SMAI-MODE 2020

A DoC APPROACH FOR CHANCE-CONSTRAINED PROBLEMS

Yassine LAGUEL[★] - Joint work with J. Malick[▲] and W. Van Ackooij[◆]

[★] Université Grenoble Alpes - [▲] CNRS - [◆] EDF R&D

Collaboration with

CNRS

J. MALICK

EDF R&D

W. VAN ACKOOIJ

Definition of Chance Constraints

- A chance constraint problem is of the form:

$$\min_{x \in \mathbb{R}^d} f(x)$$
$$\text{s.t.} \ \ \mathbb{P}[g(x, \xi) \leq 0] \geq p$$

Definition of Chance Constraints

- A chance constraint problem is of the form:

Objective function

Random variable $\xi : \Omega \to \mathbb{R}^m$

$$\min_{x \in \mathbb{R}^d} f(x)$$
$$\text{s.t. } \mathbb{P}[g(x, \xi) \leq 0] \geq p$$

Safety probability level $p \in [0, 1)$

Chance constraint

Definition of Chance Constraints

- A chance constraint problem is of the form:

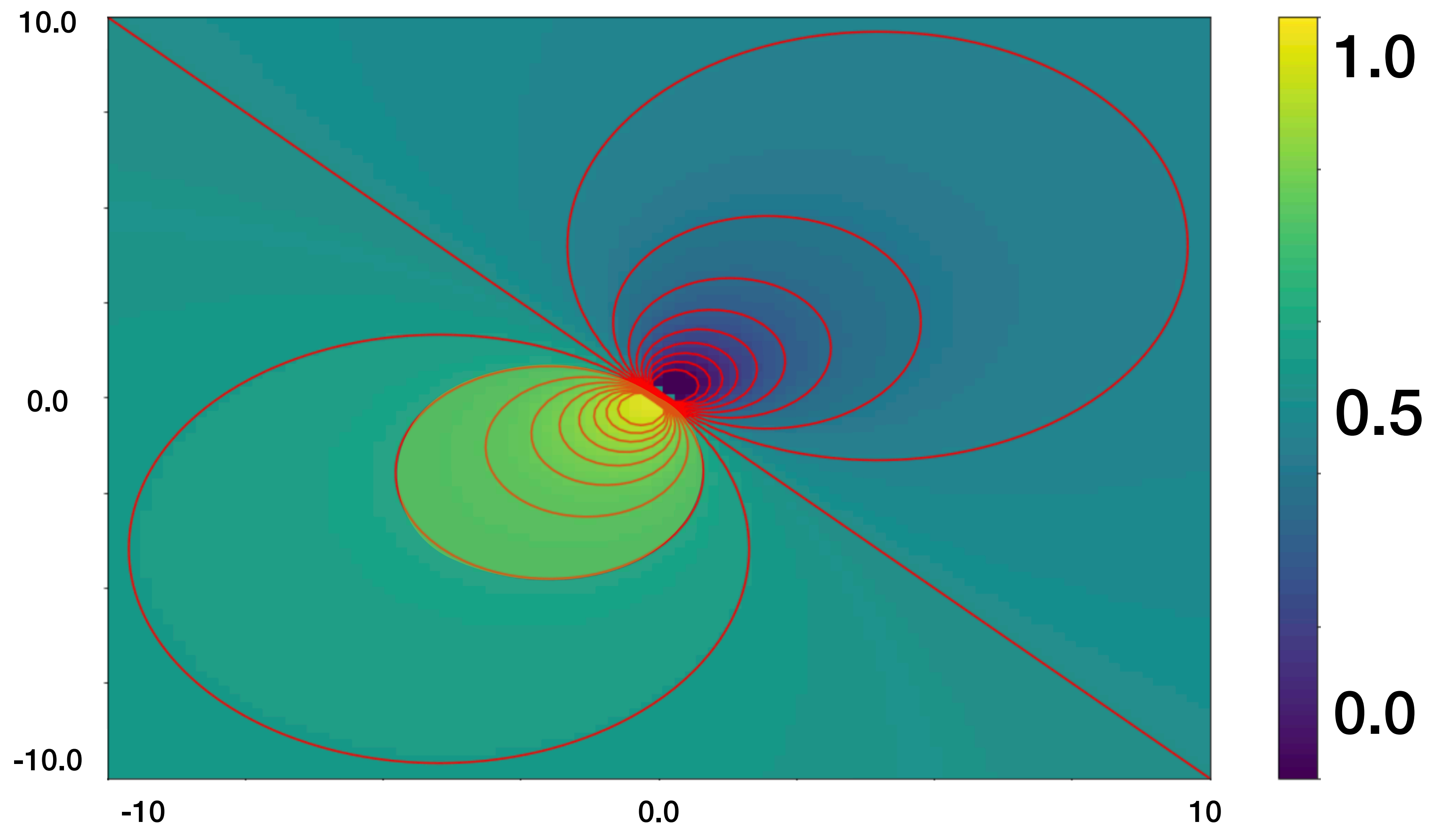
$$\min_{x \in \mathbb{R}^d} f(x)$$
$$\text{s.t. } \mathbb{P}[g(x,\xi) \leq 0] \geq p$$

- Chance constrained optimization problems are difficult:
  - non-convex
  - non-smooth

Non-convexity of Chance Constraints

- A classical result
  [Kataoka 1963]
  - Take $g : (x, \xi) \mapsto x^\top \xi$
  - Take $\xi \sim \mathcal{N}(\mu, \Sigma)$

Eventual Convexity on 2D Example



$$\mu = \mathbb{E}[\xi] = (1,1)^\top, \mathbb{V}ar(\xi) = I_2$$

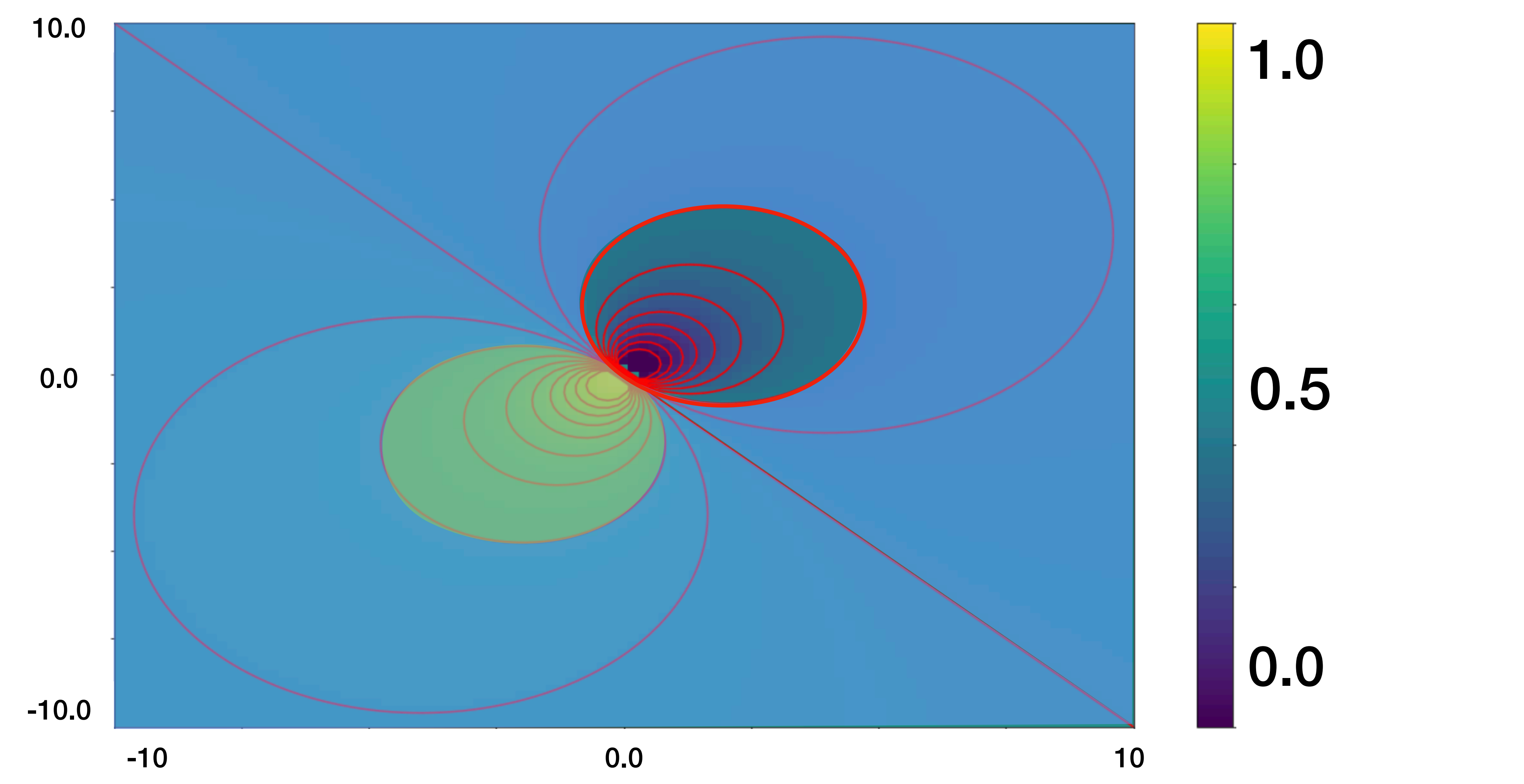
- Studying under which conditions chance constraints are convex

Henrion, Strugarek 06’ | Van Ackooij ‘15 | Van Ackooij, Malick ‘19

Non-convexity of Chance Constraints

- A classical result
  [Kataoka 1963]
  - Take $g : (x, \xi) \mapsto x^\top \xi$
  - Take $\xi \sim \mathcal{N}(\mu, \Sigma)$

Eventual Convexity on 2D Example



$$\mu = \mathbb{E}[\xi] = (1,1)^\top, \mathbb{V}ar(\xi) = I_2$$

- Studying under which conditions chance constraints are convex

Henrion, Strugarek 06'   Van Ackooij '15   Van Ackooij, Malick '19

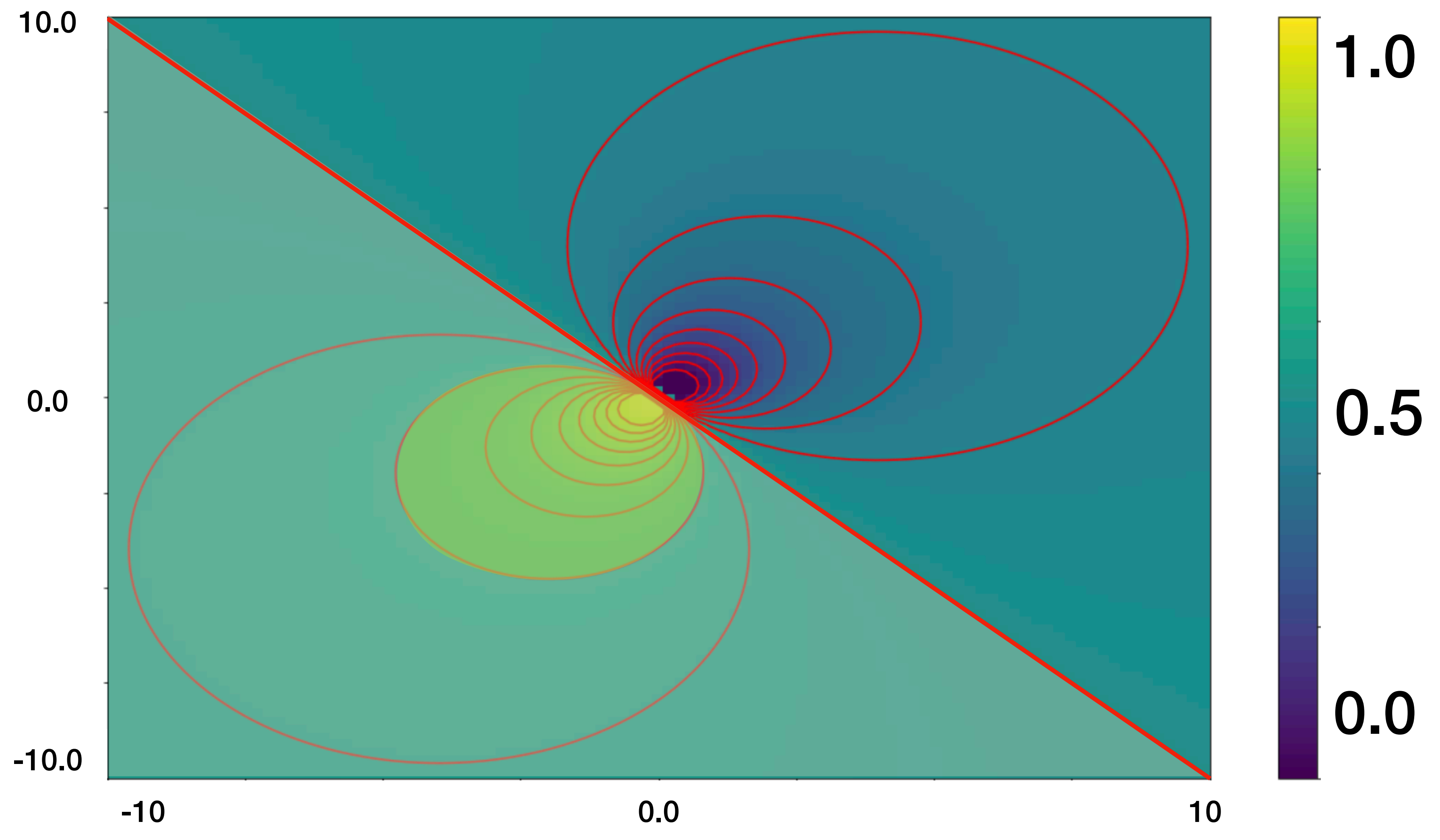
Non-convexity of Chance Constraints

- A classical result

  [Kataoka 1963]

  - Take $g : (x, \xi) \mapsto x^\top \xi$
  - Take $\xi \sim \mathcal{N}(\mu, \Sigma)$

Eventual Convexity on 2D Example



$$\mu = \mathbb{E}[\xi] = (1, 1)^\top, \mathbb{V}ar(\xi) = I_2$$

- Studying under which conditions chance constraints are convex

Henrion, Strugarek 06' | Van Ackooij ‘15 | Van Ackooij, Malick ‘19

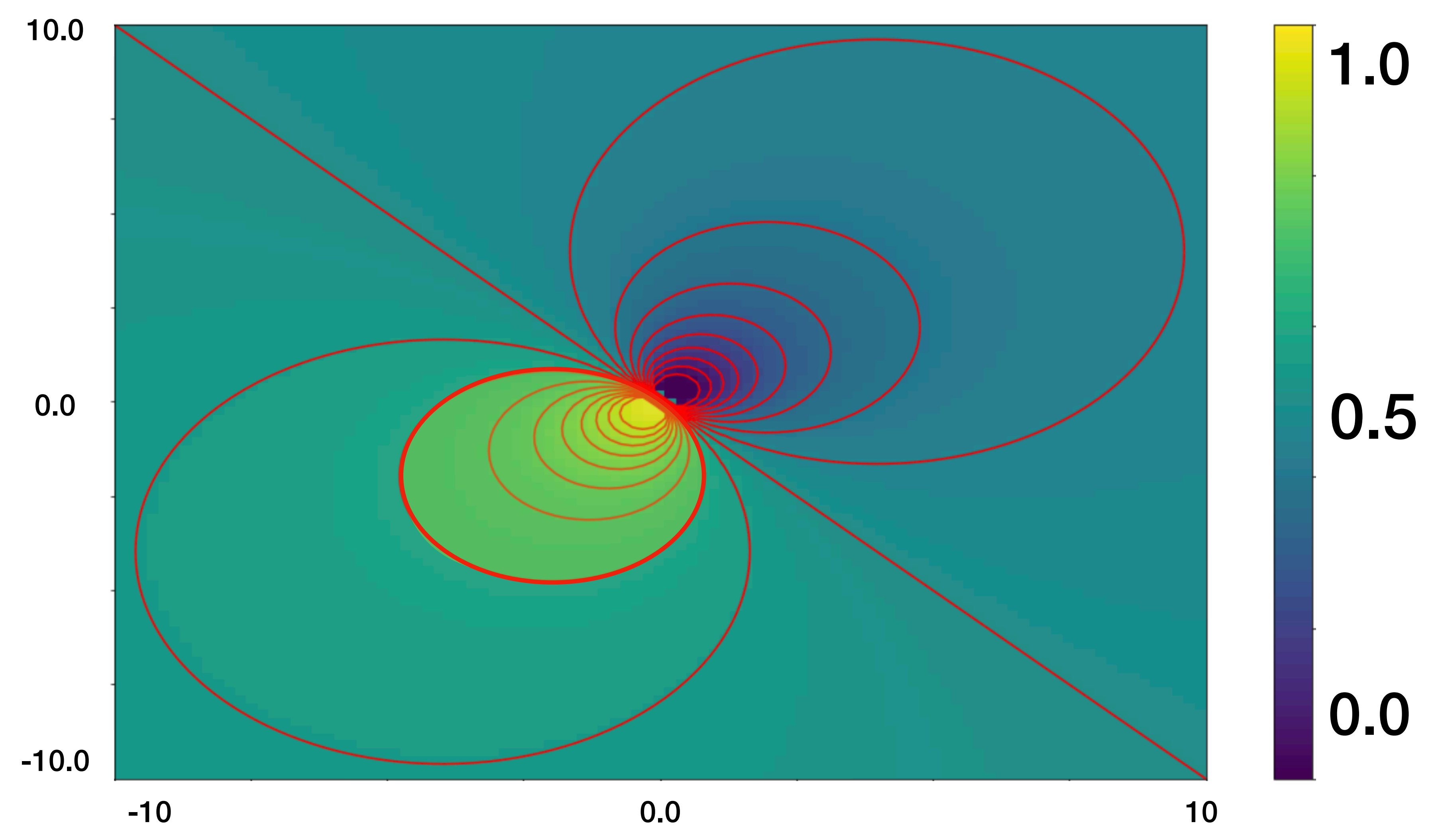
Non-convexity of Chance Constraints

- A classical result

  [Kataoka 1963]

  - Take $g : (x, \xi) \mapsto x^\top \xi$
  - Take $\xi \sim \mathcal{N}(\mu, \Sigma)$

Eventual Convexity on 2D Example



$$\mu = \mathbb{E}[\xi] = (1, 1)^\top, \mathbb{V}ar(\xi) = I_2$$

- Studying under which conditions chance constraints are convex

  Henrion, Strugarek 06’ | Van Ackooij ‘15 | Van Ackooij, Malick ‘19

Non-smoothness of Chance Constraints

- Consider the discrete case : $\xi \in \{\xi_1, \ldots, \xi_n\}$

$$\mathbb{P}[g(x,\xi) \leq 0] = \frac{1}{n} \sum_{i=1}^{n} \underbrace{\mathbb{1}_{g(x,\xi_i) \leq 0}}$$

Not even continuous !

- Recent works study the generalized differentiability properties of chance constraints

Van Ackooij, Henrion, '17 | **Geletu, Hoffmann, '19** | **Heitsch, '19**

A data-driven setting

- Many existing approaches
  - MINLP approaches — **Pagnoncelli, Ahmed, Shapiro(2009)**
  - Boolean Methods — **Kogan, Lejeune (2014)**
  - DoC approaches — **Hong, Yang, Zhang (2009)**

    Paul Javal's talk yesterday

- In this talk,
  - f is convex.
  - $g(., z)$ is convex for all $z \in \mathbb{R}^m$
  - ξ is discrete : $\xi \in \{\xi_1, \ldots, \xi_n\} \subset \mathbb{R}^m$

PLANNING

1 Chance Constraints are Bilevel Programs

2 Penalization Method

3 TACO

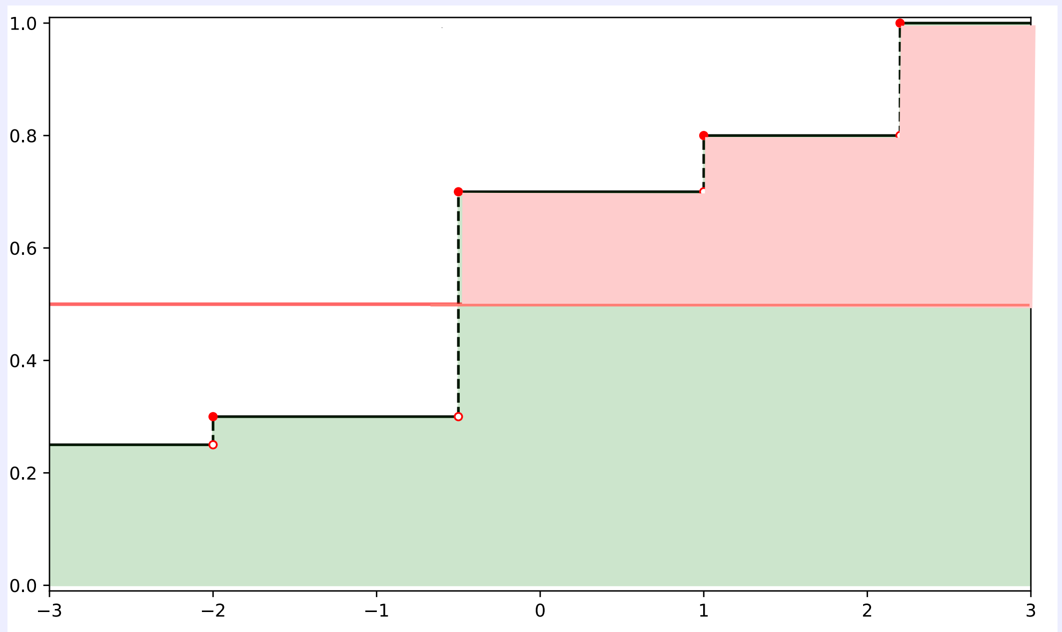
4 Numerical Illustrations

1 Revealing the bilevel structure of Chance Constraints

CDFs, Quantiles & Superquantiles

- Recall that for any real random variable U,
  - its cumulative distribution function, $F_U : \mathbb{R} \to [0, 1]$:

