Submodular Optimization: From Discrete to Continuous and Back

Hamed Hassani

Amin Karbasi

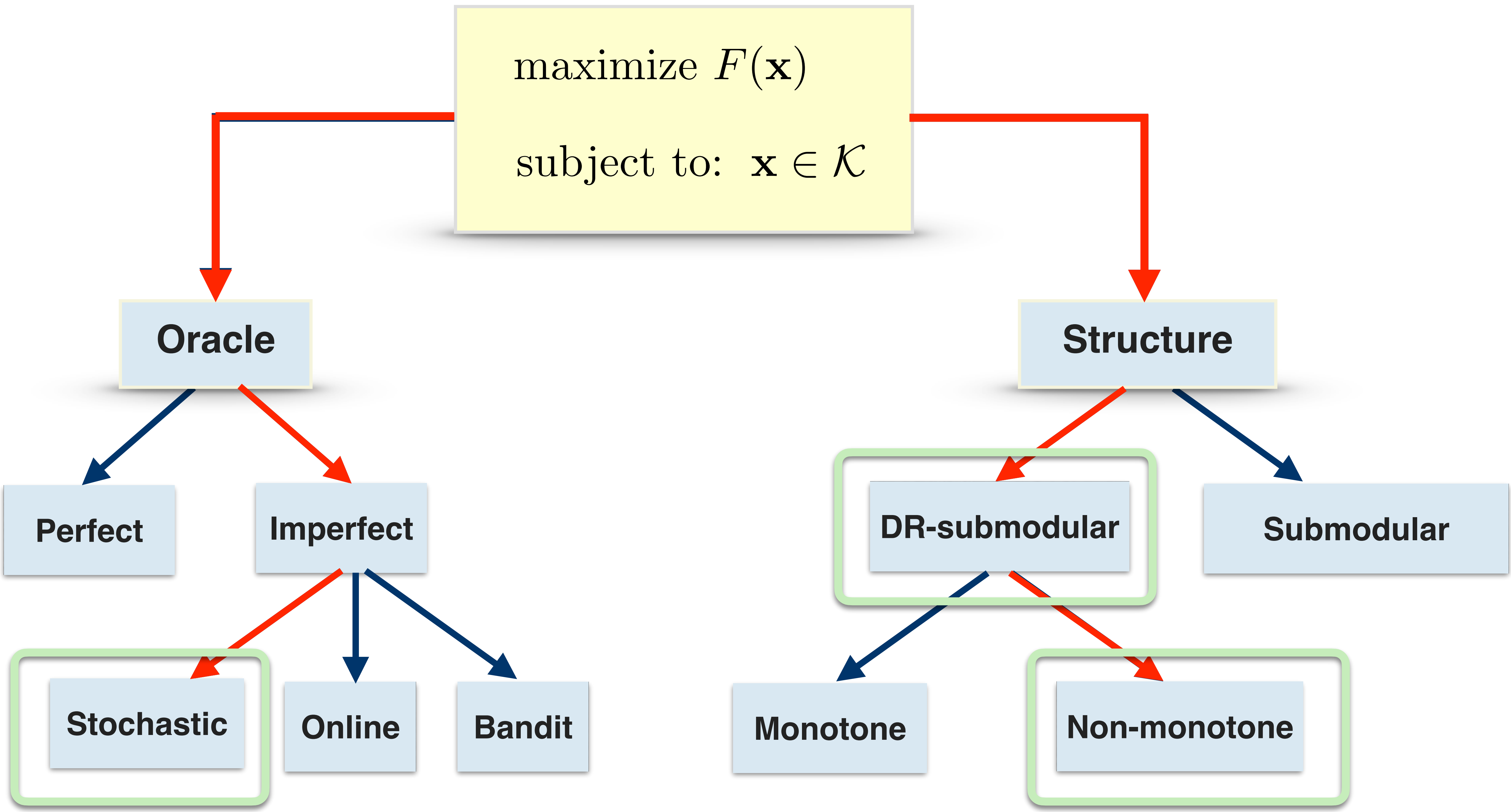
Penn

Yale

Slides + references: http://iid.yale.edu/icml/icml-20.md/

Submodular Maximization

- We consider the following optimization problem:



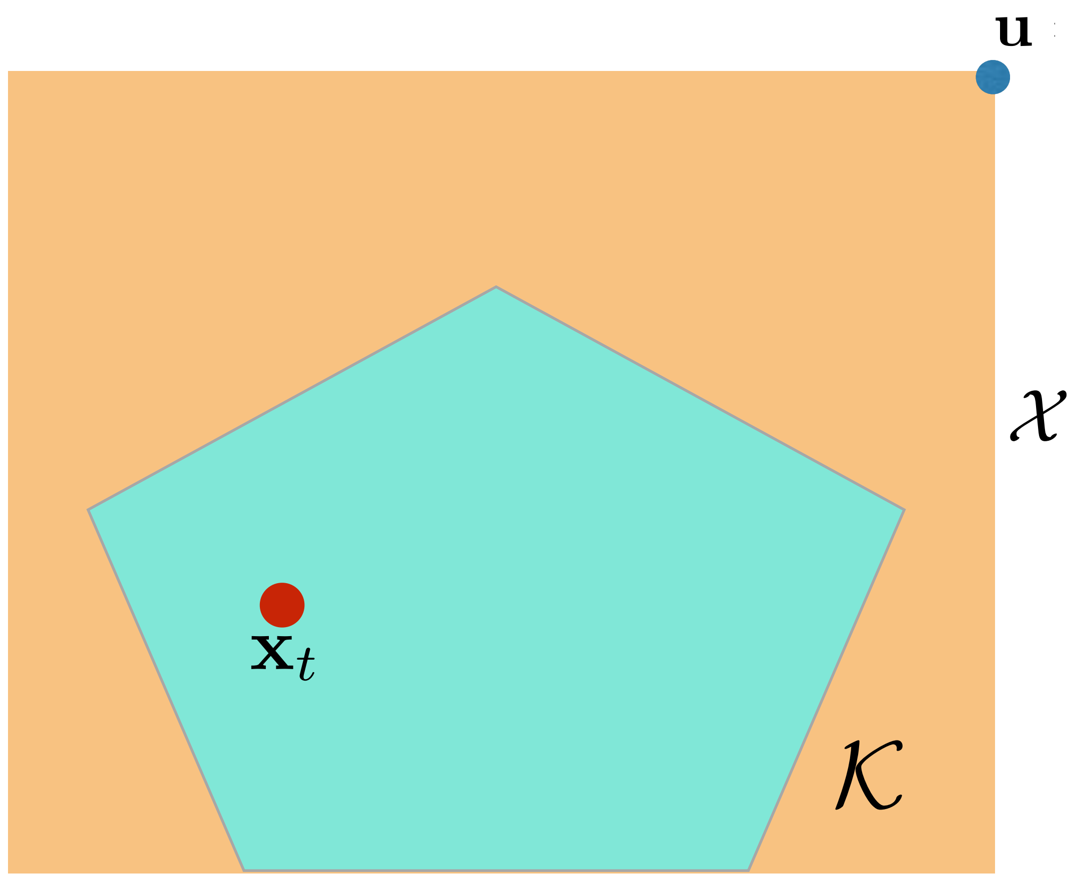
Yale

Penn

Non-monotone and DR-submodular

- measured continuous greedy:

> - Initialize at $\mathbf{x}_0 = \mathbf{0}$
> - Repeat for T iterations:
>
> $$\mathbf{v}_t = \arg\max_{\substack{\mathbf{v}\in\mathcal{K} \\ \mathbf{v} \leq \mathbf{u} - \mathbf{x}_t}} \langle \nabla F(\mathbf{x}_t), \mathbf{v} \rangle$$
>
> $$\mathbf{x}_{t+1} = \mathbf{x}_t + \frac{1}{T}\mathbf{v}_t$$



"A Unified Continuous Greedy Algorithm for Submodular Maximization", Feldmann, Noar, Schwartz, 2011

"Continuous DR-submodular Maximization: Structure and Algorithms", Bian, Levy, Krause, Buhmann, 2017

Yale Penn

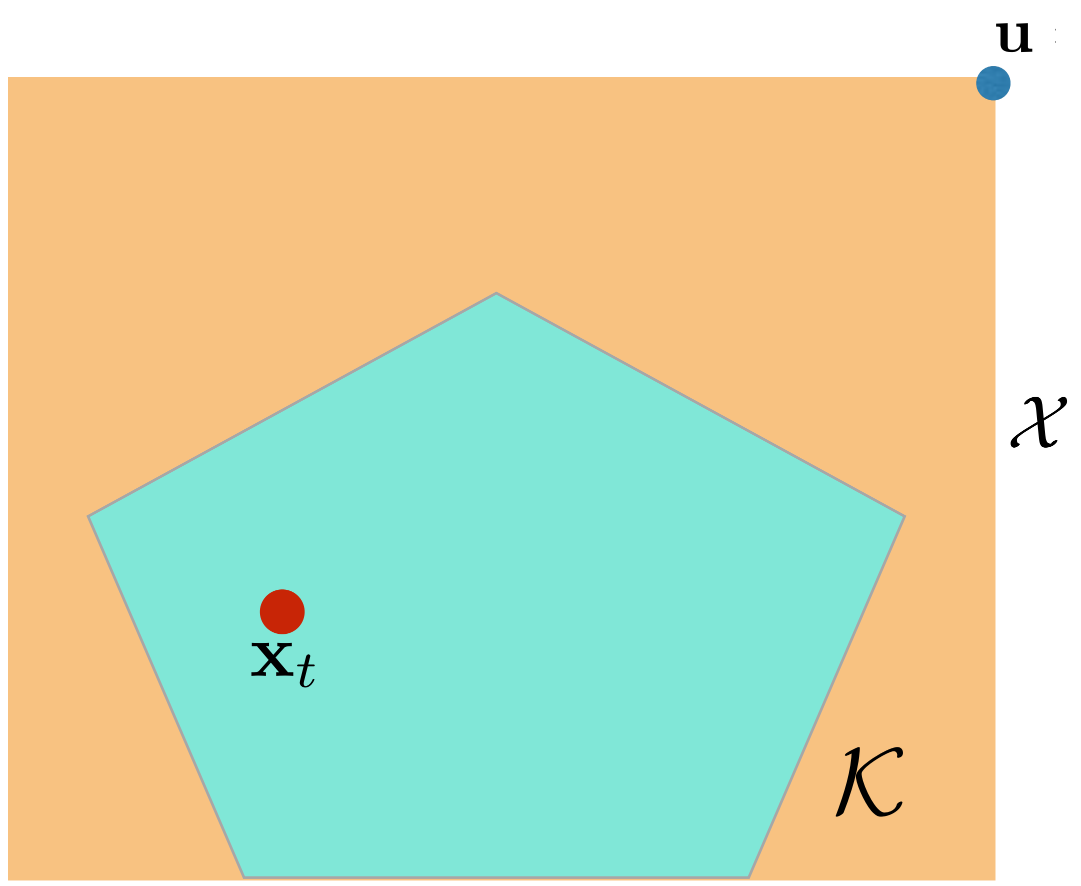
Non-monotone and DR-submodular

- measured continuous greedy:

  - Initialize at $\mathbf{x}_0 = \mathbf{0}$
  - Repeat for T iterations:

$$\mathbf{v}_t = \arg\max_{\substack{\mathbf{v}\in\mathcal{K} \\ \mathbf{v} \le \mathbf{u} - \mathbf{x}_t}} \langle \nabla F(\mathbf{x}_t), \mathbf{v} \rangle$$

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \frac{1}{T}\mathbf{v}_t$$



"A Unified Continuous Greedy Algorithm for Submodular Maximization", Feldmann, Noar, Schwartz, 2011

"Continuous DR-submodular Maximization: Structure and Algorithms", Bian, Levy, Krause, Buhmann, 2017

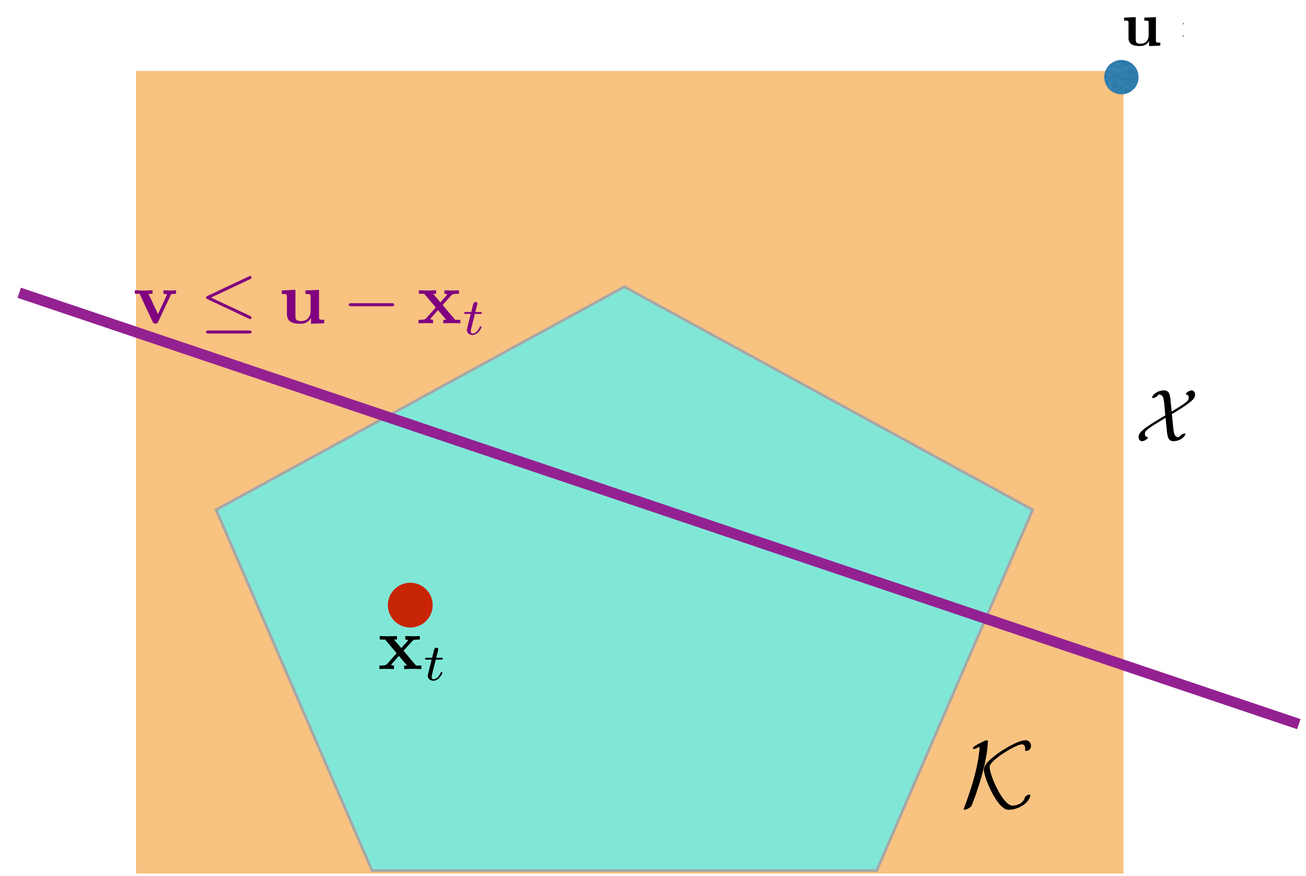
Non-monotone and DR-submodular

- measured continuous greedy:

  - Initialize at $\mathbf{x}_0 = \mathbf{0}$
  - Repeat for T iterations:

$$\mathbf{v}_t = \underset{\substack{\mathbf{v}\in\mathcal{K} \\ \mathbf{v}\leq \mathbf{u}-\mathbf{x}_t}}{\arg\max} \langle \nabla F(\mathbf{x}_t), \mathbf{v}\rangle$$

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \frac{1}{T}\mathbf{v}_t$$



"A Unified Continuous Greedy Algorithm for Submodular Maximization", Feldmann, Noar, Schwartz, 2011

"Continuous DR-submodular Maximization: Structure and Algorithms", Bian, Levy, Krause, Buhmann, 2017

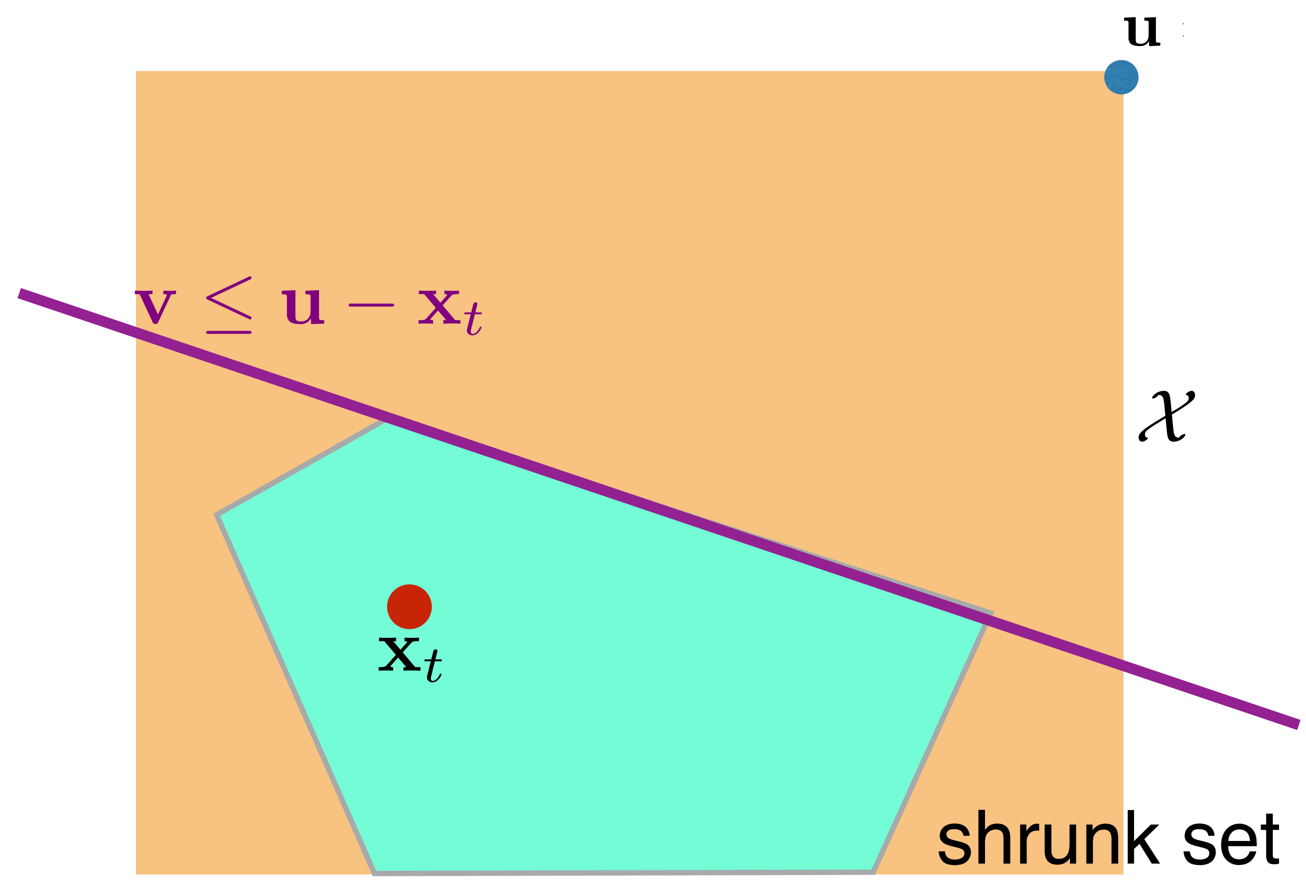
Non-monotone and DR-submodular

- measured continuous greedy:

- Initialize at $\mathbf{x}_0 = \mathbf{0}$
- Repeat for T iterations:

$$\mathbf{v}_t = \arg\max_{\substack{\mathbf{v}\in\mathcal{K} \\ \mathbf{v} \le \mathbf{u} - \mathbf{x}_t}} \langle \nabla F(\mathbf{x}_t), \mathbf{v} \rangle$$

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \frac{1}{T}\mathbf{v}_t$$



"A Unified Continuous Greedy Algorithm for Submodular Maximization", Feldmann, Noar, Schwartz, 2011

"Continuous DR-submodular Maximization: Structure and Algorithms", Bian, Levy, Krause, Buhmann, 2017

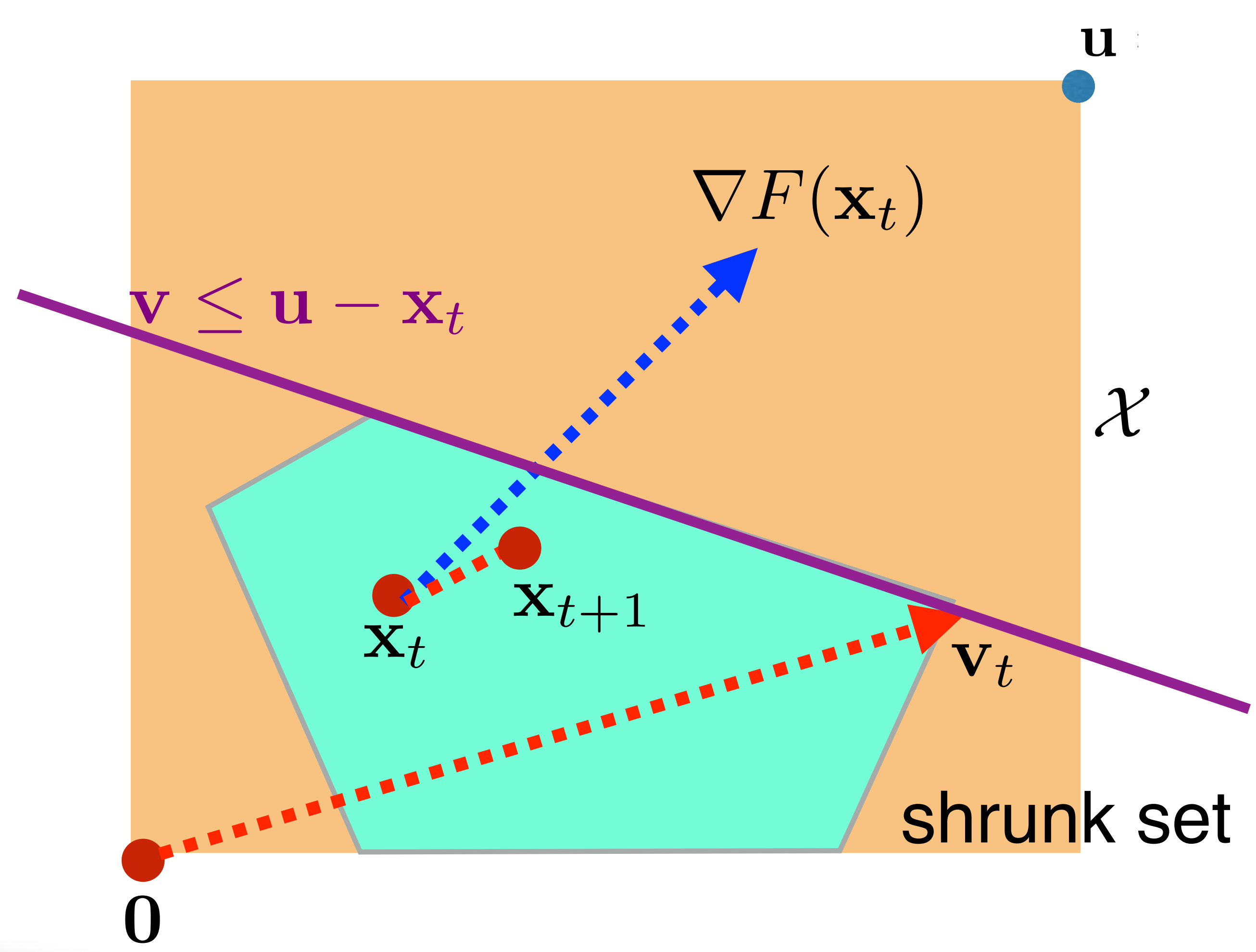
Non-monotone and DR-submodular

- measured continuous greedy:

- Initialize at $\mathbf{x}_0 = \mathbf{0}$
- Repeat for T iterations:

$$\mathbf{v}_t = \arg\max_{\substack{\mathbf{v} \in \mathcal{K} \\ \mathbf{v} \leq \mathbf{u} - \mathbf{x}_t}} \langle \nabla F(\mathbf{x}_t), \mathbf{v} \rangle$$

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \frac{1}{T}\mathbf{v}_t$$



[Bian, Levy, Krause, Buhmann]

The measured continuous greedy algorithm achieves $\frac{1}{e}\mathrm{OPT}$ for constrained maximization of non-monotone DR-submodular functions

