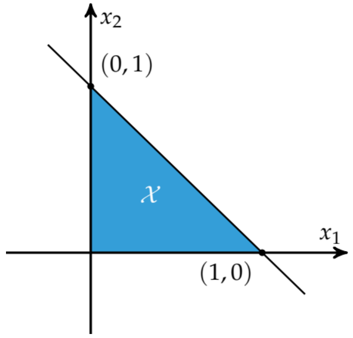
Tecnológico Nacional de México
Instituto Tecnológico de Orizaba/Celaya

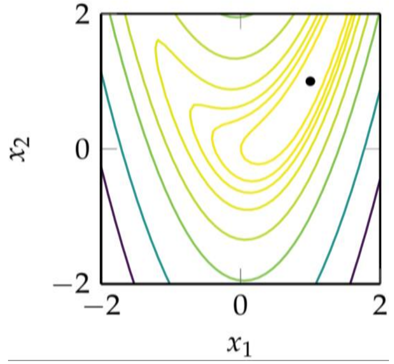
Modern Computing Algorithms for Process Optimization with Julia Programming. Part I

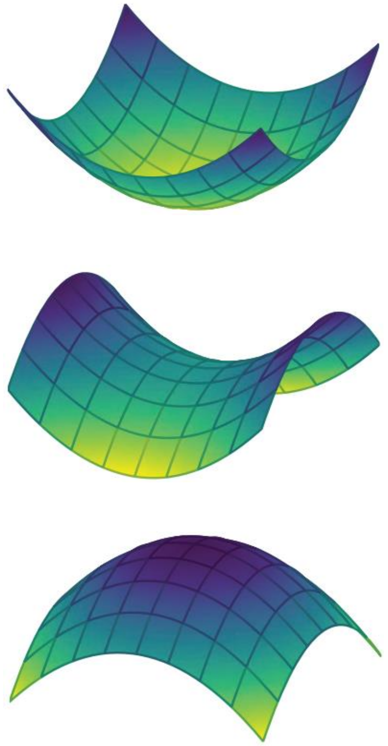


"4 – Local Descent"

By:

Dr. Kelvyn Baruc Sánchez Sánchez
Postdoctoral Researcher/I.T. Celaya





Descent Direction Iteration

- Descent Direction Methods use a local model to incrementally improve design point until some convergence criteria is met

1. Check termination conditions at $\mathbf{x}^{(k)}$; if not met, continue.
2. Decide descent direction $\mathbf{d}^{(k)}$ using local information
3. Decide step size $\alpha^{(k)}$
4. Compute next design point $\mathbf{x}^{(k+1)}$

$$\mathbf{x}^{(k+1)} \leftarrow \mathbf{x}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)}$$

- There are many different optimization methods, each with their own ways of determining α and **d.**