$$F_U(t) = \mathbb{P}[U \leq t]$$



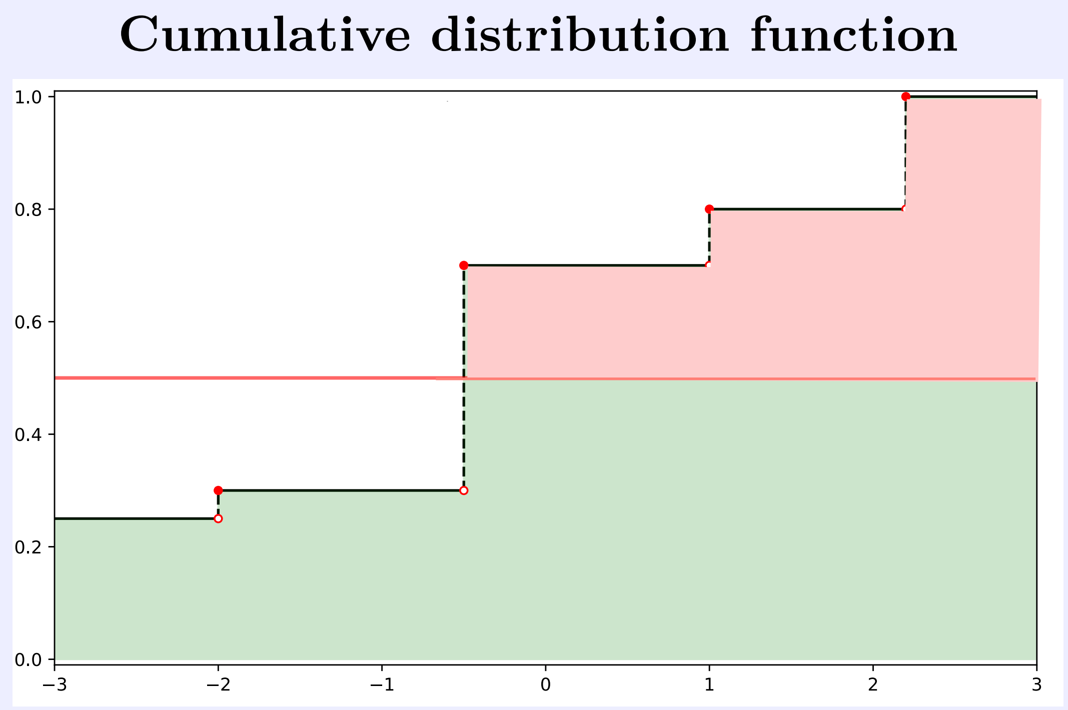
CDFs, Quantiles & Superquantiles

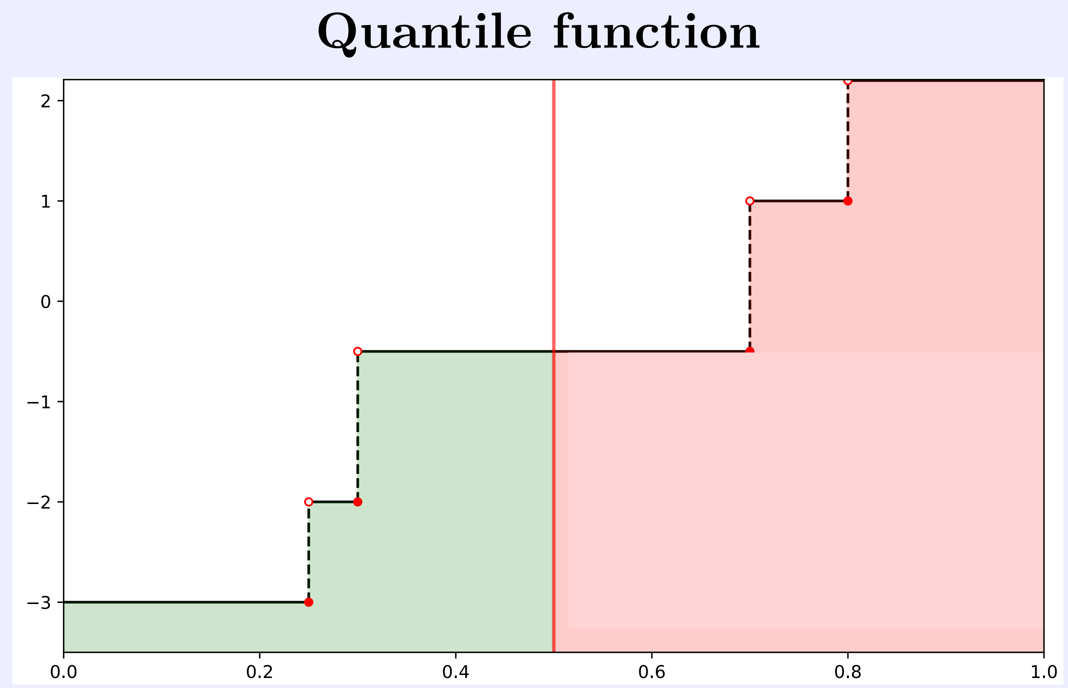
- Recall that for any real random variable U,
  - its cumulative distribution function, $F_U : \mathbb{R} \to [0,1]$:

$$F_U(t) = \mathbb{P}[U \le t]$$

  - for any $p \in [0,1)$, its p-quantile $Q_p(U)$:

$$Q_p(U) = \inf\{t \in \mathbb{R}, \mathbb{P}[U \le t] \ge p\}$$

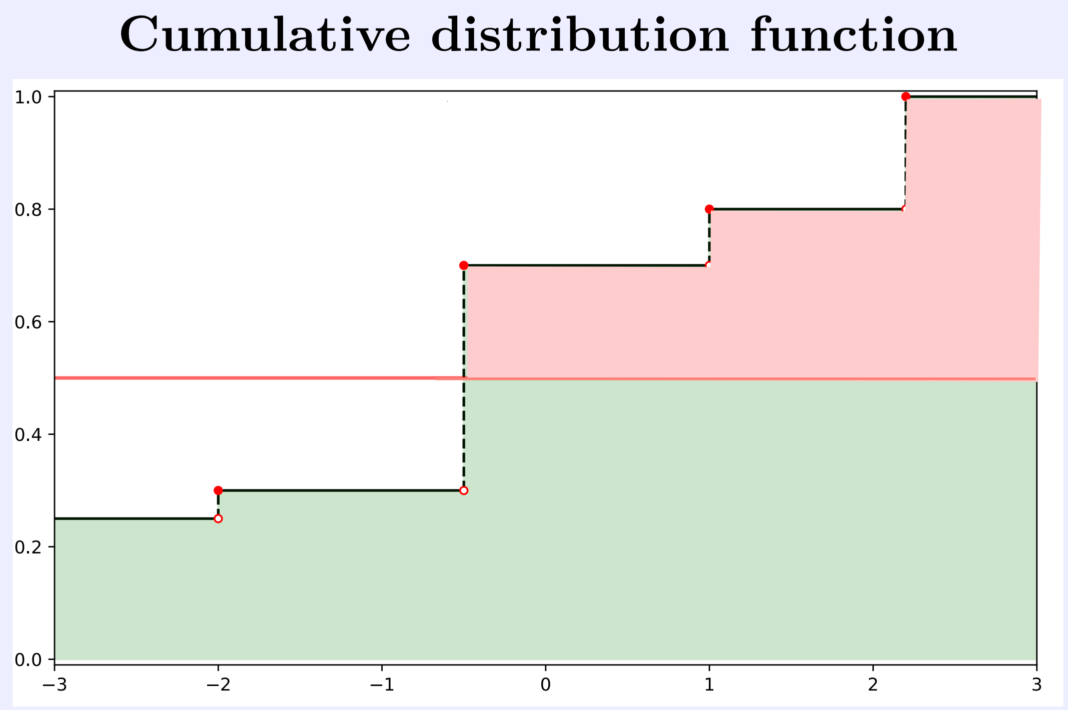




CDFs, Quantiles & Superquantiles

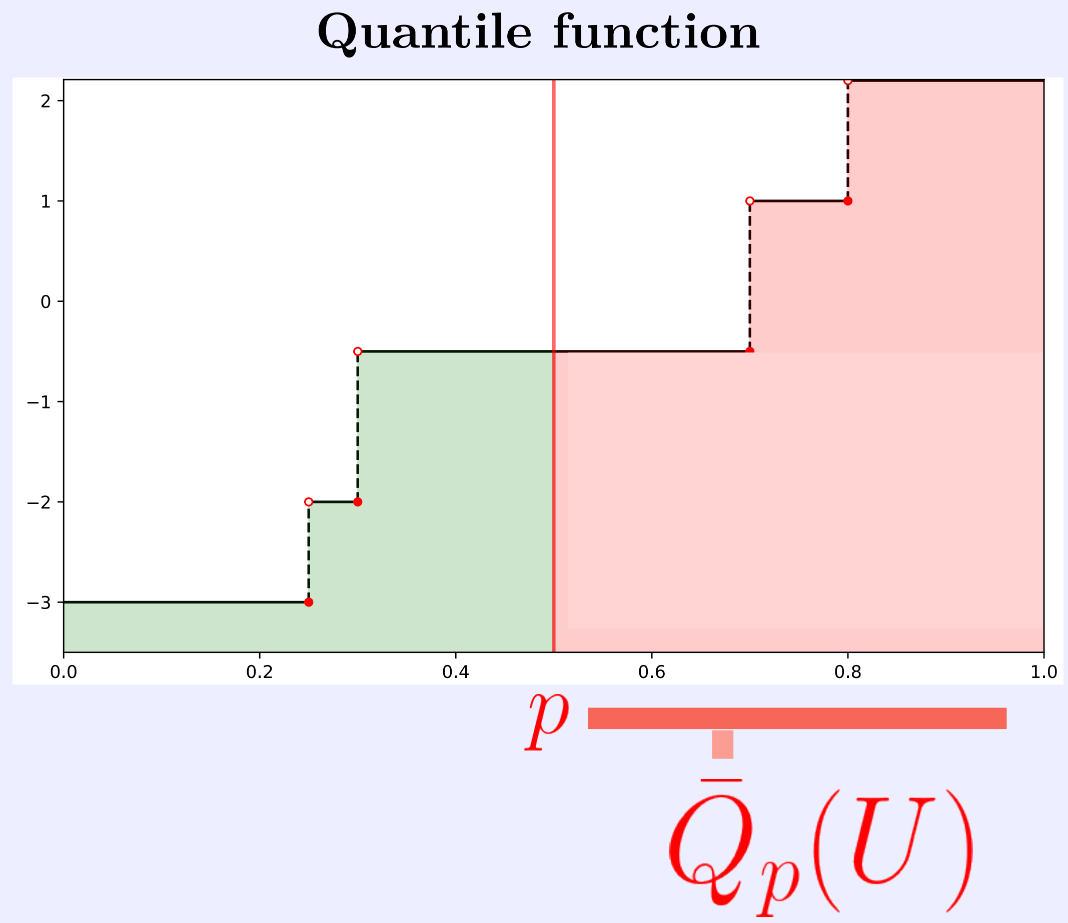
- Recall that for any real random variable U,
  - its cumulative distribution function, $F_U : \mathbb{R} \to [0,1]$:

$$F_U(t) = \mathbb{P}[U \leq t]$$



  - for any $p \in [0,1)$, its p-quantile $Q_p(U)$:

$$Q_p(U) = \inf\{t \in \mathbb{R}, \mathbb{P}[U \leq t] \geq p\}$$



  - for any $p \in [0,1)$, its p-superquantile $\bar{Q}_p(U)$:

$$\bar{Q}_p(U) = \frac{1}{1-p}\int_{p'=p}^{1} Q_{p'}(U)dp'$$

CDFs, Quantiles & Superquantiles

- Recall that for any real random variable U,
  - its cumulative distribution function, $F_U : \mathbb{R} \to [0, 1]$:

Invert

$$F_U(t) = \mathbb{P}[U \leq t]$$

  - for any $p \in [0, 1)$, its p-quantile $Q_p(U)$:

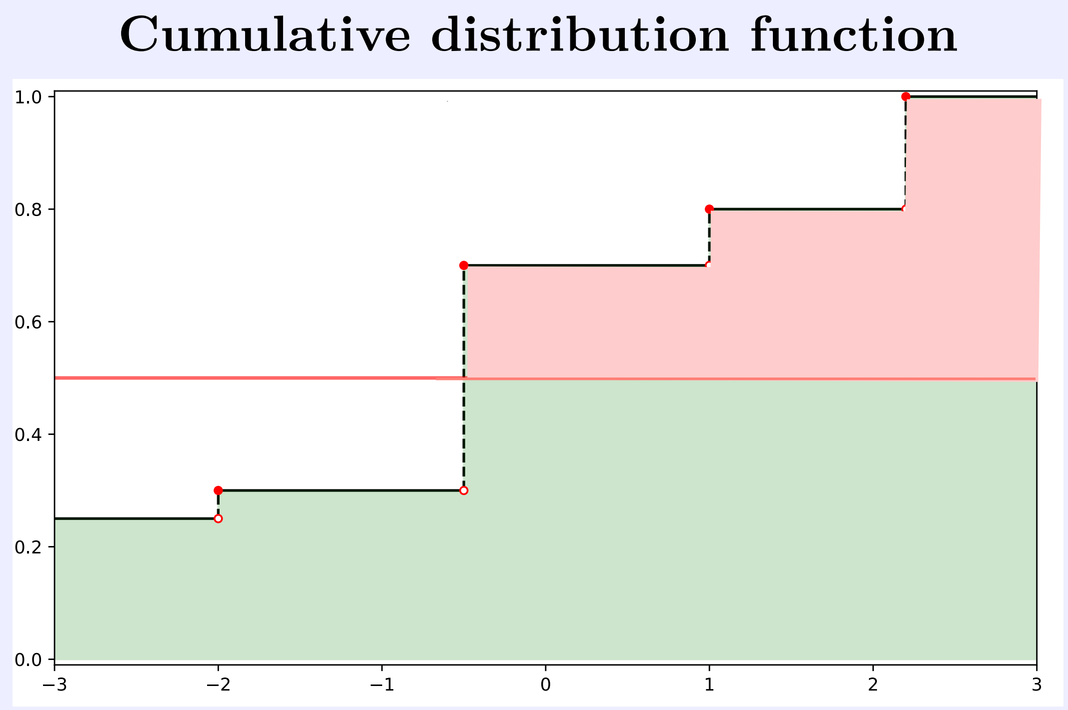
Integrate

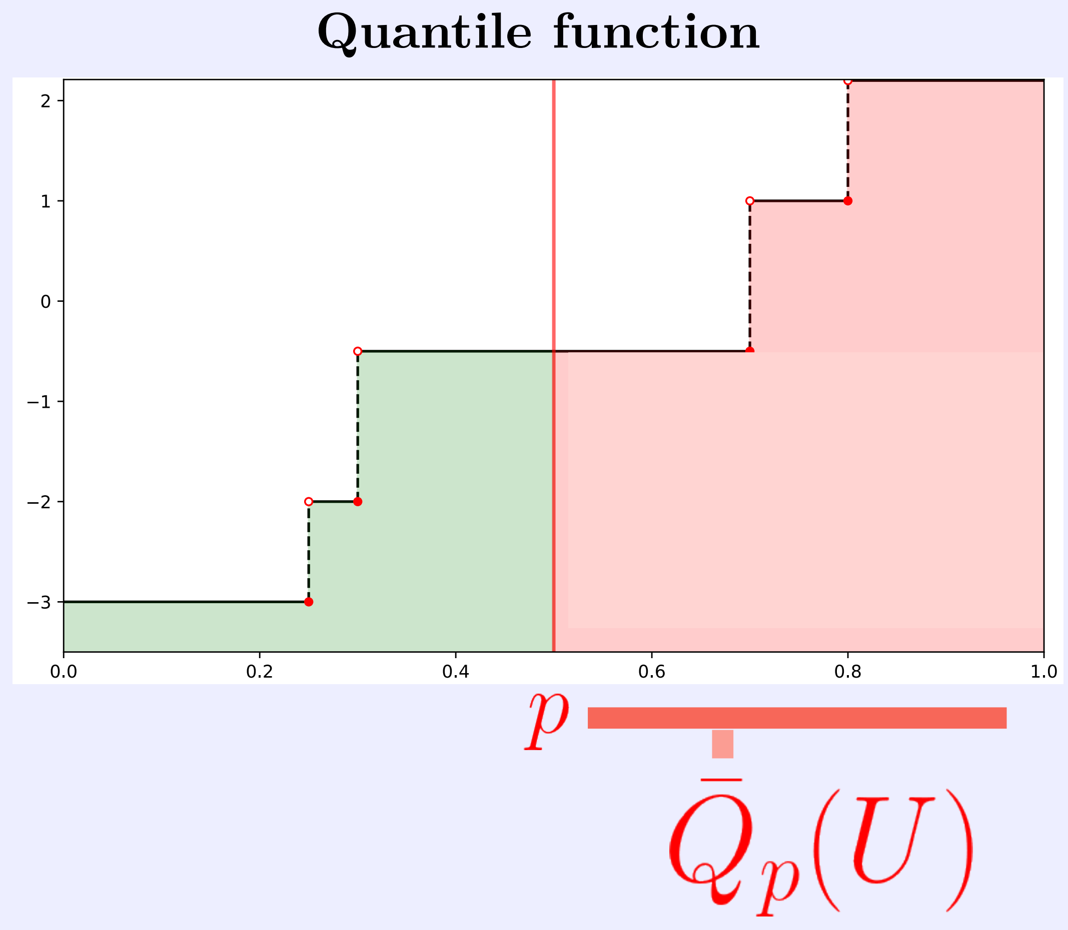
Differentiate

$$Q_p(U) = \inf \{t \in \mathbb{R}, \mathbb{P}[U \leq t] \geq p\}$$

  - for any $p \in [0, 1)$, its p-superquantile $\bar{Q}_p(U)$:

$$\bar{Q}_p(U) = \frac{1}{1-p} \int_{p'=p}^{1} Q_{p'}(U) dp'$$





Rockafellar's Duality Result

- **Rockafellar, Uryasev (2000)**: Superquantile and quantiles are optimal value and optimal solutions resp. of a same one-dimensional convex optimization problem.

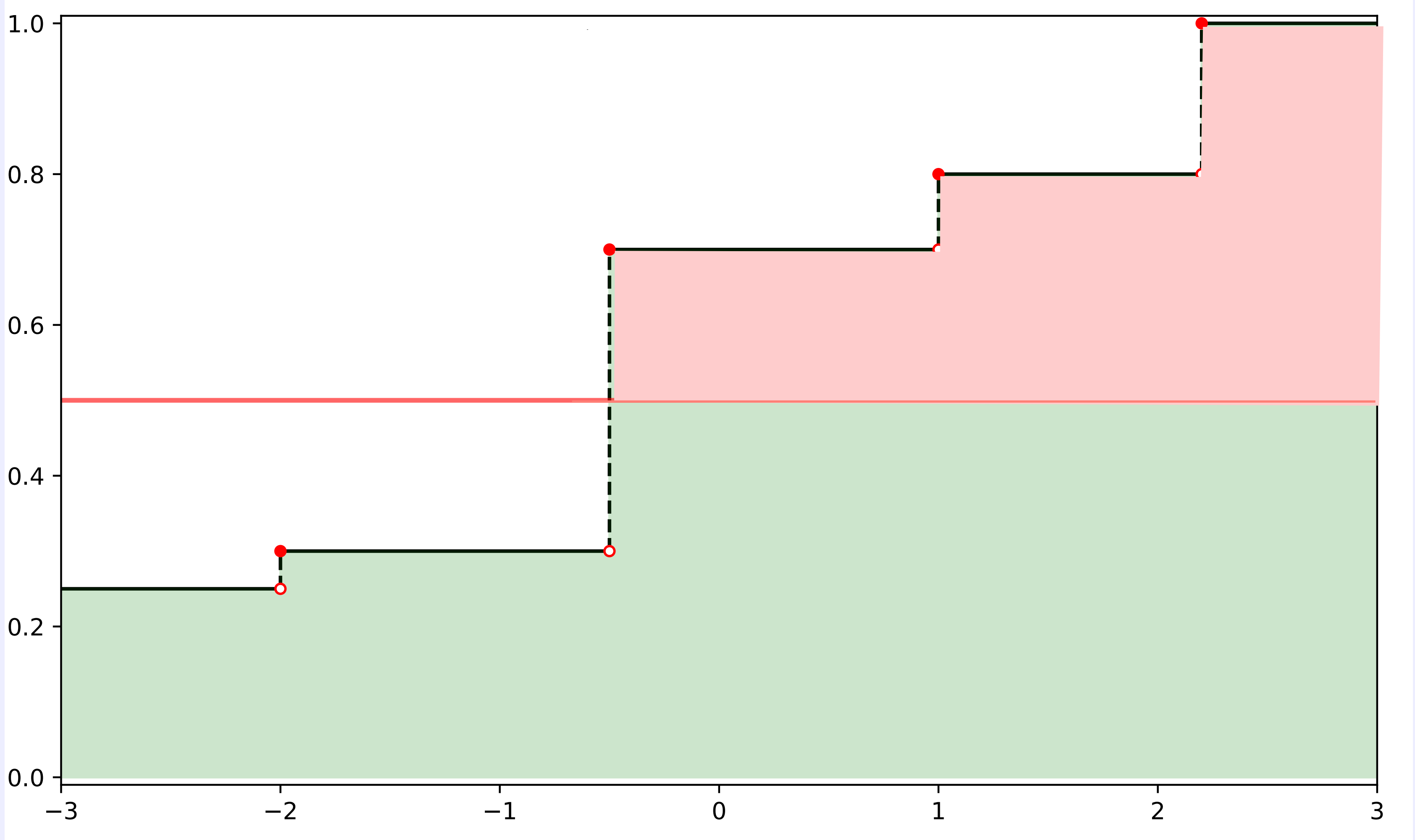
$$\bar{Q}_p(U) = \min_{\eta \in \mathbb{R}} \eta + \frac{1}{1-p}\mathbb{E}[\max(U-\eta, 0)]$$
$$Q_p(U) = \operatorname*{argmin}_{\eta \in \mathbb{R}} \eta + \frac{1}{1-p}\mathbb{E}[\max(U-\eta, 0)]$$