"Continuous DR-submodular Maximization: Structure and Algorithms", NIPS'17

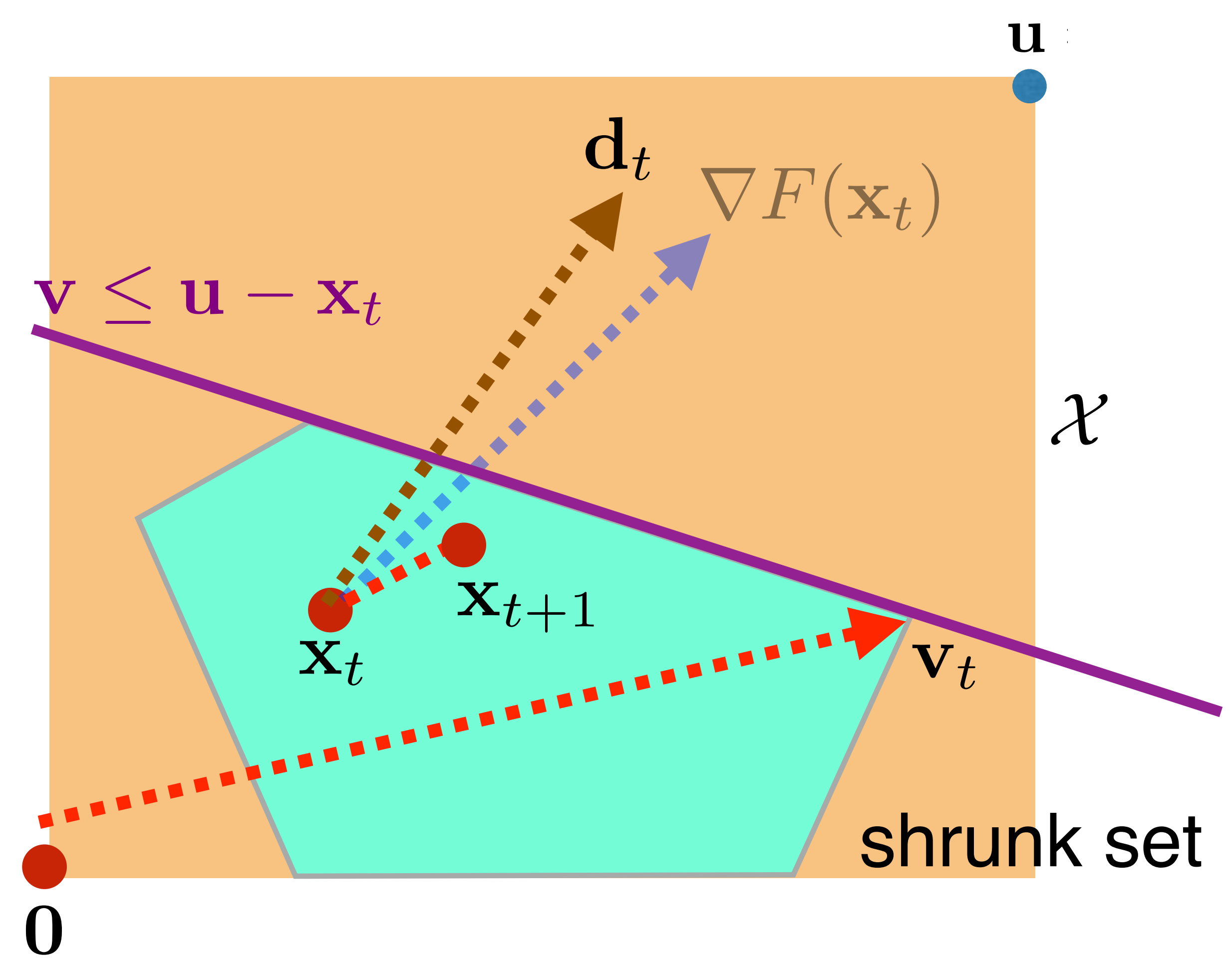
Non-monotone and DR-submodular

- Initialize at $\mathbf{x}_0 = \mathbf{0}$
- Repeat for T iterations:

$$\mathbf{v}_t = \arg\max_{\substack{\mathbf{v}\in\mathcal{K} \\ \mathbf{v} \leq \mathbf{1} - \mathbf{x}_t}} \langle \nabla F(\mathbf{x}_t), \mathbf{v} \rangle$$

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \frac{1}{T}\mathbf{v}_t$$

$$\mathbf{d}_{t+1} = (1-\rho_t)\,\mathbf{d}_t + \rho_t\,\mathbf{g}_t$$



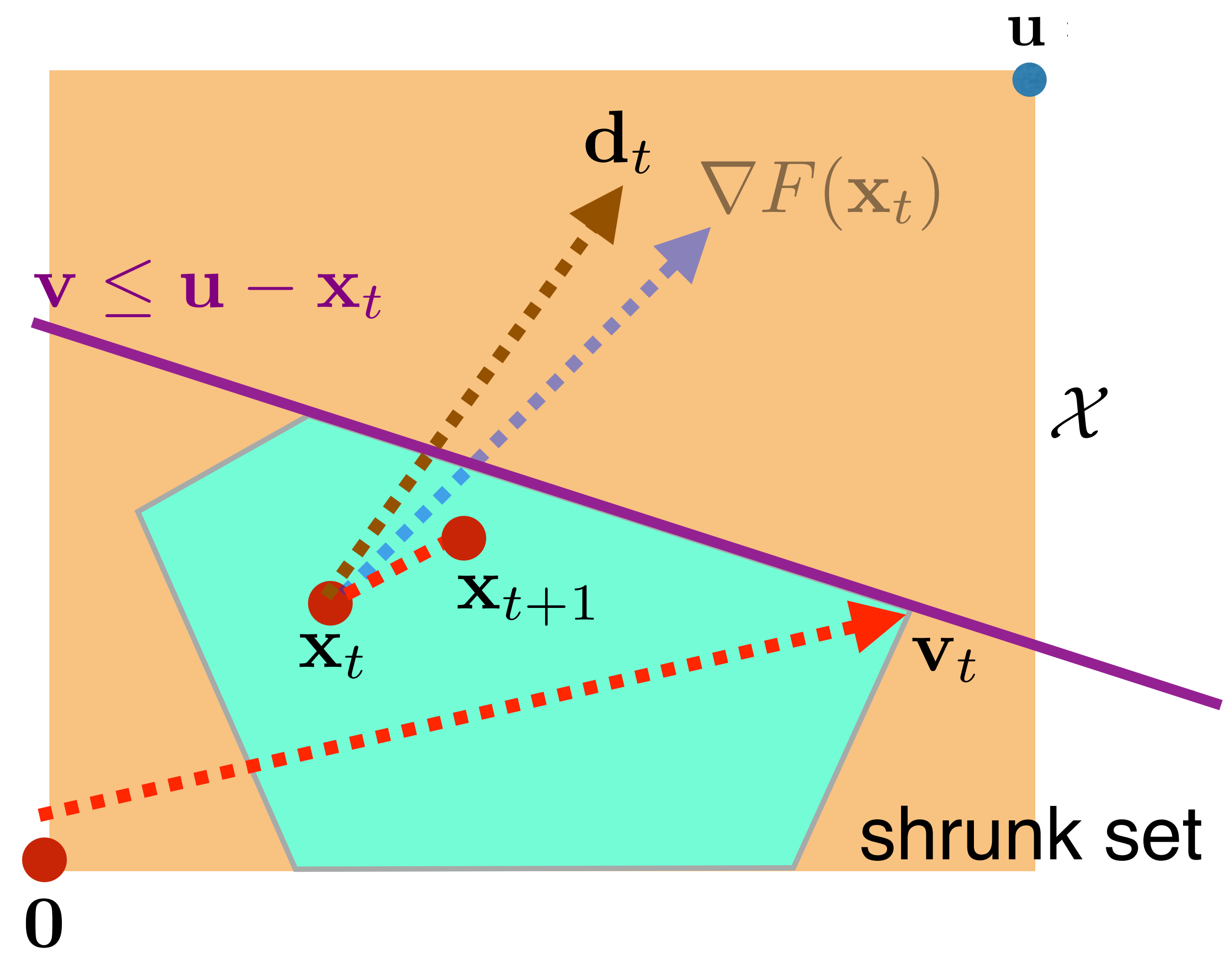
Non-monotone and DR-submodular

- Initialize at $\mathbf{x}_0 = \mathbf{0}$
- Repeat for T iterations:

$$\mathbf{v}_t = \underset{\substack{\mathbf{v} \in \mathcal{K} \\ \mathbf{v} \leq \mathbf{1} - \mathbf{x}_t}}{\arg\max} \langle \nabla F(\mathbf{x}_t), \mathbf{v} \rangle$$

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \frac{1}{T}\mathbf{v}_t$$

$$\mathbf{d}_{t+1} = (1 - \rho_t)\,\mathbf{d}_t + \rho_t\,\mathbf{g}_t$$



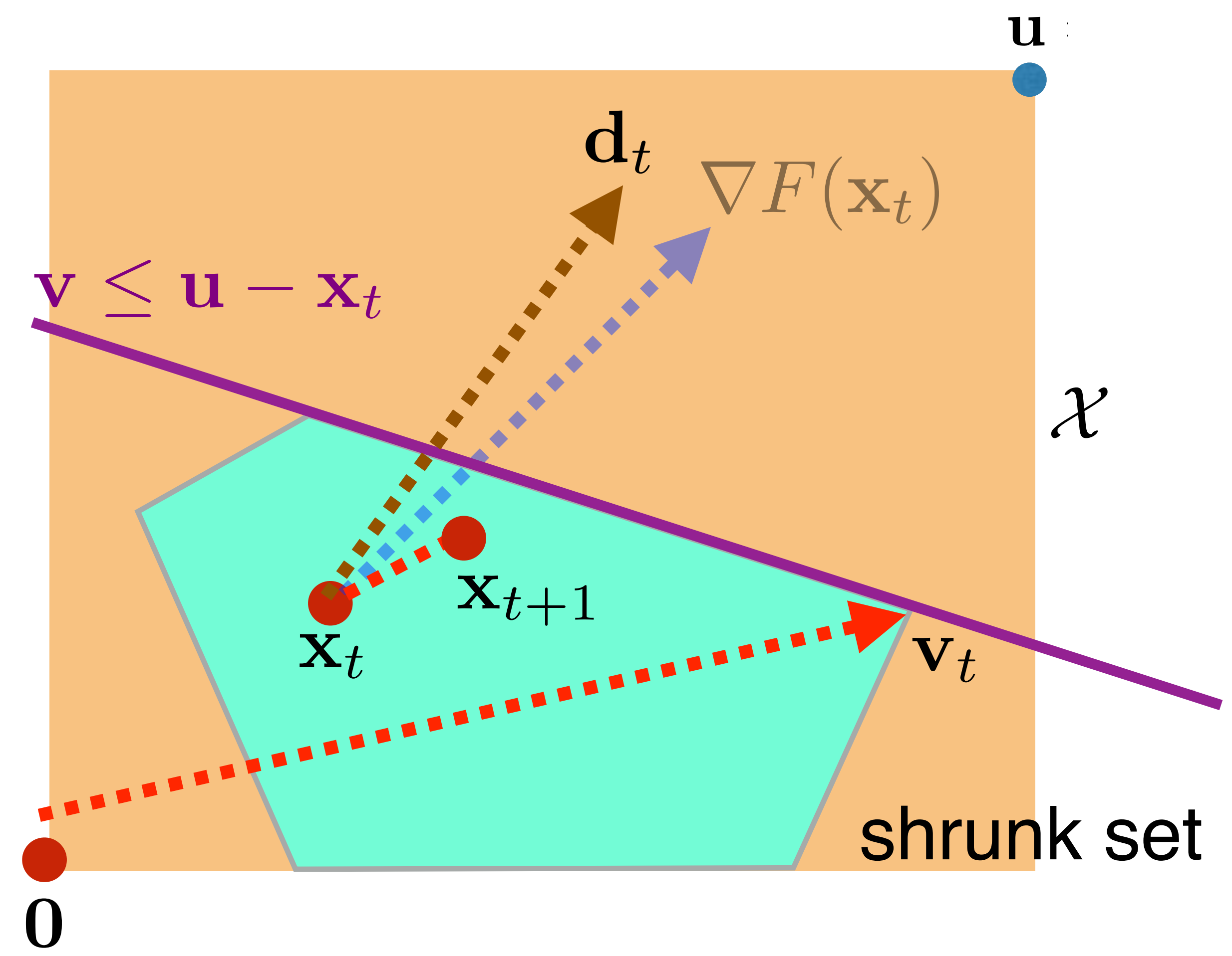
Non-monotone and DR-submodular

- Initialize at $\mathbf{x}_0 = \mathbf{0}$
- Repeat for T iterations:

$$\mathbf{v}_t = \arg\max_{\substack{\mathbf{v}\in\mathcal{K} \\ \mathbf{v} \le \mathbf{1} - \mathbf{x}_t}} \langle \nabla F(\mathbf{x}_t), \mathbf{v} \rangle$$

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \frac{1}{T}\mathbf{v}_t$$

$$\mathbf{d}_{t+1} = (1-\rho_t)\,\mathbf{d}_t + \rho_t\,\mathbf{g}_t$$



[Hassani, Karbasi, Mokhtari, Shen]

The stochastic measured continuous greedy algorithm achieves $\frac{1}{e}\mathrm{OPT} - \epsilon$ with sample complexity $O(\frac{1}{\epsilon^2})$

"Stochastic Conditional Gradient Methods: From Convex Minimization to Submodular Maximization", JMLR '20

"Stochastic Conditional Gradient++: (Non-)Convex Minimization and Continuous Submodular Maximization", preprint '20

- Relevant work:

"Non-monotone DR-submodular Maximization: Approximation and Regret Guarantees", Dürr, Nguyen Kim Thang, Abhinav Srivastav, Léo Tible, 2019

Non-monotone and DR-submodular

- Constrained case:

Non-monotone and DR-submodular

- Constrained case:

Non-monotone and DR-submodular

- Constrained case:

- Unconstrained case:

[Bian, Buhmann, Krause]

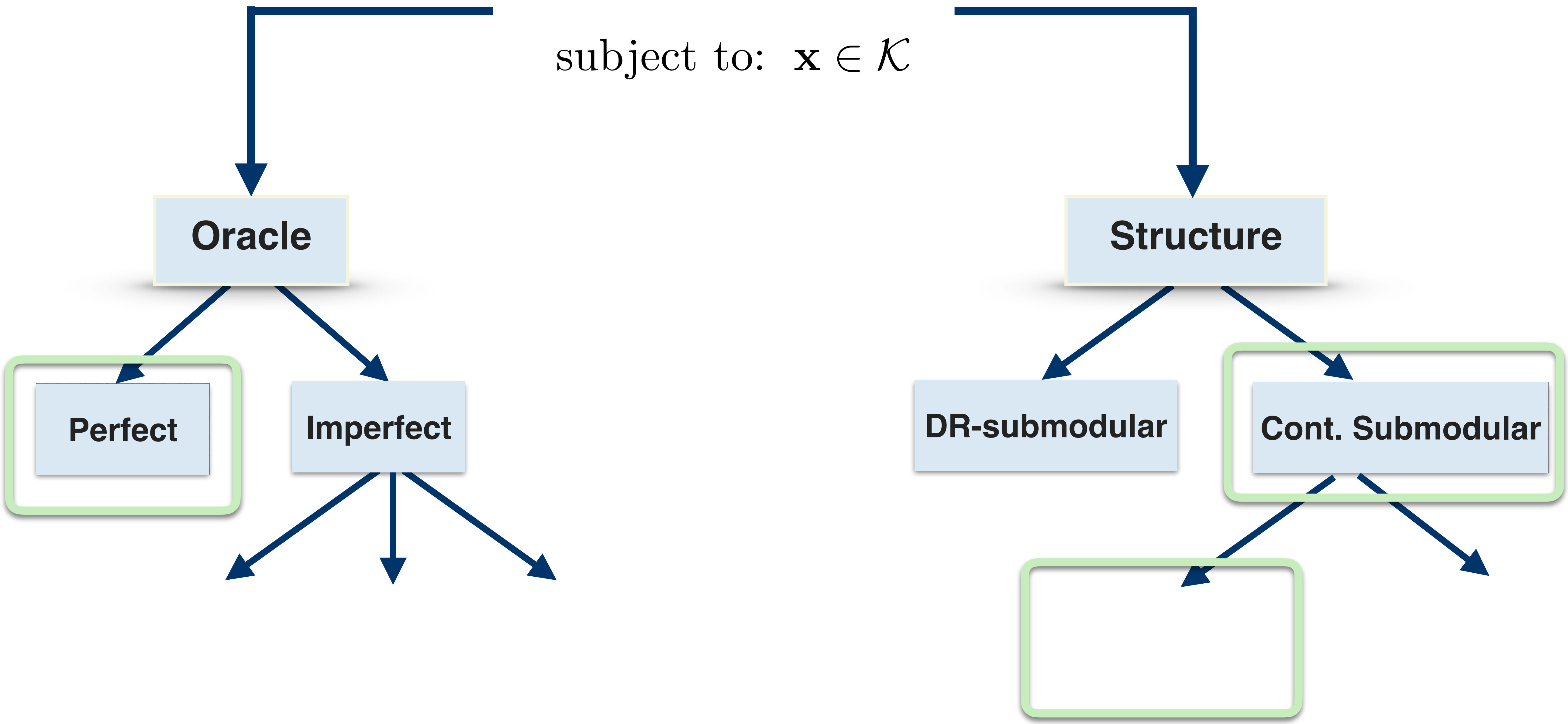
For the the case where $\mathcal{K}$ is a box, an extension of the double-greedy method (covered in part 1) achieves a tight guarantee $\frac{1}{2}\mathrm{OPT}$.

"Optimal Continuous DR-Submodular Maximization and Applications to Provable Mean Field Inference", JMLR '20

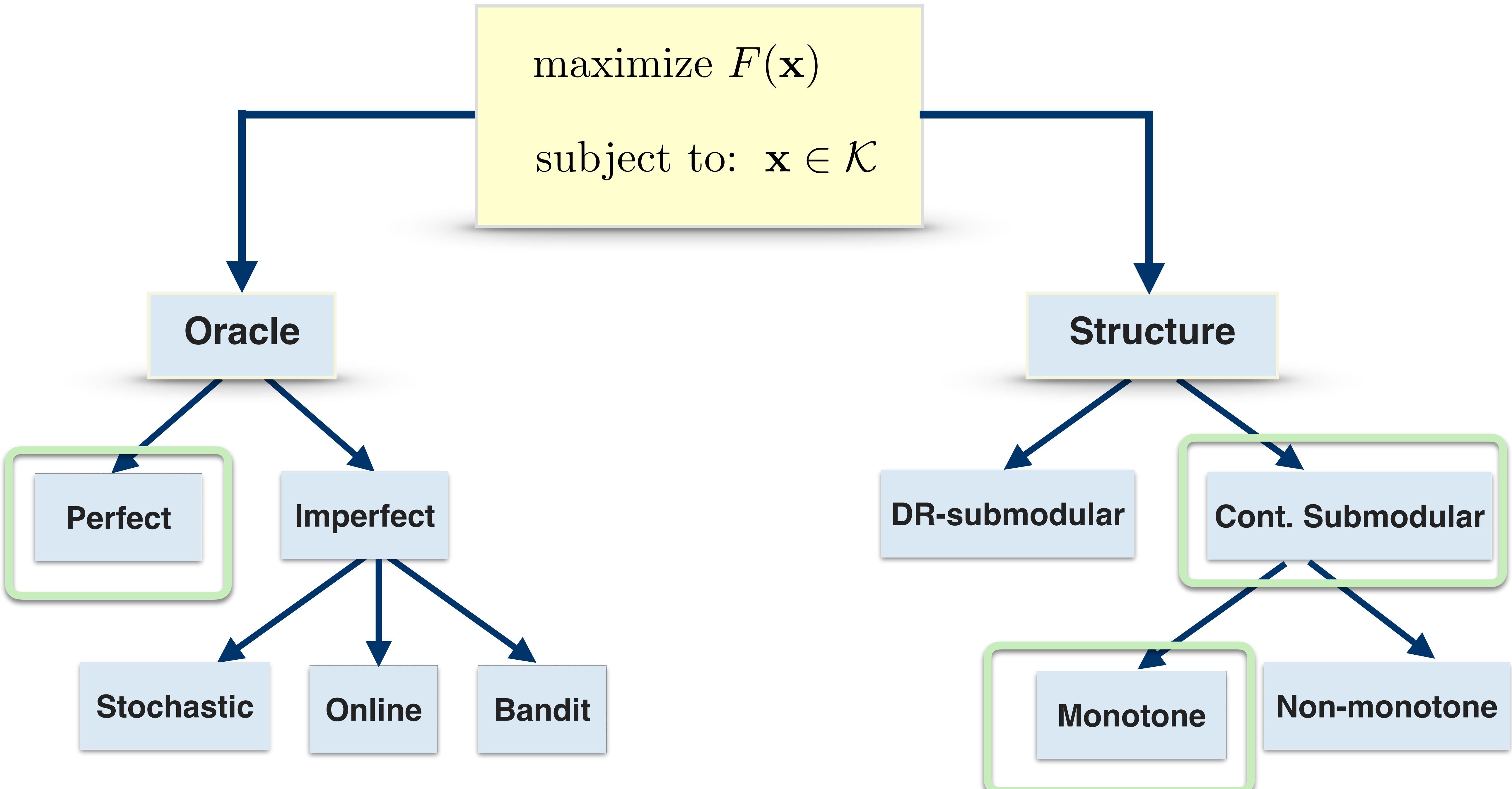
Yale Penn

Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } \mathbf{x} \in \mathcal{K}$$



Submodular Maximization



Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$

$$\text{subject to: } \mathbf{x} \in \mathcal{K}$$

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$

$$\text{subject to: } \mathbf{x} \in \mathcal{K}$$

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } \mathbf{x} \in \mathcal{K}$$

- Very recent progress: Coordinate-ascent methods

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$

$$\text{subject to: } ||\mathbf{x}||_1 \leq B$$

- Very recent progress: Coordinate-ascent methods

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$

$$\text{subject to: } ||\mathbf{x}||_1 \leq B$$

- Basic idea:

while the budget B is not exhausted

for each coordinate i:

find: $\alpha_i = \arg\max_{\alpha \geq 0} \dfrac{F(\mathbf{x} + \alpha \mathbf{e}_i) - F(\mathbf{x})}{\alpha}$

choose coordinate i^* with the largest α_{i^*}

update $\mathbf{x} = \mathbf{x} + \alpha_{i^*} e_{i^*}$

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } ||\mathbf{x}||_1 \leq B$$

- Basic idea:

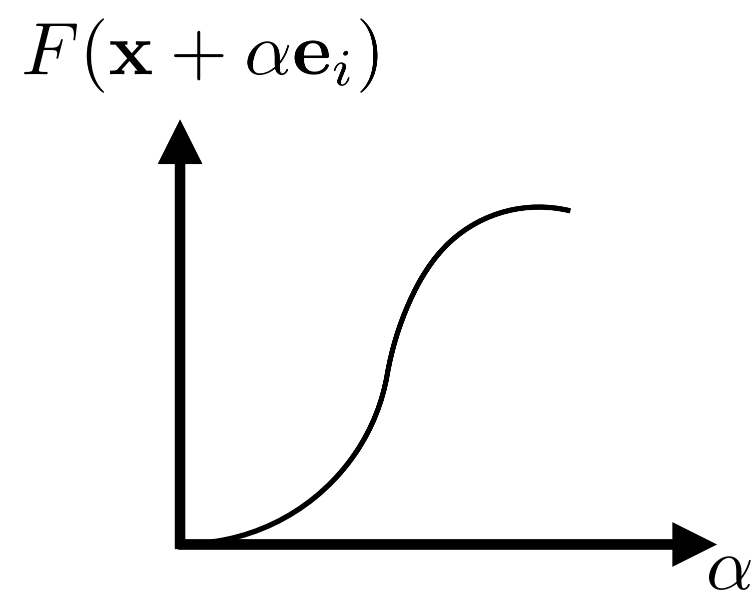
while the budget B is not exhausted

for each coordinate i:

find: $\alpha_i = \arg\max_{\alpha \geq 0} \frac{F(\mathbf{x} + \alpha \mathbf{e}_i) - F(\mathbf{x})}{\alpha}$

choose coordinate i^* with the largest α_{i^*}

update $\mathbf{x} = \mathbf{x} + \alpha_{i^*} e_{i^*}$



Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } ||\mathbf{x}||_1 \leq B$$

- Basic idea:

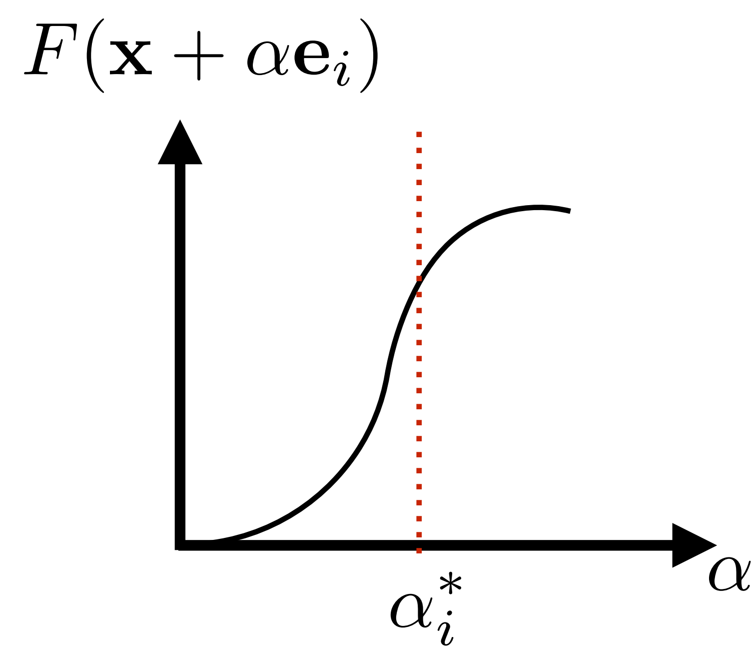
while the budget B is not exhausted

for each coordinate i:

find: $\alpha_i = \arg\max_{\alpha \geq 0} \frac{F(\mathbf{x} + \alpha \mathbf{e}_i) - F(\mathbf{x})}{\alpha}$

choose coordinate i^* with the largest α_{i^*}

update $\mathbf{x} = \mathbf{x} + \alpha_{i^*} e_{i^*}$



Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } ||\mathbf{x}||_1 \leq B$$

- Basic idea:

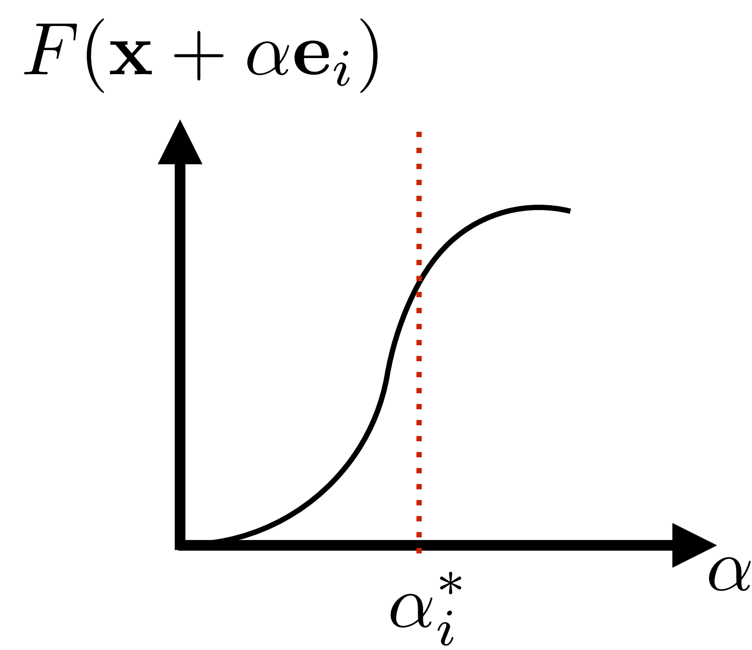
while the budget B is not exhausted

for each coordinate i:

find: $\alpha_i = \arg\max_{\alpha \geq 0} \frac{F(\mathbf{x} + \alpha \mathbf{e}_i) - F(\mathbf{x})}{\alpha}$

choose coordinate i^* with the largest α_{i^*}

update $\mathbf{x} = \mathbf{x} + \alpha_{i^*} e_{i^*}$



[Feldmann, Karbasi]