Line Search

- Used to compute α
- Using the techniques discussed in Chapter 3,

$$\underset{\alpha}{\text{minimize}}\; f(\mathbf{x} + \alpha \mathbf{d})$$

- Often this is computed approximately to reduce cost

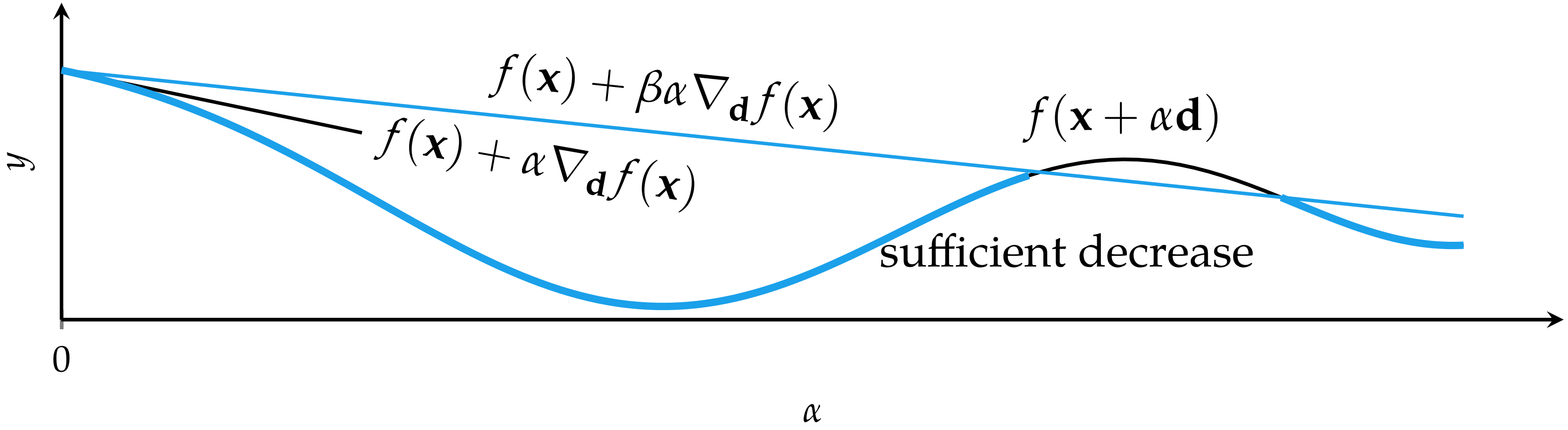
See example 4.1 & 4.2.ipynb

Approximate Line Search

- If function calls are expensive, rather than finding the minimum along a search direction, find a point of sufficient decrease

$$f(\mathbf{x}^{(k+1)}) \leq f(\mathbf{x}^{(k)}) + \beta\alpha\nabla_{\mathbf{d}^{(k)}} f(\mathbf{x}^{(k)})$$