Rockafellar’s Duality Result

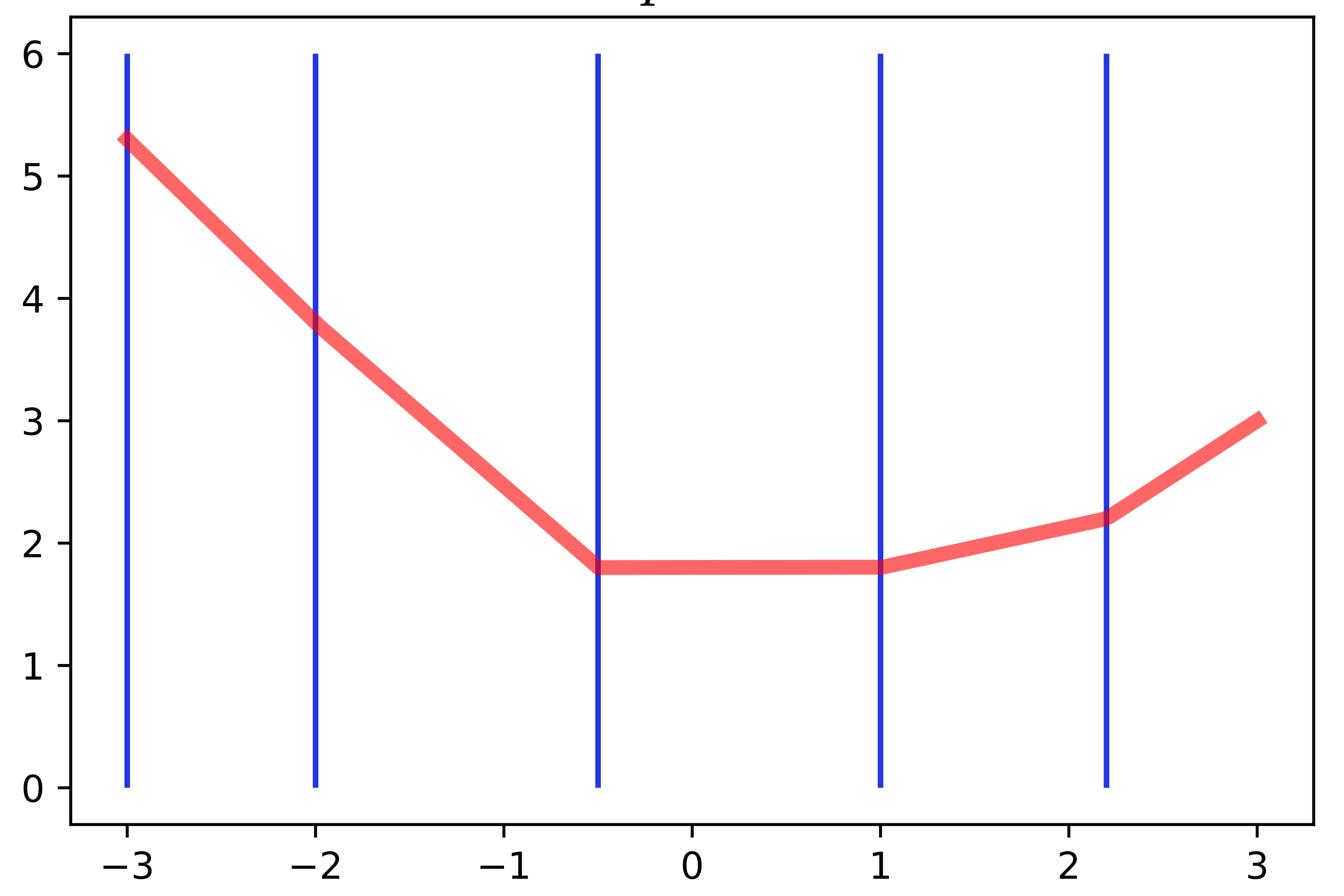
- **Rockafellar, Uryasev (2000)**: Superquantile and Quantiles are optimal value and optimal solutions resp. of a same one-dimensional convex optimization problem.

$$\bar{Q}_p(U) = \min_{\eta \in \mathbb{R}} \eta + \frac{1}{1-p}\mathbb{E}[\max(U-\eta,0)]$$
$$Q_p(U) = \operatorname*{argmin}_{\eta \in \mathbb{R}} \eta + \frac{1}{1-p}\mathbb{E}[\max(U-\eta,0)]$$

Cumulative distribution function



$$\eta \mapsto \eta + \frac{1}{1-p}\mathbb{E}[\max(U-\eta,0)]$$



From Chance Constraints to Bilevel Programs

- **Our approach**: rewrite chance constraints as

$$\mathbb{P}[g(x,\xi) \leq 0] \geq p \;\Leftrightarrow\; Q_p(g(x,\xi)) \leq 0$$

From Chance Constraints to Bilevel Programs

- **Our approach**: rewrite chance constraints as

$$\mathbb{P}[g(x,\xi) \leq 0] \geq p \iff Q_p(g(x,\xi)) \leq 0$$

$$\iff \eta \leq 0$$

$$\eta \in \underset{s \in \mathbb{R}}{\operatorname{argmin}}\ s + \frac{1}{1-p}\mathbb{E}\left[\max(g(x,\xi) - s, 0)\right]$$

- We obtain the following bilevel program:

Upper Level

$$\min_{x \in \mathbb{R}^d, \eta \in \mathbb{R}} \quad f(x)$$
$$\text{s.t.} \quad \eta \leq 0$$

Lower Level

$$\eta \in \underset{s \in \mathbb{R}}{\operatorname{argmin}}\ s + \frac{1}{1-p}\mathbb{E}\left[\max(g(x,\xi) - s, 0)\right]$$

2 A double penalization method for Chance Constraints

Recall Penalization on a Picture

- The penalization procedure

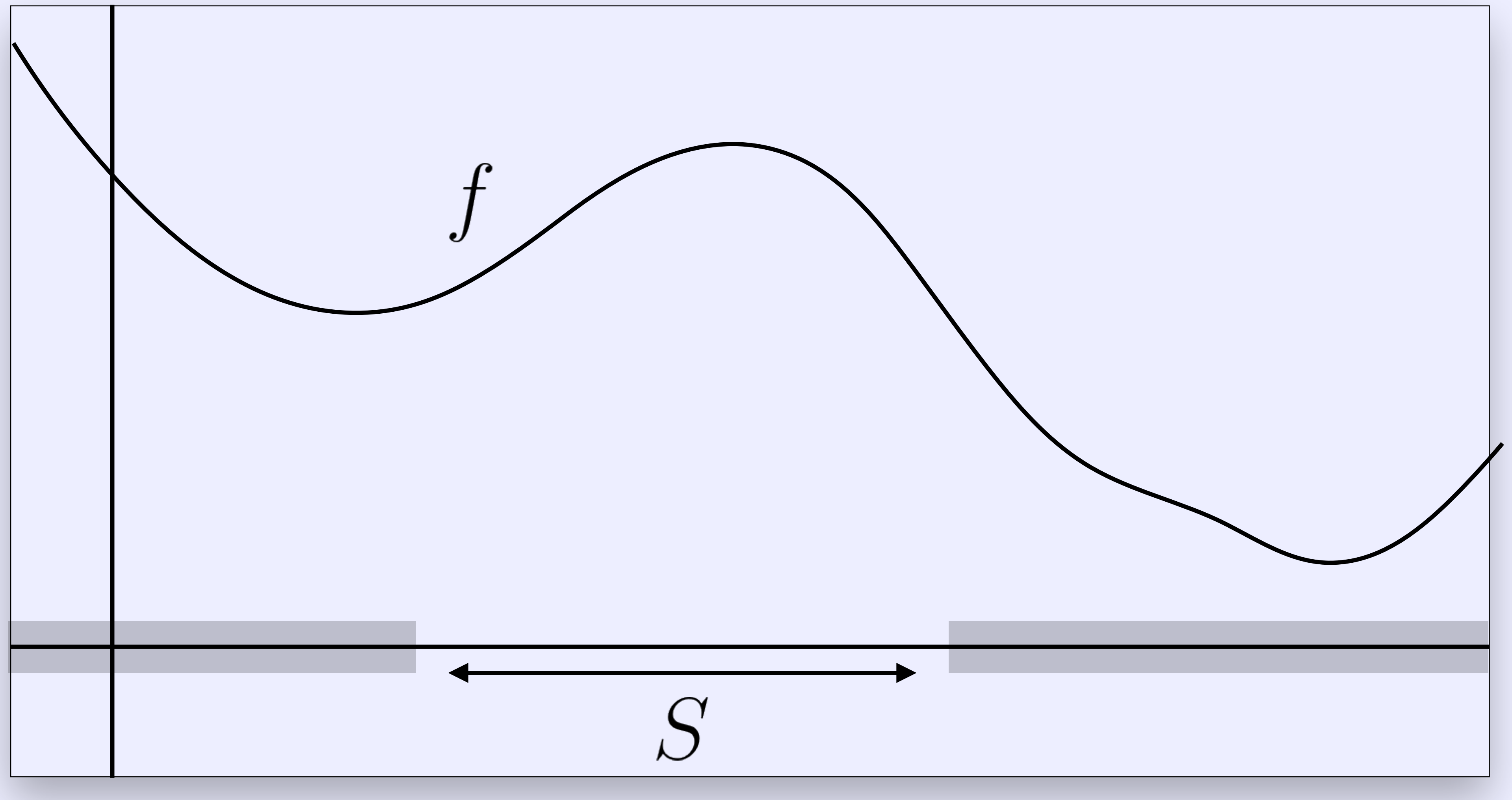
$$\min_{x\in\mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

Penalty function

$$P(x) \begin{cases} = 0 \text{ if } x \in S \\ > 0 \text{ if } x \notin S \end{cases}$$

Penalized Problem

$$\min_{x\in\mathbb{R}^d} f(x) + \mu P(x)$$



Recall Penalization on a Picture

- The penalization procedure

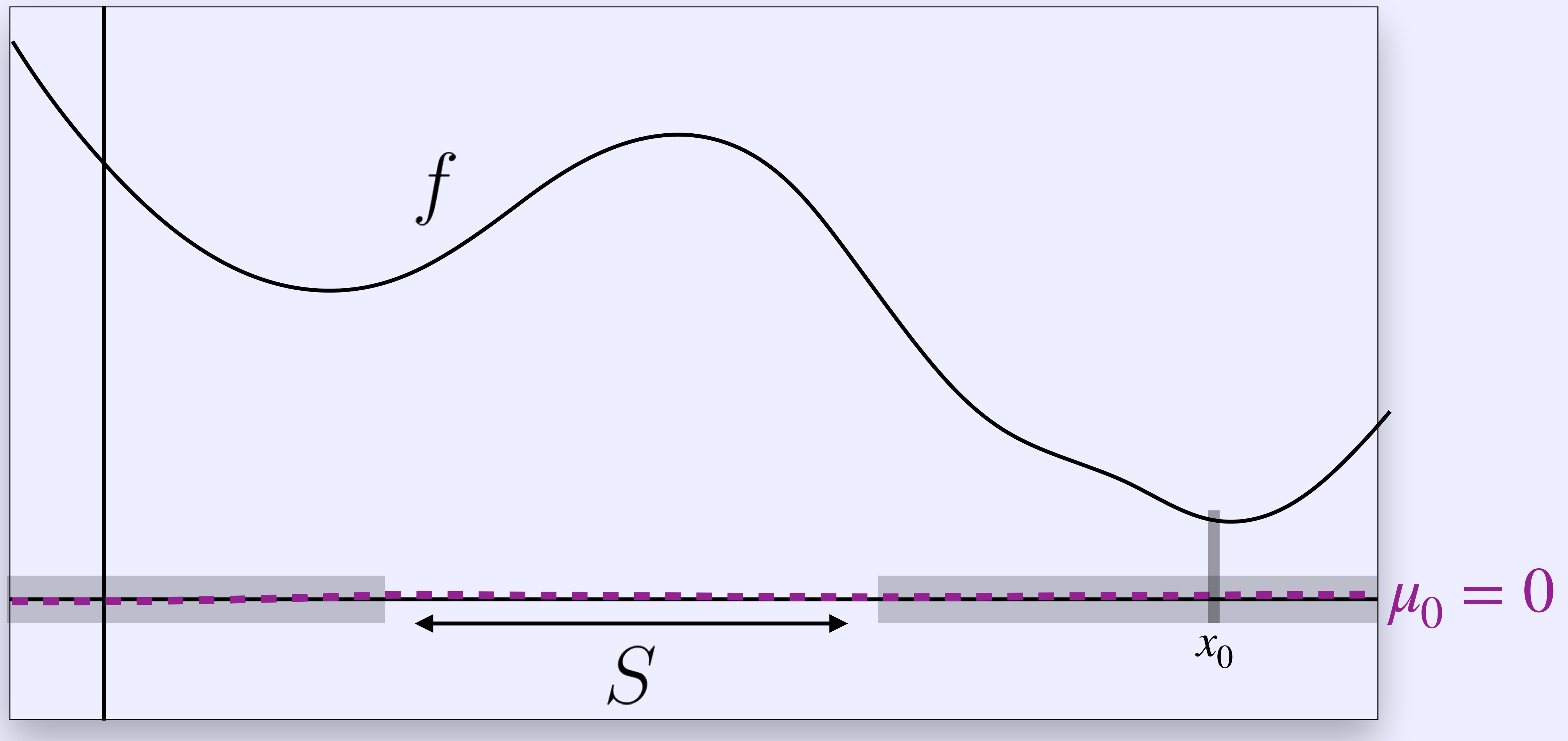
$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

Penalty function

$$P(x) \begin{cases} = 0 \text{ if } x \in S \\ > 0 \text{ if } x \notin S \end{cases}$$

Penalized Problem

$$\min_{x \in \mathbb{R}^d} f(x) + \mu P(x)$$



Recall Penalization on a Picture

- The penalization procedure

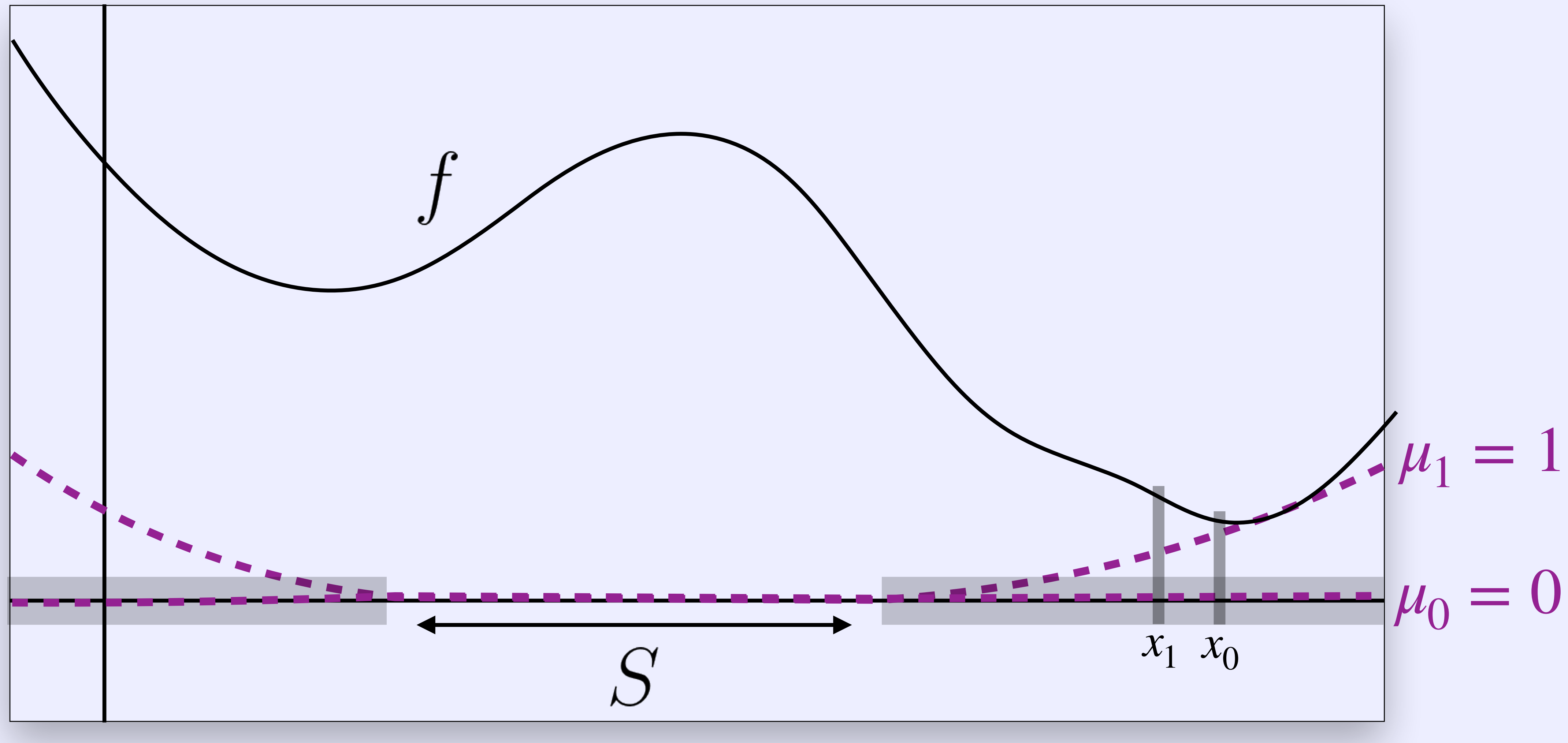
$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

Penalty function

$$P(x) \begin{cases} = 0 \text{ if } x \in S \\ > 0 \text{ if } x \notin S \end{cases}$$

Penalized Problem

$$\min_{x \in \mathbb{R}^d} f(x) + \mu P(x)$$



Recall Penalization on a Picture

- The penalization procedure

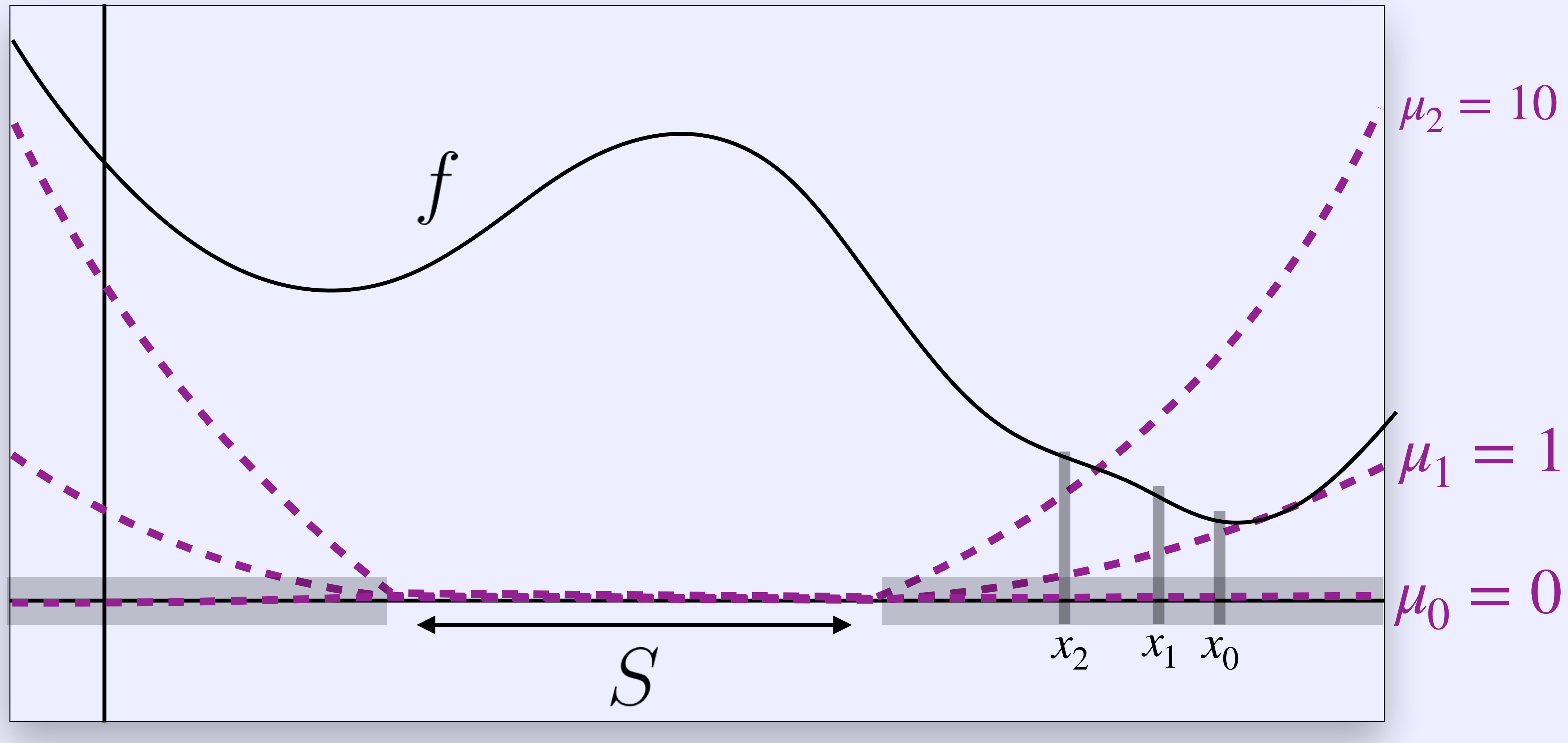
$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

Penalty function

$$P(x) \begin{cases} = 0 \text{ if } x \in S \\ > 0 \text{ if } x \notin S \end{cases}$$

Penalized Problem

$$\min_{x \in \mathbb{R}^d} f(x) + \mu P(x)$$



Recall Penalization on a Picture

- The penalization procedure

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

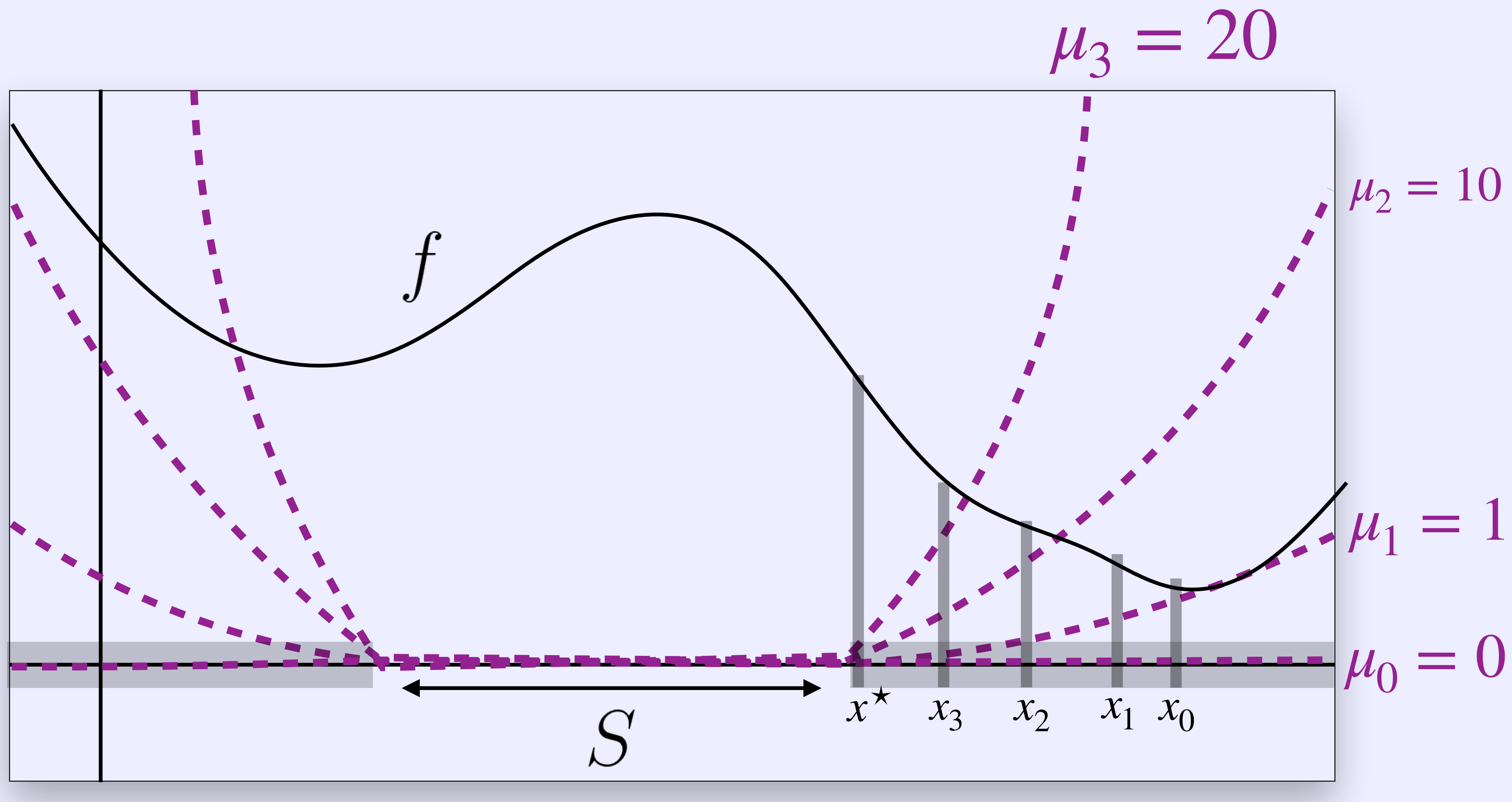
Penalty function

$$P(x) \begin{cases} = 0 \text{ if } x \in S \\ > 0 \text{ if } x \notin S \end{cases}$$

Penalized Problem

$$\min_{x \in \mathbb{R}^d} f(x) + \mu P(x)$$

Any cluster point of the sequence of solutions $(x_k)_{k \geq 0}$ is a solution of the constrained problem.