Coordinate descent methods obtain a $(1 - \frac{1}{e} - \epsilon)$ OPT solution to the above problem with computational complexity $O(\frac{n^3}{\epsilon^{2.5}})$

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$

$$\text{subject to: } \mathbf{x} \in \mathcal{K}$$

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$

$$\text{subject to: } \mathbf{x} \in \mathcal{K}$$

Continuous Submodular Maximization

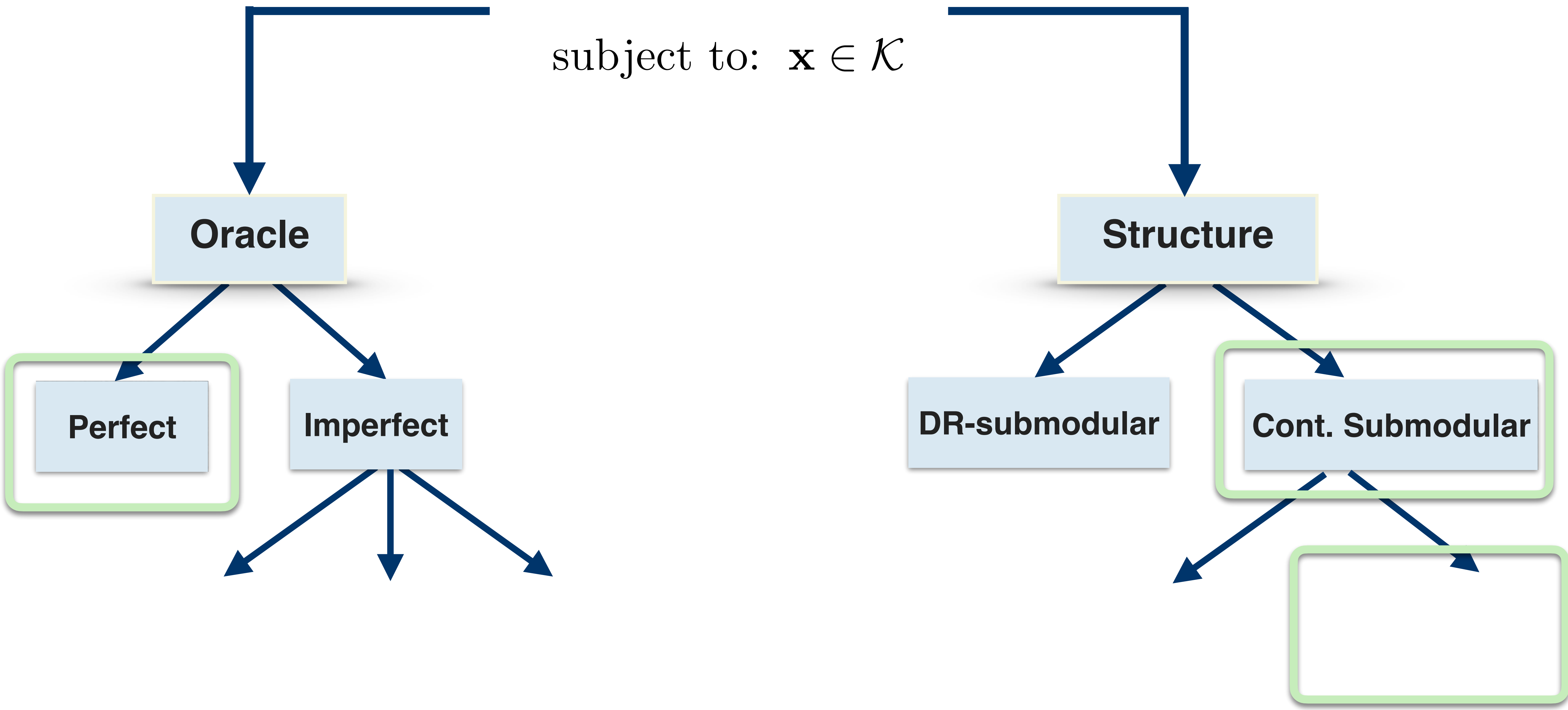
$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } \mathbf{x} \in \mathcal{K}$$

- Many open questions:
  General convex constraints, stochastic, better algorithms, etc

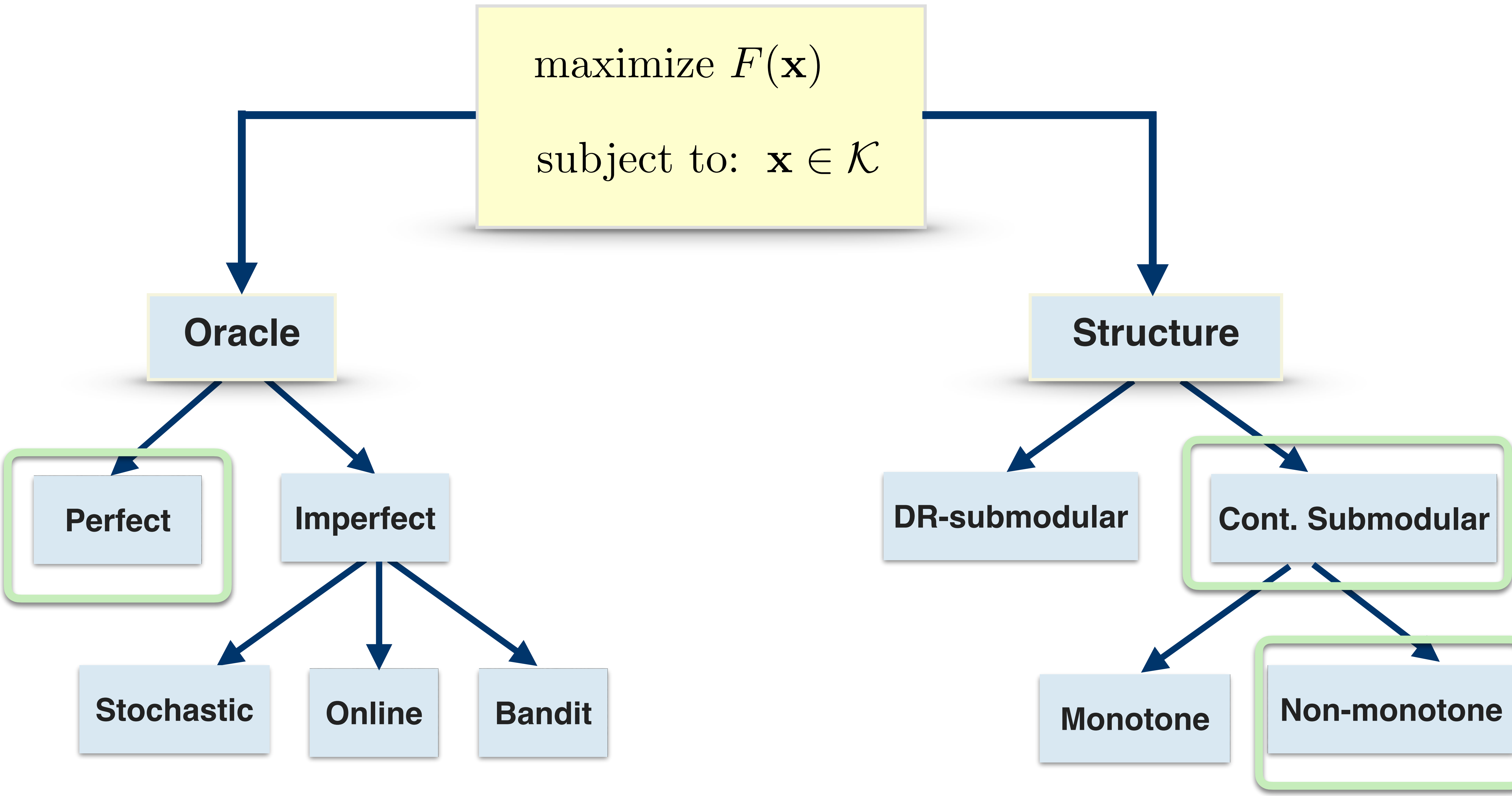
Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$

$$\text{subject to: } \mathbf{x} \in \mathcal{K}$$



Submodular Maximization



Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$

$$\text{subject to: } \mathbf{x} \in \mathcal{K}$$

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$

$$\text{subject to: } \mathbf{x} \in [0,1]^n$$

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } \mathbf{x} \in [0,1]^n$$

- Initialize $\mathbf{x} = (0, 0, \cdots, 0)$
  $\mathbf{y} = (1, 1, \cdots, 1)$
- For $i = 1$ to n

  Find $z_i \in [0,1]$ as the value of the final solution at the i-th coordinate

"Optimal Algorithms for Continuous Non-monotone Submodular and DR-Submodular Maximization", Niazadeh, Roughgarden, Wang '18

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } \mathbf{x} \in [0,1]^n$$

- Initialize $\mathbf{x} = (0, 0, \cdots, 0)$
  $\mathbf{y} = (1, 1, \cdots, 1)$
- For $i = 1$ to n

  Find $z_i \in [0,1]$ as the value of the final solution at the i-th coordinate

“Optimal Algorithms for Continuous Non-monotone Submodular and DR-Submodular Maximization ”, Niazadeh, Roughgarden, Wang ‘18

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } \mathbf{x} \in [0,1]^n$$

- Initialize $\begin{array}{l}\mathbf{x} = (0, 0, \cdots, 0) \\ \mathbf{y} = (1, 1, \cdots, 1)\end{array}$
- For $i = 1 \text{ to } n$

  Find $z_i \in [0,1]$ as the value of the final solution at the i-th coordinate

  $$\mathbf{x} = (z_1, 0, \cdots, 0)$$
  $$\mathbf{y} = (z_1, 1, \cdots, 1)$$

"Optimal Algorithms for Continuous Non-monotone Submodular and DR-Submodular Maximization ", Niazadeh, Roughgarden, Wang '18

Yale Penn

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } \mathbf{x} \in [0,1]^n$$

- Initialize $\mathbf{x} = (0, 0, \cdots, 0)$
  $\mathbf{y} = (1, 1, \cdots, 1)$
- For $i = 1$ to n

  Find $z_i \in [0,1]$ as the value of the final solution at the i-th coordinate

$$\mathbf{x} = (z_1, z_2, 0, \cdots, 0)$$
$$\mathbf{y} = (z_1, z_2, 1, \cdots, 1)$$

"Optimal Algorithms for Continuous Non-monotone Submodular and DR-Submodular Maximization ", Niazadeh, Roughgarden, Wang '18

Yale

Penn

Continuous Submodular Maximization

maximize $F(\mathbf{x})$

subject to: $\mathbf{x} \in [0,1]^n$

- Initialize $\mathbf{x} = (0, 0, \cdots, 0)$
  $\mathbf{y} = (1, 1, \cdots, 1)$
- For $i = 1$ to n

  Find $z_i \in [0,1]$ as the value of the final solution at the i-th coordinate

$$\mathbf{x} = (z_1, z_2, \cdots, z_{i-1}, z_i, 0, \cdots, 0)$$
$$\mathbf{y} = (z_1, z_2, \cdots, z_{i-1}, z_i, 1, \cdots, 1)$$

"Optimal Algorithms for Continuous Non-monotone Submodular and DR-Submodular Maximization ", Niazadeh, Roughgarden, Wang '18

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } \mathbf{x} \in [0,1]^n$$

- Initialize $\mathbf{x} = (0, 0, \cdots, 0)$
  $\mathbf{y} = (1, 1, \cdots, 1)$
- For $i = 1$ to n

  Find $z_i \in [0,1]$ as the value of the final solution at the i-th coordinate

  $$\mathbf{x} = (z_1, z_2, \cdots, z_{i-1}, z_i, 0, \cdots, 0)$$
  $$\mathbf{y} = (z_1, z_2, \cdots, z_{i-1}, z_i, 1, \cdots, 1)$$

  z_i is found by drawing an analogy to a two-player zero-sum game

"Optimal Algorithms for Continuous Non-monotone Submodular and DR-Submodular Maximization ", Niazadeh, Roughgarden, Wang '18

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } \mathbf{x} \in [0,1]^n$$

- Initialize $\mathbf{x} = (0, 0, \cdots, 0)$
  $\mathbf{y} = (1, 1, \cdots, 1)$
- For $i = 1$ to n

  Find $z_i \in [0,1]$ as the value of the final solution at the i-th coordinate

  $$\mathbf{x} = (z_1, z_2, \cdots, z_{i-1}, z_i, 0, \cdots, 0)$$
  $$\mathbf{y} = (z_1, z_2, \cdots, z_{i-1}, z_i, 1, \cdots, 1)$$

  z_i is found by drawing an analogy to a two-player zero-sum game

- Output: $(z_1, z_2, \cdots, z_n)$

"Optimal Algorithms for Continuous Non-monotone Submodular and DR-Submodular Maximization ", Niazadeh, Roughgarden, Wang '18

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$

$$\text{subject to: } \mathbf{x} \in \mathcal{K}$$

[Niazadeh, Roughgarden, Wang]

For the the case where $\mathcal{K}$ is a box, the above algorithm achieves a tight guarantee $\frac{1}{2}\text{OPT} - \epsilon$ with complexity $O(\frac{n^2}{\epsilon})$.

"Optimal Algorithms for Continuous Non-monotone Submodular and DR-Submodular Maximization ", NIPS '18

Continuous Submodular Maximization

$$\text{maximize } F(\mathbf{x})$$
$$\text{subject to: } \mathbf{x} \in \mathcal{K}$$

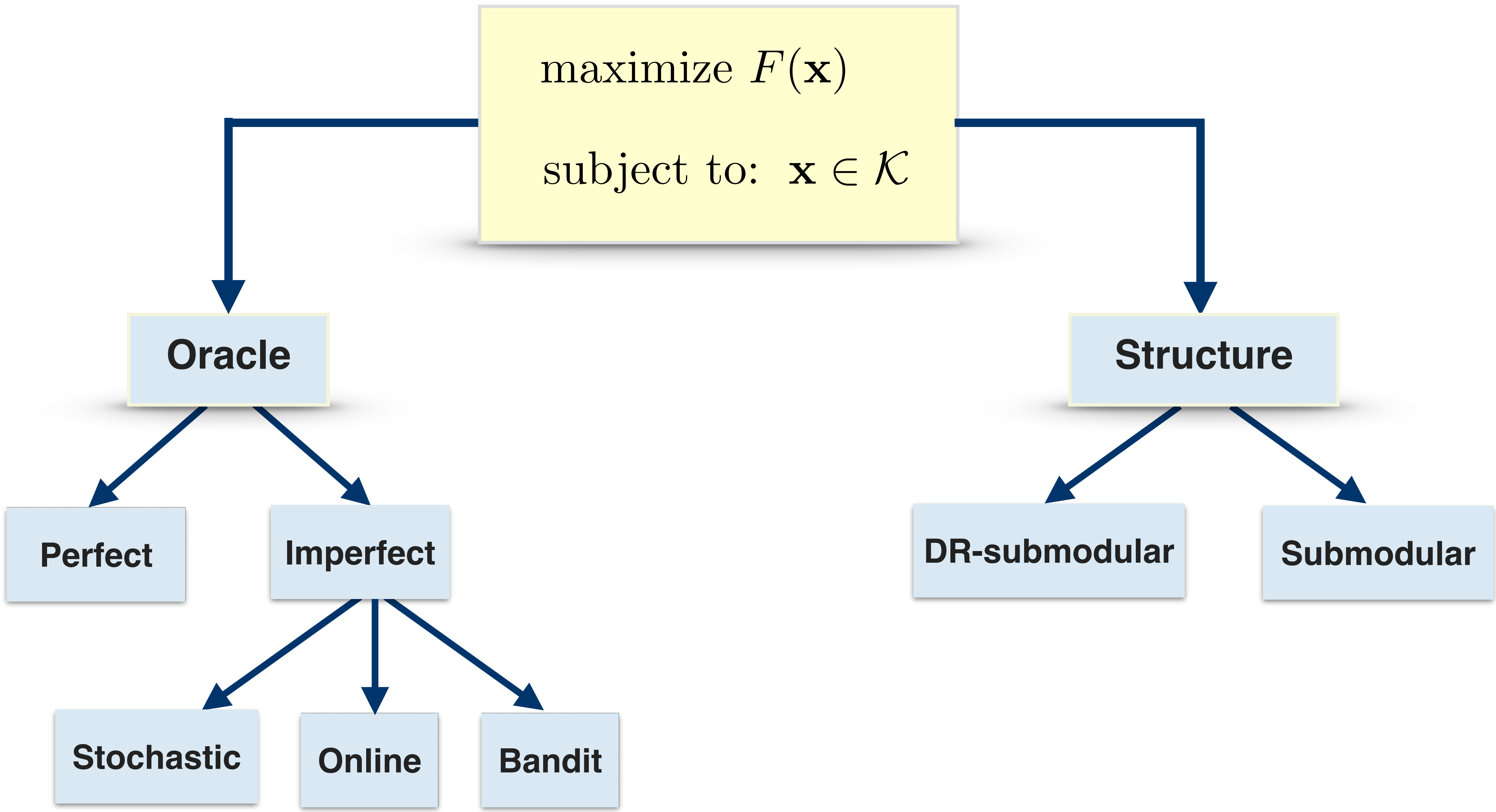
[Niazadeh, Roughgarden, Wang]

For the the case where $\mathcal{K}$ is a box, the above algorithm achieves a tight guarantee $\frac{1}{2}\mathrm{OPT} - \epsilon$ with complexity $O(\frac{n^2}{\epsilon})$.

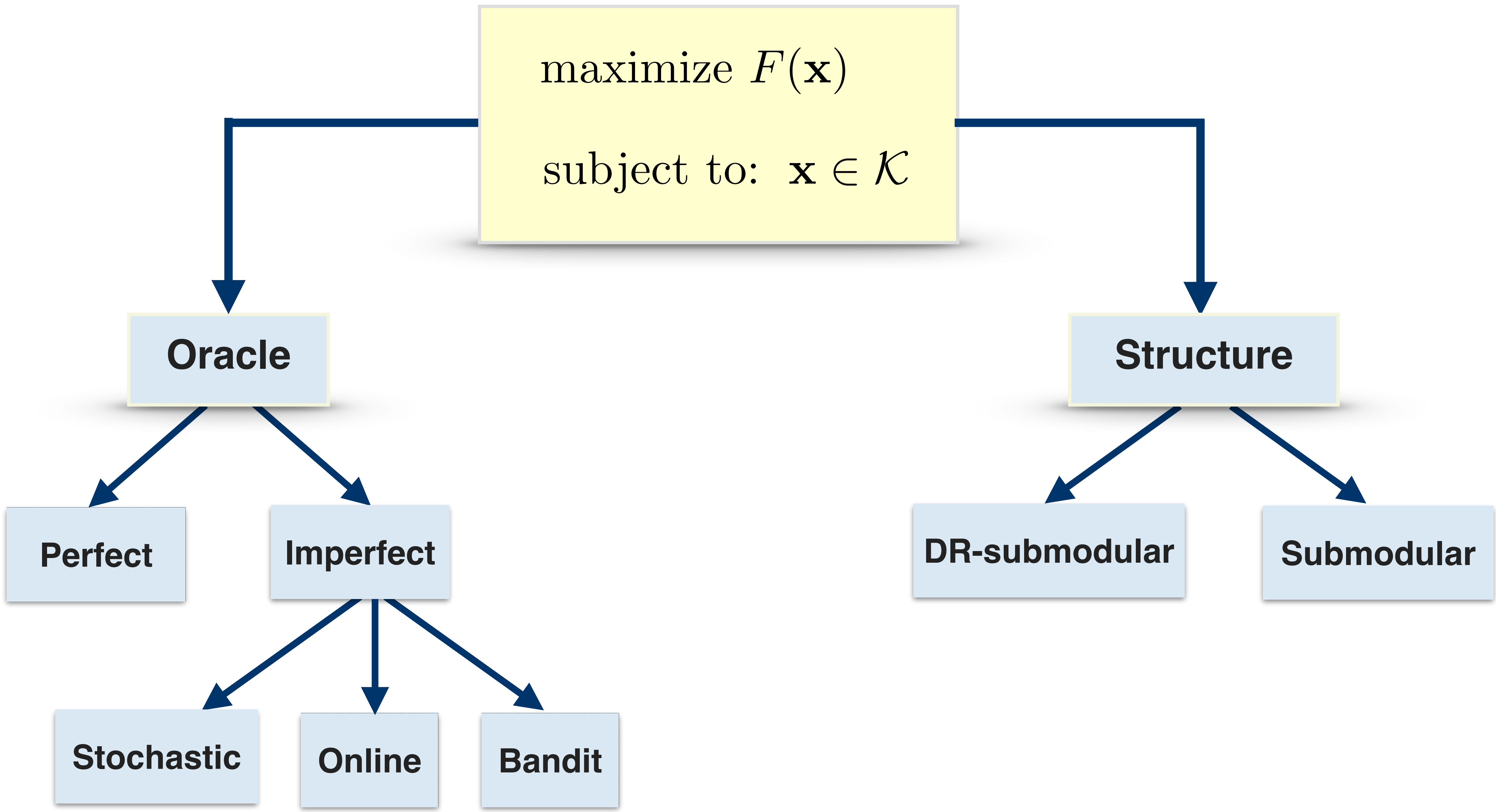
"Optimal Algorithms for Continuous Non-monotone Submodular and DR-Submodular Maximization ", NIPS '18

- Many open questions:
non-monotone and constrained, stochastic, etc

Submodular Maximization

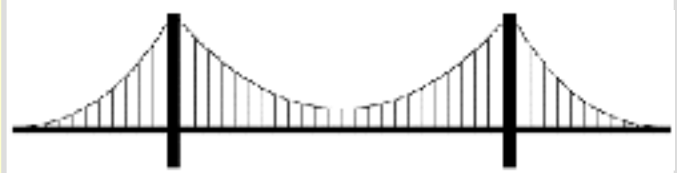


Submodular Maximization



Bridging Discrete and Continuous Settings

$$\max_{S \in \mathcal{I}} f(S)$$



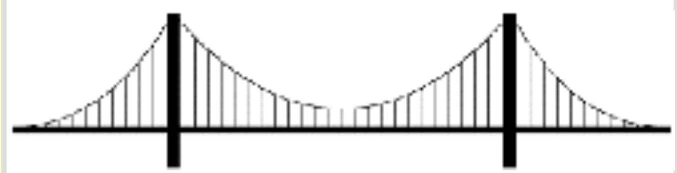
$$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

"Maximizing a monotone submodular function subject to a matroid constraint", Calinescu, Chekuri, Pal, Vondrák, 2011

Bridging Discrete and Continuous Settings

submodular

$$\max_{S \in \mathcal{I}} f(S)$$

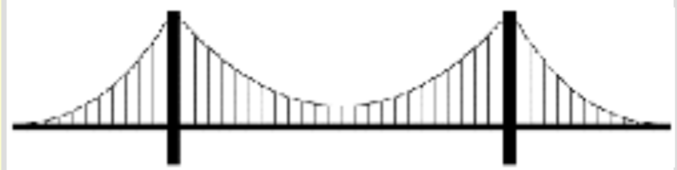


$$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

Bridging Discrete and Continuous Settings

submodular

$$\max_{S \in \mathcal{I}} f(S)$$



DR-submodular

$$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

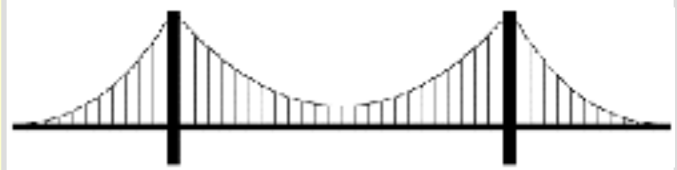
"Maximizing a monotone submodular function subject to a matroid constraint", Calinescu, Chekuri, Pal, Vondrák, 2011

Bridging Discrete and Continuous Settings

submodular

$$\max_{S \in \mathcal{I}} f(S)$$

cardinality
matroid
knapsack



DR-submodular

$$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

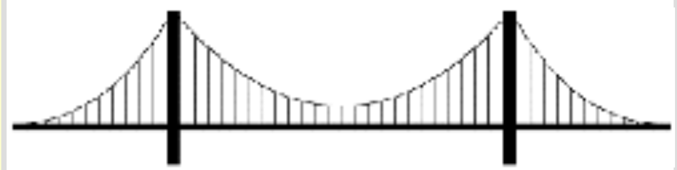
"Maximizing a monotone submodular function subject to a matroid constraint", Calinescu, Chekuri, Pal, Vondrák, 2011

Bridging Discrete and Continuous Settings

submodular

$$\max_{S \in \mathcal{I}} f(S)$$

cardinality
matroid
knapsack



DR-submodular

$$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

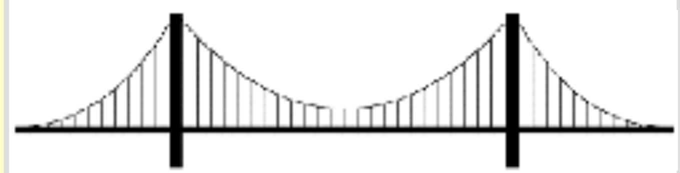
polytope

“Maximizing a monotone submodular function subject to a matroid constraint”, Calinescu, Chekuri, Pal, Vondrák, 2011