- $\beta \in [0,1]$, usually $\beta = 1\times10^{-4}$

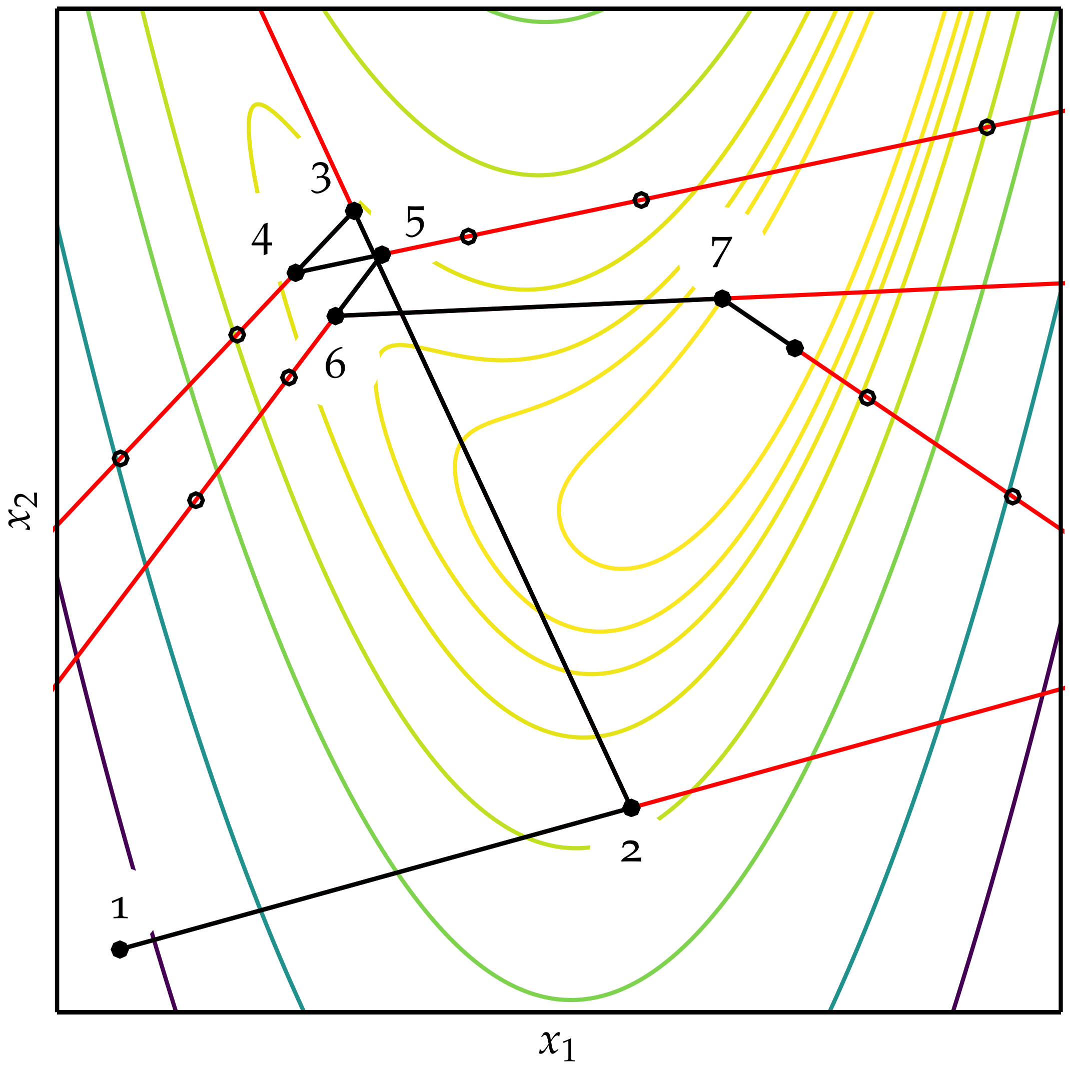


- Backtracking line search starts with a large step and then backs off

See example 4.3.ipynb

Approximate Line Search

Backtracking line search
on Rosenbrock function



See example 4.4a.ipynb

Approximate Line Search