Recall Penalization on a Picture

The penalization procedure

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

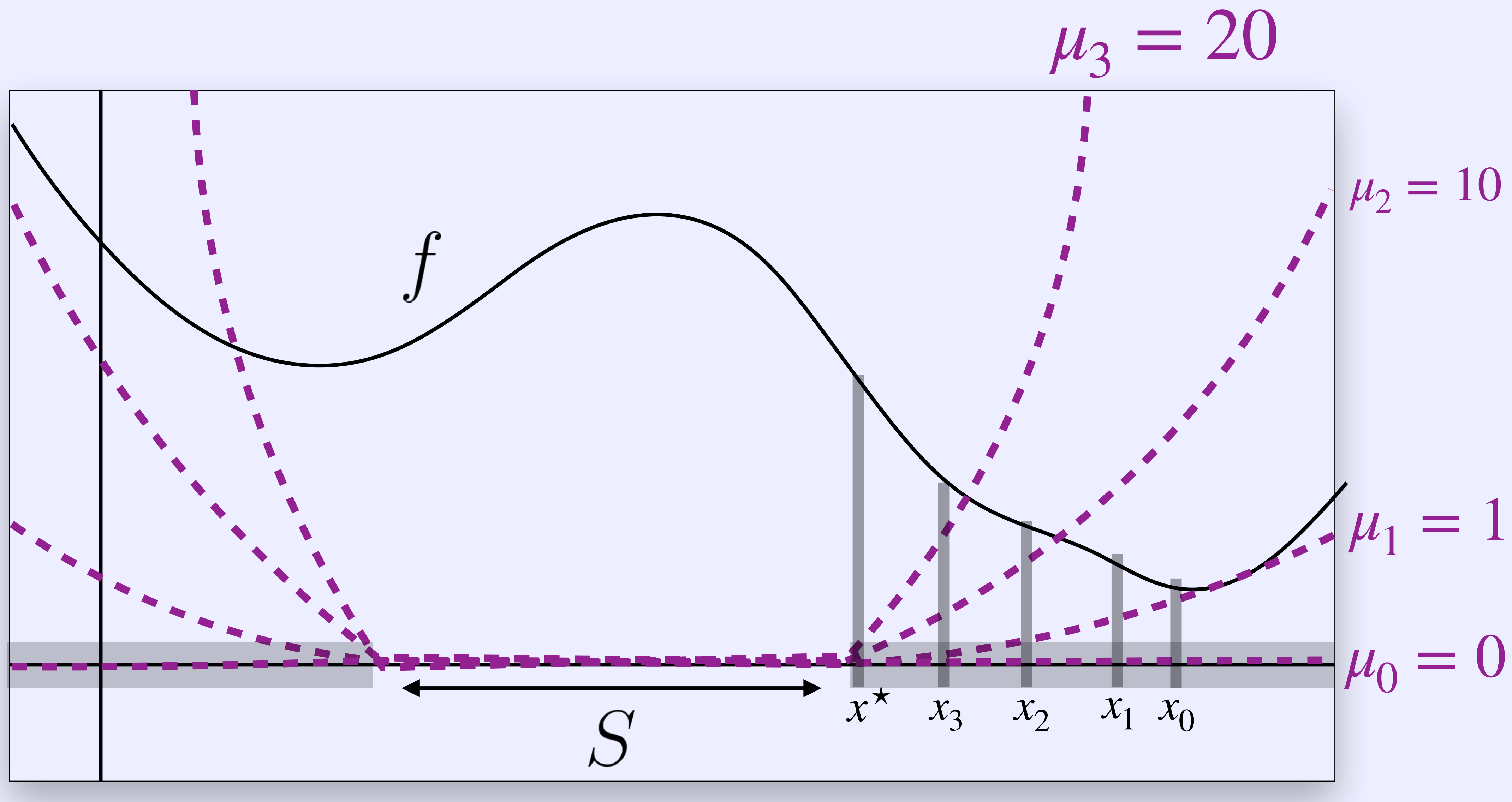
Penalty function

$$P(x) \begin{cases} = 0 \text{ if } x \in S \\ > 0 \text{ if } x \notin S \end{cases}$$

Penalized Problem

$$\min_{x \in \mathbb{R}^d} f(x) + \mu P(x)$$

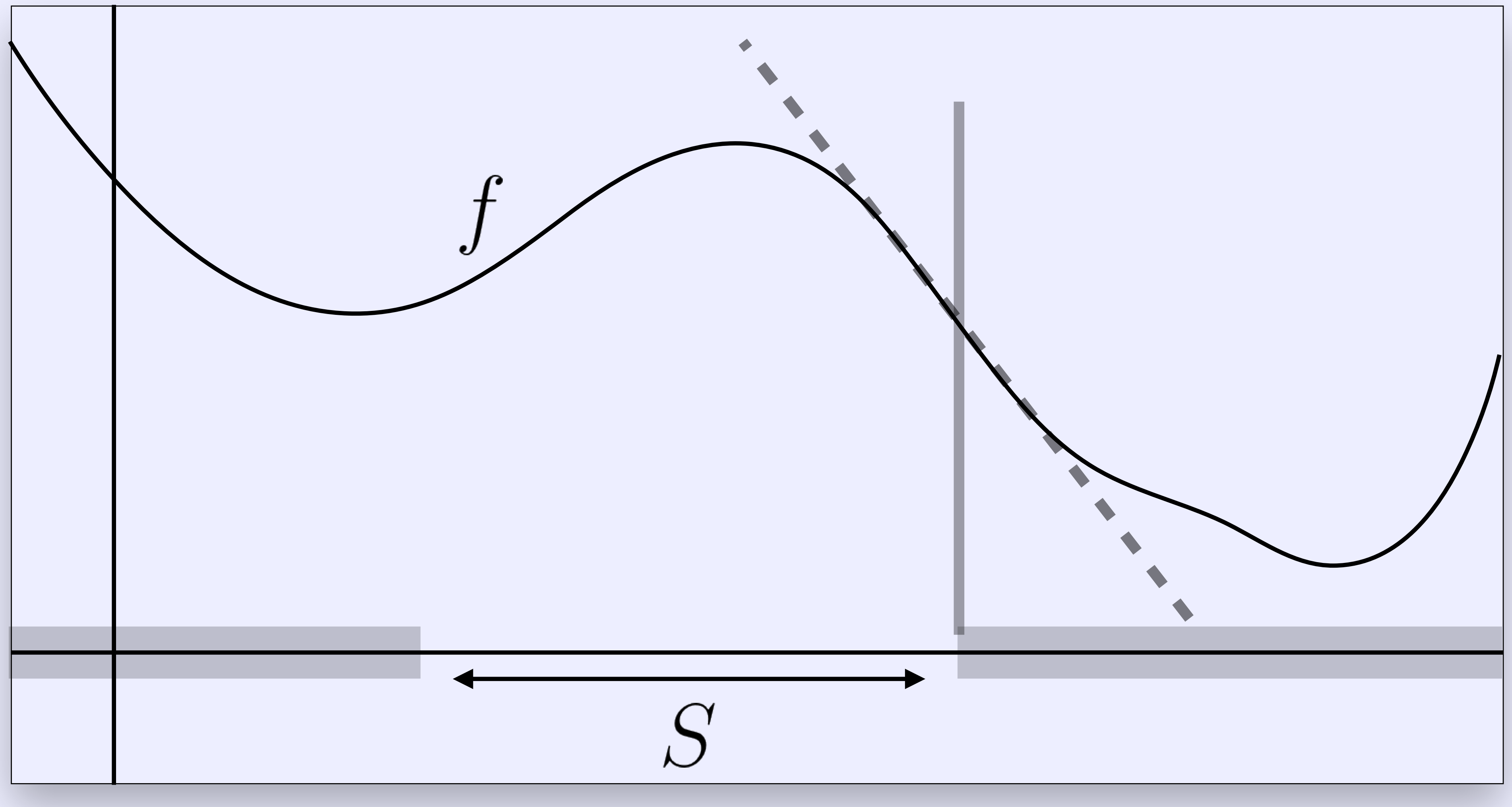
Any cluster point of the sequence of solutions $(x_k)_{k \geq 0}$ is a solution of the constrained problem.



Exact penalization

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

Assume f to be K-lipschitz on $\mathbb{R}^d$.



Recall Penalization on a Picture

The penalization procedure

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

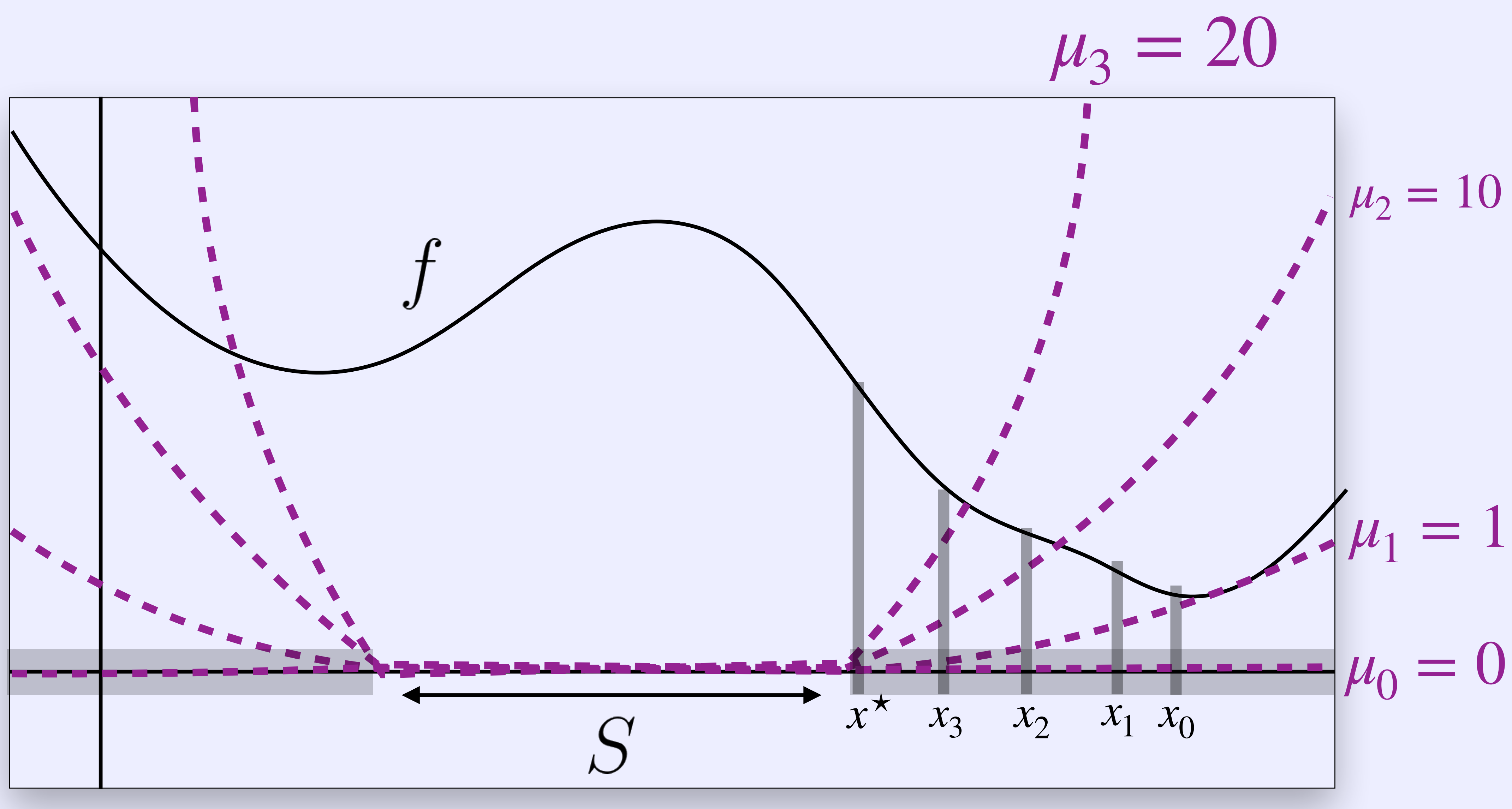
Penalty function

$$P(x) \begin{cases} = 0 \text{ if } x \in S \\ > 0 \text{ if } x \notin S \end{cases}$$

Penalized Problem

$$\min_{x \in \mathbb{R}^d} f(x) + \mu P(x)$$

Any cluster point of the sequence of solutions $(x_k)_{k \geq 0}$ is a solution of the constrained problem.

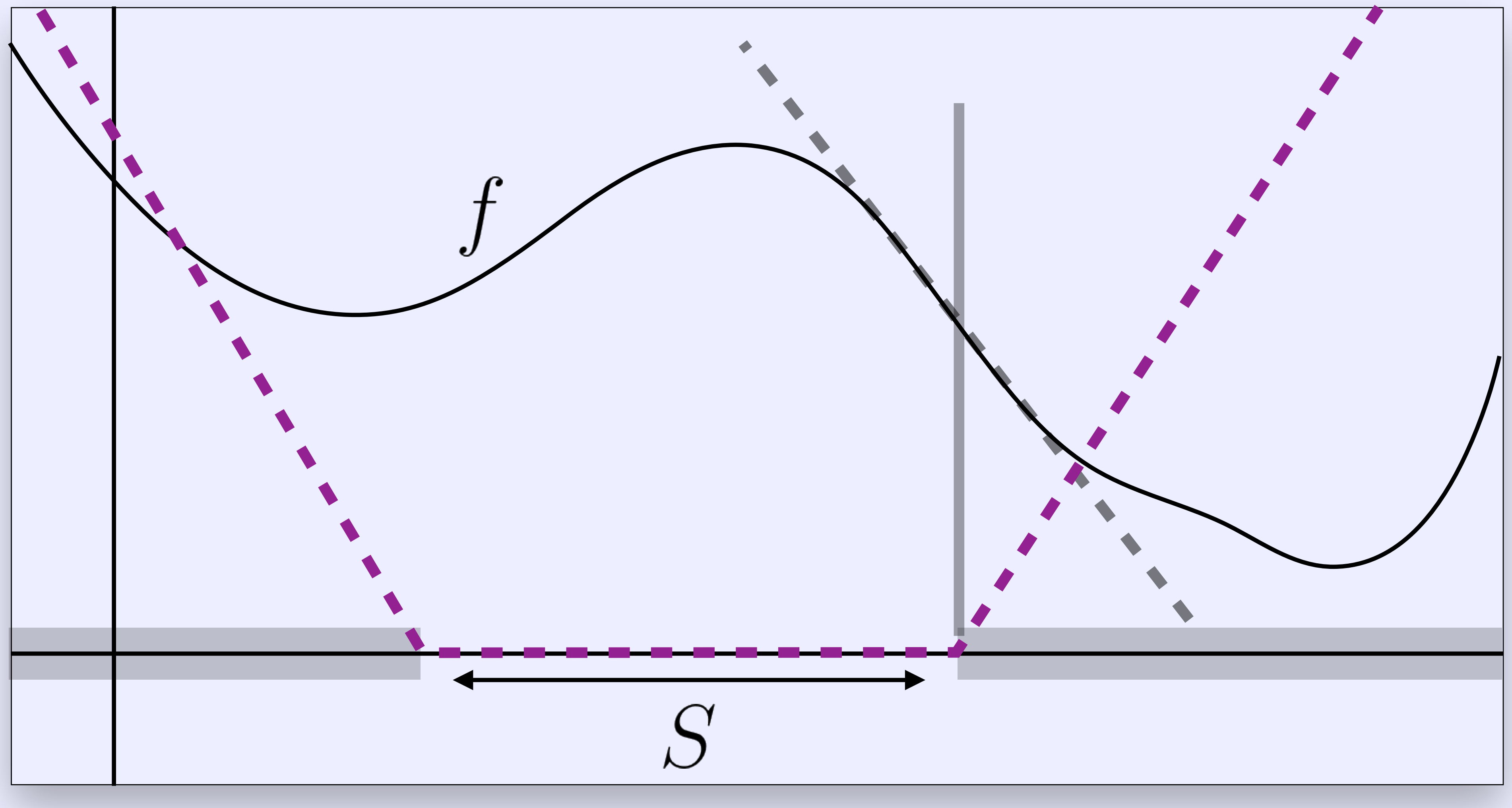


Exact penalization

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

Assume f to be K-lipschitz on $\mathbb{R}^d$.

Then, for any $K' > K$, this problem has the same set of minimisers as $\min_{x \in \mathbb{R}^d} f(x) + K' d_S(x)$



Recall Penalization on a Picture

- The penalization procedure

$$\min_{x\in\mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

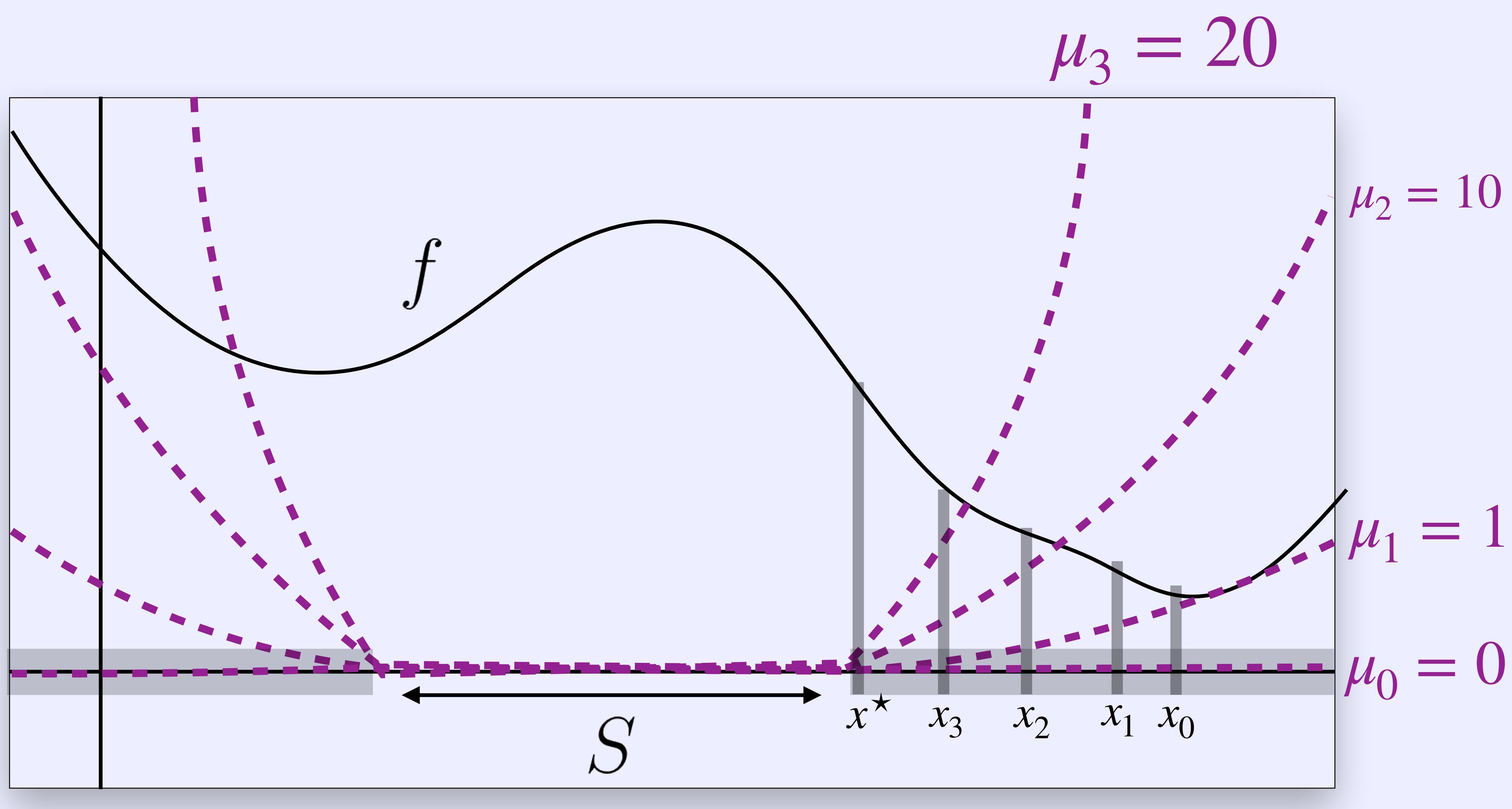
Penalty function

$$P(x) \begin{cases} = 0 \text{ if } x \in S \\ > 0 \text{ if } x \notin S \end{cases}$$

Penalized Problem

$$\min_{x\in\mathbb{R}^d} f(x) + \mu P(x)$$

Any cluster point of the sequence of solutions $(x_k)_{k\geq 0}$ is a solution of the constrained problem.