Bridging Discrete and Continuous Settings

submodular

$$\max_{S \in \mathcal{I}} f(S)$$



DR-submodular

$$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

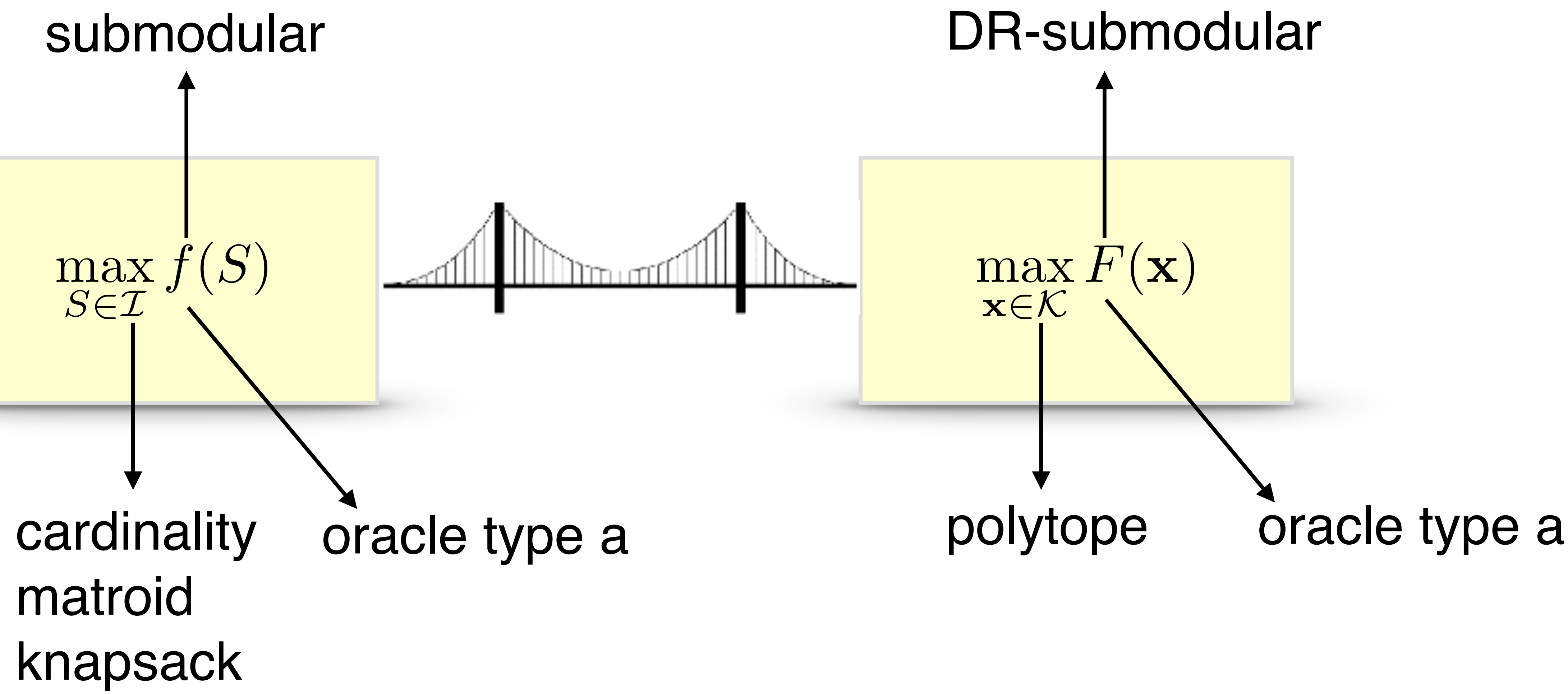
cardinality
matroid
knapsack

oracle type a

polytope

"Maximizing a monotone submodular function subject to a matroid constraint", Calinescu, Chekuri, Pal, Vondrák, 2011

Bridging Discrete and Continuous Settings



"Maximizing a monotone submodular function subject to a matroid constraint", Calinescu, Chekuri, Pal, Vondrák, 2011

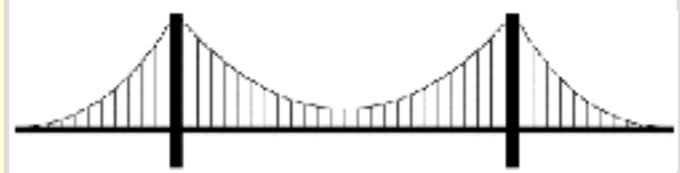
Bridging Discrete and Continuous Settings

submodular

$$\max_{S \in \mathcal{I}} f(S)$$

cardinality
matroid
knapsack

oracle type a



DR-submodular

$$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

polytope

oracle type a

Building blocks:

- Multi-linear extension
- Equivalent formulations
- Rounding

"Maximizing a monotone submodular function subject to a matroid constraint", Calinescu, Chekuri, Pal, Vondrák, 2011

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ $\Rightarrow$ $F : \mathcal{X} \to \mathbb{R}$

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ ➡ $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$

Yale Penn

Multi-linear Extension

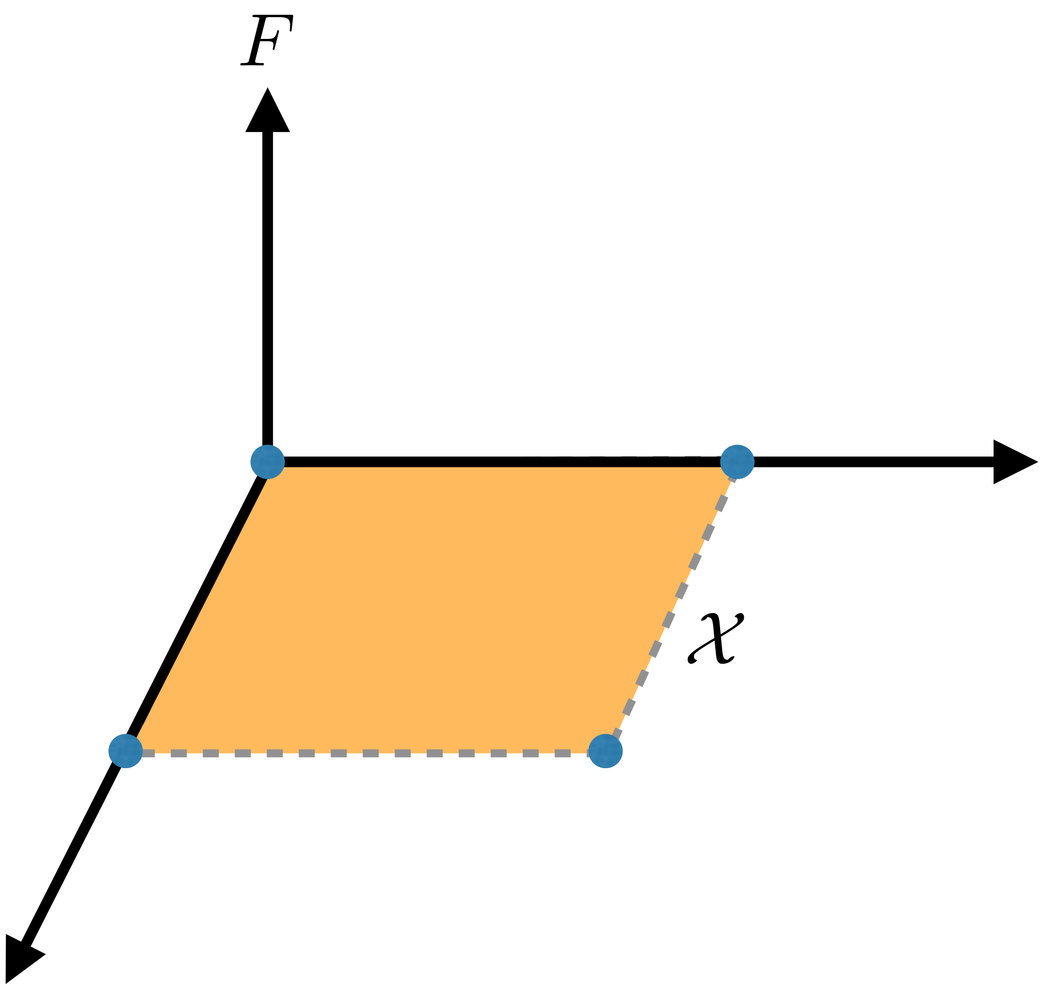
$f : 2^V \to \mathbb{R}$ $\Rightarrow$ $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\mathcal{X} = [0, 1]^n$

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ $\longrightarrow$ $F : \mathcal{X} \to \mathbb{R}$

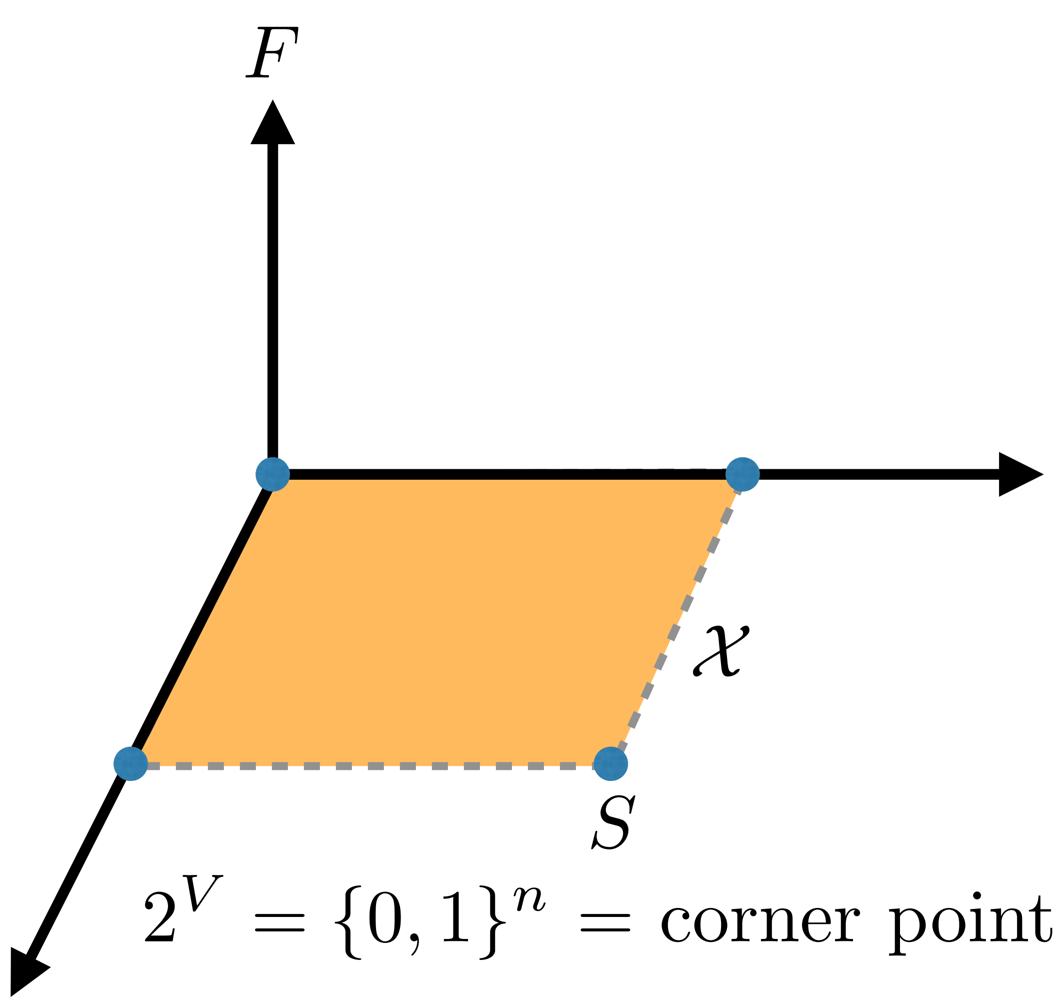
$V = \{1, 2, \cdots, n\}$ $\qquad$ $\mathcal{X} = [0, 1]^n$



Multi-linear Extension

$f : 2^V \to \mathbb{R}$ → $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\mathcal{X} = [0, 1]^n$

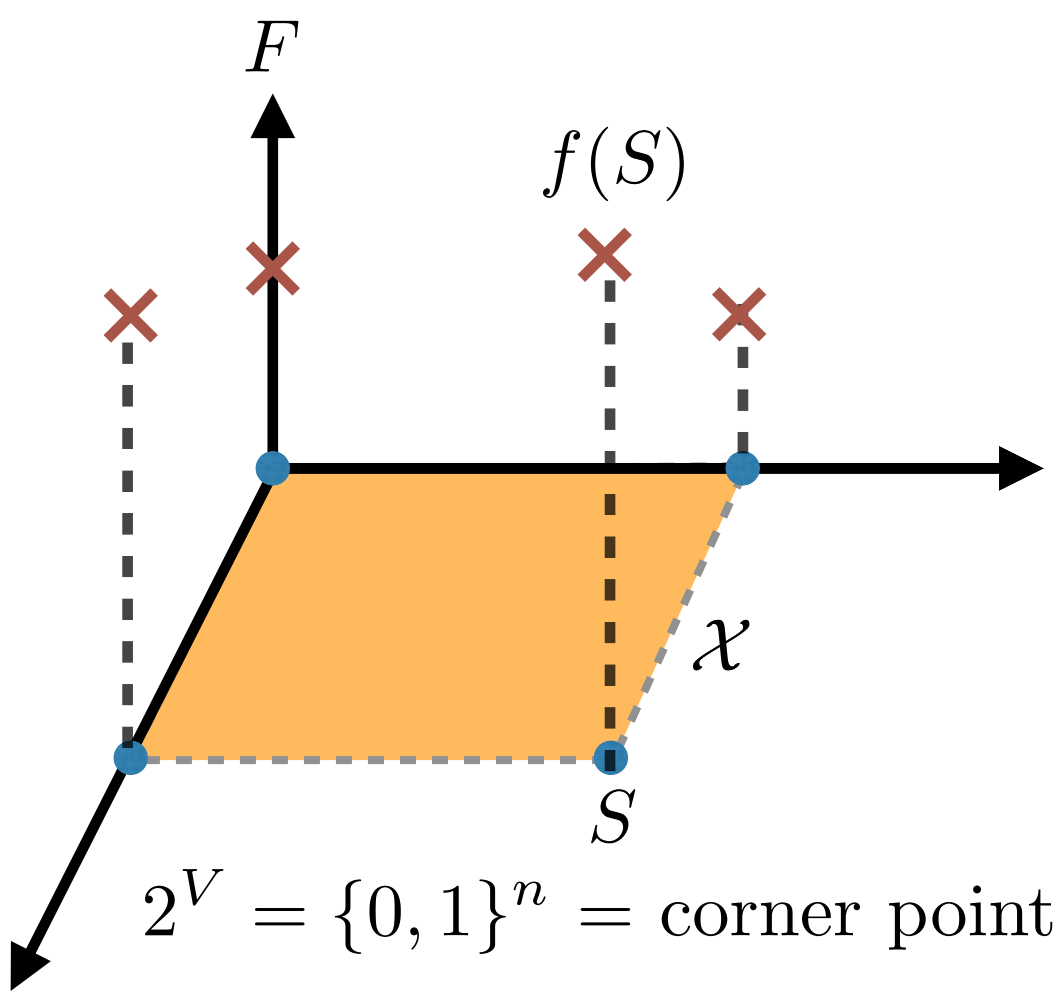


$2^V = \{0, 1\}^n =$ corner points of $[0, 1]^n$

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ → $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\mathcal{X} = [0, 1]^n$

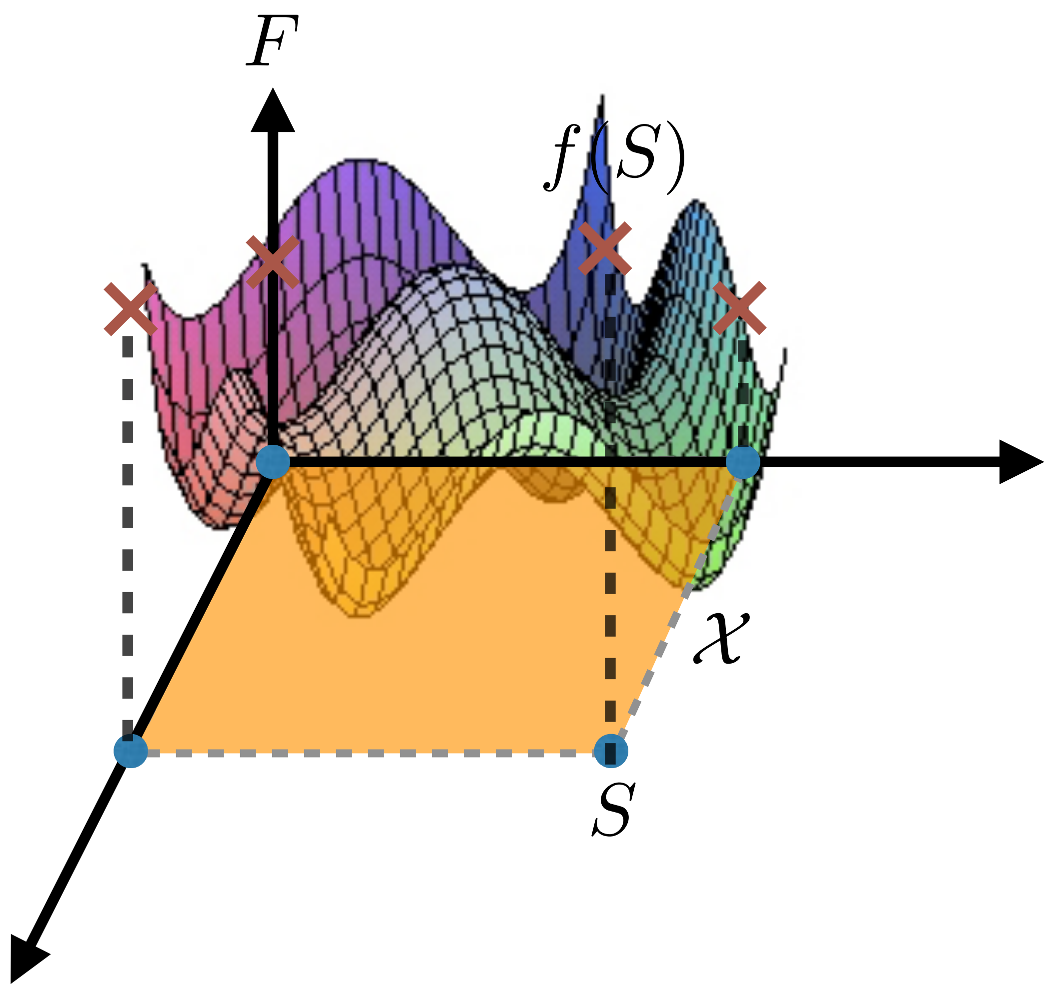


$2^V = \{0, 1\}^n =$ corner points of $[0, 1]^n$

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ $\rightarrow$ $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\qquad$ $\mathcal{X} = [0, 1]^n$

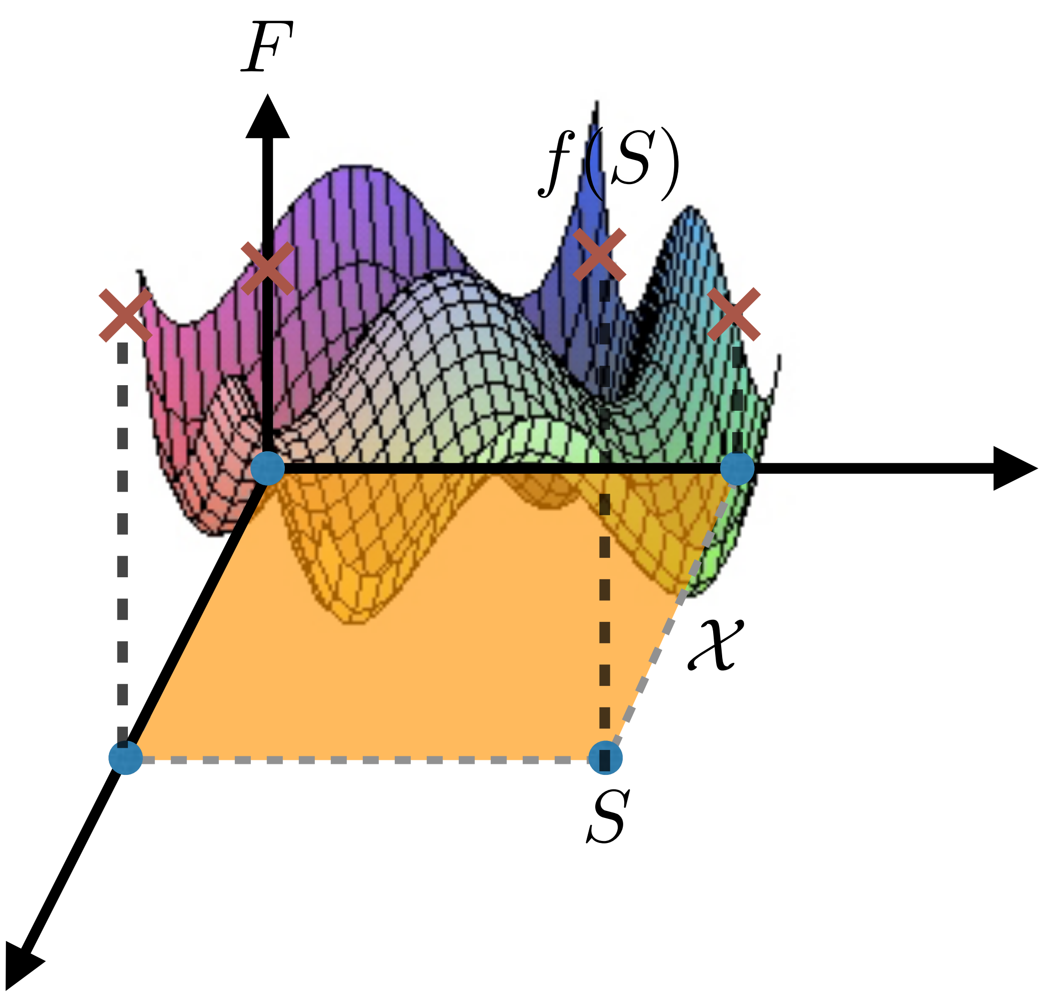


Multi-linear Extension

$f : 2^V \to \mathbb{R}$ $\longrightarrow$ $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\qquad$ $\mathcal{X} = [0, 1]^n$

$\mathbf{x} \in [0, 1]^n$

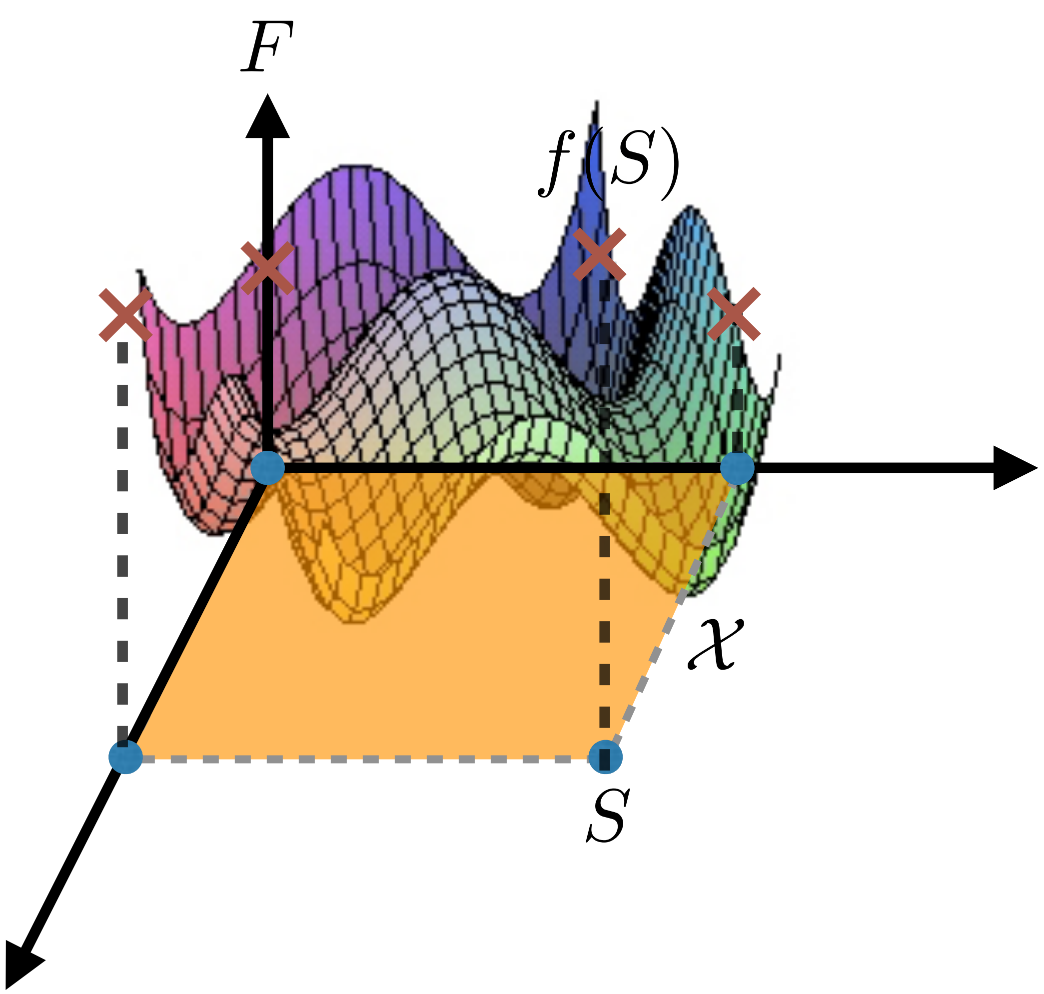


Multi-linear Extension

$f : 2^V \to \mathbb{R}$ → $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$

$\mathcal{X} = [0, 1]^n$



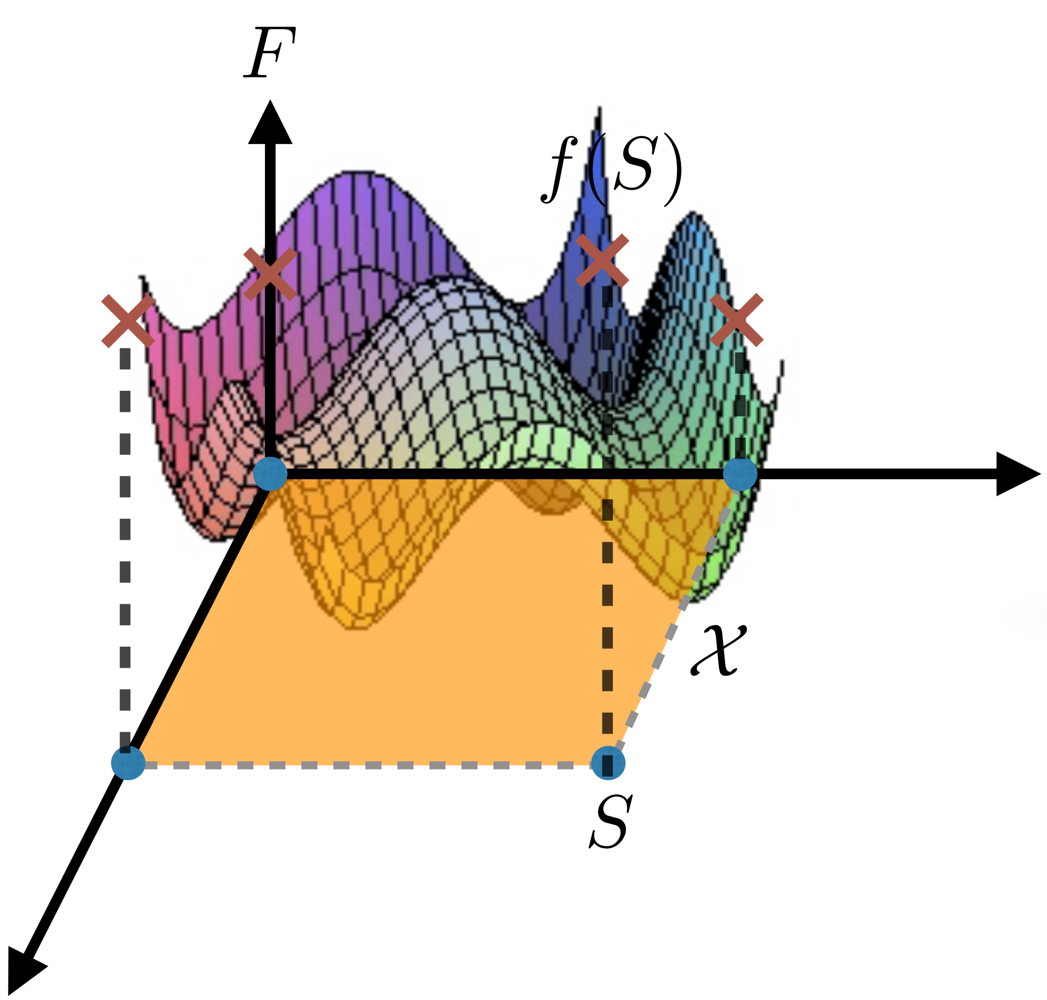
$\mathbf{x} \in [0, 1]^n$

$\mathbf{x} = (x_1, x_2, \cdots, x_n) \qquad x_i \in [0, 1]$

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ → $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\mathcal{X} = [0,1]^n$



