)
Time
Probability of being inside the bounds
0.1
0.01
0.001
0.01
0.1
1


Time for sampling (dim= 100)
Rejection Sampler
mvnTrunc (Matlab)
mvnTrunc (Fortran)
Time
Probability of being inside the bounds
0.1
0.01
0.001
0.01
0.1
1



BreakEven Probabilities
mvnTrunc (Matlab)
mvnTrunc (Fortran)
BreakEven Probability
0.1
0.001
Dimension
10
100
1000

Illustrative examples: Sphere function cut by a hyperplane (tangent problem) and CMA-ES

$$\min_{\mathbf{x}\in\mathbb{R}^n} \quad f_{\mathrm{TR}}(\mathbf{x}) = \sum_{i=1}^{n} x_i^2$$

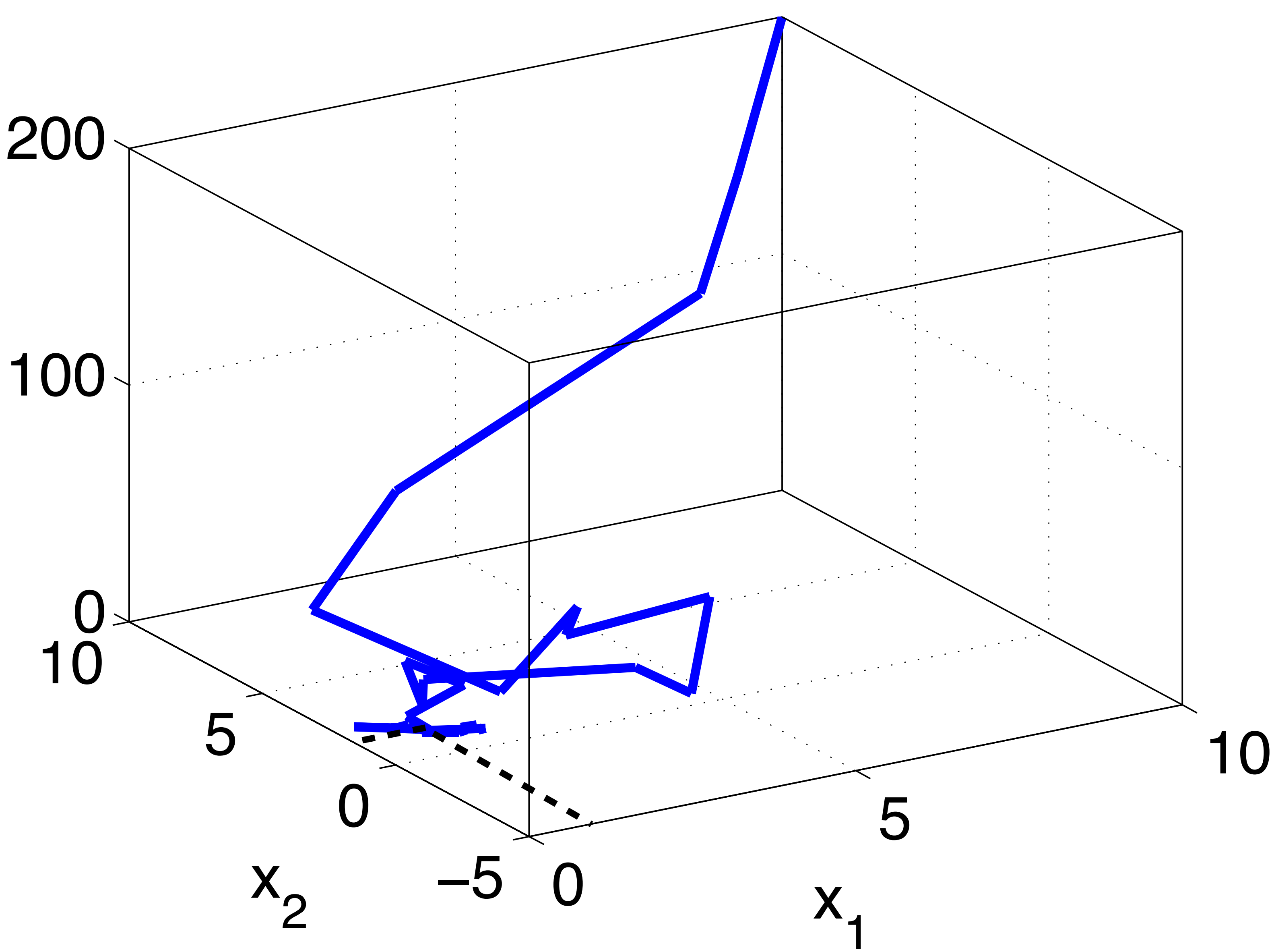
$$\text{s.t.} \quad -\sum_{i=1}^{n} x_i + n \leq 0\,.$$

The optimal solution is $\mathbf{x}^* = [1, \ldots, 1]^T$
with objective function value $f_{\mathrm{TR}}(\mathbf{x}^*) = n$.