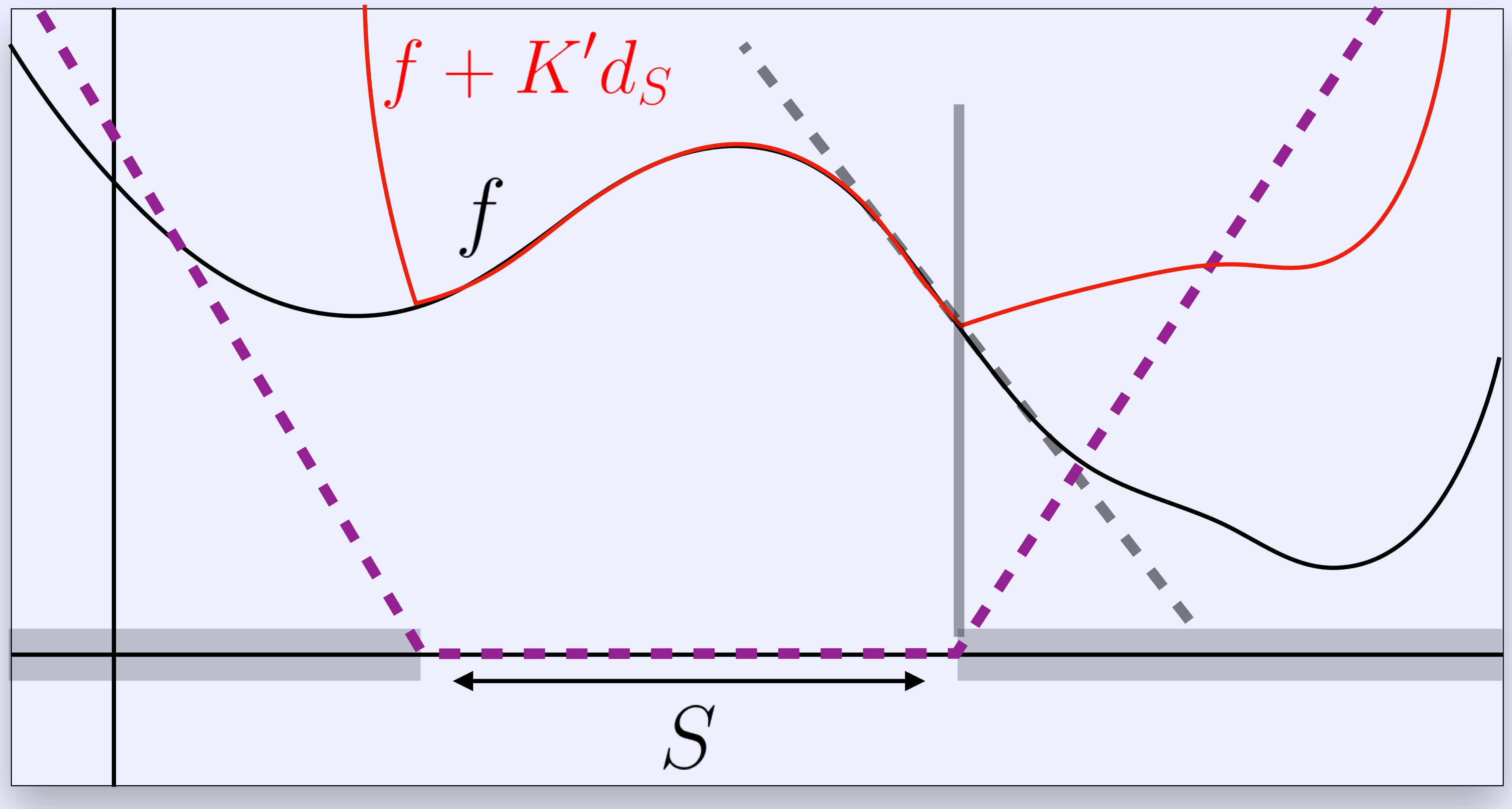


- Exact penalization

$$\min_{x\in\mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

Assume f to be K-lipschitz on $\mathbb{R}^d$.

Then, for any $K' > K$, this problem has the same set of minimisers as $\min_{x\in\mathbb{R}^d} f(x) + K' d_S(x)$



Recall Penalization on a Picture

- The penalization procedure

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

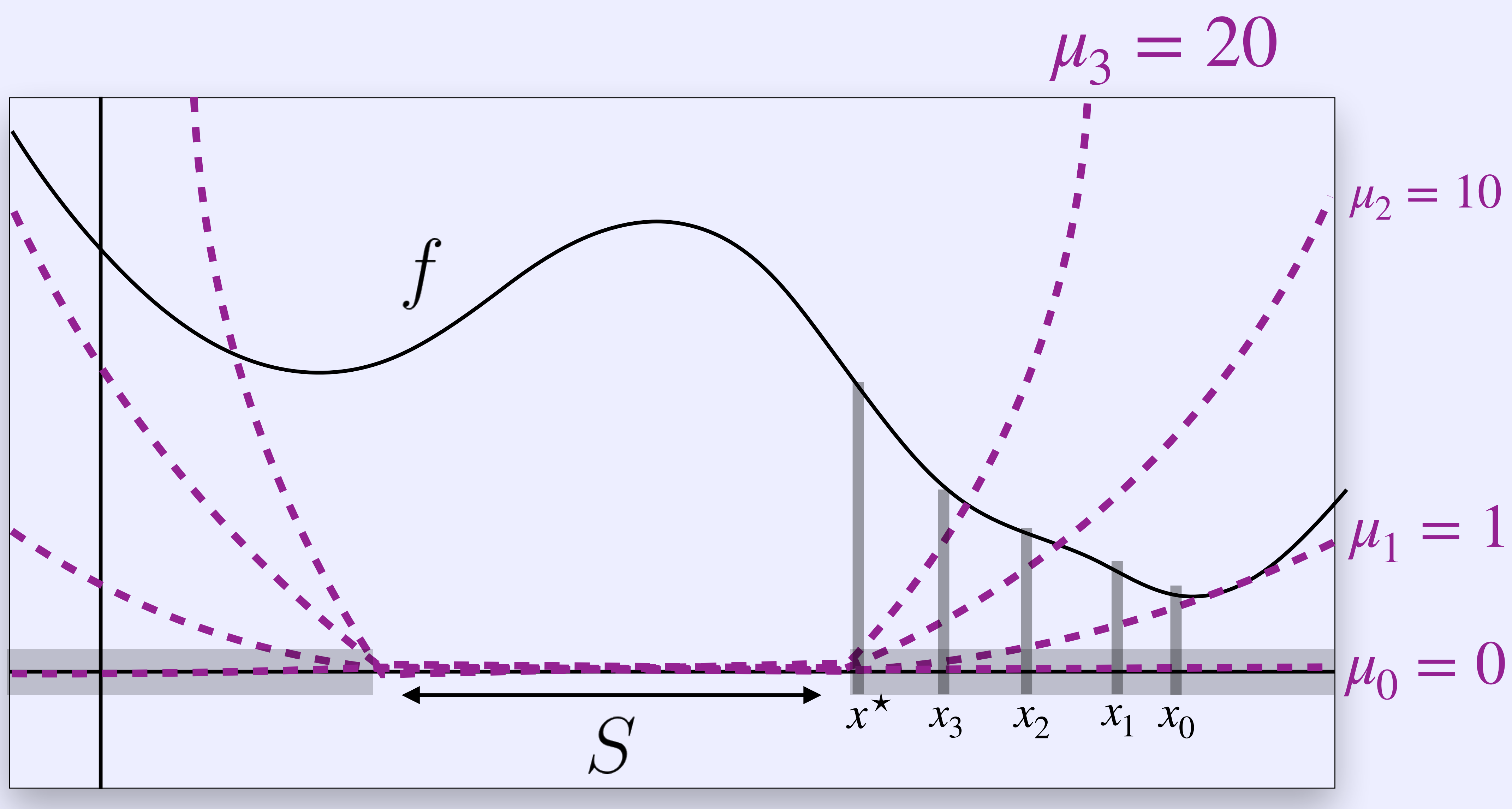
Penalty function

$$P(x) \begin{cases} = 0 \text{ if } x \in S \\ > 0 \text{ if } x \notin S \end{cases}$$

Penalized Problem

$$\min_{x \in \mathbb{R}^d} f(x) + \mu P(x)$$

Any cluster point of the sequence of solutions $(x_k)_{k \geq 0}$ is a solution of the constrained problem.



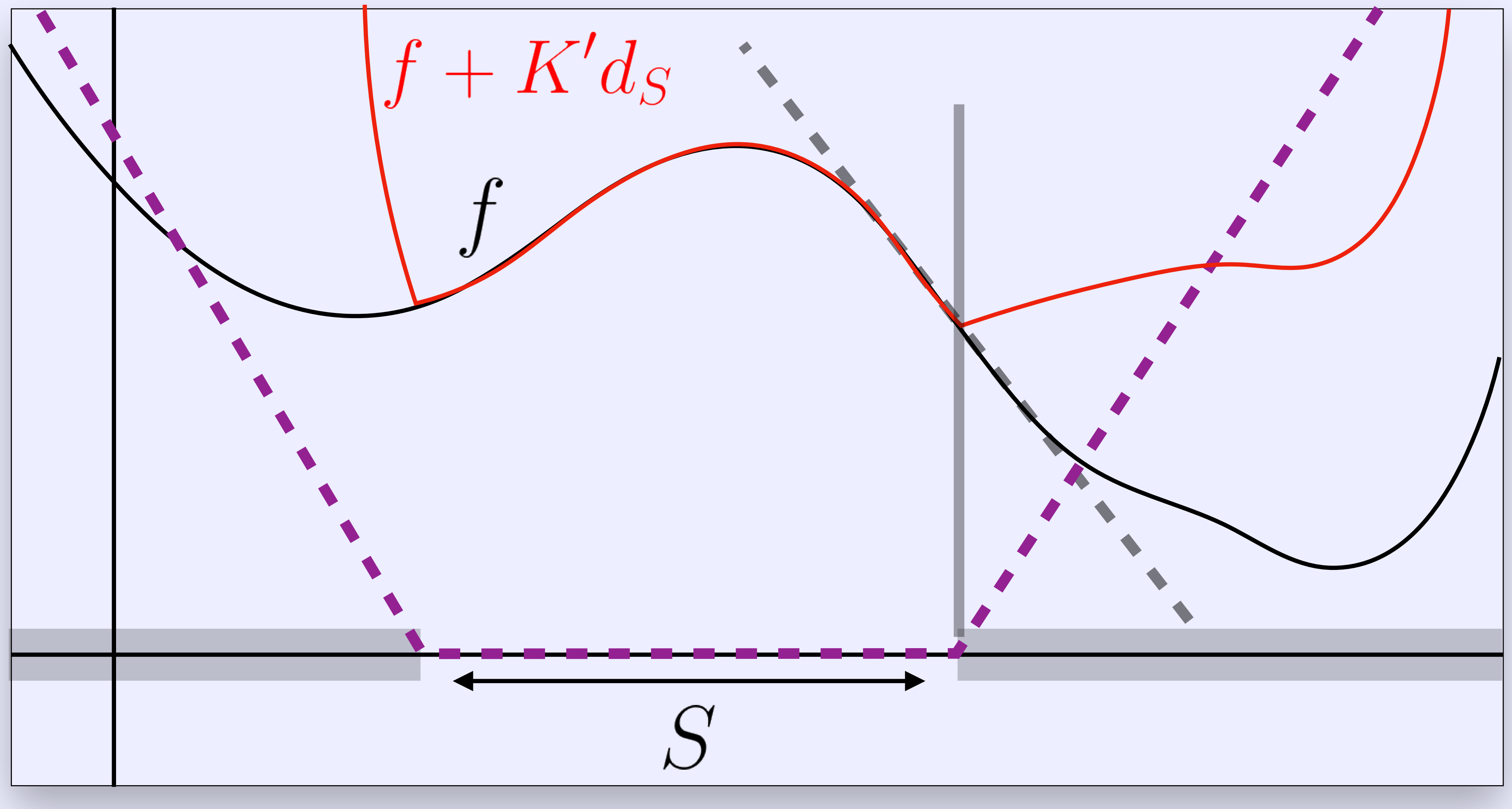
- Exact penalization

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } x \in S$$

Assume f to be K-lipschitz on $\mathbb{R}^d$.

Then, for any $K' > K$, this problem has the same set of minimisers as $\min_{x \in \mathbb{R}^d} f(x) + K' d_S(x)$

Uniform Parametric Error Bound	$h(x) \geq \delta\, d_S(x)\ \forall x \in \mathbb{R}^d$ $h(x) = 0 \Leftrightarrow x \in S$



We propose a Double Penalization Procedure

- First penalization

$$(\mathcal{P})$$

$$\begin{aligned} \min_{x \in \mathbb{R}^d, \eta \in \mathbb{R}} \quad & f(x) \\ \text{s.t.} \quad & \eta \leq 0 \\ & \eta \in \underset{s \in \mathbb{R}}{\operatorname{argmin}}\ s + \frac{1}{1-p} \mathbb{E}\left[\max(g(x,\xi) - s, 0)\right] \end{aligned}$$

We propose a Double Penalization Procedure

- First penalization

$(\mathcal{P})$

$$\min_{x \in \mathbb{R}^d, \eta \in \mathbb{R}} \quad f(x)$$
$$\text{s.t.} \quad \eta \leq 0$$
$$\eta \in \underset{s \in \mathbb{R}}{\operatorname{argmin}} \; s + \frac{1}{1-p} \mathbb{E}\left[\max(g(x,\xi) - s, 0)\right]$$

$\Rightarrow$

$(\mathcal{P}_\mu)$

$$\min_{x \in \mathbb{R}^d, \eta \in \mathbb{R}} \quad f(x) + \mu \max(\eta, 0)$$
$$\text{s.t.} \quad \eta \in \underset{s \in \mathbb{R}}{\operatorname{argmin}} \; s + \frac{1}{1-p} \mathbb{E}\left[\max(g(x,\xi) - s, 0)\right]$$

We propose a Double Penalization Procedure

- First penalization

$(\mathcal{P})$

$$\begin{aligned} \min_{x \in \mathbb{R}^d, \eta \in \mathbb{R}} \quad & f(x) \\ \text{s.t.} \quad & \eta \leq 0 \\ & \eta \in \underset{s \in \mathbb{R}}{\operatorname{argmin}} \; s + \frac{1}{1-p} \mathbb{E}\left[\max(g(x,\xi) - s, 0)\right] \end{aligned}$$

$\Rightarrow$

$(\mathcal{P}_\mu)$

$$\begin{aligned} \min_{x \in \mathbb{R}^d, \eta \in \mathbb{R}} \quad & f(x) + \mu \max(\eta, 0) \\ \text{s.t.} \quad & \eta \in \underset{s \in \mathbb{R}}{\operatorname{argmin}} \; s + \frac{1}{1-p} \mathbb{E}\left[\max(g(x,\xi) - s, 0)\right] \end{aligned}$$

- In practice, the constant μ is a hyperparameter to tune.

We propose a Double Penalization Procedure

- First penalization

$(\mathcal{P})$

$$\begin{aligned} \min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \quad & f(x) \\ \text{s.t.} \quad & \eta \leq 0 \\ & \eta \in \underset{s\,\in\,\mathbb{R}}{\operatorname{argmin}}\ s + \frac{1}{1-p}\mathbb{E}\left[\max(g(x,\xi) - s, 0)\right] \end{aligned}$$

$\Rightarrow$

$(\mathcal{P}_\mu)$

$$\begin{aligned} \min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \quad & f(x) + \mu \max(\eta, 0) \\ \text{s.t.} \quad & \eta \in \underset{s\,\in\,\mathbb{R}}{\operatorname{argmin}}\ \underbrace{s + \frac{1}{1-p}\mathbb{E}\left[\max(g(x,\xi) - s, 0)\right]} \end{aligned}$$

$$= G(x, s)$$

$$\geq \bar{Q}_p(g(x,\xi))$$

- In practice, the constant μ is a hyperparameter to tune.

- Using Rockafellar property

$$\begin{aligned} \min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \quad & f(x) + \mu \max(\eta, 0) \\ \text{s.t.} \quad & G(x,\eta) - \bar{Q}_p(g(x,\xi)) \leq 0 \end{aligned}$$

We propose a Double Penalization Procedure

- Second penalization

$(\mathcal{P}_\mu)$

$$\begin{aligned} \min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \quad & f(x) + \mu \max(\eta, 0) \\ \text{s.t.} \quad & G(x,\eta) - \bar{Q}_p(g(x,\xi)) \leq 0 \end{aligned}$$

$\Rightarrow$

$(\mathcal{P}_{\lambda,\mu})$

$$\min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} f(x) + \mu \max(\eta, 0) + \lambda \left(G(x,\eta) - \bar{Q}_p(g(x,\xi))\right)$$

We propose a Double Penalization Procedure

- Second penalization

$(\mathcal{P}_\mu)$

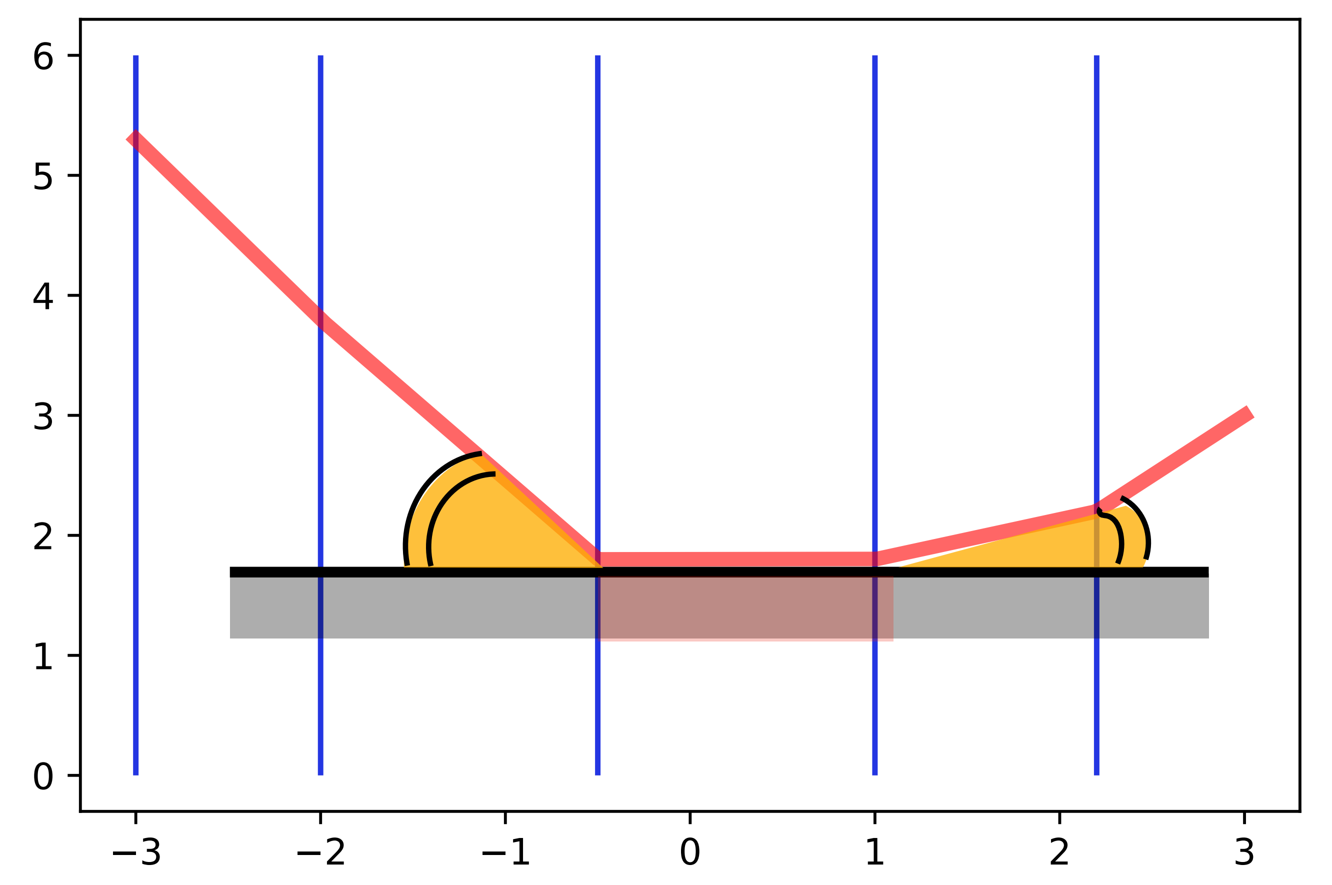
$$\begin{aligned} \min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \quad & f(x) + \mu \max(\eta, 0) \\ \text{s.t.} \quad & G(x,\eta) - \bar{Q}_p(g(x,\xi)) \leq 0 \end{aligned}$$

$(\mathcal{P}_{\lambda,\mu})$

$$\min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \quad f(x) + \mu \max(\eta, 0) + \lambda \left(G(x,\eta) - \bar{Q}_p(g(x,\xi))\right)$$

- This penalization is exact.

$$\eta \mapsto G(x,\eta) = \eta + \frac{1}{1-p}\mathbb{E}[\max(g(x,\xi) - \eta, 0)]$$



Theorem Let $\mu > 0$ be given and fixed and assume that the solution set of problem $(\mathcal{P}_\mu)$ is not empty. Then for any $\lambda > \lambda_\mu = \frac{\mu}{\delta}$ where:

$$\delta = \begin{cases} \frac{1}{n(1-p)} & \text{if } p \in \mathcal{I} \\ \frac{d_{\mathcal{I}}(p)}{1-p} & \text{otherwise.} \end{cases}$$

the solution set of $(\mathcal{P}_\mu)$ coincides with the solution set of $(\mathcal{P}_{\lambda,\mu})$

Solving of the doubly-penalized problem

- Second penalization

$$(\mathcal{P}_\mu) \quad \min_{x\in\mathbb{R}^d, \eta\in\mathbb{R}} \; f(x) + \mu \max(\eta, 0) \quad \text{s.t.} \quad G(x,\eta) - \bar{Q}_p(g(x,\xi)) \leq 0$$

$$(\mathcal{P}_{\lambda,\mu}) \quad \min_{x\in\mathbb{R}^d, \eta\in\mathbb{R}} \; \underbrace{f(x) + \mu \max(\eta, 0)}_{\text{Convex}} + \lambda \underbrace{\left(G(x,\eta) - \bar{Q}_p(g(x,\xi))\right)}_{\text{Non convex}}$$

Convex

Non convex

but…

Solving of the doubly-penalized problem

- Second penalization

$(\mathcal{P}_\mu)$

$$\min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \quad f(x) + \mu\max(\eta,0)$$
$$\text{s.t.} \quad G(x,\eta) - \bar{Q}_p(g(x,\xi)) \leq 0$$

$(\mathcal{P}_{\lambda,\mu})$

$$\min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \underbrace{f(x) + \mu\max(\eta,0)}_{\text{Convex}} + \lambda\underbrace{\left(G(x,\eta) - \bar{Q}_p(g(x,\xi))\right)}_{\text{Non convex}}$$

Convex

Non convex but...

$$G(x,\eta) = \eta + \frac{1}{1-p}\mathbb{E}[\max(g(x,\xi)-\eta,0)]$$

Solving of the doubly-penalized problem

- Second penalization

$(\mathcal{P}_\mu)$

$$\min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \; f(x) + \mu \max(\eta, 0) \quad \text{s.t.} \quad G(x,\eta) - \bar{Q}_p(g(x,\xi)) \leq 0$$