$\mathbf{x} \in [0,1]^n$

$\mathbf{x} = (x_1, x_2, \cdots, x_n) \qquad x_i \in [0,1]$

$$F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)$$

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ → $F : \mathcal{X} \to \mathbb{R}$

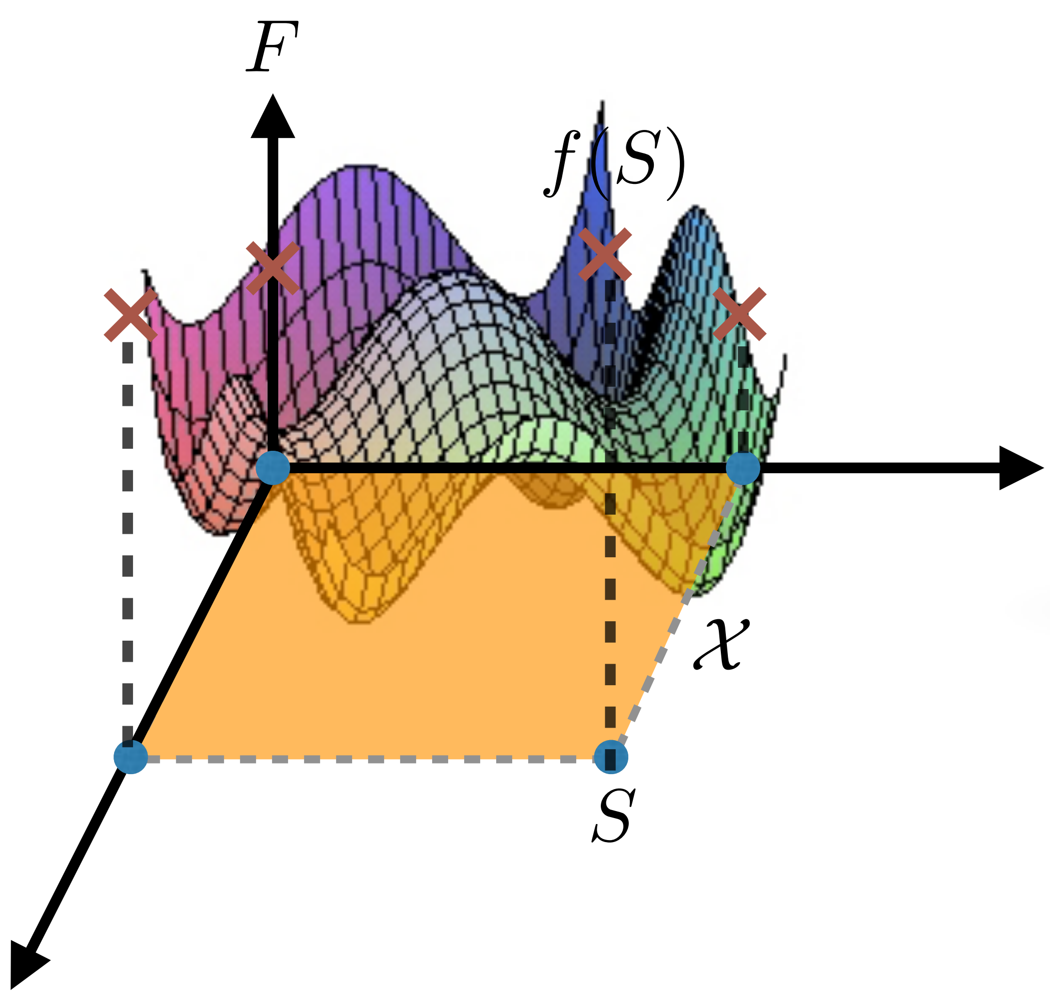
$V = \{1, 2, \cdots, n\}$ $\mathcal{X} = [0,1]^n$

$\mathbf{x} \in [0,1]^n$

$\mathbf{x} = (x_1, x_2, \cdots, x_n) \qquad x_i \in [0,1]$

$$F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)$$

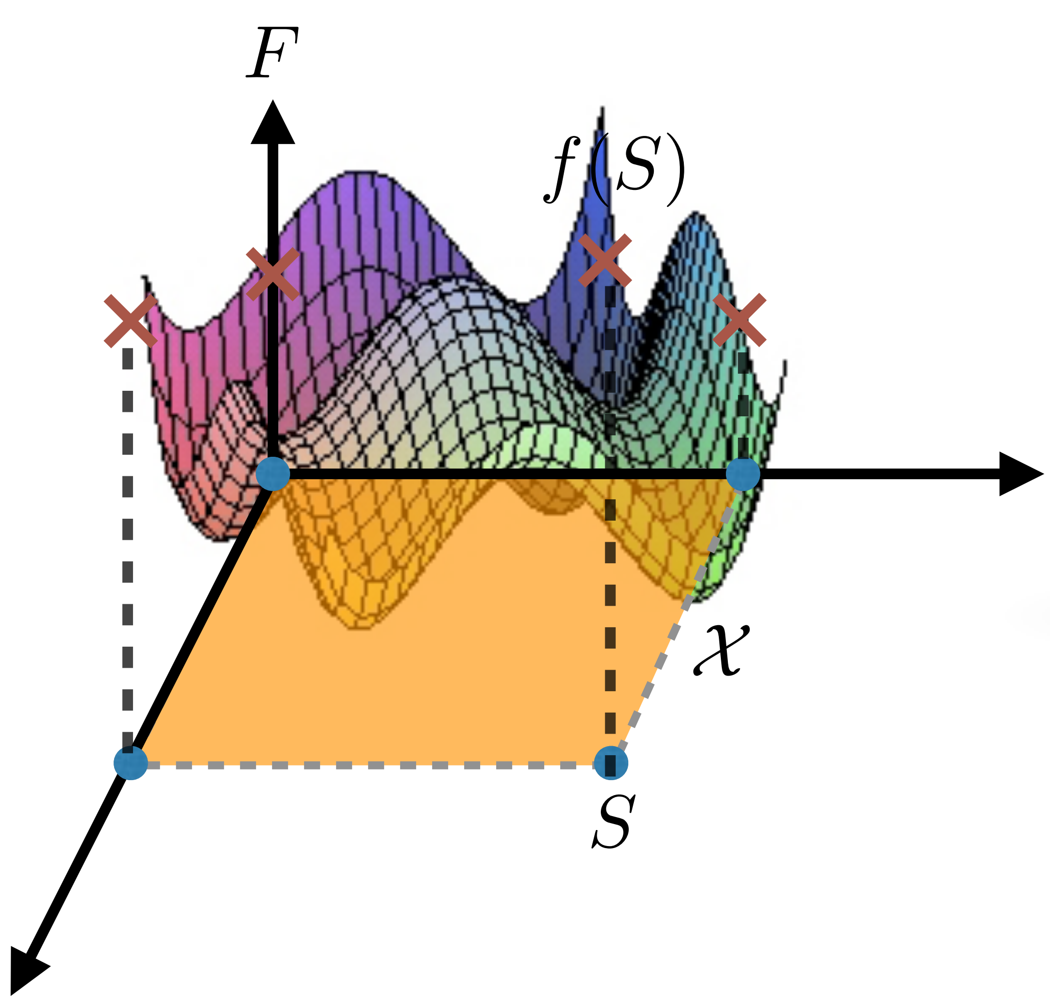
$$= \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$



Multi-linear Extension

$f : 2^V \to \mathbb{R}$ $\qquad$ $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\qquad$ $\mathcal{X} = [0, 1]^n$



$\mathbf{x} \in [0, 1]^n$

$\mathbf{x} = (x_1, x_2, \cdots, x_n) \qquad x_i \in [0, 1]$

$$F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)$$

$$= \mathbb{E}_{S \sim (x_1, \cdots, x_m)} [f(S)]$$

$1 \quad 2 \quad 3 \quad \cdots \quad i \quad \cdots \quad n$

Multi-linear Extension

$f : 2^V \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$

$F : \mathcal{X} \to \mathbb{R}$

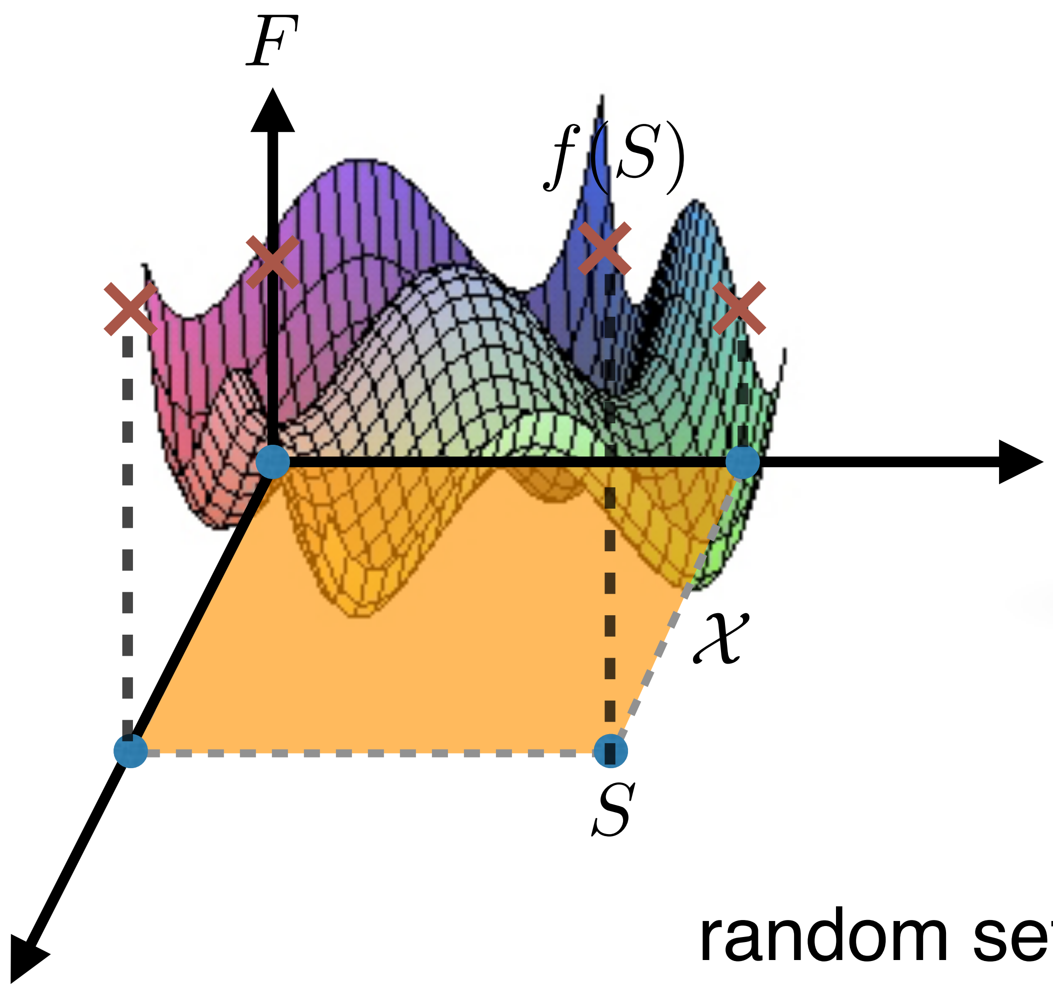
$\mathcal{X} = [0, 1]^n$

$\mathbf{x} \in [0, 1]^n$

$\mathbf{x} = (x_1, x_2, \cdots, x_n) \qquad x_i \in [0, 1]$

$$F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)$$

$$= \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$



$1 \quad 2 \quad 3 \quad \cdots \quad i \quad \cdots \quad n$

random set $S = \{ \qquad \}$

Multi-linear Extension

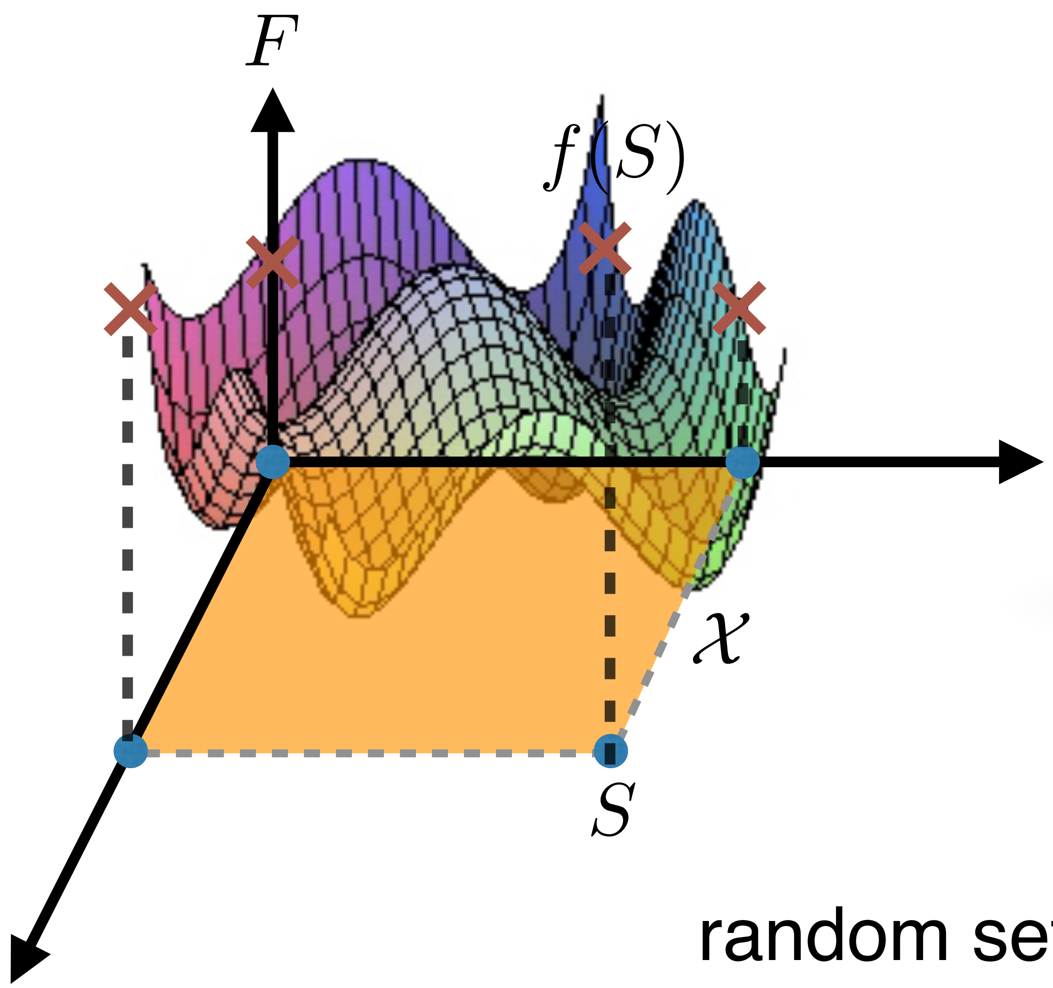
$f : 2^V \to \mathbb{R}$ $\quad\longrightarrow\quad$ $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\qquad$ $\mathcal{X} = [0,1]^n$

$\mathbf{x} \in [0,1]^n$

$\mathbf{x} = (x_1, x_2, \cdots, x_n) \qquad x_i \in [0,1]$

$$F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)$$
$$= \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$



$1 \quad 2 \quad 3 \quad \cdots \quad i \quad \cdots \quad n$

$\downarrow$ wp x_1

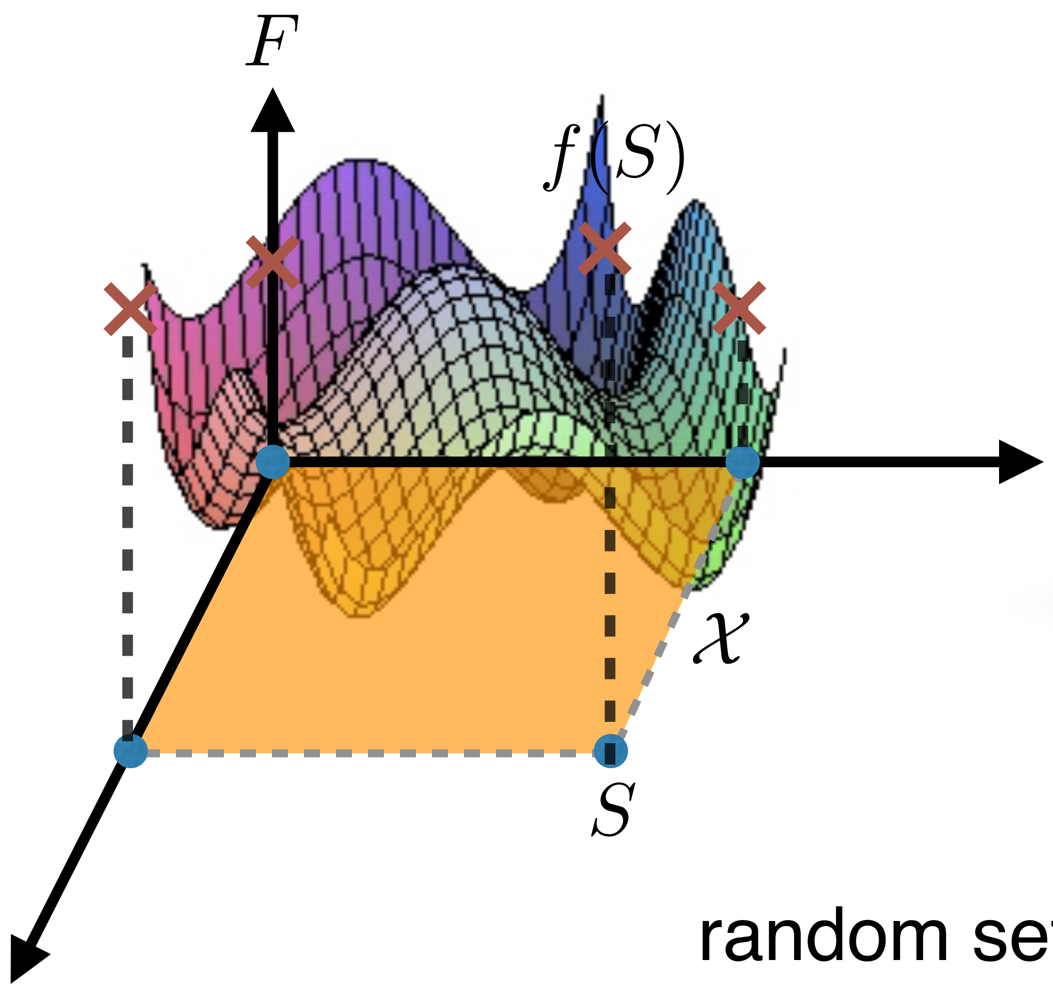
random set $S = \{ \qquad \}$

Yale $\qquad$ Penn

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ → $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\qquad$ $\mathcal{X} = [0,1]^n$



$\mathbf{x} \in [0,1]^n$

$\mathbf{x} = (x_1, x_2, \cdots, x_n) \qquad x_i \in [0,1]$

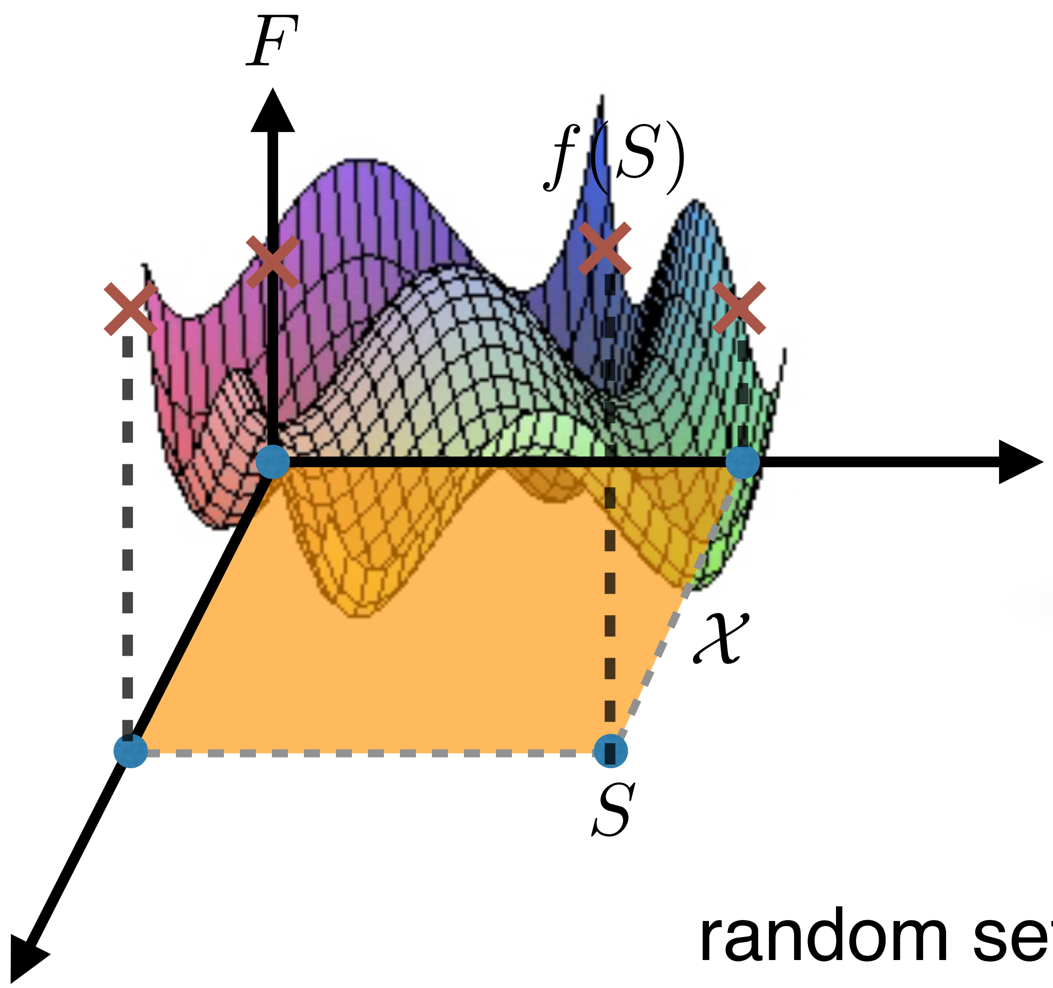
$$F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)$$
$$= \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$

$$\text{random set } S = \{ \overset{1}{\downarrow}_{\text{wp } x_1} \quad \overset{2}{\downarrow}_{x_2} \quad \overset{3}{\downarrow}_{x_3} \quad \cdots \quad i \quad \cdots \quad n \}$$

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ → $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\mathcal{X} = [0,1]^n$



$\mathbf{x} \in [0,1]^n$

$\mathbf{x} = (x_1, x_2, \cdots, x_n) \qquad x_i \in [0,1]$

$$F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)$$

$$= \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$

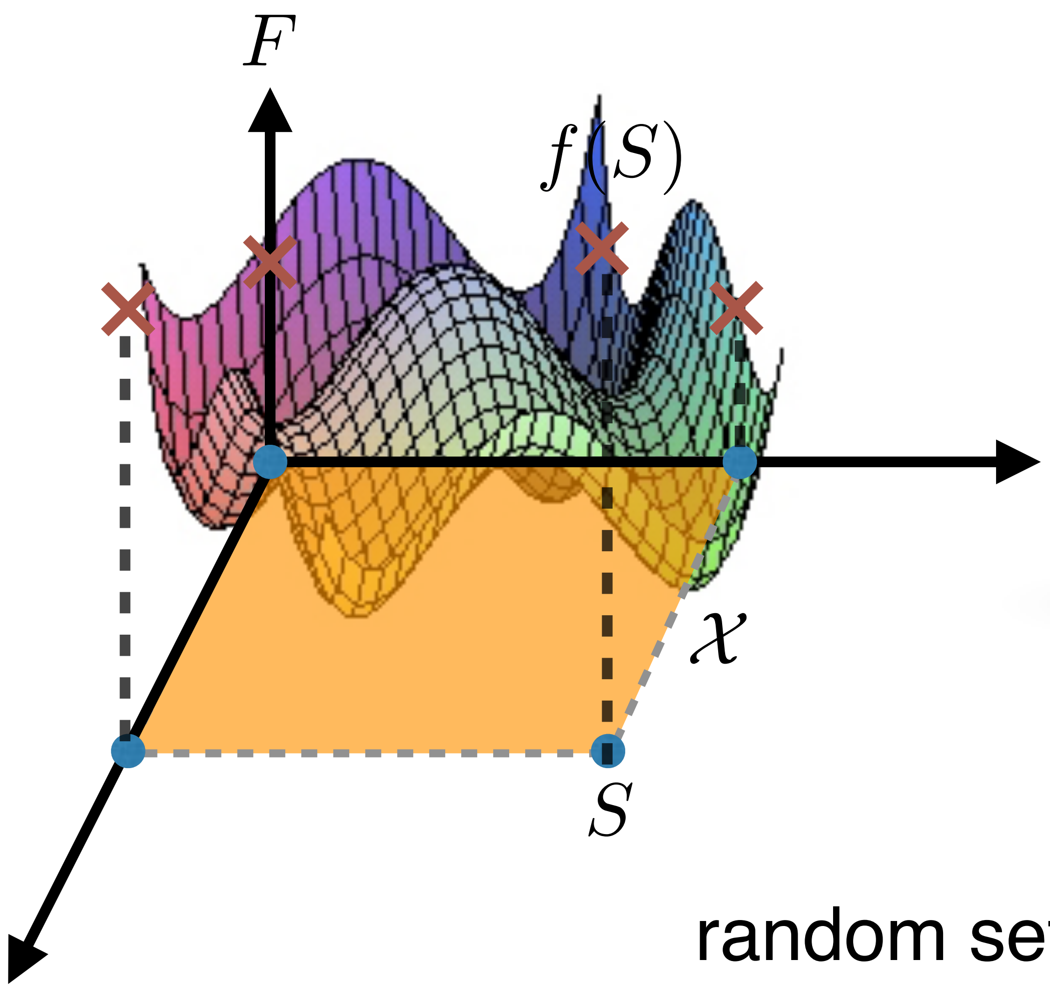
$$\begin{array}{ccccccc} 1 & 2 & 3 & \cdots & i & \cdots & n \\ \downarrow \text{wp } x_1 & \downarrow x_2 & \downarrow x_3 & & \downarrow x_i & & \end{array}$$

random set $S = \{ \qquad \}$

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ $\quad\longrightarrow\quad$ $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\qquad$ $\mathcal{X} = [0,1]^n$