- Building on backtracking line search are the Wolfe Conditions

1. First Wolfe Condition: Sufficient Decrease

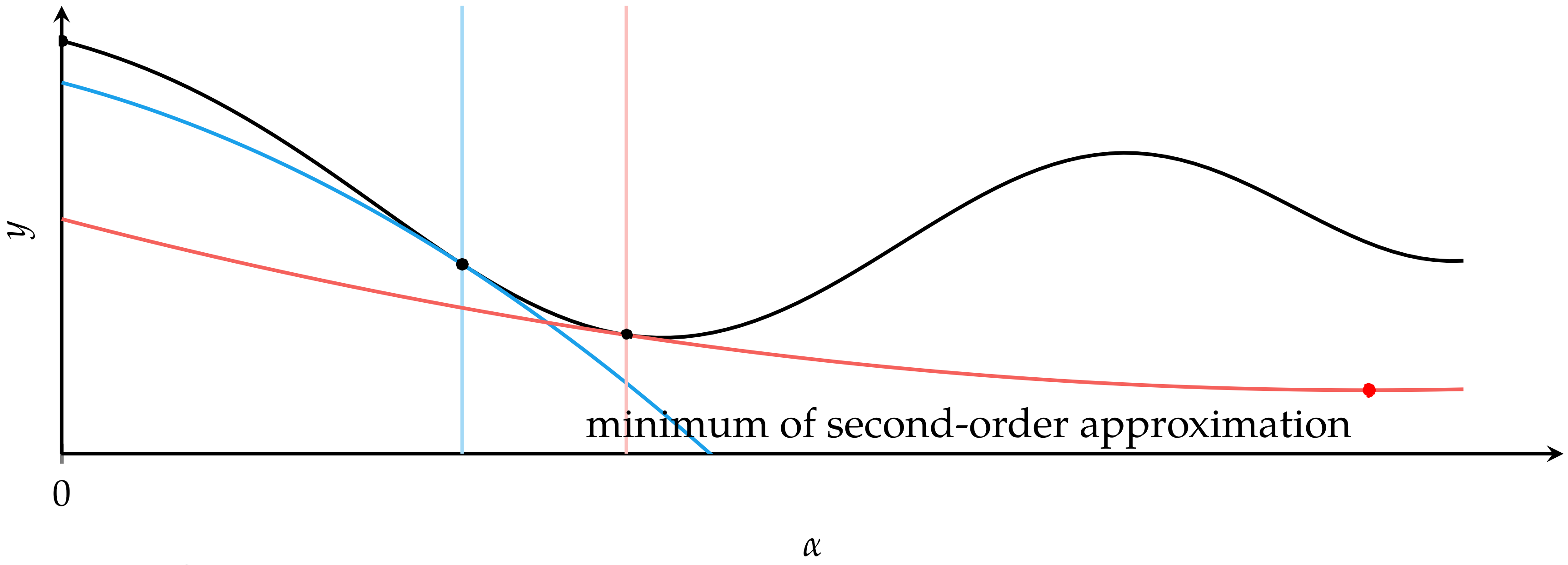
$$f\left(\mathbf{x}^{(k+1)}\right) \leq f\left(\mathbf{x}^{(k)}\right) + \beta\alpha\nabla_{\mathbf{d}^{(k)}} f\left(\mathbf{x}^{(k)}\right)$$