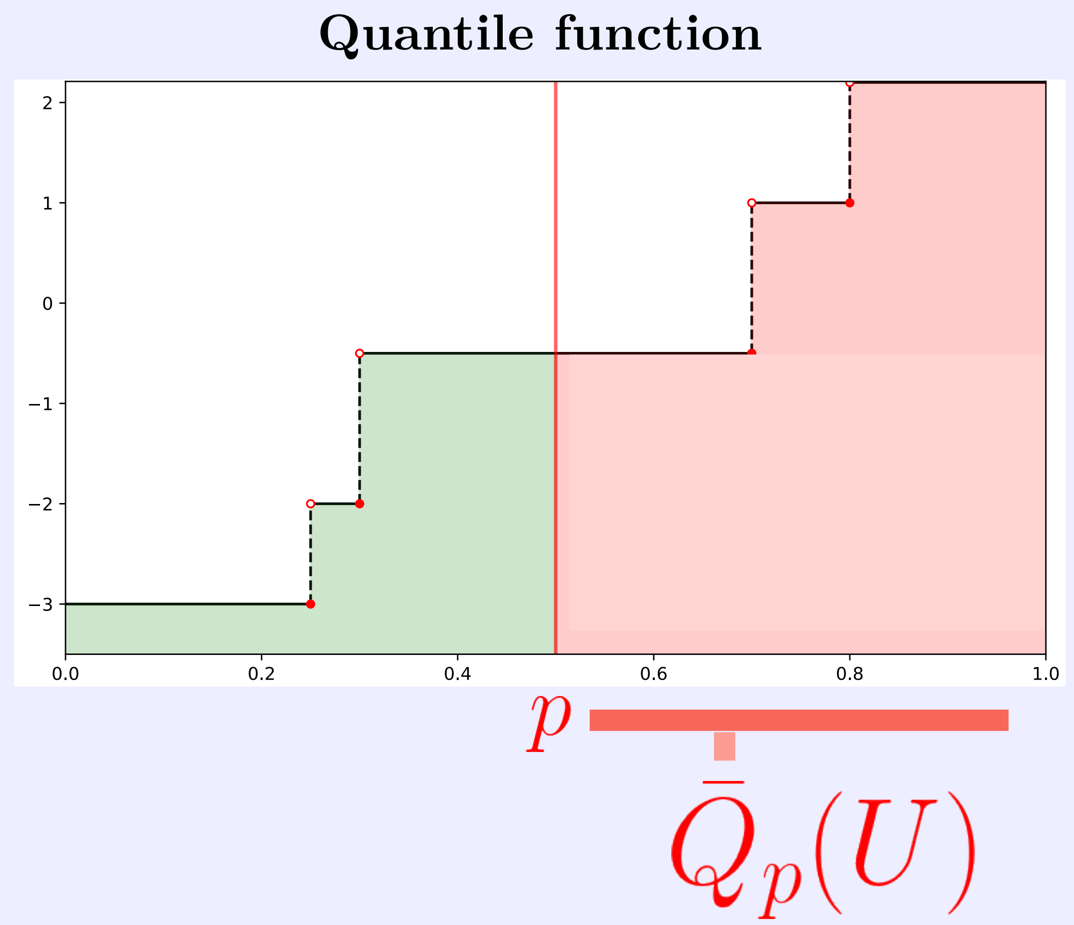
$(\mathcal{P}_{\lambda,\mu})$

$$\min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \; \underbrace{f(x) + \mu \max(\eta, 0) + \lambda \left(G(x,\eta)\right.}_{\text{Convex}} - \underbrace{\left.\bar{Q}_p(g(x,\xi))\right)}_{\text{Convex}}$$

$$G(x,\eta) = \eta + \frac{1}{1-p}\mathbb{E}[\max(g(x,\xi) - \eta, 0)]$$

- $\bar{Q}_p$ is convex ! $(\mathcal{P}_{\lambda,\mu})$ is a Difference of Convex problem.

$$\bar{Q}_p(U) = \frac{1}{1-p}\int_{p'=p}^{1} Q_{p'}(U)dp'$$



Solving of the doubly-penalized problem

- Second penalization

$(\mathcal{P}_\mu)$

$$\min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \quad f(x) + \mu\max(\eta,0)$$
$$\text{s.t.} \quad G(x,\eta) - \bar{Q}_p(g(x,\xi)) \leq 0$$

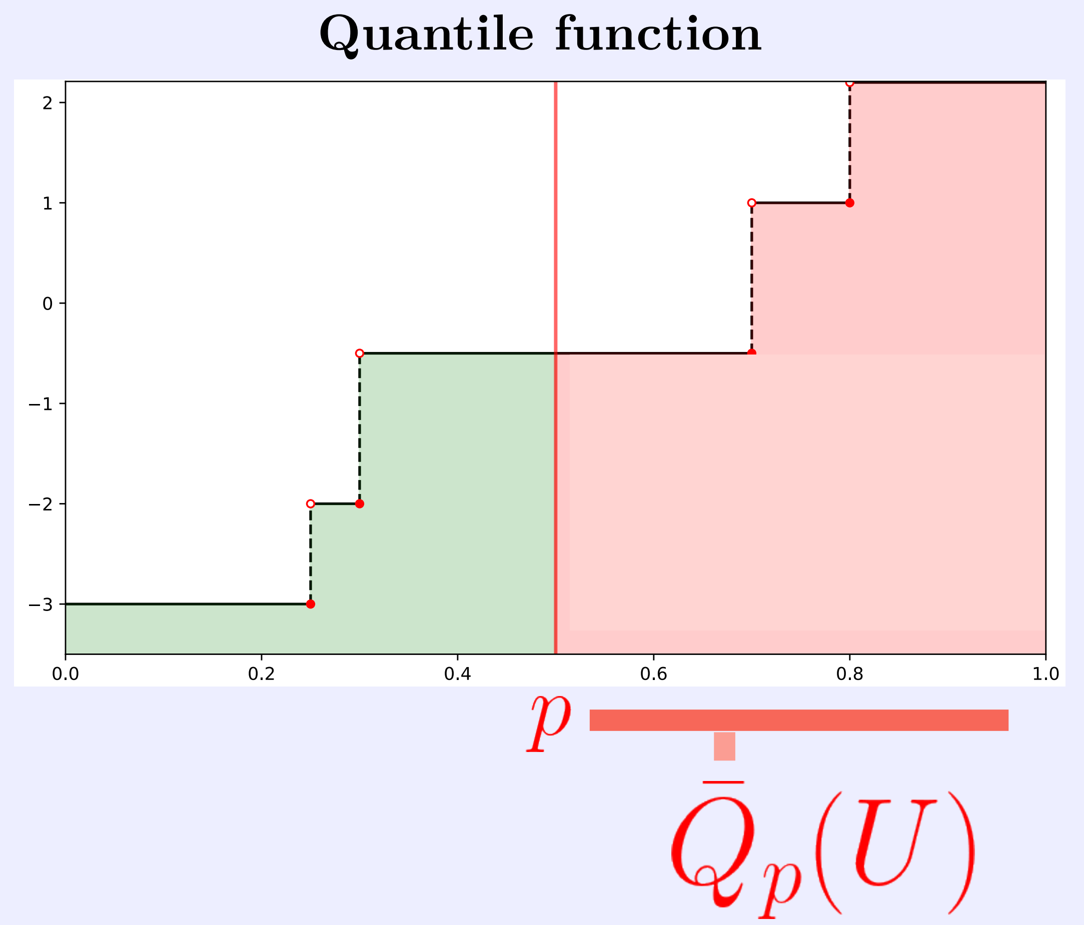
$(\mathcal{P}_{\lambda,\mu})$

$$\min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \quad \underbrace{f(x) + \mu\max(\eta,0) + \lambda\left(G(x,\eta)\right.}_{\text{Convex}} - \underbrace{\left.\bar{Q}_p(g(x,\xi))\right)}_{\text{Convex}}$$

$$G(x,\eta) = \eta + \frac{1}{1-p}\mathbb{E}[\max(g(x,\xi)-\eta,0)]$$

- $\bar{Q}_p$ is convex ! $(\mathcal{P}_{\lambda,\mu})$ is a Difference of Convex problem.

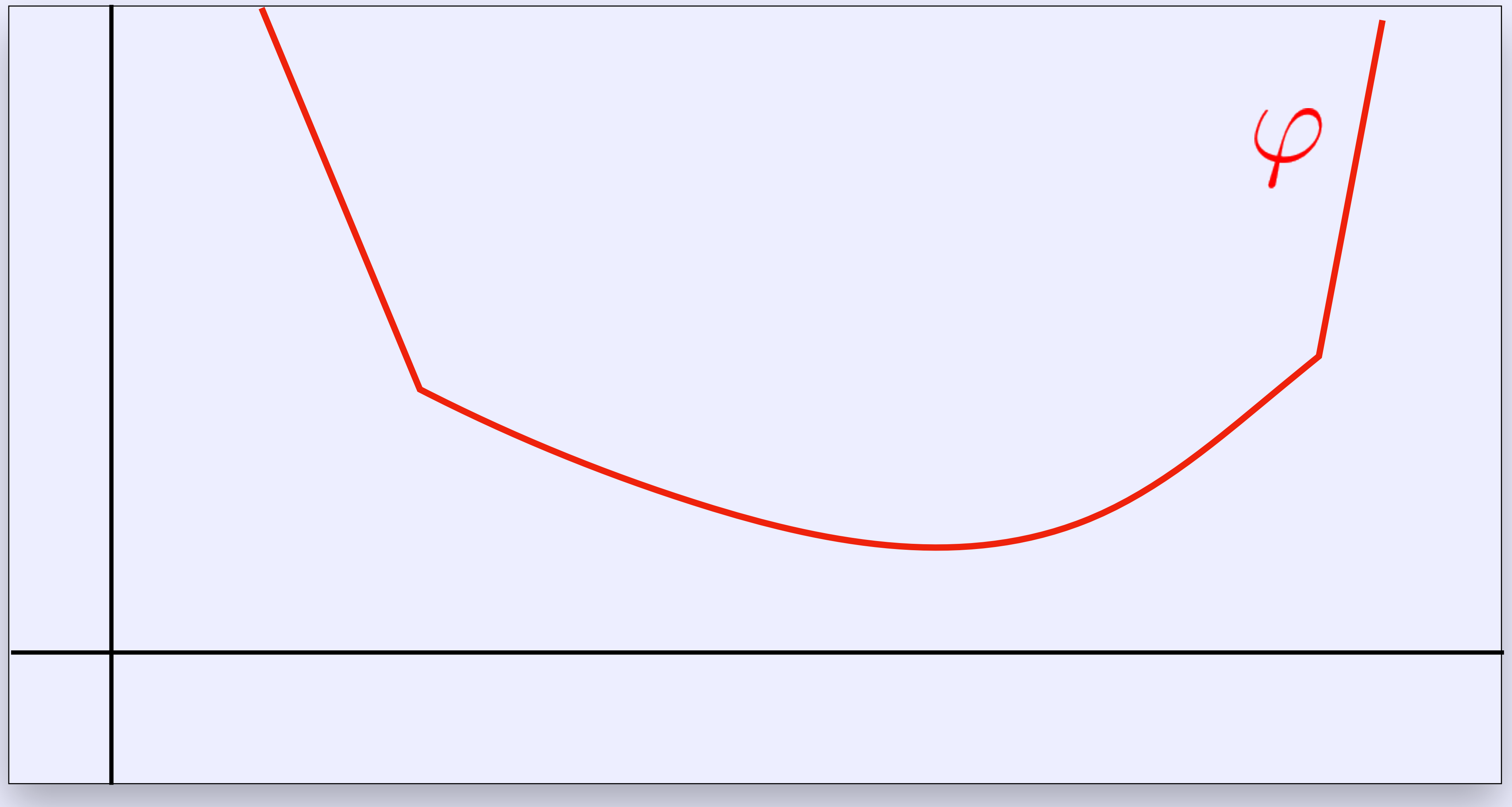
$$\bar{Q}_p(U) = \frac{1}{1-p}\int_{p'=p}^{1} Q_{p'}(U)dp' = \sup_{\substack{q\,\in\,\mathbb{R}^n \\ \sum_{i=1}^n q_i = 1 \\ 0\leq q_i \leq \frac{1}{n(1-p)}}} \sum_{i=1}^n q_i U_i$$



3 TACO
A Python Toolbox for Chance Constrained Problems

Recall: Solving DC programs by Bundle

- Bundle methods in a nutshell
    - Minimization of non-smooth problems



Recall: Solving DC programs by Bundle

- Bundle methods in a nutshell
  - Minimization of non-smooth problems
  - Maintains:
    - the **Bundle Information.**
    - the **Polyhedral Approximation.**

$$x \mapsto \varphi(x_i) + g_\varphi^\top (x - x_i)$$

$$\check{\varphi}(x) = \max_{i \in \text{Bundle}} \varphi(x_i) + g_\varphi^\top (x - x_i)$$

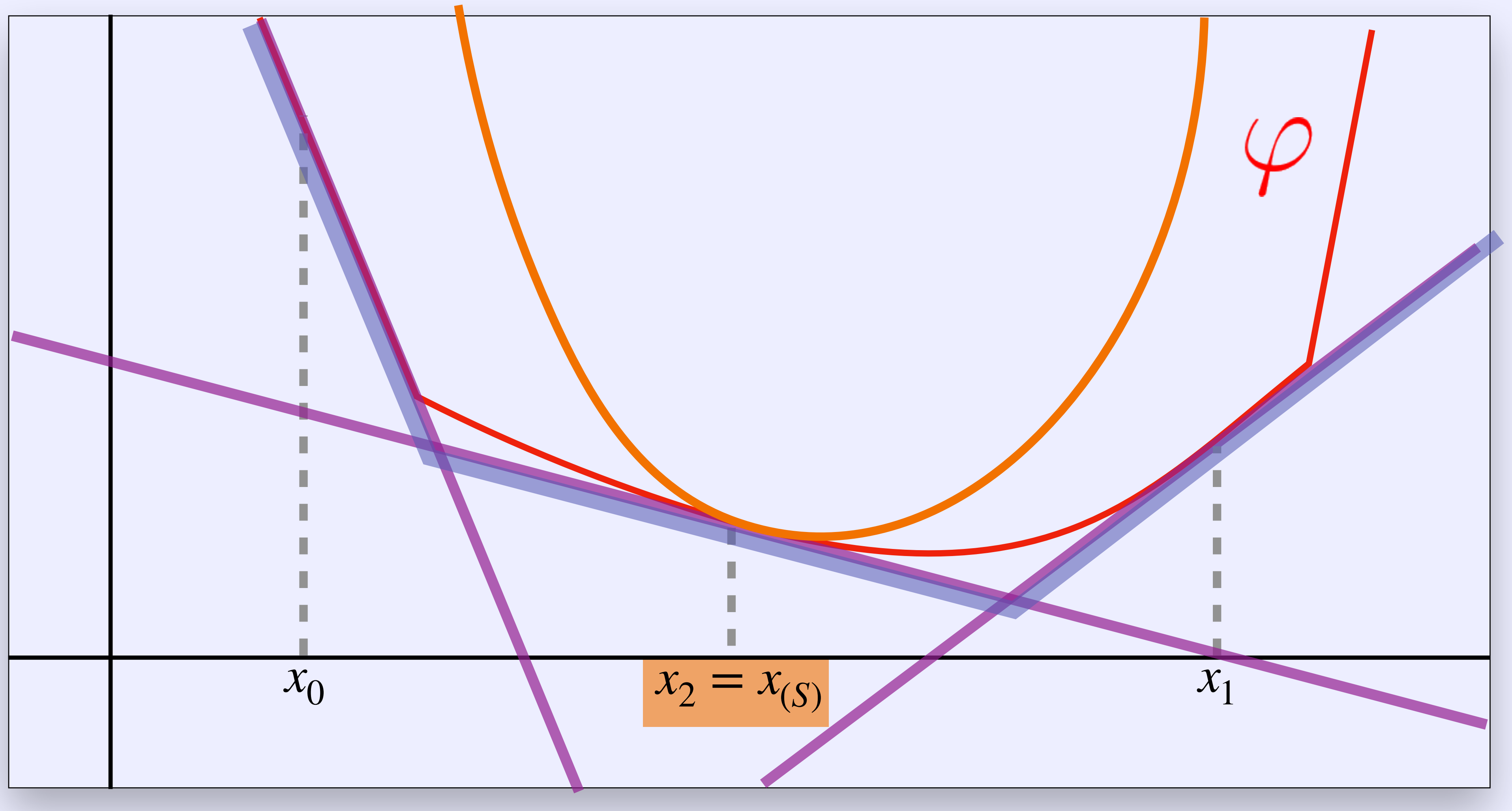
Recall: Solving DC programs by Bundle

- Bundle methods in a nutshell
  - Minimization of non-smooth problems
  - Maintains:
    - the **Bundle Information.**
    - the **Polyhedral Approximation.**
    - the **Stability Center** in the bundle.

$$x \mapsto \varphi(x_i) + g_\varphi^\top (x - x_i)$$

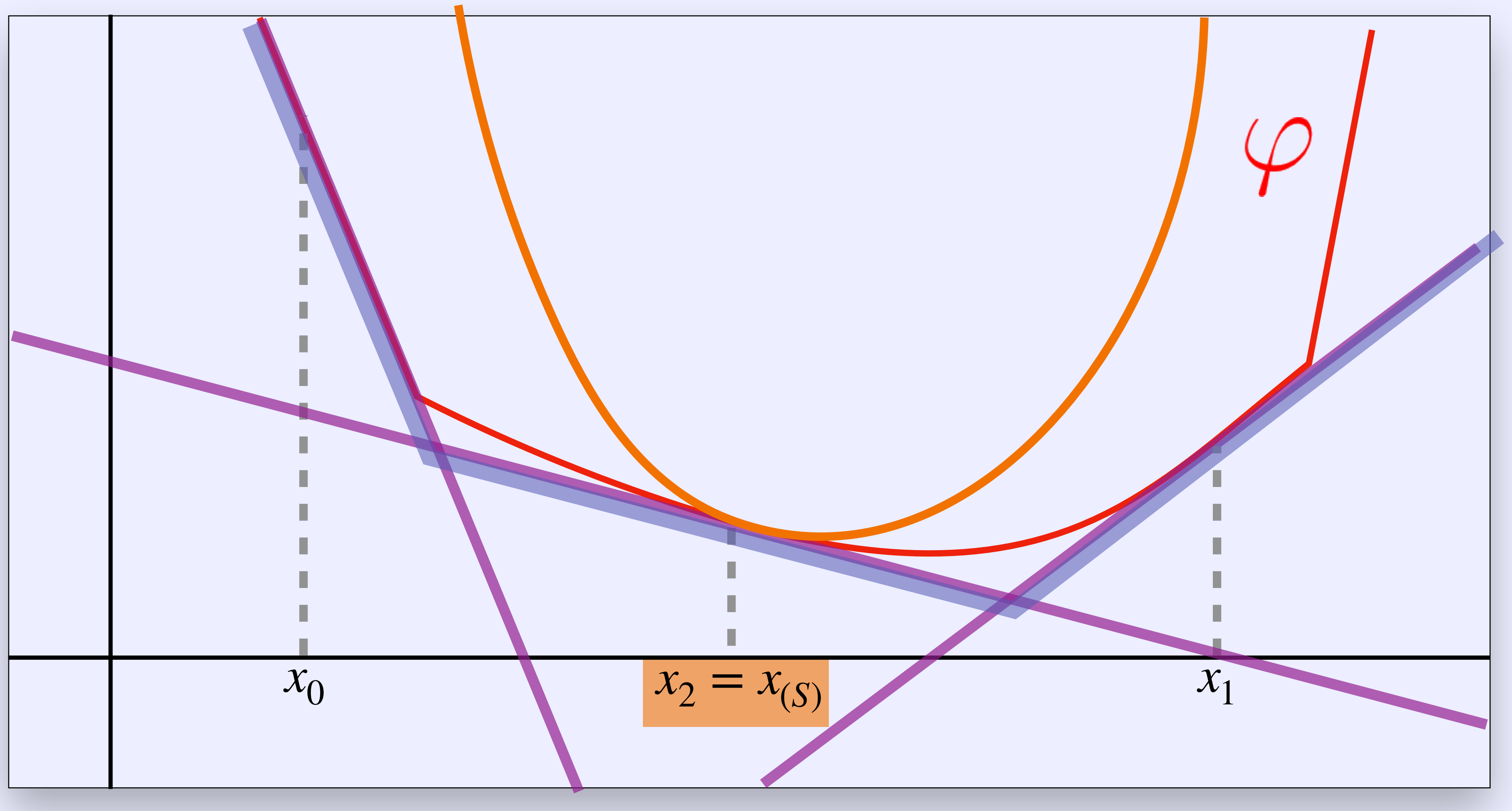
$$\check{\varphi}(x) = \max_{i \in \text{Bundle}} \varphi(x_i) + g_\varphi^\top (x - x_i)$$

$$\min_{x \in \mathbb{R}^d} \check{\varphi}(x) + \alpha \|x - x_{(S)}\|_2^2$$



Recall: Solving DC programs by Bundle

- Bundle methods in a nutshell
  - Minimization of non-smooth problems
  - Maintains:
    - the **Bundle Information.**
    - the **Polyhedral Approximation.**
    - the **Stability Center** in the bundle.



- State-of-the-art methods for DC problems

  [De Oliveira 19']

  - Function of the form $\varphi(x) = \varphi_1(x) - \varphi_2(x)$
  - We now solve at each iteration :

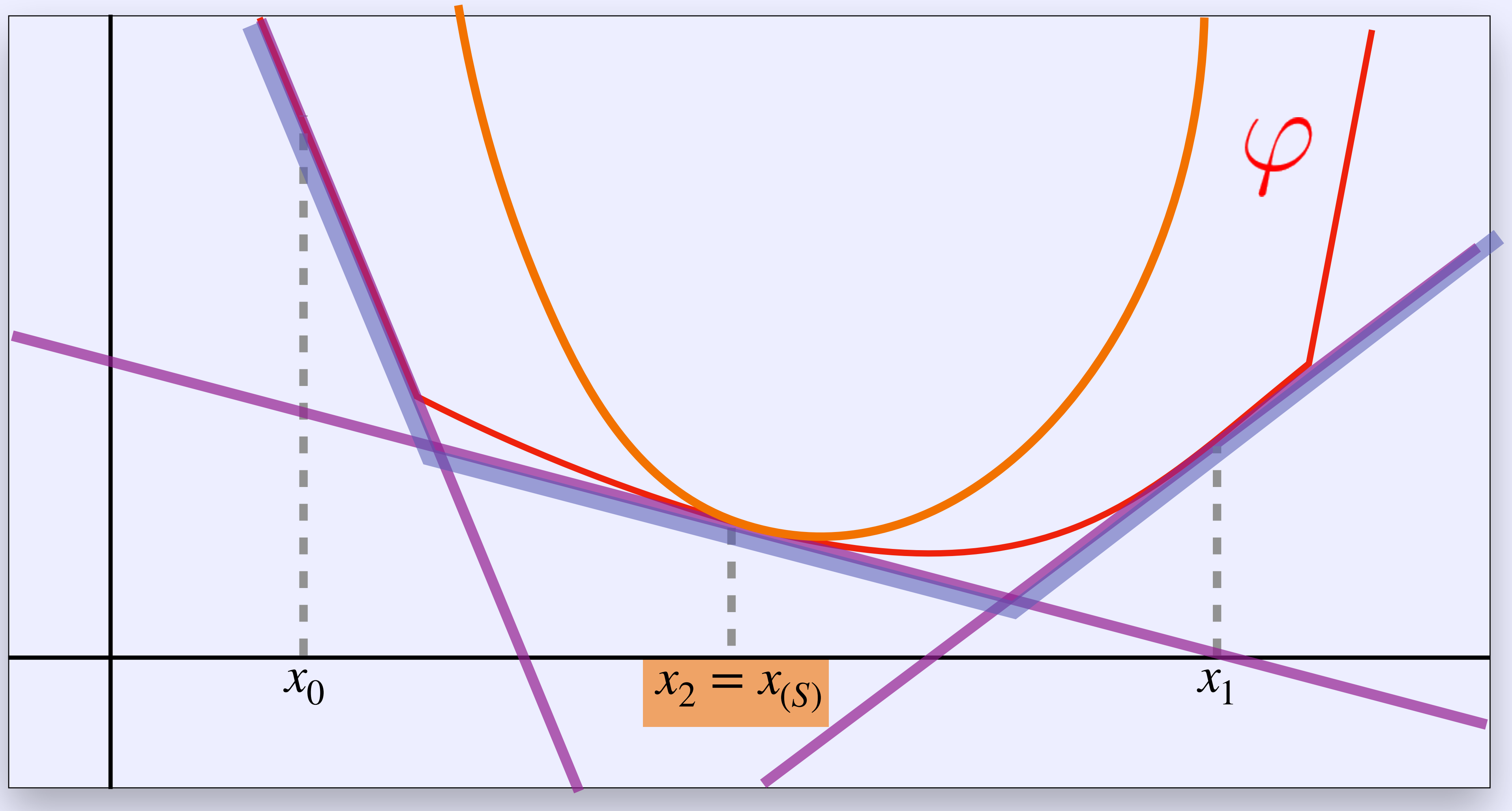
$$\min_{x \in \mathbb{R}^d} \check{\varphi}_1(x) - g_{\varphi_2}^\top (x - x_{(S)}) + \alpha \|x - x_{(S)}\|_2^2$$

  - Update rule for the stability center:

$$\varphi(x_{k+1}) \leq \varphi(x_{(s)}) - \beta \|x_{k+1} - x_{(S)}\|^2$$

Recall: Solving DC programs by Bundle

- Bundle methods in a nutshell
  - Minimization of non-smooth problems
  - Maintains:
    - the **Bundle Information.**
    - the **Polyhedral Approximation.**
    - the **Stability Center** in the bundle.