Illustrative examples: Sphere function cut by a hyperplane (tangent problem) and CMA-ES

- We chose $x_0 = [10, 10]^T$ and step size $\sigma_0 = 1$ and conducted 25 experiments.
- Constrained CMA-ES has been stopped when the optimum was reached within $\varepsilon < 1e{-}16$
- Minimum, median, and maximum number of function evaluations are 764, 992, and 1346.
- For comparison, the best combination of CMA-ES and a sophisticated meta-model strategy for the constraints needs on average 3,432 function evaluations and 5,326 constraint evaluations (Kramer, 2010)

Illustrative examples: Sphere function cut by a hyperplane (tangent problem) and CMA-ES

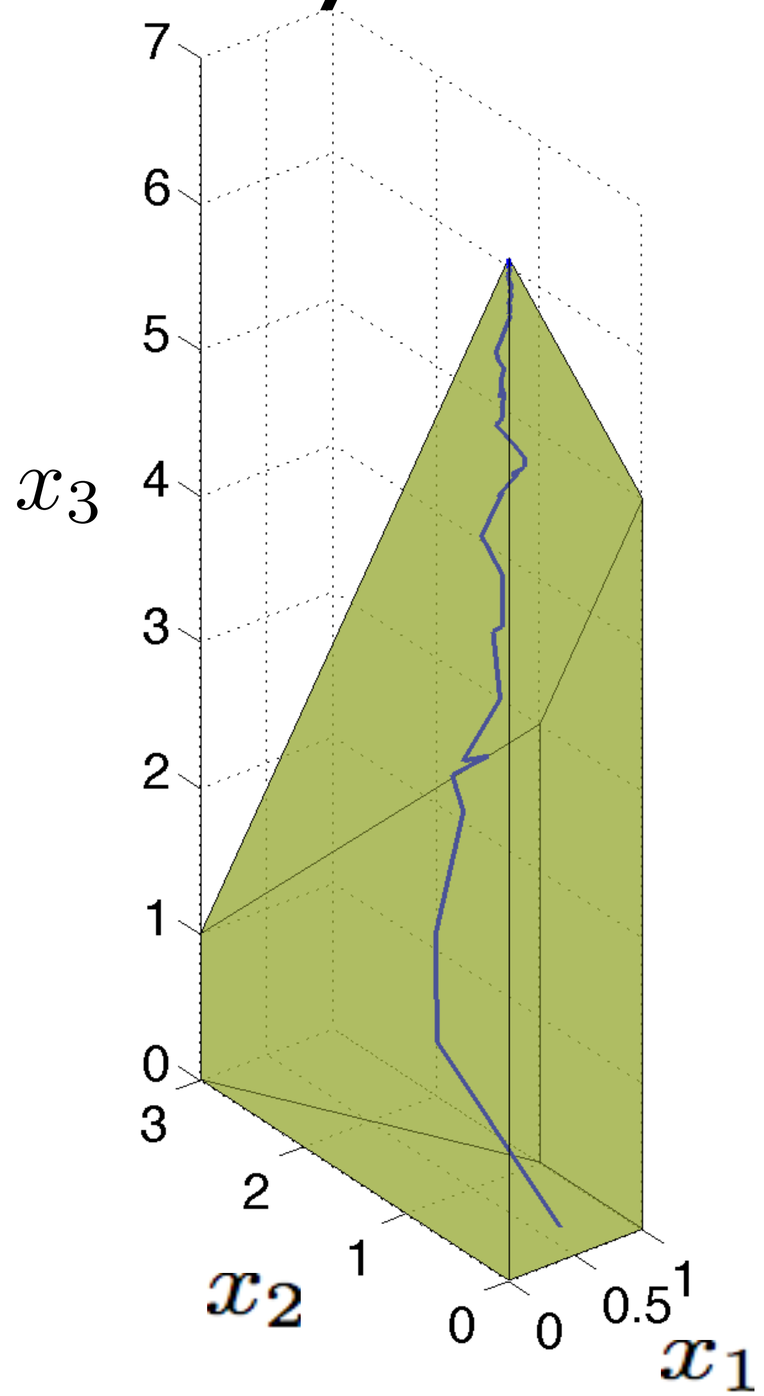


Illustrative examples: Linear functions over Klee-Minty cubes and GaA

$$
\begin{aligned}
\max_{\mathbf{x}\in\mathbb{R}^n} \quad & f_{\mathrm{KM}}(\mathbf{x}) = \textstyle\sum_{i=1}^{n} x_i \\
\text{s.t.} \quad & x_1 \leq 1 \\
& 2\textstyle\sum_{i=1}^{k-1} x_i + x_k \leq 2^k - 1, \quad k = 2, \ldots, n \\
& \mathbf{x} \geq 0
\end{aligned}
$$