2. Second Wolfe Condition: Curvature Condition

$$\nabla_{\mathbf{d}^{(k)}} f\left(\mathbf{x}^{(k+1)}\right) \geq \sigma\nabla_{\mathbf{d}^{(k)}} f\left(\mathbf{x}^{(k)}\right)$$

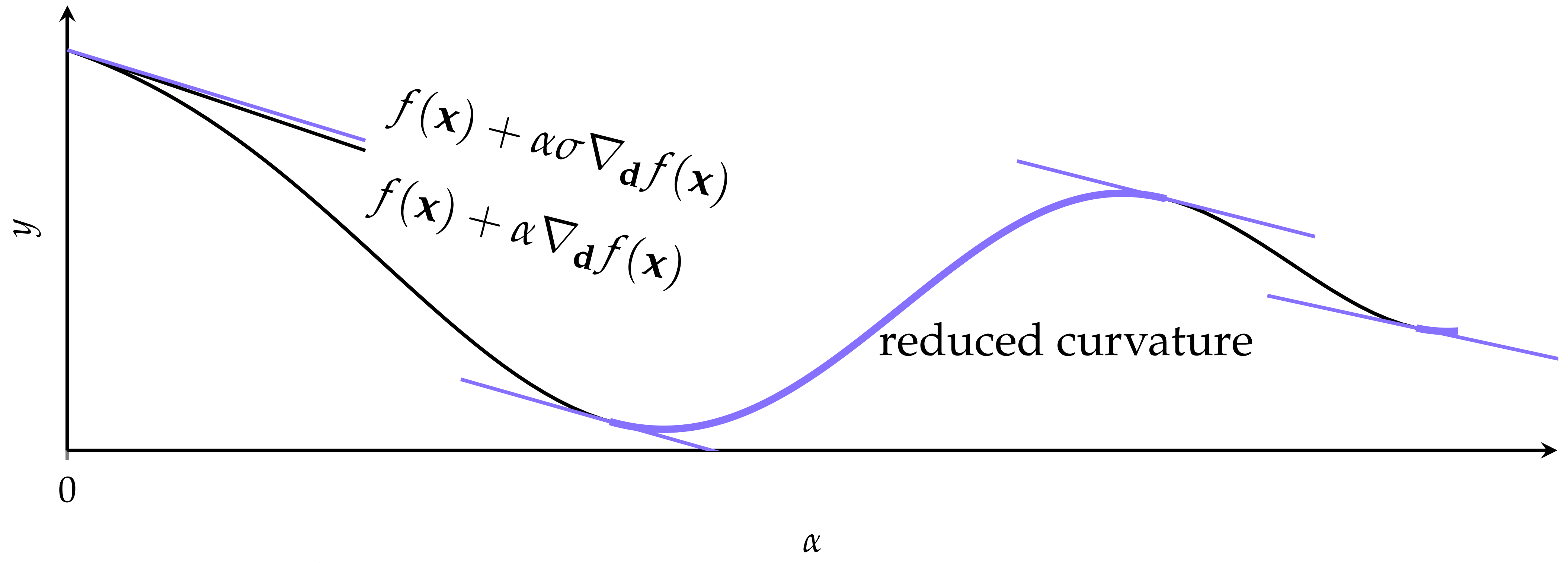
Approximate Line Search

- The curvature condition ensures the second-order function approximations have positive curvature



Approximate Line Search

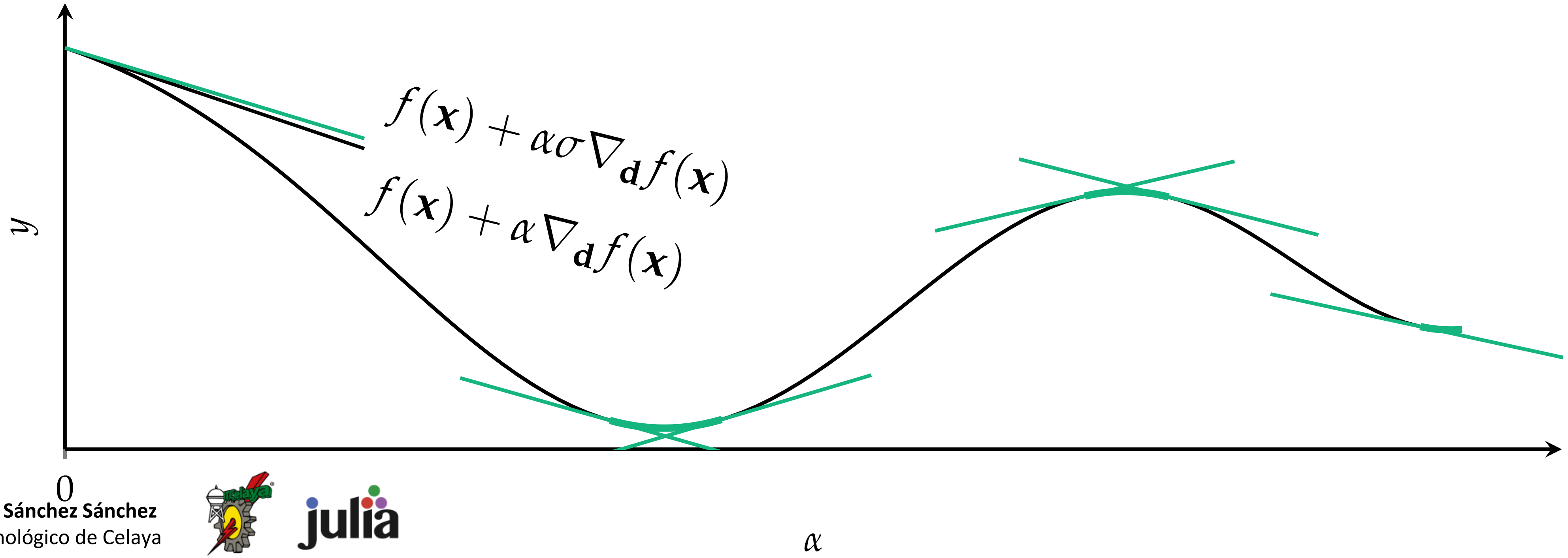
- Regions satisfying the curvature condition



Approximate Line Search

- Regions where the strong curvature condition is satisfied

$$|\nabla_{\mathbf{d}^{(k)}} f(\mathbf{x}^{(k+1)})| \leq -\sigma \nabla_{\mathbf{d}^{(k)}} f(\mathbf{x}^{(k)})$$