- State-of-the-art methods for DC problems

  [De Oliveira 19']

  - Function of the form $\varphi(x) = \varphi_1(x) - \varphi_2(x)$
  - We now solve at each iteration :

$$\min_{x \in \mathbb{R}^d} \check{\varphi}_1(x) - g_{\varphi_2}^\top (x - x_{(S)}) + \alpha \|x - x_{(S)}\|_2^2$$

  - Update rule for the stability center:

$$\varphi(x_{k+1}) \leq \varphi(x_{(s)}) - \beta \|x_{k+1} - x_{(S)}\|^2$$

- Convergence property

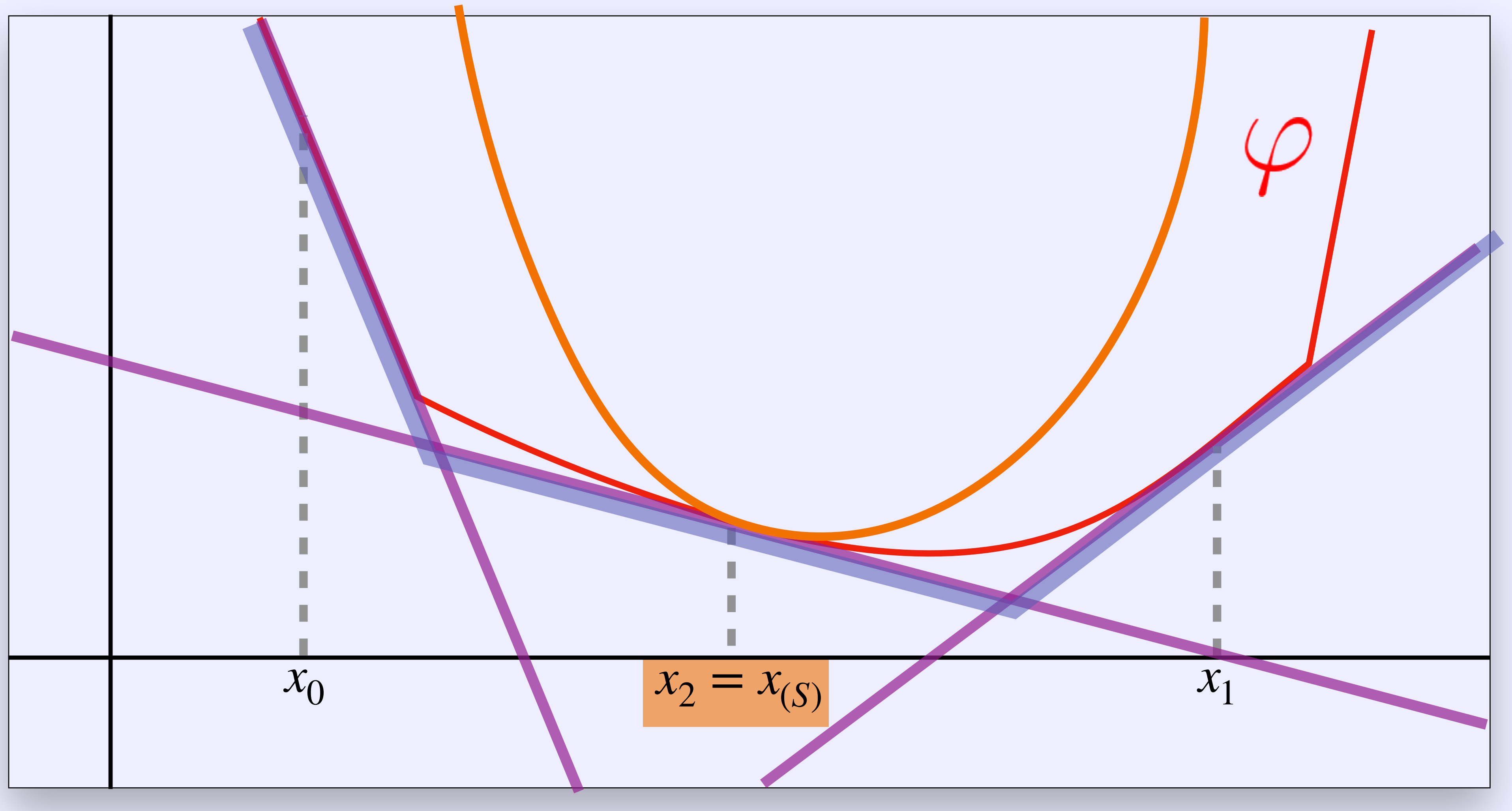
Global → Local → d-stationary → Critical

d-stationary: $\partial\varphi_2(x) \subset \partial\varphi_1(x)$

Critical: $\partial\varphi_2(x) \cap \partial\varphi_1(x) \neq \emptyset$

Recall: Solving DC programs by Bundle

- Bundle methods in a nutshell
  - Minimization of non-smooth problems
  - Maintains:
    - the **Bundle Information.**
    - the **Polyhedral Approximation.**
    - the **Stability Center** in the bundle.



- State-of-the-art methods for DC problems

[De Oliveira 19']

  - Function of the form $\varphi(x) = \varphi_1(x) - \varphi_2(x)$
  - We now solve at each iteration :

$$\min_{x \in \mathbb{R}^d} \check{\varphi}_1(x) - g_{\varphi_2}^\top (x - x_{(S)}) + \alpha \|x - x_{(S)}\|_2^2$$

  - Update rule for the stability center:

$$\varphi(x_{k+1}) \leq \varphi(x_{(s)}) - \beta \|x_{k+1} - x_{(S)}\|^2$$

- Convergence property

Global → Local → d-stationary → Critical

Improvement when φ_2 is smooth: $\partial\varphi_2(x) \subset \partial\varphi_1(x)$

For the bundle method: $\partial\varphi_2(x) \cap \partial\varphi_1(x) \neq \emptyset$

For Our DC Problem

- The DC problem

$$(\mathcal{P}_{\lambda,\mu}) \quad \min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \overbrace{f(x) + \mu \max(\eta, 0) + \lambda \left(G(x,\eta)\right.}^{\varphi_1} - \overbrace{\left.\bar{Q}_p(g(x,\xi))\right)}^{\varphi_2}$$

For Our DC Problem

- The DC problem

$$(\mathcal{P}_{\lambda,\mu}) \quad \min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \overbrace{f(x) + \mu \max(\eta, 0) + \lambda \left(G(x,\eta)\right.}^{\varphi_1} - \overbrace{\bar{Q}_p(g(x,\xi))}^{\varphi_2}\left.\right)$$

- Smoothing of the superquantile [L., Malick, Harchaoui 20']

  - Smoothing of f_2 based on Nesterov's technique.

$$\bar{Q}_p(U) = \sup_{\substack{q\in\mathbb{R}^n \\ \sum_{i=1}^n q_i = 1 \\ 0\le q_i \le \frac{1}{n(1-p)}}} \sum_{i=1}^n q_i U_i$$

For Our DC Problem

- The DC problem

$$(\mathcal{P}_{\lambda,\mu}) \quad \min_{x\in\mathbb{R}^d,\eta\in\mathbb{R}} \overbrace{f(x) + \mu\max(\eta,0) + \lambda\Big(G(x,\eta)}^{\varphi_1} - \overbrace{\bar{Q}_p(g(x,\xi))}^{\varphi_2}\Big)$$

- Smoothing of the superquantile [L., Malick, Harchaoui 20']

  - Smoothing of f_2 based on Nesterov's technique.

$$\bar{Q}_p(U) \simeq \sup_{\substack{q\in\mathbb{R}^n \\ \sum_{i=1}^n q_i = 1 \\ 0\le q_i\le \frac{1}{n(1-p)}}} \sum_{i=1}^n q_i U_i - \frac{\alpha}{2}\|q - \frac{1}{n}(1,\ldots,1)^\top\|^2$$

What a long process !

- To sum up

CHANCE CONSTRAINED PROBLEM → BILEVEL PROGRAM → DOUBLE PENALIZATION → BUNDLE METHOD

- What about the implementation ?

What a long process !

- To sum up

CHANCE CONSTRAINED PROBLEM → BILEVEL PROGRAM → DOUBLE PENALIZATION → BUNDLE METHOD

- What about the implementation ?

TACO : a Toolbox for chAnce Constrained Optimization

- Goal : solve a problem of the form $$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } \mathbb{P}[g(x,\xi) \leq 0] \geq p$$
- Input : the class `Problem`
  - First-order oracles for f and g.
  - A sampled dataset for the values of ξ.
  - A python dictionary of parameters.

Example : Kataoka's Example

In[1]:

```
import numpy as np

class Kataoka:

    def __init__(self, nb_samples=10000, nb_features=2, seed=42):

        np.random.seed(seed)
        mean = np.array([1.0, 1.0])
        cov =  np.eye(2)
        self.data = np.random.multivariate_normal(mean, cov,
size=self.nb_samples)

    def objective_func(self, x):
         return 0.5*np.dot(x,x)

    def objective_grad(self,x):
        return x

    def constraint_func(self, x, z):
        return np.dot(x,z)

    def constraint_grad(self, x, z):
        return z
```

TACO : a Toolbox for chAnce Constrained Optimization

- Goal : solve a problem of the form
$$\min_{x\in\mathbb{R}^d} f(x) \quad \text{s.t. } \mathbb{P}[g(x,\xi)\leq 0] \geq p$$
- Input : the class `Problem`
  - First-order oracles for f and g.
  - A sampled dataset for the values of ξ.
  - A python dictionary of parameters.
- The class `Optimizer`
  - Instantiate with the inputs.
  - Optimization launched with the method `run`.

Example : Kataoka's Example

In[2]:
```
optimizer = Optimizer(problem, params=params)
optimiser.run()
```

TACO : a Toolbox for chAnce Constrained Optimization

- Goal : solve a problem of the form
$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } \mathbb{P}[g(x,\xi) \leq 0] \geq p$$
- Input : the class `Problem`
  - First-order oracles for f and g.
  - A sampled dataset for the values of ξ.
  - A python dictionary of parameters.
- The class `Optimizer`
  - Instantiate with the inputs.
  - Optimization launched with the method `run`.
- Output
  - Retrieved from the `Optimizer` class.

Example : Kataoka's Example

In[2]:
```
optimizer = Optimizer(problem, params=params)
optimiser.run()
```

In[3]:
```
sol = optimizer.solution
```

TACO : a Toolbox for chAnce Constrained Optimization

- Goal : Solve a problem of the form

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t. } \mathbb{P}[g(x,\xi) \leq 0] \geq p$$

- Input : the class `Problem`
  - First-order oracles for f and g.
  - A sampled dataset for the values of ξ.
  - A python dictionary of parameters.
- The class `Optimizer`
  - Instantiate with the inputs.
  - Optimization launched with the method `run`.
- Output
  - Retrieved from the `Optimizer` class.

Example : Kataoka's Example

In[2]:
```
optimizer = Optimizer(problem, params=params)
optimiser.run()
```

In[3]:
```
sol = optimizer.solution
```

- Hyperparameters
  - Probability threshold p
  - Penalization parameters μ, λ
  - Number of iterations, starting point, target precision, etc.

4 Numerical Illustrations

Proof of concept on a quadratic Chance constraint Problem

- 2D quadratic problem

$$\min_{x \in \mathbb{R}^d} f(x)$$
$$\text{s.t. } \mathbb{P}[g(x,\xi) \leq 0] \geq p$$

$$f(x) = (x-c)^\top A(x-c)$$
$$g(x,z) = z^\top W(x)^\top z + p^\top z + b$$
$$\xi \sim \mathcal{N}(\mu, \Sigma)$$

Proof of concept on a quadratic Chance constraint Problem

- 2D quadratic problem

$$\min_{x\in\mathbb{R}^d} f(x) \quad \text{s.t. } \mathbb{P}[g(x,\xi)\le 0]\ge p$$

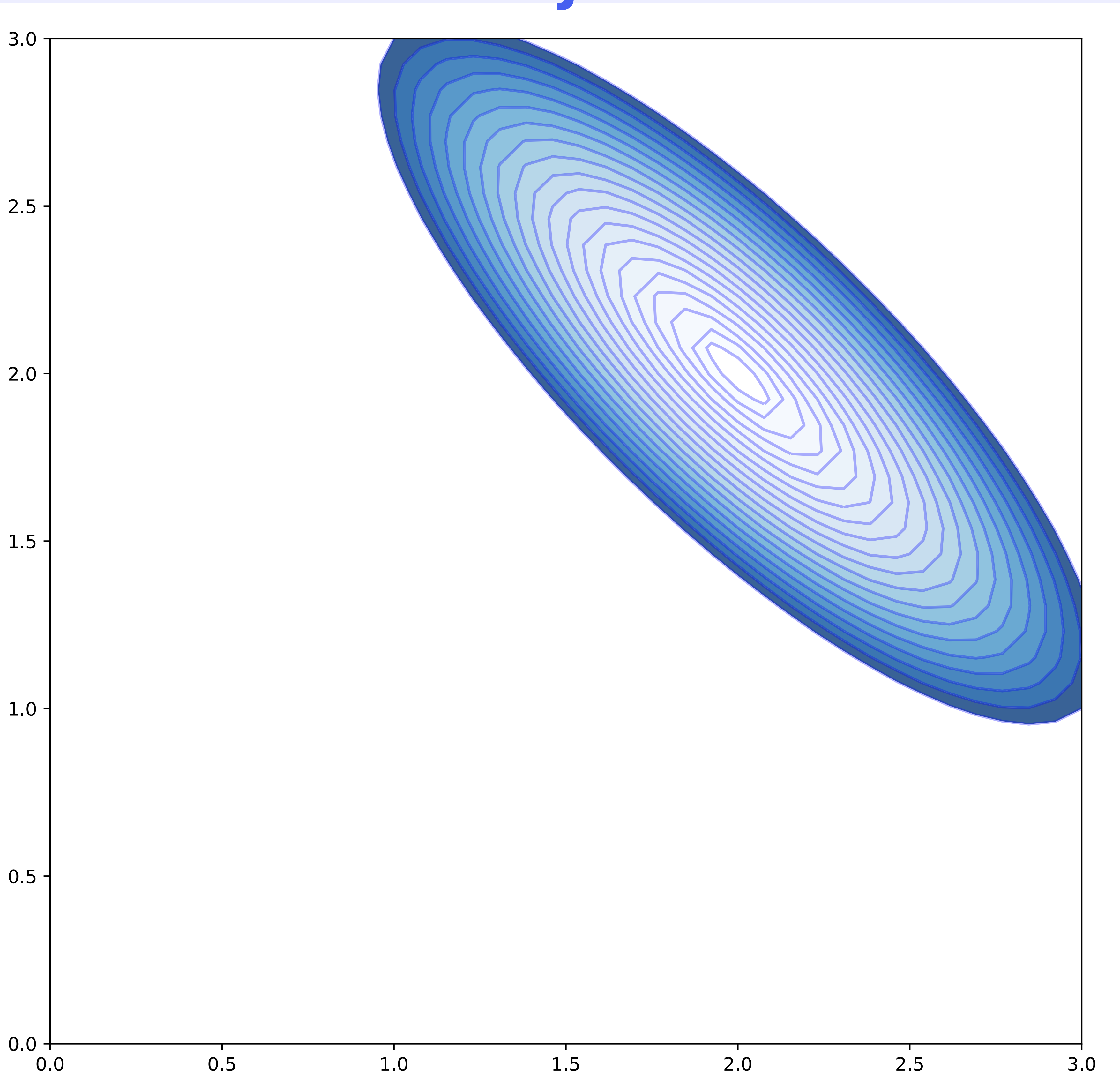
$$f(x) = (x-c)^\top A(x-c)$$

$$g(x,z) = z^\top W(x)^\top z + p^\top z + b$$

$$\xi \sim \mathcal{N}(\mu, \Sigma)$$

$$c = \begin{pmatrix} 2. \\ 2. \end{pmatrix} \quad A = \begin{pmatrix} 5.5 & 4.5 \\ 4.5 & 5.5 \end{pmatrix}$$

The Objective



Proof of concept on a quadratic Chance constraint Problem

- 2D quadratic problem

$$\min_{x \in \mathbb{R}^d} f(x) \qquad f(x) = (x-c)^\top A(x-c)$$

$$\text{s.t. } \mathbb{P}[g(x,\xi) \leq 0] \geq p \qquad g(x,z) = z^\top W(x)^\top z + q^\top z + r$$
$$\xi \sim \mathcal{N}(\mu, \Sigma)$$

$$c = \begin{pmatrix} 2. \\ 2. \end{pmatrix} \quad A = \begin{pmatrix} 5.5 & 4.5 \\ 4.5 & 5.5 \end{pmatrix}$$

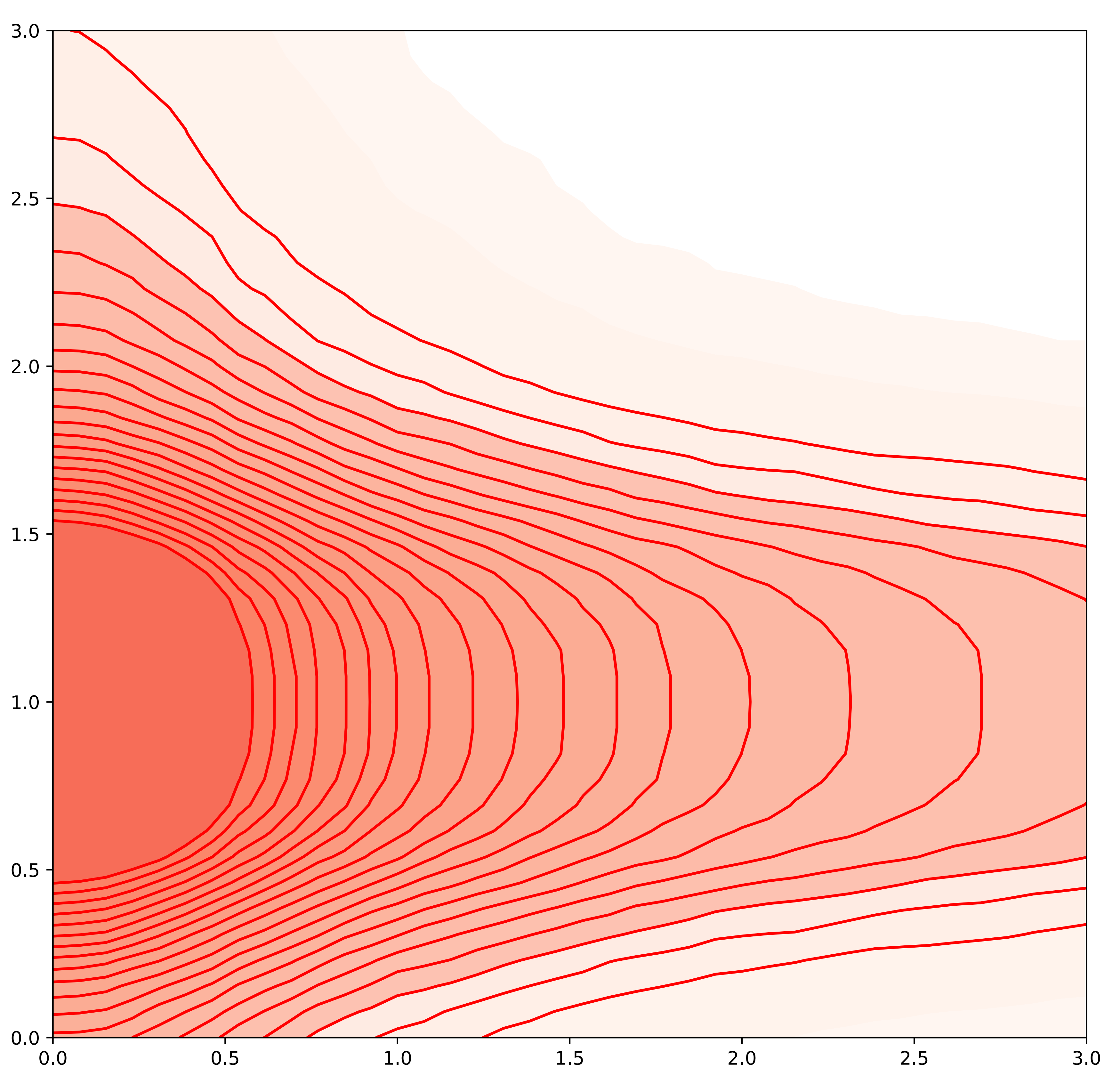
$$W : x = (x_1, x_2)^\top \mapsto \begin{pmatrix} x_1^2 + 0.5 & 0.0 \\ 0.0 & |x_2 - 1|^3 + 1. \end{pmatrix}$$

$$q = \begin{pmatrix} 1. \\ 1. \end{pmatrix}, \ r = -1$$

ξ is sampled 10000 times with parameters $\mu = \begin{pmatrix} 1. \\ 1. \end{pmatrix}$

$$\Sigma = \begin{pmatrix} 20. & 0. \\ 0. & 20. \end{pmatrix}$$

The Chance Constraint



Proof of concept on a quadratic Chance constraint Problem

- 2D quadratic problem

$$\min_{x\in\mathbb{R}^d} f(x) \qquad f(x) = (x-c)^\top A(x-c)$$

$$\text{s.t. } \mathbb{P}[g(x,\xi)\le 0] \ge p \qquad g(x,z) = z^\top W(x)^\top z + q^\top z + r$$