The optimal solution is located at $\mathbf{x}^* = [0, 0, \ldots, 1]^T$ with $f_{\mathrm{KM}}(\mathbf{x}^*) = 2^n - 1$.

Illustrative examples: Linear functions over Klee-Minty cubes and GaA



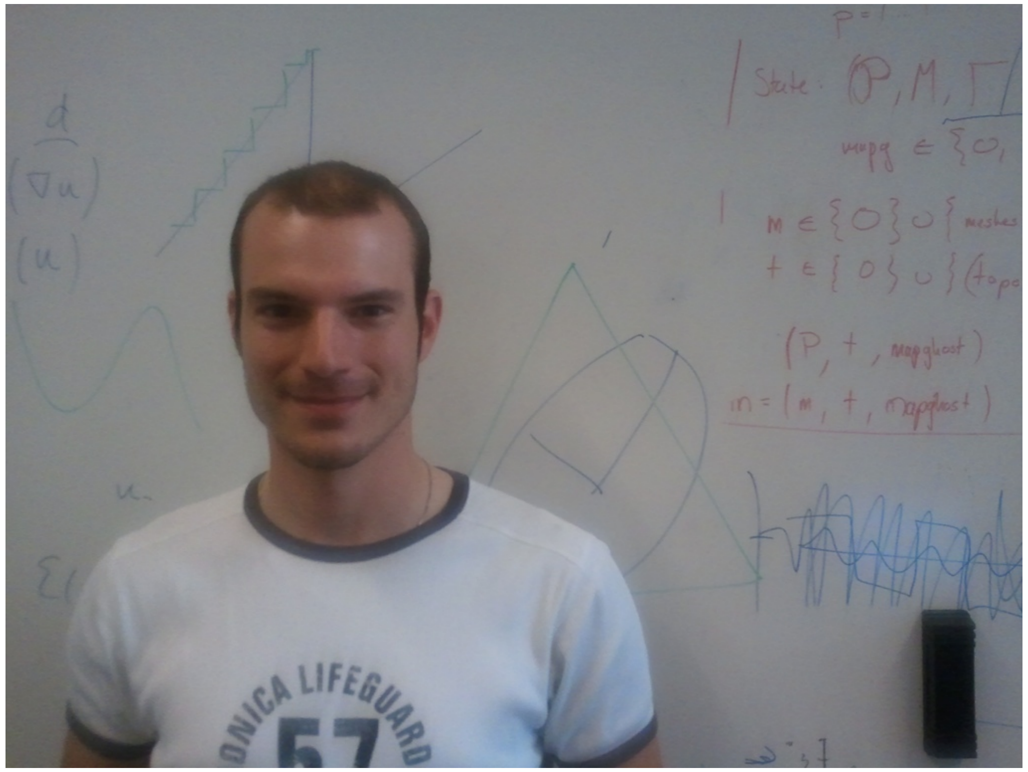
Illustrative examples: Linear functions over Klee-Minty cubes and GaA

- We chose $x_0 = [0,..,0]^T$ (this is kind of the worst case) and $r_0 = 2^{n-1}$ and conducted 25 experiments for n=3,..,16.
- Constrained GaA has been stopped when the optimum was reached within $\varepsilon <$ 1e−3.
- We observe a quadratic relation between dimension n and number of function evaluations.

Conclusions

- General framework for handling linear constraints for black-box optimizer that use the Gaussian as proposal distribution
- Key ingredients is an efficient Gibbs sampler for truncated multivariate Gaussians that has poly(n) complexity
- Efficient implementation available in MATLAB and FORTRAN 90 with MATLAB mex interfacing
- Embedding in CMA-ES and GaA available, so called cCMA-ES and cGaA
- Straight-forward generalization to convex quadratic and convex oracle constraints is possible

Acknowledgements



Patrick Plattner

Merci beaucoup! Thank you for your time!

http://www.mosaic.ethz.ch

christian.mueller@inf.ethz.ch