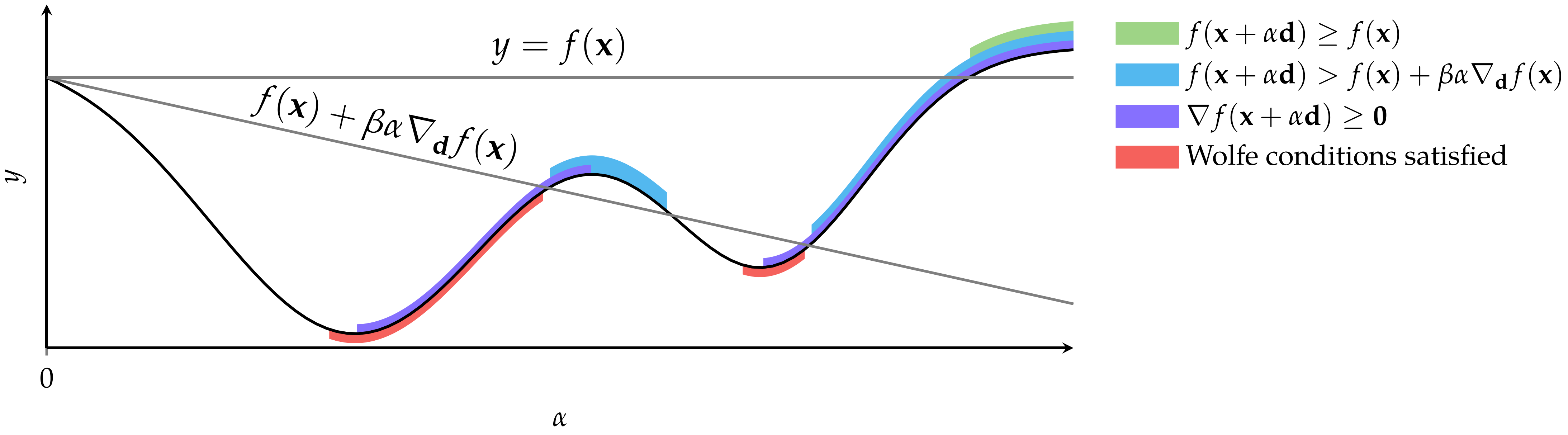
Dr. Kelvyn B. Sánchez Sánchez
Instituto Tecnológico de Celaya

Approximate Line Search

1. Bracketing Phase: test successively larger step sizes to find interval guaranteed to contain step lengths satisfying Wolfe conditions
2. Zoom Phase: shrink the interval using bisection to find point satisfying Wolfe conditions