$$\xi \sim \mathcal{N}(\mu,\Sigma)$$

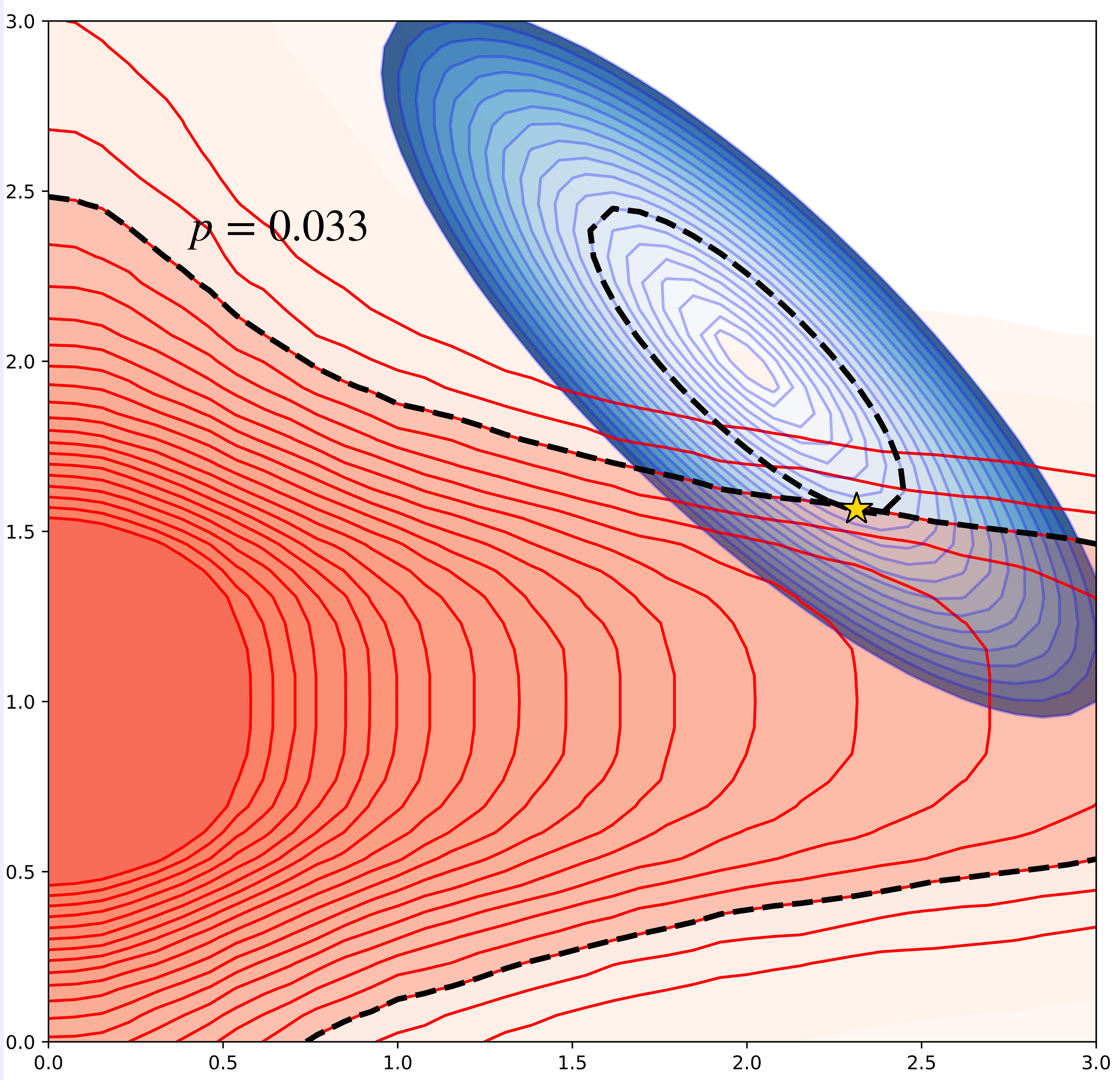
$$c = \begin{pmatrix} 2. \\ 2. \end{pmatrix} \quad A = \begin{pmatrix} 5.5 & 4.5 \\ 4.5 & 5.5 \end{pmatrix}$$

$$W : x = (x_1, x_2)^\top \mapsto \begin{pmatrix} x_1^2 + 0.5 & 0.0 \\ 0.0 & |x_2 - 1|^3 + 1. \end{pmatrix}$$

$$q = \begin{pmatrix} 1. \\ 1. \end{pmatrix}, \ r = -1$$

ξ is sampled 10000 times with parameters $\mu = \begin{pmatrix} 1. \\ 1. \end{pmatrix}$

$$\Sigma = \begin{pmatrix} 20. & 0. \\ 0. & 20. \end{pmatrix}$$



Proof of concept on a quadratic Chance constraint Problem

- 2D quadratic problem

$$\min_{x \in \mathbb{R}^d} f(x) \qquad f(x) = (x-c)^\top A(x-c)$$

$$\text{s.t. } \mathbb{P}[g(x,\xi) \le 0] \ge p \qquad g(x,z) = z^\top W(x)^\top z + q^\top z + r$$

$$\xi \sim \mathcal{N}(\mu, \Sigma)$$

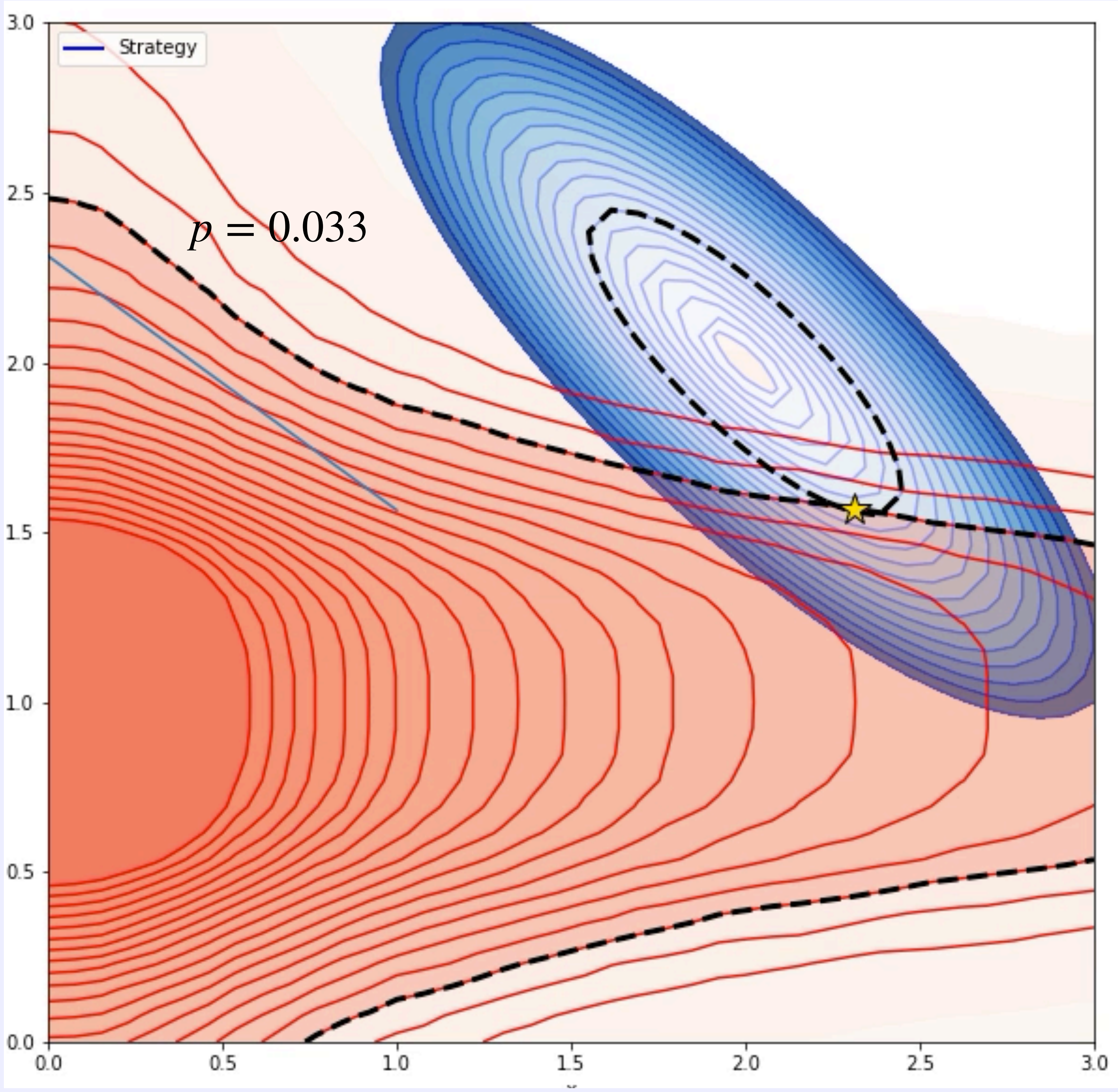
$$c = \begin{pmatrix} 2. \\ 2. \end{pmatrix} \quad A = \begin{pmatrix} 5.5 & 4.5 \\ 4.5 & 5.5 \end{pmatrix}$$

$$W : x = (x_1, x_2)^\top \mapsto \begin{pmatrix} x_1^2 + 0.5 & 0.0 \\ 0.0 & |x_2 - 1|^3 + 1. \end{pmatrix}$$

$$q = \begin{pmatrix} 1. \\ 1. \end{pmatrix}, \; r = -1$$

ξ is sampled 10000 times with parameters $\mu = \begin{pmatrix} 1. \\ 1. \end{pmatrix}$

$$\Sigma = \begin{pmatrix} 20. & 0. \\ 0. & 20. \end{pmatrix}$$



Proof of concept on a quadratic Chance constraint Problem

- 2D quadratic problem

$$\min_{x\in\mathbb{R}^d} f(x) \qquad f(x) = (x-c)^\top A(x-c)$$

$$\text{s.t. } \mathbb{P}[g(x,\xi)\le 0] \ge p \qquad g(x,z) = z^\top W(x)^\top z + q^\top z + r$$

$$\xi \sim \mathcal{N}(\mu, \Sigma)$$

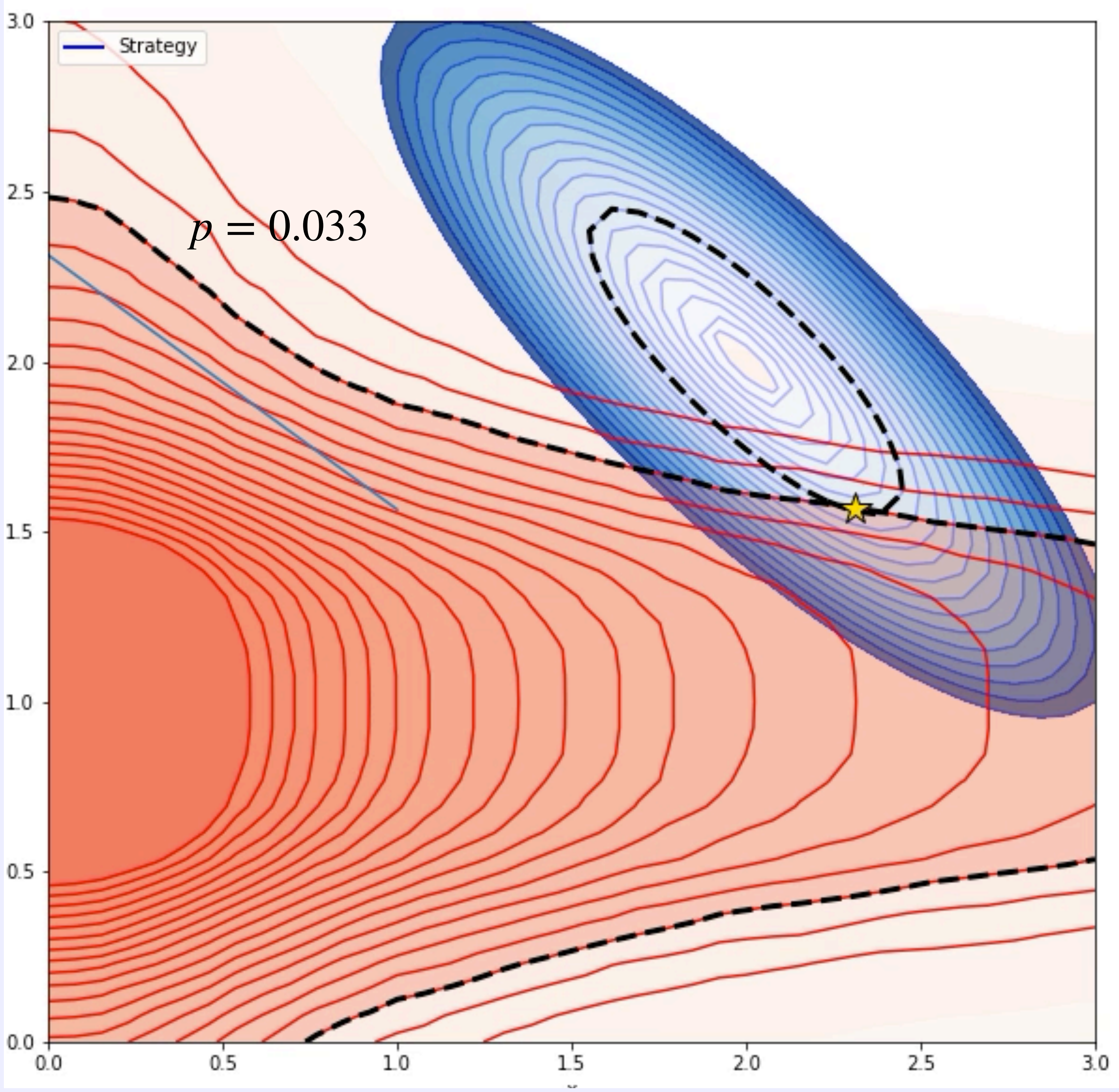
$$c = \begin{pmatrix} 2. \\ 2. \end{pmatrix} \quad A = \begin{pmatrix} 5.5 & 4.5 \\ 4.5 & 5.5 \end{pmatrix}$$

$$W : x = (x_1, x_2)^\top \mapsto \begin{pmatrix} x_1^2 + 0.5 & 0.0 \\ 0.0 & |x_2 - 1|^3 + 1. \end{pmatrix}$$

$$q = \begin{pmatrix} 1. \\ 1. \end{pmatrix}, \; r = -1$$

ξ is sampled 10000 times with parameters $\mu = \begin{pmatrix} 1. \\ 1. \end{pmatrix}$

$$\Sigma = \begin{pmatrix} 20. & 0. \\ 0. & 20. \end{pmatrix}$$



Numerical Experiments on Second Toy Problem

- A norm optimization problem

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t.} \quad \mathbb{P}[g(x, \xi) \leq 0] \geq p$$

$$f(x) = -\|x\|_1$$

$$g : \mathbb{R}^d \times \mathcal{M}_{n,d} \to \mathbb{R}$$

$$x, Z \mapsto \max_{i \in [n]} \sum_{j=1}^{d} Z_{i,j}^2 x_j^2$$

$$\xi_{i,j} \sim \mathcal{N}(0, 1)$$

$$p = 0.8$$

Numerical Experiments on Second Toy Problem

- A norm optimization problem

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{s.t.} \quad \mathbb{P}[g(x,\xi) \le 0] \ge p$$

$$f(x) = -\|x\|_1$$

$$g : \mathbb{R}^d \times \mathcal{M}_{n,d} \to \mathbb{R}$$

$$x, Z \mapsto \max_{i \in [n]} \sum_{j=1}^{d} Z_{i,j}^2 x_j^2$$

$$\xi_{i,j} \sim \mathcal{N}(0,1)$$

$$p = 0.8$$

- Optimal value and solution

$$f^\star = \frac{10d}{\sqrt{F_{\chi_d^2}^{-1}(p^{\frac{1}{10}})}} \qquad x_i^\star = \frac{10}{\sqrt{F_{\chi_d^2}^{-1}(p^{\frac{1}{10}})}}, i \in \{1, \dots, d\}$$

$F_{\chi_d^2}^{-1}$: Quantile function of a χ^2 distribution with d degrees of freedom

Numerical Experiments on Second Toy Problem

- A family of norm optimization problems
  Hong, Yang, Zhang (2009)

$$\min_{x \in \mathbb{R}^d} f(x) \qquad f(x) = -\|x\|_1$$

$$\text{s.t. } \mathbb{P}[g(x,\xi) \leq 0] \geq p$$

$$g : \mathbb{R}^d \times \mathcal{M}_{n,d} \to \mathbb{R}$$

$$x, Z \mapsto \max_{i \in [n]} \sum_{j=1}^{d} Z_{i,j}^2 x_j^2$$

$$\xi_{i,j} \sim \mathcal{N}(0,1)$$

$$p = 0.8$$

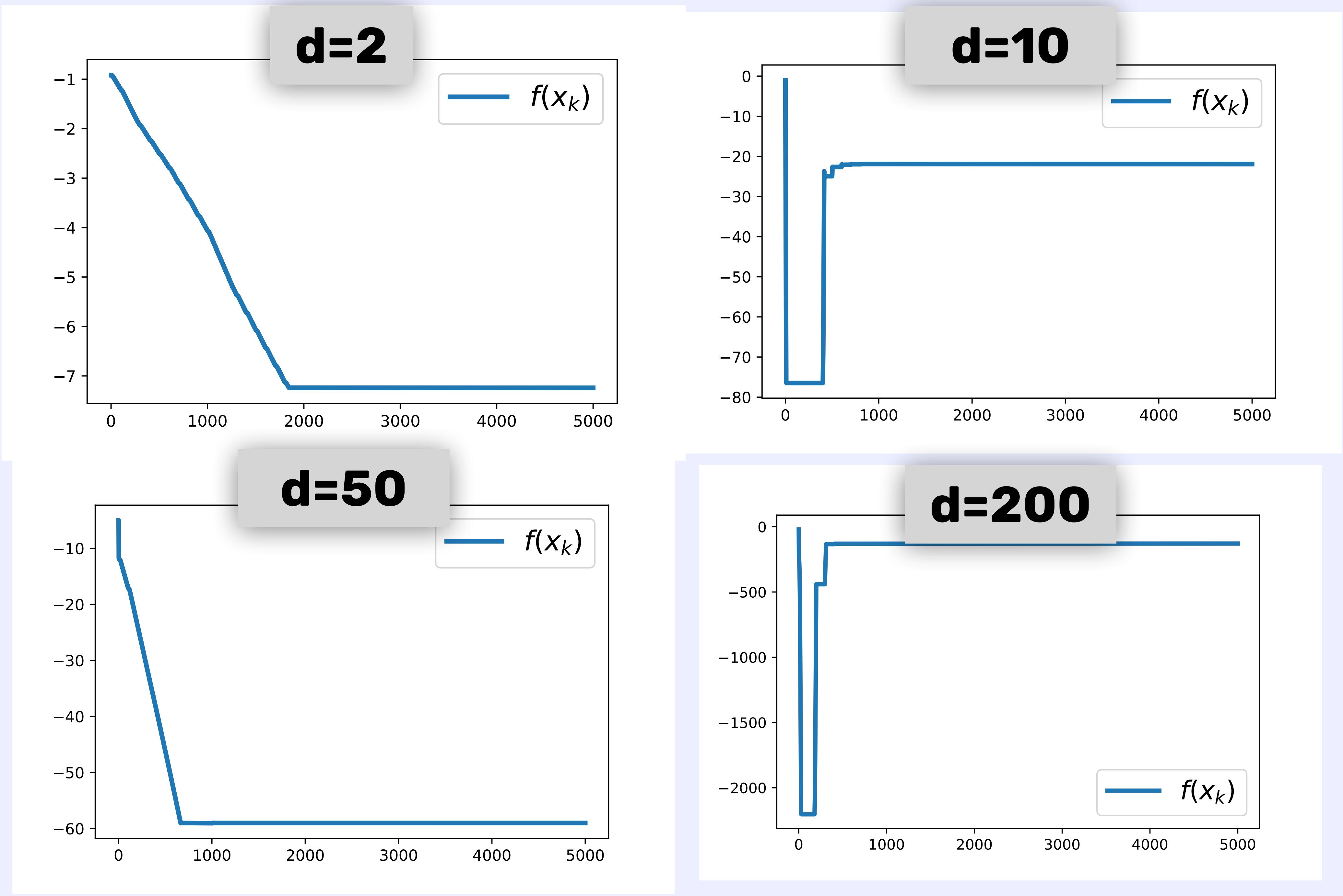
Dimension	Final Sub-optimality	$\mathbb{P}[g(x,\xi) \leq 0]$	μ	λ
$d = 2$	5.1×10^{-4}	0.7992	0.01	10.0
$d = 10$	2.4×10^{-2}	0.8	1.0	0.01
$d = 50$	1.2×10^{-1}	0.7999	1.0	10.0
$d = 200$	2.8×10^{-1}	0.7997	1.0	0.01

- Optimal value and solution

$$f^\star = \frac{10d}{\sqrt{F_{\chi_d^2}^{-1}(p^{\frac{1}{10}})}} \qquad x_i^\star = \frac{10}{\sqrt{F_{\chi_d^2}^{-1}(p^{\frac{1}{10}})}}, i \in \{1, \ldots, d\}$$

Quantile function of a χ^2 distribution with d degrees of freedom

- Numerical Results



Conclusion

- We propose a new approach to chance constraints via Bilevel Programming.
- We derive a double penalization method for this approach, with an exact penalty for the hard constraint.
- We propose a python toolbox to test out your problems.
- Derive more methods from the bilevel approach

yassine.laguel@univ-grenoble-alpes.fr

References

- S. Kataoka.: A stochastic programming model. Econometrica **31**, (1963)
- R. Henrion, C. Strugarek: Convexity of chance constraints with independent random variables. Computational Optimization and Applications (2008)
- W. Van Ackooij: Eventual convexity of chance constrained feasible sets. Optimization, (2015)
- W. Van Ackooij, J. Malick: Eventual convexity of probability constraints with elliptical distributions. Mathematical Programming, (2019)
- W. Van Ackooij, J. Malick: Second-order differentiability of probability functions. Optimization Letters (2017)
- W. Van Ackooij, R. Henrion: (Sub-)Gradient formulae for probability functions of random inequality systems under Gaussian distributionOptimization Letters (2017)
- A. Geletu, A. Hoffmann: Analytic approximation and differentiability of joint chance constraints. Optimization (2019)
- W. Van Ackooij, P. Pérez-Aros: Generalized differentiation of probability functions acting on an infinite system of constraints. SIAM Journal on Optimization (2019)
- H. Heitsch: On probabilistic capacity maximization in a stationary gas networks. Optimization (2019)
- RT Rockafellar, S Uryasev: Optimization of conditional value-at-risk. Journal of risk (2000)
- W de Oliveira: Proximal bundle methods for nonsmooth DC programming. Journal of Global Optimization (2019)
- Y. Laguel, J Malick, Z. Harchaoui: First-rrder optimization for Superquantile-based learning. MLSP (2020)