$\mathbf{x} \in [0,1]^n$

$\mathbf{x} = (x_1, x_2, \cdots, x_n) \qquad x_i \in [0,1]$

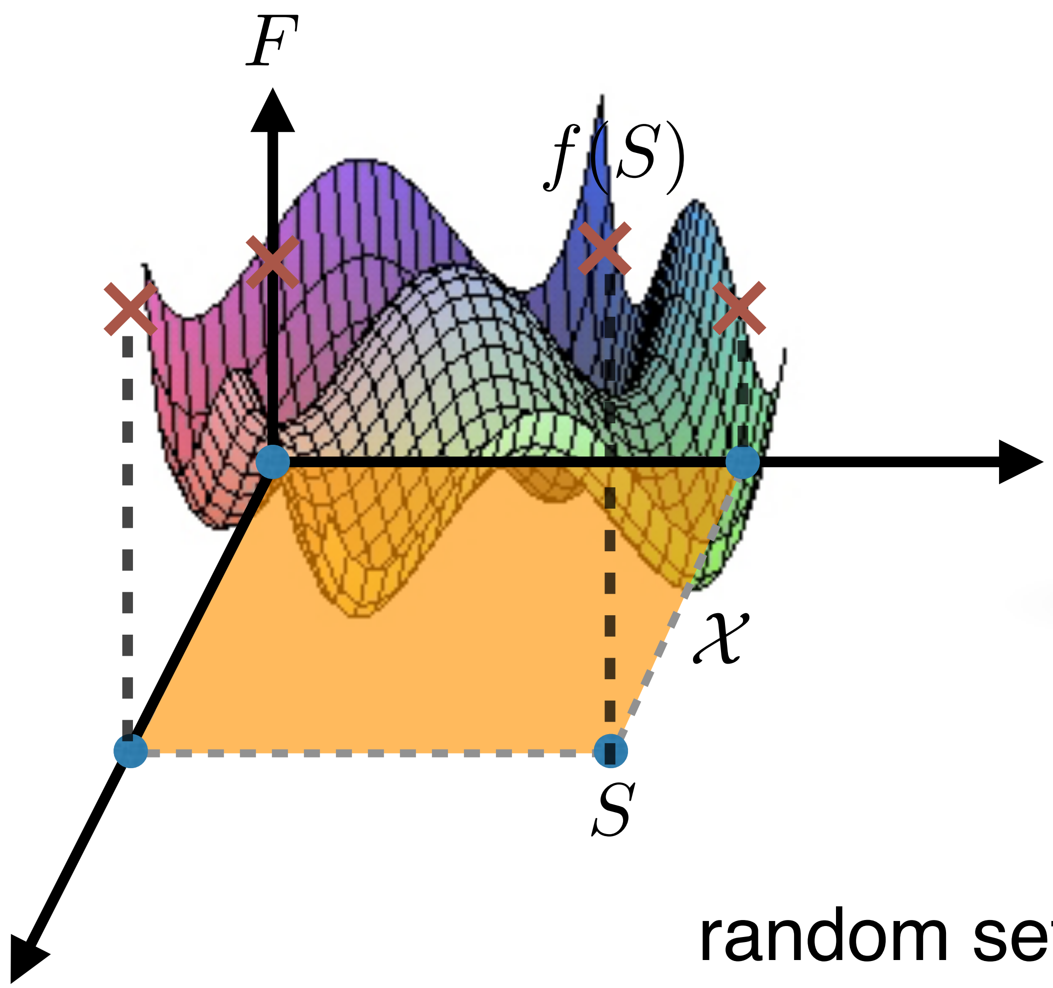
$$
\begin{aligned}
F(\mathbf{x}) &= \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \\
&= \mathbb{E}_{S \sim (x_1, \cdots, x_m)} [f(S)]
\end{aligned}
$$

$$
\begin{array}{cccccccc}
 & 1 & 2 & 3 & \cdots & i & \cdots & n \\
 & \downarrow \text{wp } x_1 & \downarrow x_2 & \downarrow x_3 & & \downarrow x_i & & \downarrow x_n \\
\text{random set } S = \{ & & & & & & & \}
\end{array}
$$

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ → $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\mathcal{X} = [0, 1]^n$



$\mathbf{x} \in [0, 1]^n$

$\mathbf{x} = (x_1, x_2, \cdots, x_n) \qquad x_i \in [0, 1]$

$$\boxed{\begin{aligned} F(\mathbf{x}) &= \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \\ &= \mathbb{E}_{S \sim (x_1, \cdots, x_m)} [f(S)] \end{aligned}}$$

$$\begin{array}{cccccccc} 1 & 2 & 3 & \cdots & i & \cdots & n \\ \downarrow \text{wp } x_1 & \downarrow x_2 & \downarrow x_3 & & \downarrow x_i & & \downarrow x_n \end{array}$$

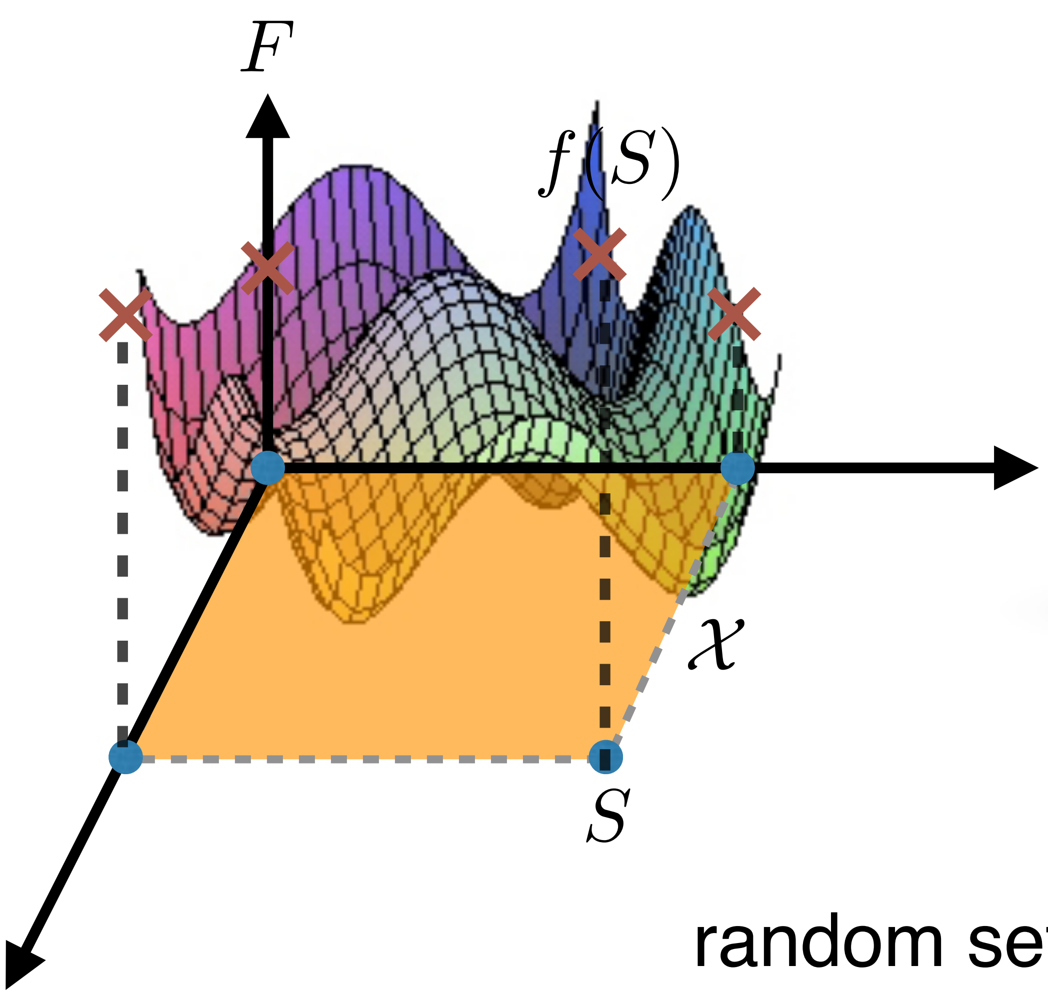
random set $S = \{ \quad \}$

probability of $S = \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)$

Multi-linear Extension

$f : 2^V \to \mathbb{R}$ $\longrightarrow$ $F : \mathcal{X} \to \mathbb{R}$

$V = \{1, 2, \cdots, n\}$ $\qquad$ $\mathcal{X} = [0,1]^n$



$\mathbf{x} \in [0,1]^n$

$\mathbf{x} = (x_1, x_2, \cdots, x_n) \qquad x_i \in [0,1]$

$$F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \boxed{\prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)}$$

$$= \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$

$1 \quad 2 \quad 3 \quad \cdots \quad i \quad \cdots \quad n$

$\downarrow$ wp x_1 $\quad \downarrow x_2 \quad \downarrow x_3 \quad \downarrow x_i \quad \downarrow x_n$

random set $S = \{ \qquad \}$

probability of $S = \boxed{\prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)}$

Properties of Multi-linear Extension

f is submodular $\longleftrightarrow$ F is DR-submodular

Properties of Multi-linear Extension

f is submodular $\Longleftrightarrow$ F is DR-submodular

$$\nabla^2 F(\mathbf{x}) = \begin{pmatrix} 0 & \leq 0 & \leq 0 \\ \leq 0 & 0 & \leq 0 \\ \leq 0 & \leq 0 & 0 \end{pmatrix}$$

Properties of Multi-linear Extension

f is submodular $\longleftrightarrow$ F is DR-submodular

f is monotone $\longleftrightarrow$ F is monotone

Properties of Multi-linear Extension

f is submodular $\longleftrightarrow$ F is DR-submodular

f is monotone $\longleftrightarrow$ F is monotone

multi-linear extension is a linear operator: $\alpha f_1 + \beta f_2 \longrightarrow \alpha F_1 + \beta F_2$

Properties of Multi-linear Extension

f is submodular $\longleftrightarrow$ F is DR-submodular

f is monotone $\longleftrightarrow$ F is monotone

multi-linear extension is a linear operator: $\alpha f_1 + \beta f_2 \longrightarrow \alpha F_1 + \beta F_2$

F is concave along positive directions and convex along cross directions

Properties of Multi-linear Extension

f is submodular $\longleftrightarrow$ F is DR-submodular

f is monotone $\longleftrightarrow$ F is monotone

multi-linear extension is a linear operator: $\alpha f_1 + \beta f_2 \longrightarrow \alpha F_1 + \beta F_2$

F is concave along positive directions and convex along cross directions

$(v_1, \cdots, v_n), v_i \geq 0$

Properties of Multi-linear Extension

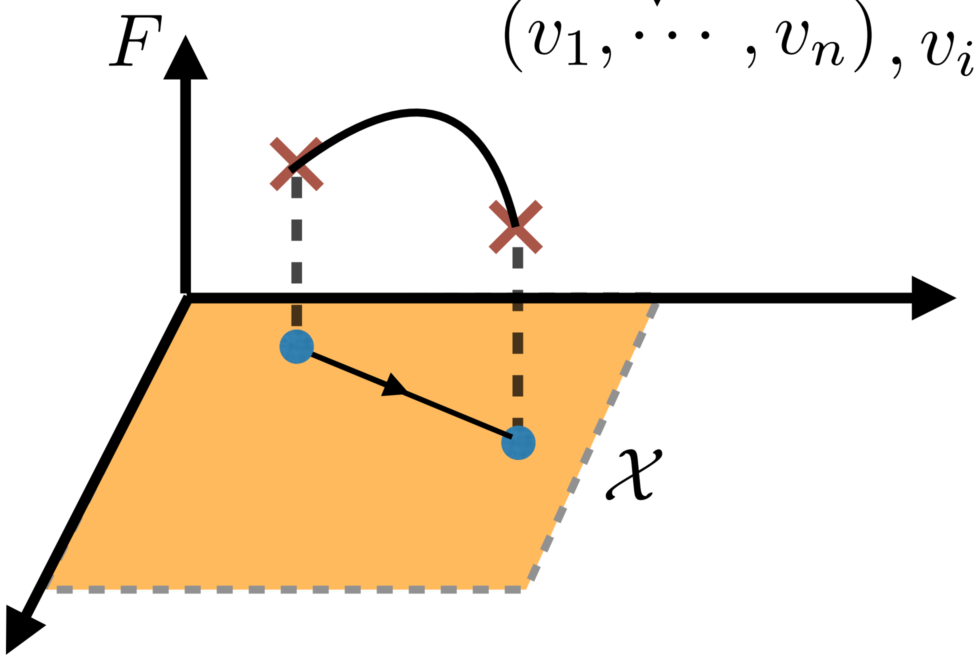
f is submodular $\longleftrightarrow$ F is DR-submodular

f is monotone $\longleftrightarrow$ F is monotone

multi-linear extension is a linear operator: $\alpha f_1 + \beta f_2 \longrightarrow \alpha F_1 + \beta F_2$

F is concave along positive directions and convex along cross directions

$(v_1, \cdots, v_n), v_i \geq 0$



Properties of Multi-linear Extension

f is submodular $\longleftrightarrow$ F is DR-submodular

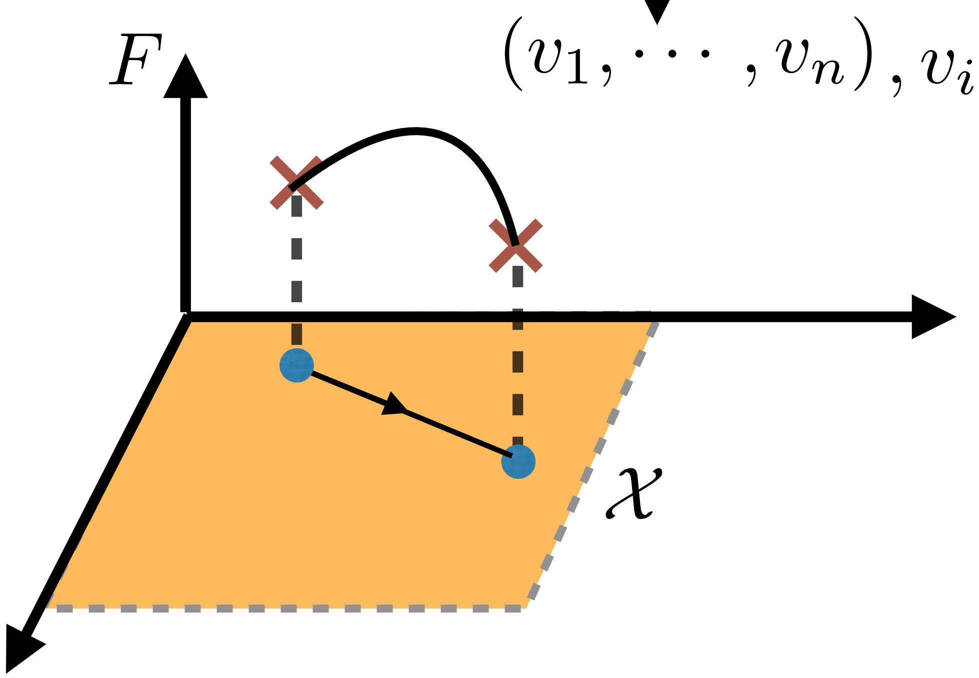
f is monotone $\longleftrightarrow$ F is monotone

multi-linear extension is a linear operator: $\alpha f_1 + \beta f_2 \longrightarrow \alpha F_1 + \beta F_2$

F is concave along positive directions and convex along cross directions

$(v_1, \cdots, v_n), v_i \geq 0$

$e_i - e_j$



Properties of Multi-linear Extension

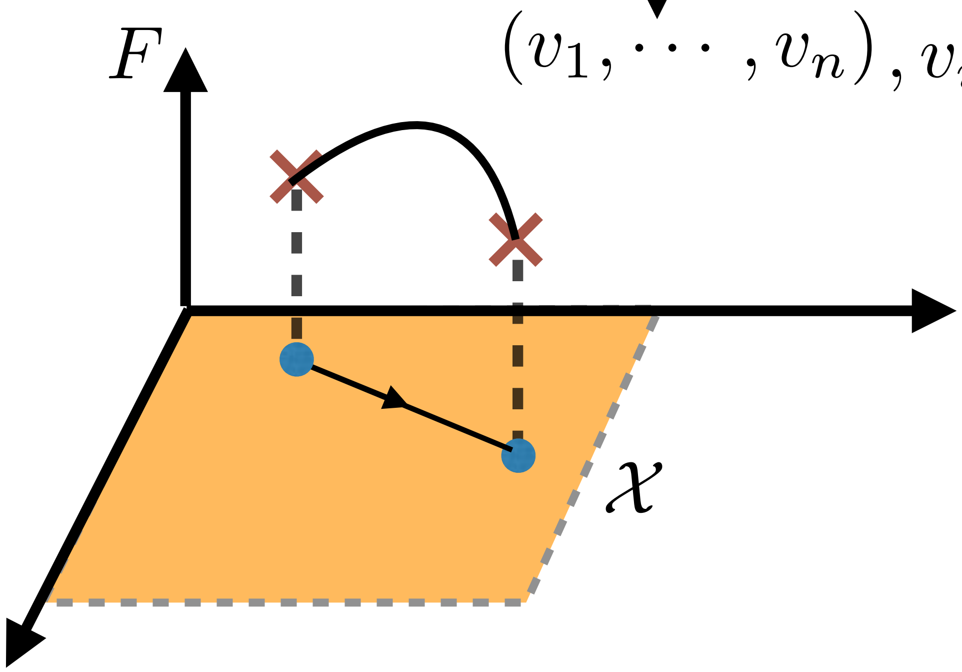
f is submodular $\longleftrightarrow$ F is DR-submodular

f is monotone $\longleftrightarrow$ F is monotone

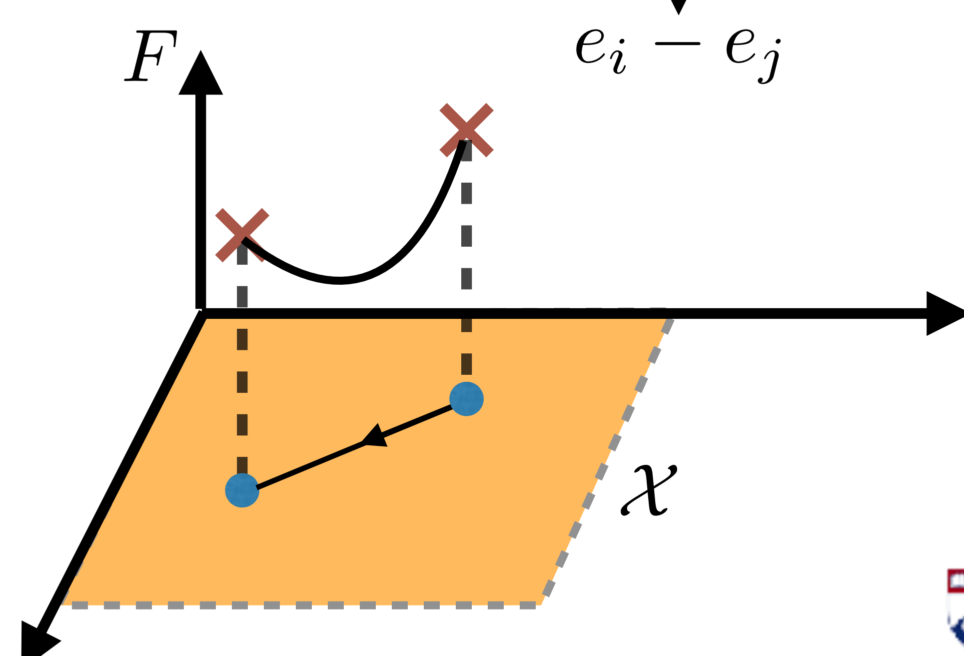
multi-linear extension is a linear operator: $\alpha f_1 + \beta f_2 \longrightarrow \alpha F_1 + \beta F_2$

F is concave along positive directions and convex along cross directions

$(v_1, \cdots, v_n), v_i \geq 0$



$e_i - e_j$



Equivalent Formulations

Let f be a submodular set function with multi-linear extension F, then:

$$\max_{S \in \mathcal{I}} f(S) \quad = \quad \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

Equivalent Formulations

Let f be a submodular set function with multi-linear extension F, then:

$$\max_{S \in \mathcal{I}} f(S) \quad = \quad \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

Equivalent Formulations

Let f be a submodular set function with multi-linear extension F, then:

$$\max_{S \in \mathcal{I}} f(S) \quad = \quad \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

$$\mathcal{I} = \{S \subseteq V : |S| \leq k\} \quad \Longrightarrow \quad \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} x_i \leq k\}$$

(cardinality)

Equivalent Formulations

Let f be a submodular set function with multi-linear extension F, then:

$$\max_{S \in \mathcal{I}} f(S) \quad = \quad \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

$$\mathcal{I} = \{S \subseteq V : |S| \leq k\} \quad \longrightarrow \quad \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} x_i \leq k\}$$

(cardinality)

$$\mathcal{I} = \{S \subseteq V : \sum_{i \in S} c_i \leq B\} \quad \longrightarrow \quad \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} c_i x_i \leq B\}$$

(knapsack)

Equivalent Formulations

Let f be a submodular set function with multi-linear extension F, then:

$$\max_{S \in \mathcal{I}} f(S) \quad = \quad \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

$$\mathcal{I} = \{S \subseteq V : |S| \leq k\} \quad \Rightarrow \quad \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} x_i \leq k\}$$

(cardinality)

$$\mathcal{I} = \{S \subseteq V : \sum_{i \in S} c_i \leq B\} \quad \Rightarrow \quad \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} c_i x_i \leq B\}$$

(knapsack)

$\mathcal{I}$ is a matroid $\Rightarrow$ $\mathcal{K}$ is the base polytope of the matroid

(matroid)

Equivalent Formulations

Let f be a submodular set function with multi-linear extension F, then:

$$\max_{S \in \mathcal{I}} f(S) = \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

$$\mathcal{I} = \{S \subseteq V : |S| \leq k\} \longrightarrow \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} x_i \leq k\}$$

(cardinality)

$$\mathcal{I} = \{S \subseteq V : \sum_{i \in S} c_i \leq B\} \longrightarrow \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} c_i x_i \leq B\}$$

(knapsack)

$\mathcal{I}$ is a matroid $\longrightarrow$ $\mathcal{K}$ is the base polytope of the matroid

(matroid)

$$\max_{S \in \mathcal{I}} f(S)$$

Equivalent Formulations

Let f be a submodular set function with multi-linear extension F, then:

$$\max_{S \in \mathcal{I}} f(S) \quad = \quad \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

$$\mathcal{I} = \{S \subseteq V : |S| \leq k\} \quad \Rightarrow \quad \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} x_i \leq k\}$$

(cardinality)

$$\mathcal{I} = \{S \subseteq V : \sum_{i \in S} c_i \leq B\} \quad \Rightarrow \quad \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} c_i x_i \leq B\}$$

(knapsack)

$\mathcal{I}$ is a matroid $\Rightarrow$ $\mathcal{K}$ is the base polytope of the matroid

(matroid)

$$\boxed{\max_{S \in \mathcal{I}} f(S)} \longrightarrow \boxed{\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})}$$

Equivalent Formulations

Let f be a submodular set function with multi-linear extension F, then:

$$\max_{S \in \mathcal{I}} f(S) \quad = \quad \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

$$\mathcal{I} = \{S \subseteq V : |S| \leq k\} \Longrightarrow \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} x_i \leq k\}$$

(cardinality)

$$\mathcal{I} = \{S \subseteq V : \sum_{i \in S} c_i \leq B\} \Longrightarrow \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} c_i x_i \leq B\}$$

(knapsack)

$\mathcal{I}$ is a matroid $\Longrightarrow$ $\mathcal{K}$ is the base polytope of the matroid

(matroid)

$$\begin{tikzcd}
\boxed{\max_{S \in \mathcal{I}} f(S)} \arrow[r] & \boxed{\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})} \arrow[d] \\
 & \boxed{\mathbf{x}_{\text{sol}} \in \mathcal{K}}
\end{tikzcd}$$

Equivalent Formulations

Let f be a submodular set function with multi-linear extension F, then:

$$\max_{S \in \mathcal{I}} f(S) \quad = \quad \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

$$\mathcal{I} = \{S \subseteq V : |S| \leq k\} \quad \longrightarrow \quad \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} x_i \leq k\}$$

(cardinality)

$$\mathcal{I} = \{S \subseteq V : \sum_{i \in S} c_i \leq B\} \quad \longrightarrow \quad \mathcal{K} = \{(x_1, \cdots, x_n) \in [0,1]^n : \sum_{i=1}^{n} c_i x_i \leq B\}$$