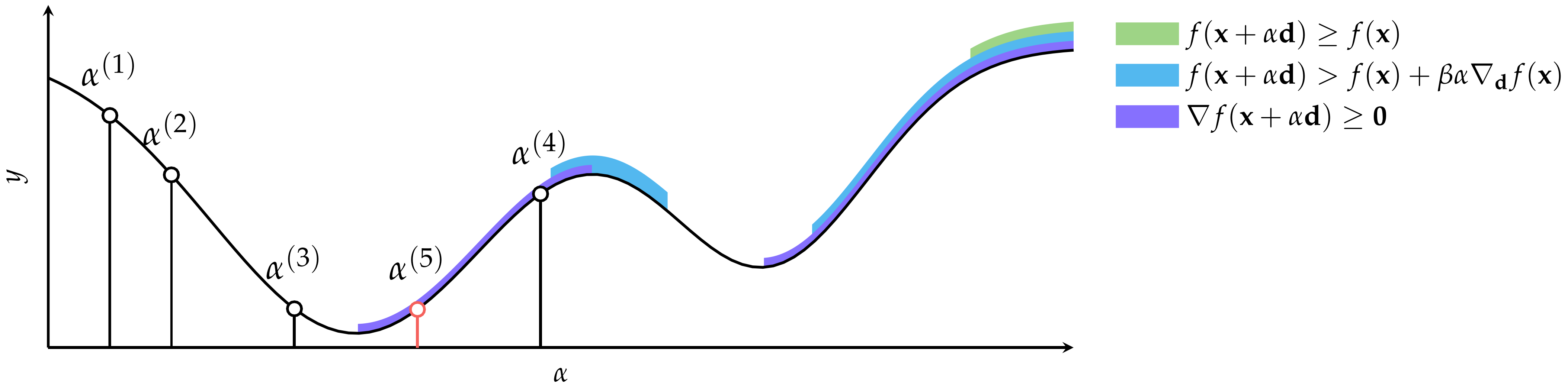
Approximate Line Search

1. Bracketing Phase:



Approximate Line Search

2. Zoom Phase:



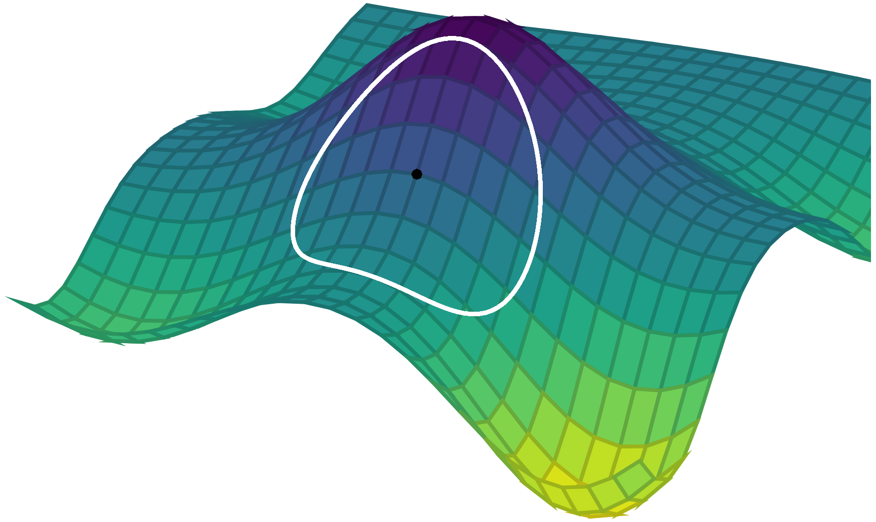
Trust Region Methods

- Descent methods can place too much trust in their first and second order information
- Trust region methods, or restricted step methods, limit the step size to ensure local approximation error is minimized

Trust Region Methods

- $\hat{f}(x')$ is local function approximation at new design point
- δ is trust region radius
- x' is new design point

$$\begin{aligned} \underset{\mathbf{x}'}{\text{minimize}} \quad & \hat{f}(\mathbf{x}') \\ \text{subject to} \quad & \|\mathbf{x} - \mathbf{x}'\| \leq \delta \end{aligned}$$