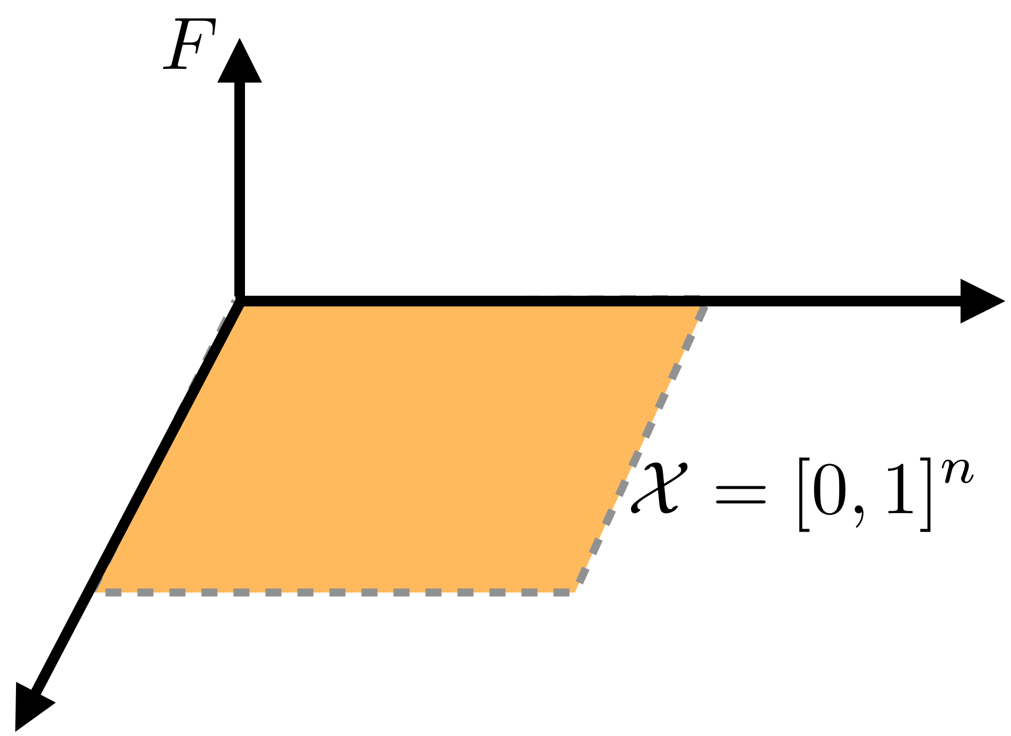
(knapsack)

$\mathcal{I}$ is a matroid $\longrightarrow$ $\mathcal{K}$ is the base polytope of the matroid

(matroid)

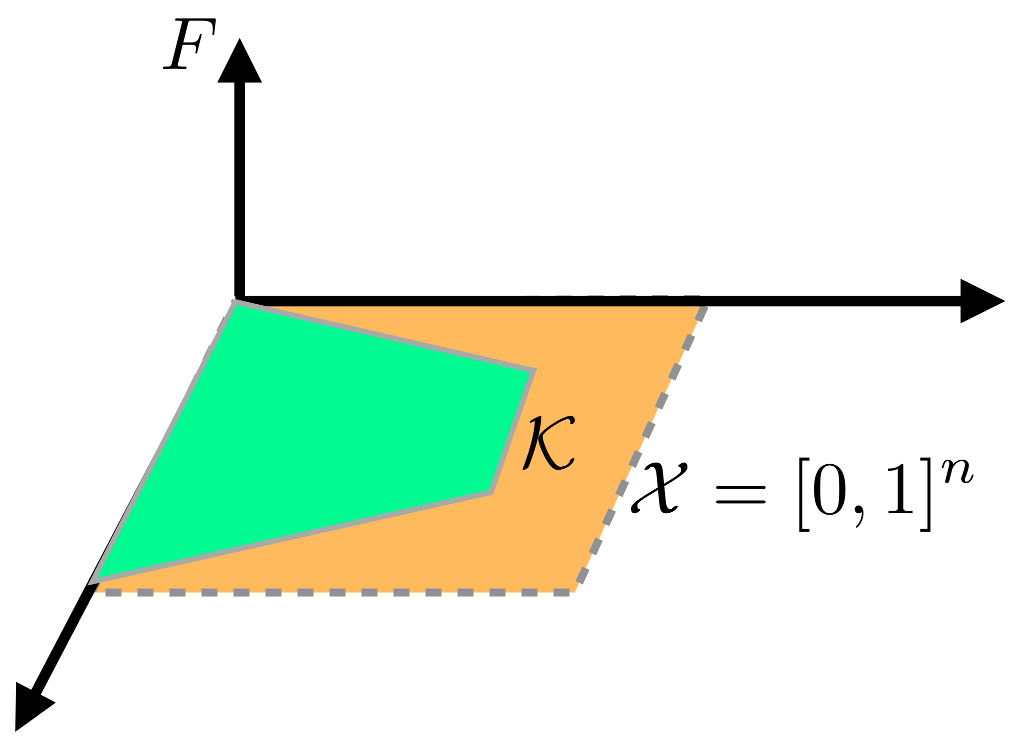
$$\begin{tikzcd}
\max_{S \in \mathcal{I}} f(S) \arrow[r] & \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x}) \arrow[d] \\
S_{\text{sol}} \in \mathcal{I} \arrow[u] & \mathbf{x}_{\text{sol}} \in \mathcal{K} \arrow[l, "\text{rounding}"']
\end{tikzcd}$$

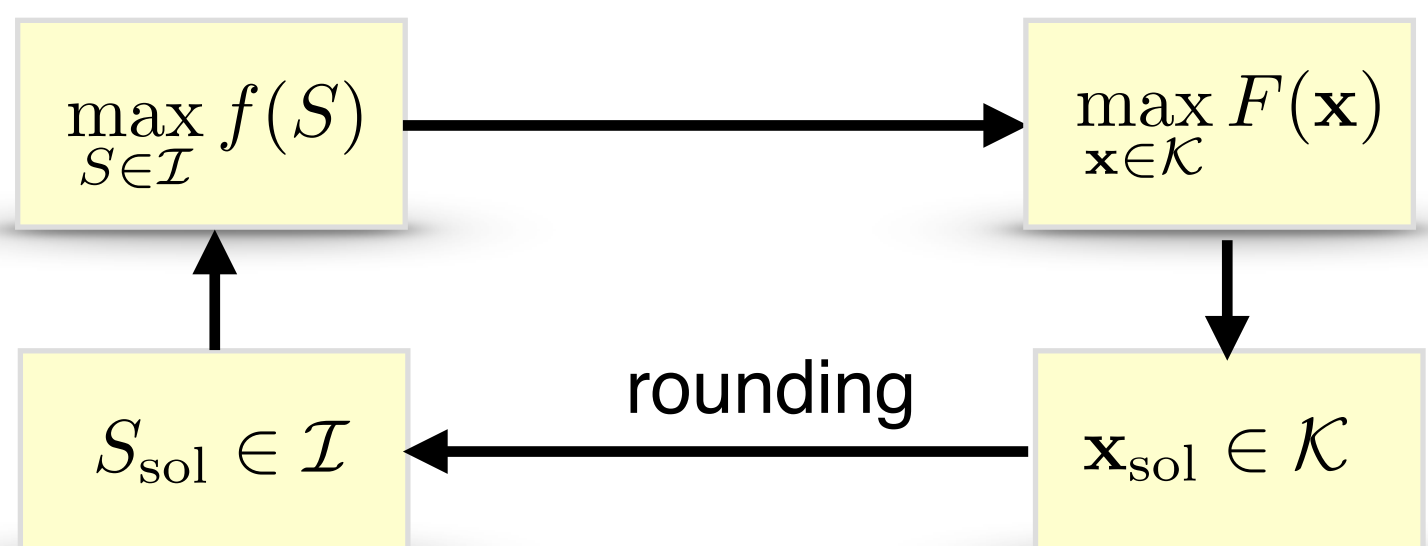
Rounding



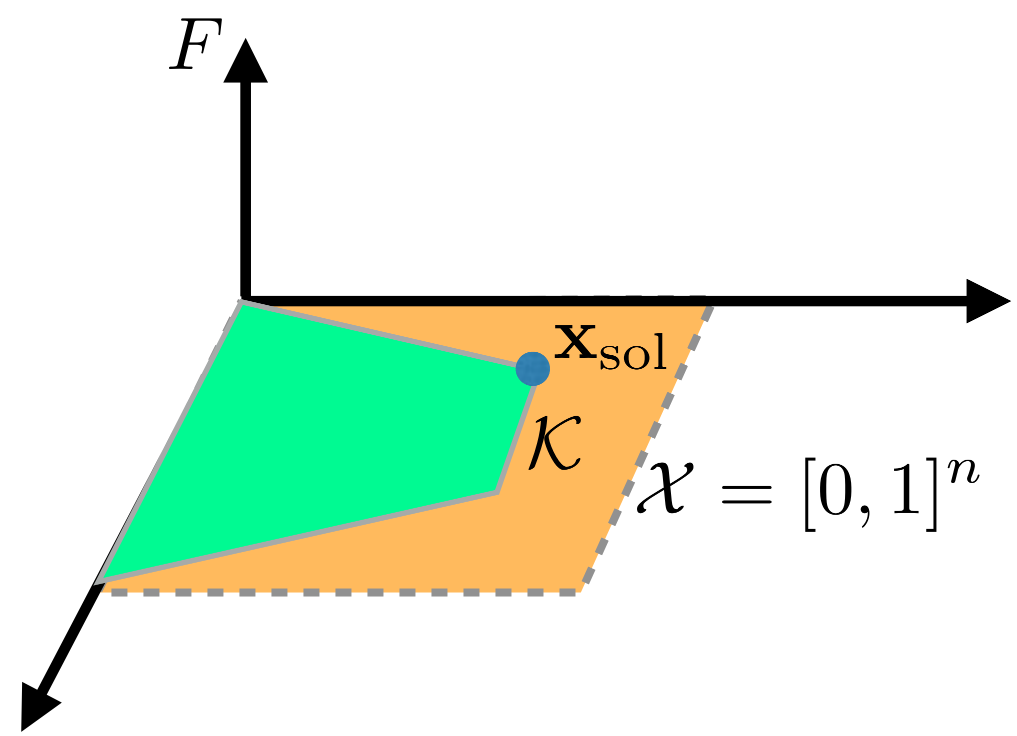
$$
\begin{tikzcd}
\max_{S \in \mathcal{I}} f(S) \arrow[r] & \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x}) \arrow[d] \\
S_{\text{sol}} \in \mathcal{I} \arrow[u] & \mathbf{x}_{\text{sol}} \in \mathcal{K} \arrow[l, "\text{rounding}"']
\end{tikzcd}
$$

Rounding


F
$\mathcal{K}$
$\mathcal{X} = [0,1]^n$

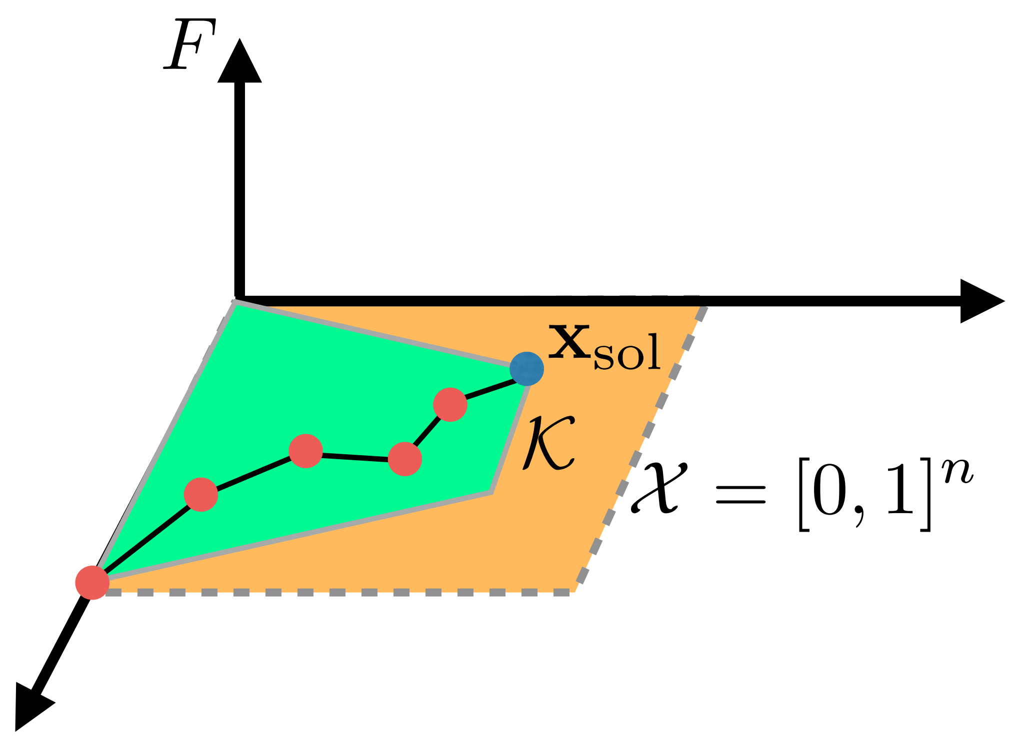

$\max_{S \in \mathcal{I}} f(S)$
$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$
$\mathbf{x}_{\text{sol}} \in \mathcal{K}$
rounding
$S_{\text{sol}} \in \mathcal{I}$

Rounding

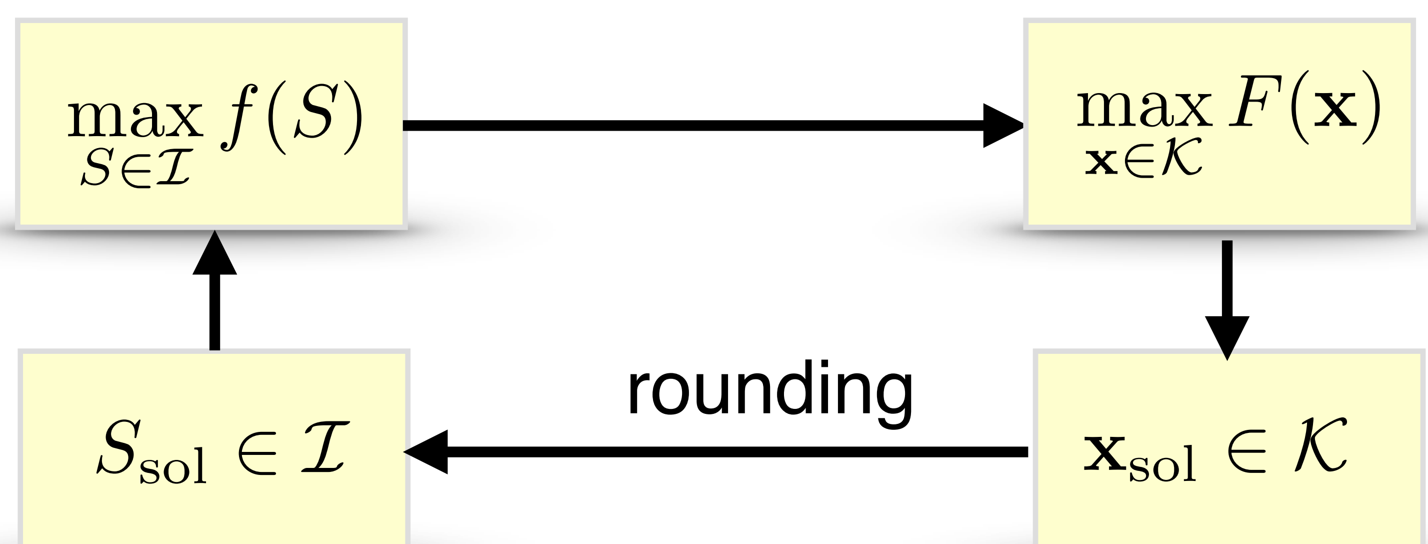


$$
\begin{tikzcd}
\max\limits_{S\in\mathcal{I}} f(S) \arrow[r] & \max\limits_{\mathbf{x}\in\mathcal{K}} F(\mathbf{x}) \arrow[d] \\
S_{\rm sol} \in \mathcal{I} \arrow[u] & \mathbf{x}_{\rm sol} \in \mathcal{K} \arrow[l, "\text{rounding}"']
\end{tikzcd}
$$

Rounding

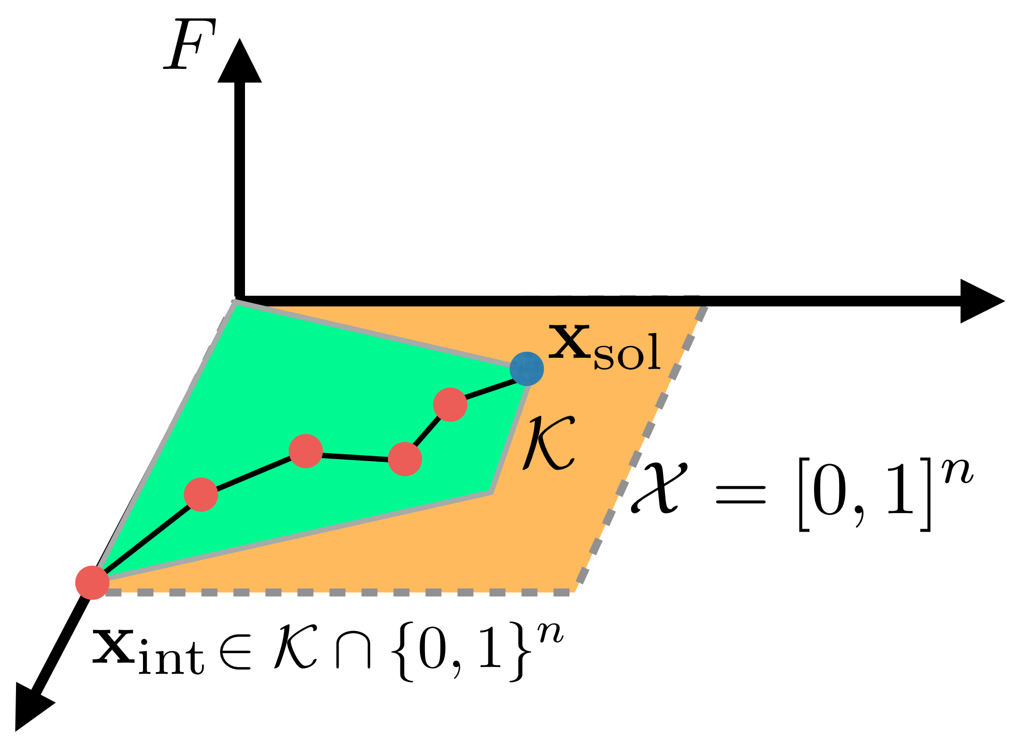


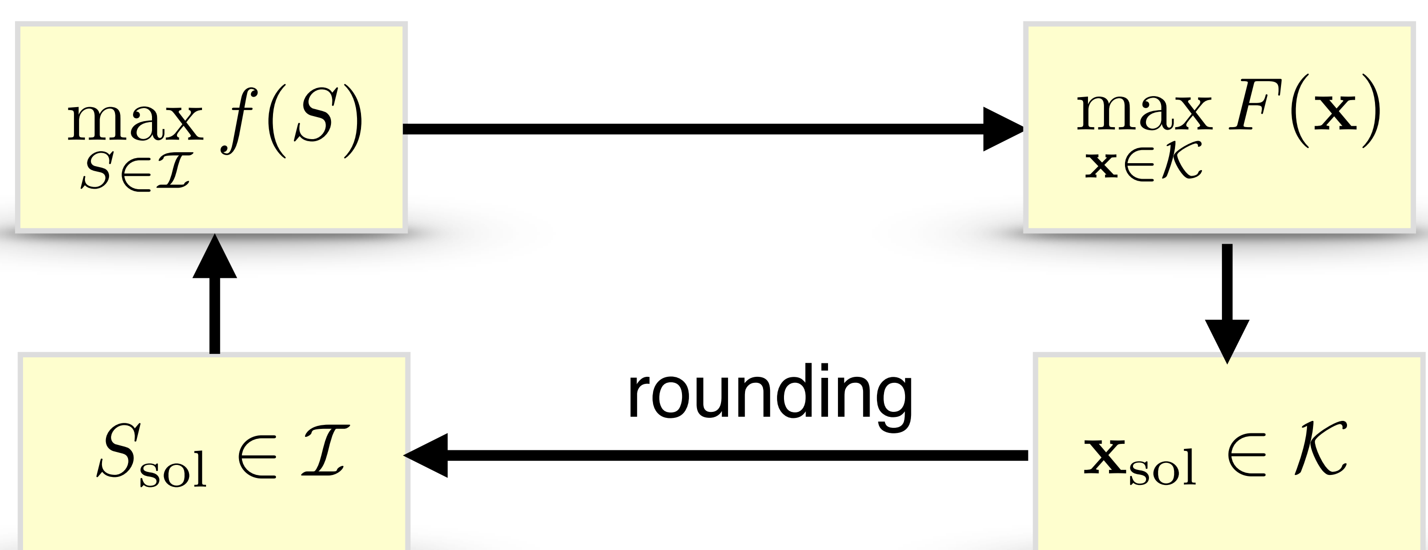
F
$\mathbf{x}_{\text{sol}}$
$\mathcal{K}$
$\mathcal{X} = [0,1]^n$



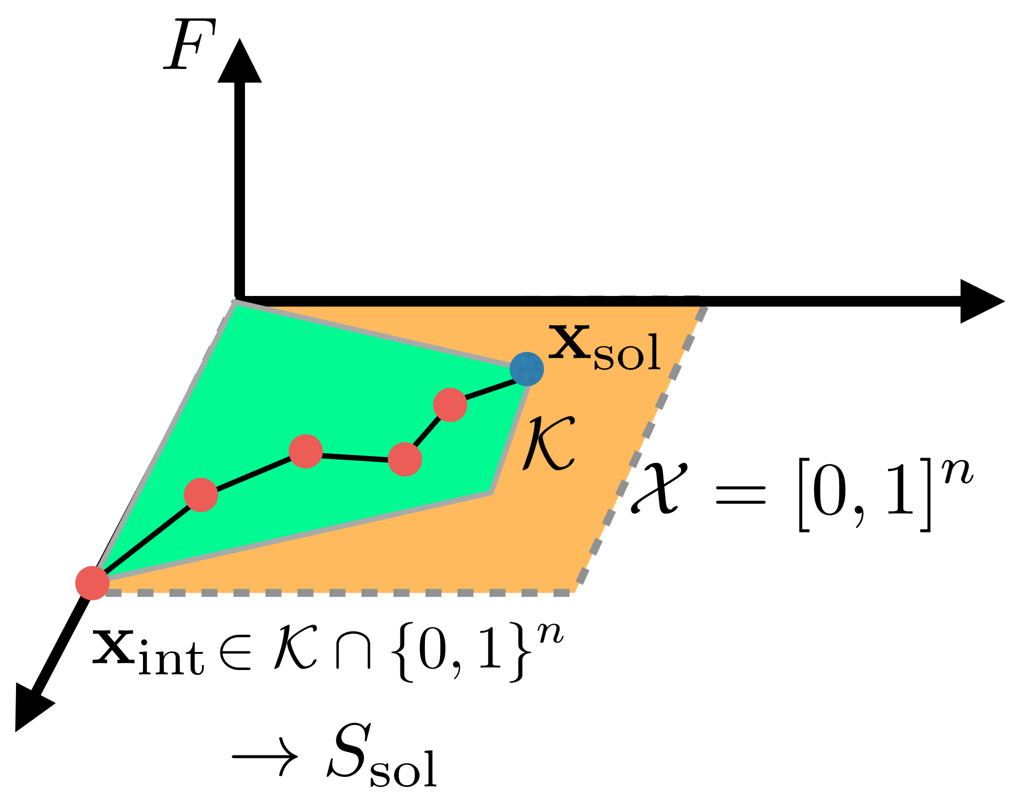
$\max_{S \in \mathcal{I}} f(S)$
$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$
$\mathbf{x}_{\text{sol}} \in \mathcal{K}$
rounding
$S_{\text{sol}} \in \mathcal{I}$

Rounding


F
$\mathbf{x}_{\text{sol}}$
$\mathcal{K}$
$\mathcal{X} = [0,1]^n$
$\mathbf{x}_{\text{int}} \in \mathcal{K} \cap \{0,1\}^n$

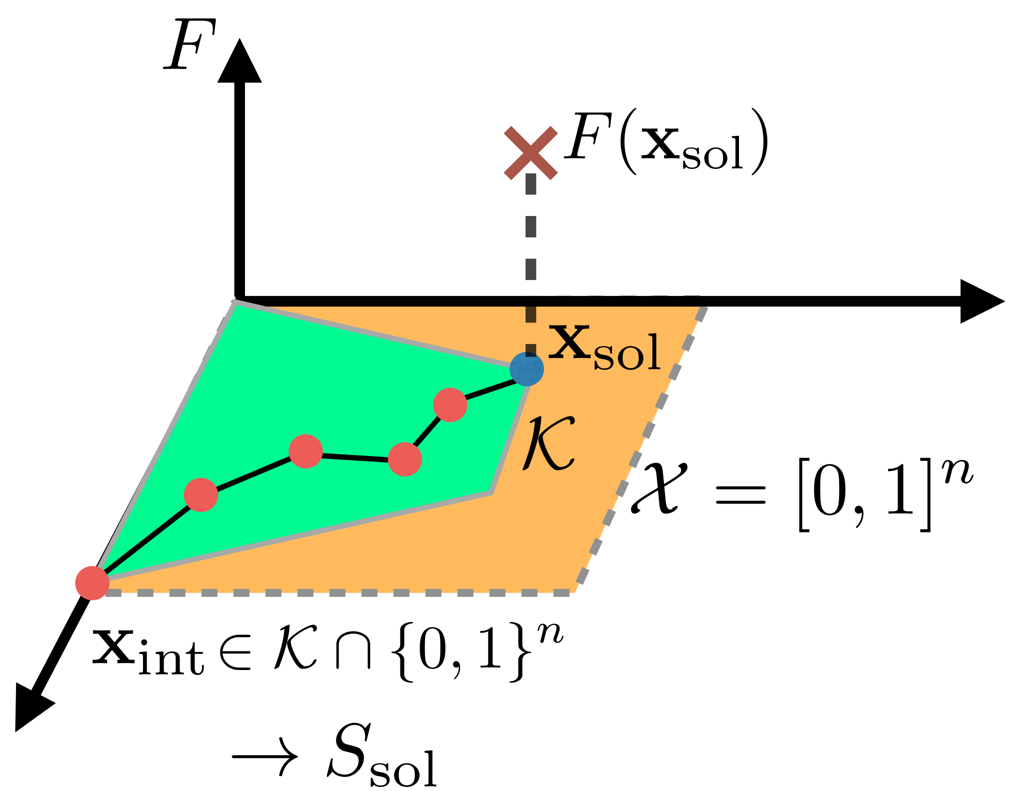

$\max_{S \in \mathcal{I}} f(S)$
$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$
rounding
$S_{\text{sol}} \in \mathcal{I}$
$\mathbf{x}_{\text{sol}} \in \mathcal{K}$