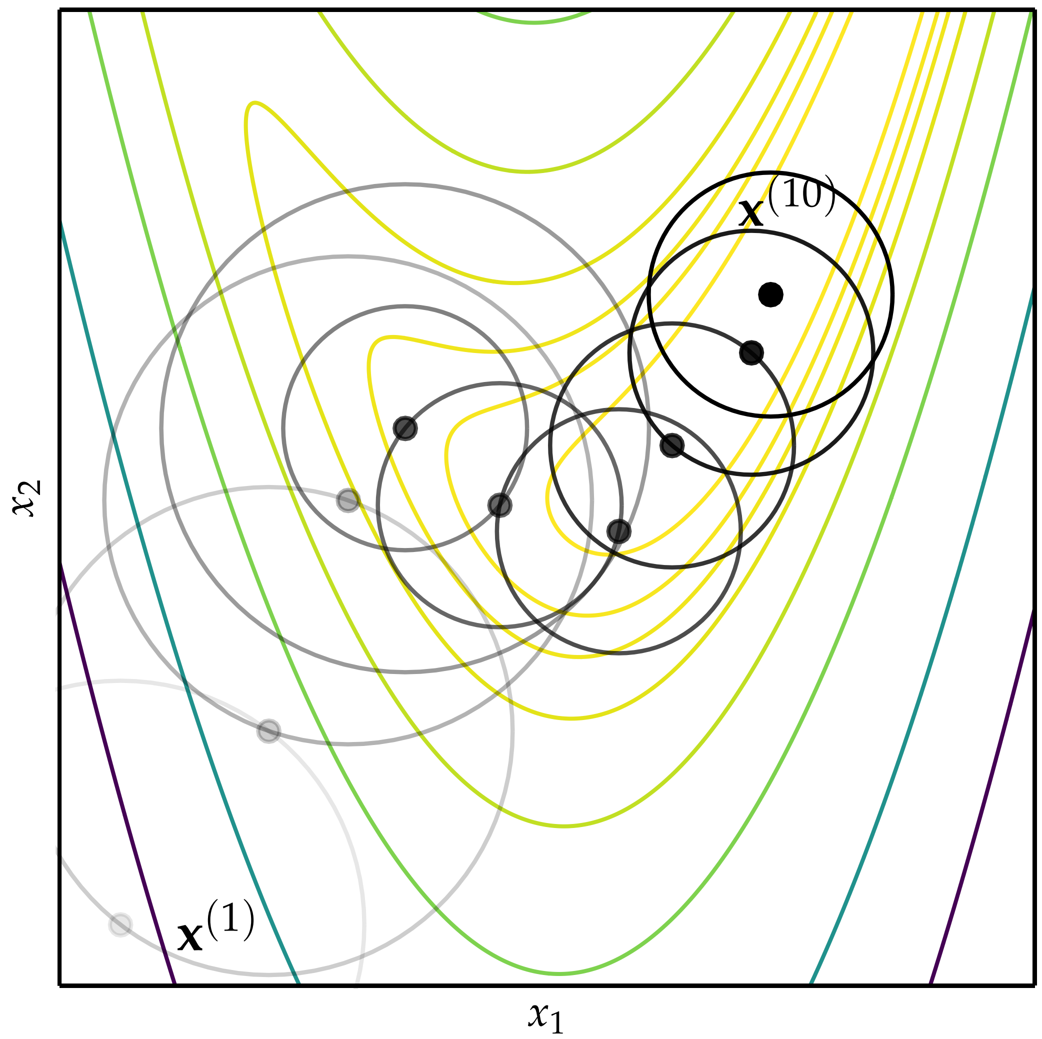
Trust Region Methods

- δ can be expanded or contracted based on performance

$$\eta = \frac{\text{actual improvement}}{\text{predicted improvement}} = \frac{f(\mathbf{x}) - f(\mathbf{x}')}{f(\mathbf{x}) - \hat{f}(\mathbf{x}')}$$

Trust Region Methods

- Trust region optimization of Rosenbrock Function



See example 4.5 & 4.6.ipynb

Termination Conditions

- Maximum Iterations

$$k > k_{\max}$$

- Absolute Improvement

$$f(\mathbf{x}^{(k)}) - f(\mathbf{x}^{(k+1)}) < \epsilon_a$$

Termination Conditions

- Relative Improvement

$$f(\mathbf{x}^{(k)}) - f(\mathbf{x}^{(k+1)}) < \epsilon_r |f(\mathbf{x}^{(k)})|$$

- Gradient Magnitude

$$\|\nabla f(\mathbf{x}^{(k+1)})\| < \epsilon_g$$

Summary

- Descent direction methods incrementally descend toward a local optimum.
- Univariate optimization can be applied during line search.
- Approximate line search can be used to identify appropriate descent step sizes.
- Trust region methods constrain the step to lie within a local region that expands or contracts based on predictive accuracy.
- Termination conditions for descent methods can be based on criteria such as the change in the objective function value or magnitude of the gradient.