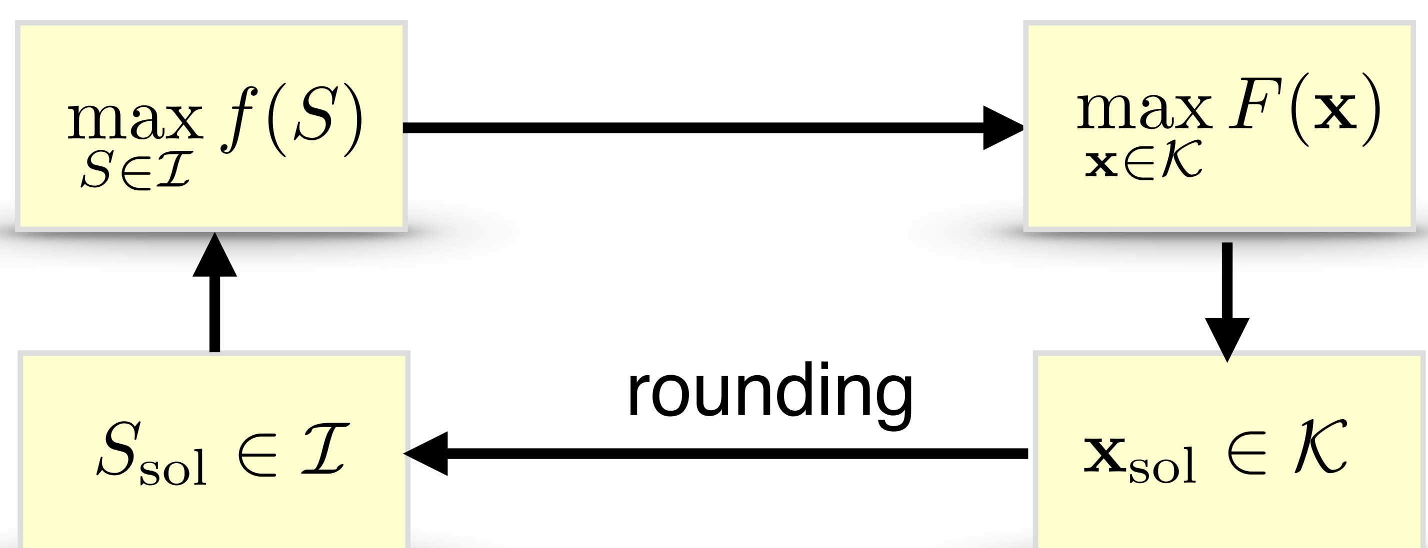
Rounding



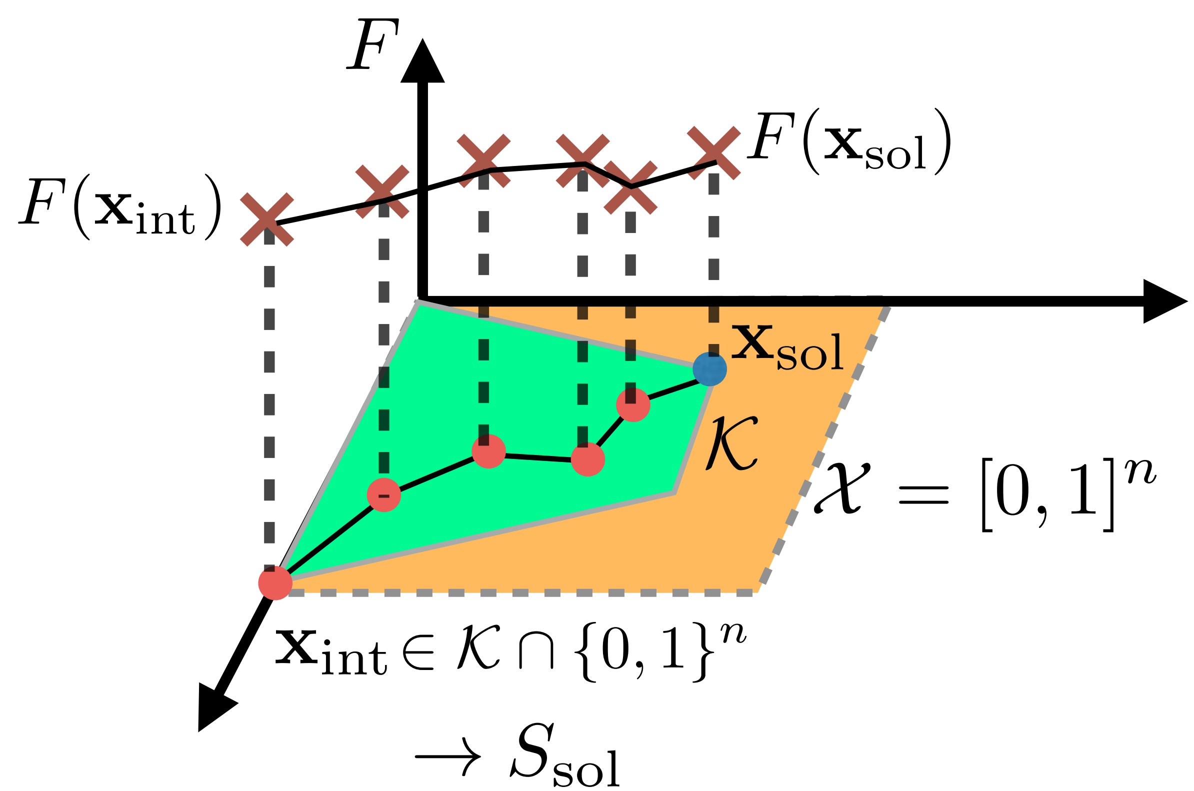
$$
\begin{tikzcd}
\max\limits_{S \in \mathcal{I}} f(S) \arrow[r] & \max\limits_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x}) \arrow[d] \\
S_{\text{sol}} \in \mathcal{I} \arrow[u] & \mathbf{x}_{\text{sol}} \in \mathcal{K} \arrow[l, "\text{rounding}"']
\end{tikzcd}
$$

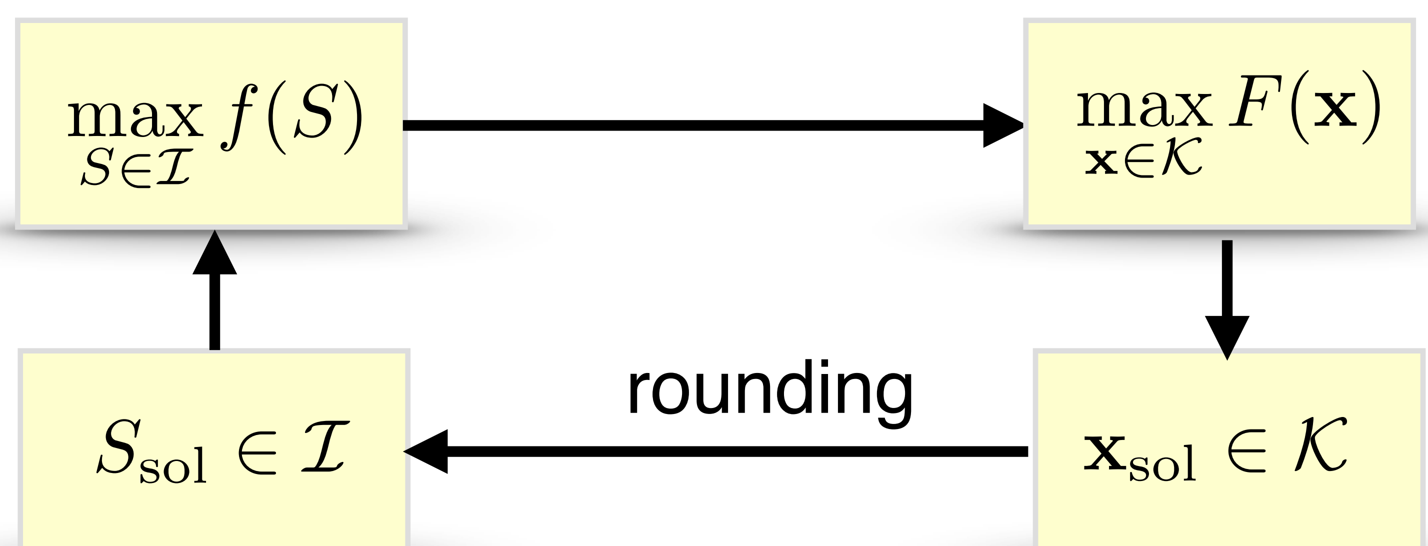
Rounding


F
$F(\mathbf{x}_{\mathrm{sol}})$
$\mathbf{x}_{\mathrm{sol}}$
$\mathcal{K}$
$\mathcal{X} = [0,1]^n$
$\mathbf{x}_{\mathrm{int}} \in \mathcal{K} \cap \{0,1\}^n$
$\to S_{\mathrm{sol}}$


$\max_{S \in \mathcal{I}} f(S)$
$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$
rounding
$S_{\mathrm{sol}} \in \mathcal{I}$
$\mathbf{x}_{\mathrm{sol}} \in \mathcal{K}$

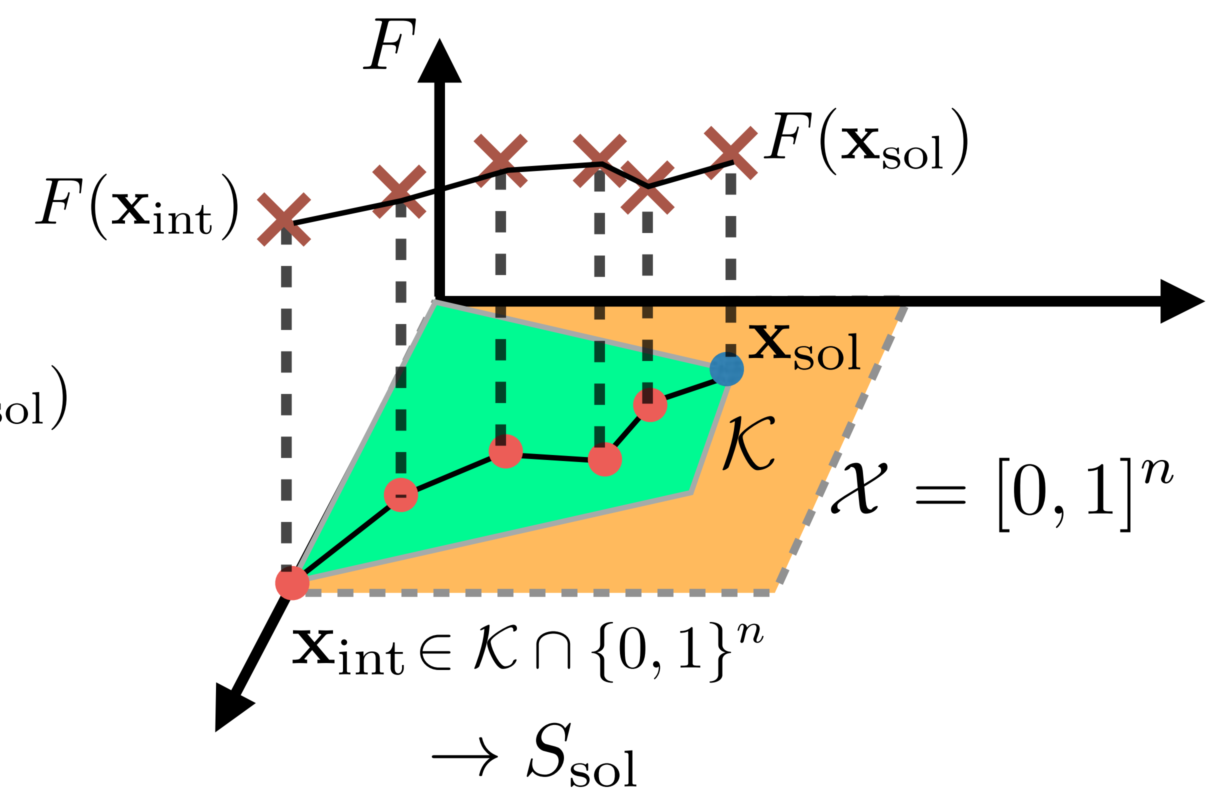
Rounding


F
$F(\mathbf{x}_{\text{int}})$
$F(\mathbf{x}_{\text{sol}})$
$\mathbf{x}_{\text{sol}}$
$\mathcal{K}$
$\mathcal{X} = [0,1]^n$
$\mathbf{x}_{\text{int}} \in \mathcal{K} \cap \{0,1\}^n$
$\to S_{\text{sol}}$


$\max_{S \in \mathcal{I}} f(S)$
$\max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$
rounding
$S_{\text{sol}} \in \mathcal{I}$
$\mathbf{x}_{\text{sol}} \in \mathcal{K}$

Rounding

$$f(S_{\text{sol}}) = F(\mathbf{x}_{\text{int}}) \geq F(\mathbf{x}_{\text{sol}})$$



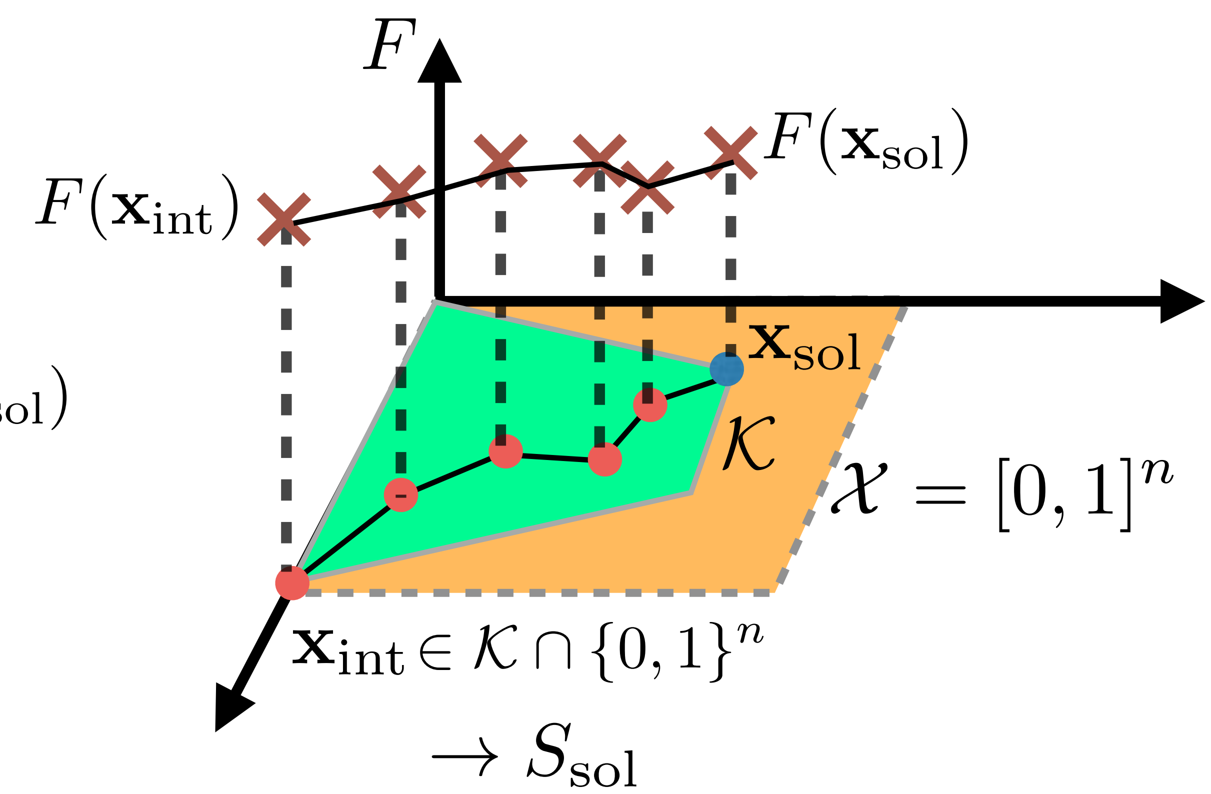
$$\begin{tikzcd}
\max_{S \in \mathcal{I}} f(S) \arrow[r] & \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x}) \arrow[d] \\
S_{\text{sol}} \in \mathcal{I} \arrow[u] & \mathbf{x}_{\text{sol}} \in \mathcal{K} \arrow[l, "\text{rounding}"']
\end{tikzcd}$$

Yale

Penn

Rounding

$$f(S_{\text{sol}}) = F(\mathbf{x}_{\text{int}}) \geq F(\mathbf{x}_{\text{sol}})$$



Several rounding techniques:

- Pipage rounding **Calinescu et al, 2007, 2011**
- Swap rounding **Chekuri et al, 2010**
- Contention resolution schemes **Chekuri et al, 2014**

Pipage Rounding

- Used for Matroid constraints
- We describe the special case of cardinality constraints (for simplicity)

Pipage Rounding

Yale Penn

Pipage Rounding

$\mathbf{x} = (x_1, \cdots, x_n) \in [0,1]^n$ → $\mathbf{x}_{\text{int}} = (x_1, \cdots, x_n) \in \{0,1\}^n$

$$\sum_{i=1}^{n} x_i = k$$

$$\sum_{i=1}^{n} x_i = k$$

Yale Penn

Pipage Rounding

$\mathbf{x} = (x_1, \cdots, x_n) \in [0,1]^n$ → $\mathbf{x}_{\text{int}} = (x_1, \cdots, x_n) \in \{0,1\}^n$

$$\sum_{i=1}^{n} x_i = k \qquad \sum_{i=1}^{n} x_i = k$$

while $\mathbf{x}$ is not integer-valued:

pick two non-integer coordinates x_i, x_j

if $x_i + x_j \leq 1$:

Let
$$\mathbf{x}_1 = \mathbf{x} + x_j(\mathbf{e}_i - \mathbf{e}_j)$$
$$\mathbf{x}_2 = \mathbf{x} - x_i(\mathbf{e}_i - \mathbf{e}_j)$$

if $F(\mathbf{x}_1) \geq F(\mathbf{x}_2)$: $\mathbf{x} \leftarrow \mathbf{x}_1$

else: $\mathbf{x} \leftarrow \mathbf{x}_2$

Pipage Rounding

$$\mathbf{x} = (x_1, \cdots, x_n) \in [0,1]^n \quad \longrightarrow \quad \mathbf{x}_{\text{int}} = (x_1, \cdots, x_n) \in \{0,1\}^n$$

$$\sum_{i=1}^{n} x_i = k \qquad\qquad \sum_{i=1}^{n} x_i = k$$

while $\mathbf{x}$ is not integer-valued:

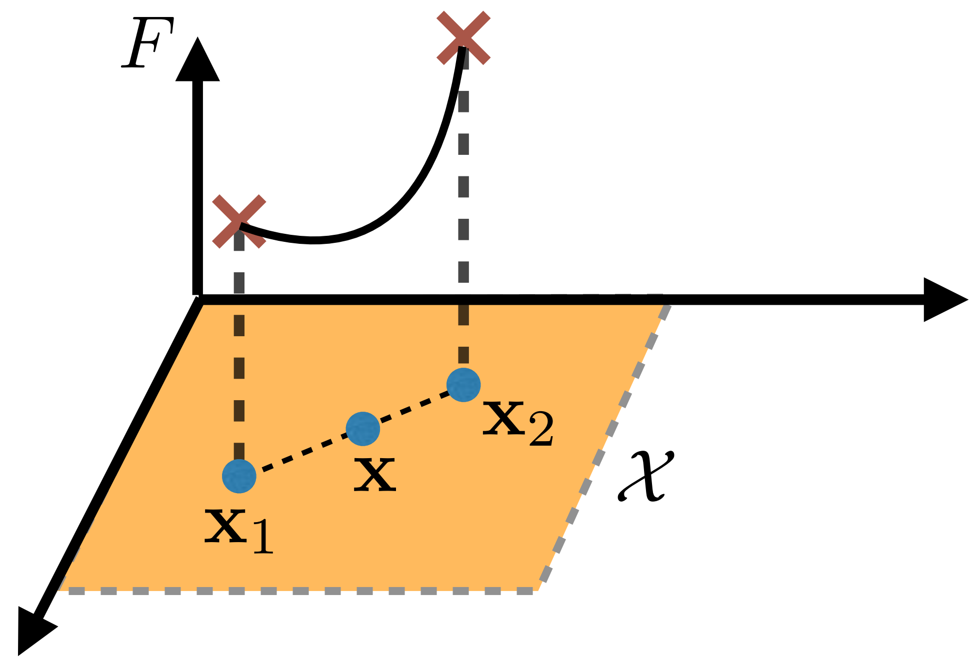
pick two non-integer coordinates x_i, x_j

if $x_i + x_j \leq 1$:

Let
$$\mathbf{x}_1 = \mathbf{x} + x_j(\mathbf{e}_i - \mathbf{e}_j)$$
$$\mathbf{x}_2 = \mathbf{x} - x_i(\mathbf{e}_i - \mathbf{e}_j)$$

if $F(\mathbf{x}_1) \geq F(\mathbf{x}_2)$: $\mathbf{x} \leftarrow \mathbf{x}_1$

else: $\mathbf{x} \leftarrow \mathbf{x}_2$



$$\begin{array}{c} \\ \mathbf{x}_1 = \end{array}\begin{array}{c} \\ (\cdots, \end{array}\begin{array}{c} i \\ \downarrow \\ x_i + x_j, \end{array}\begin{array}{c} \\ \\ \cdots, \end{array}\begin{array}{c} j \\ \downarrow \\ 0, \end{array}\begin{array}{c} \\ \\ \cdots) \end{array}$$

$$\begin{array}{c} \\ \mathbf{x}_2 = \end{array}\begin{array}{c} \\ (\cdots, \end{array}\begin{array}{c} i \\ \downarrow \\ 0, \end{array}\begin{array}{c} \\ \\ \cdots, \end{array}\begin{array}{c} j \\ \downarrow \\ x_i + x_j, \end{array}\begin{array}{c} \\ \\ \cdots) \end{array}$$

Pipage Rounding

$$\mathbf{x} = (x_1, \cdots, x_n) \in [0,1]^n \qquad \rightarrow \qquad \mathbf{x}_{\text{int}} = (x_1, \cdots, x_n) \in \{0,1\}^n$$

$$\sum_{i=1}^{n} x_i = k \qquad\qquad \sum_{i=1}^{n} x_i = k$$

while $\mathbf{x}$ is not integer-valued:

pick two non-integer coordinates x_i, x_j

if $x_i + x_j \leq 1$:

Let
$$\mathbf{x}_1 = \mathbf{x} + x_j(\mathbf{e}_i - \mathbf{e}_j)$$
$$\mathbf{x}_2 = \mathbf{x} - x_i(\mathbf{e}_i - \mathbf{e}_j)$$

if $F(\mathbf{x}_1) \geq F(\mathbf{x}_2)$: $\mathbf{x} \leftarrow \mathbf{x}_1$

else: $\mathbf{x} \leftarrow \mathbf{x}_2$

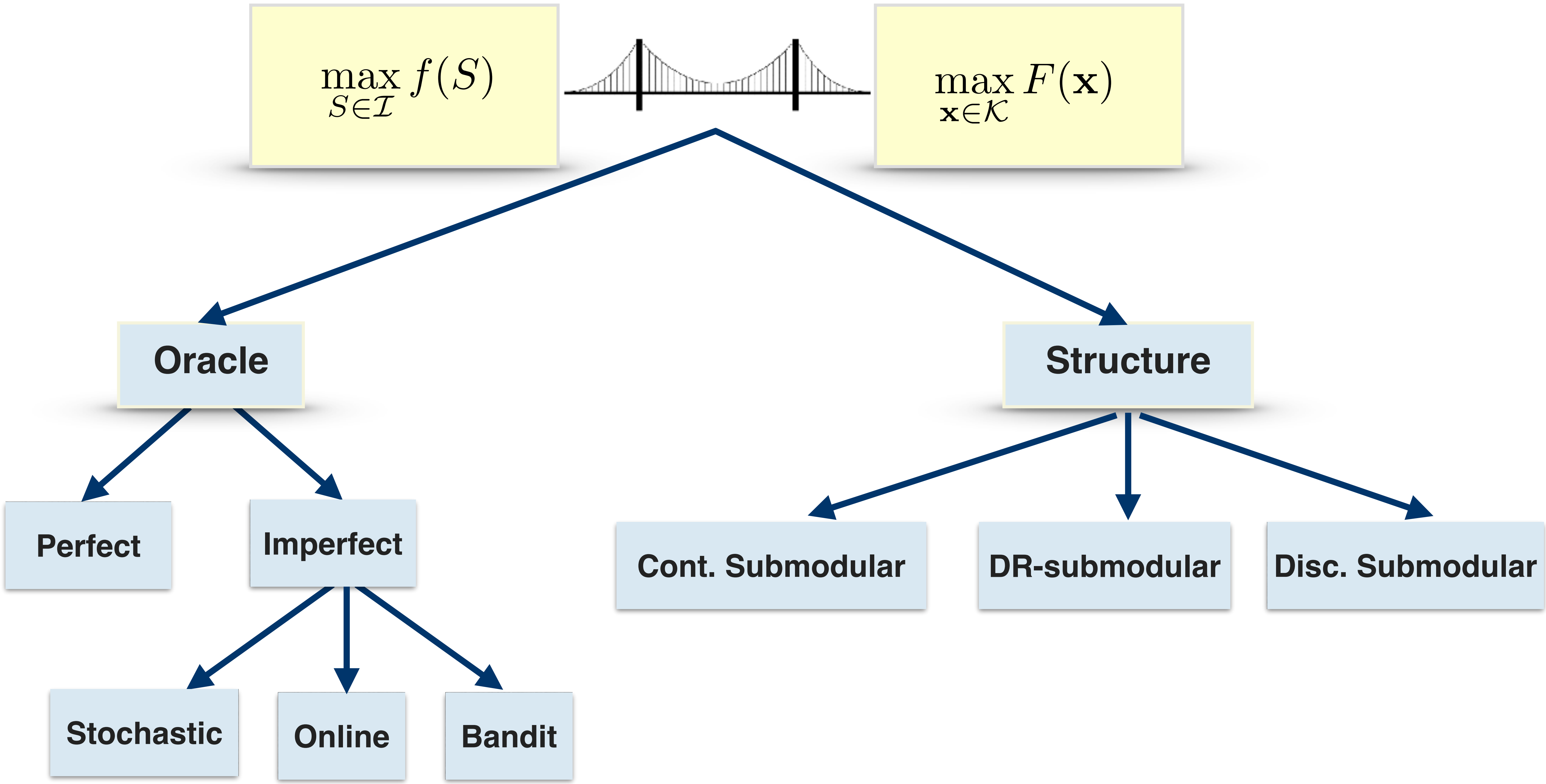
if $x_i + x_j > 1$:

Let
$$\mathbf{x}_1 = \mathbf{x} + (x_j - 1)(\mathbf{e}_i - \mathbf{e}_j)$$
$$\mathbf{x}_2 = \mathbf{x} - (x_i - 1)(\mathbf{e}_i - \mathbf{e}_j)$$

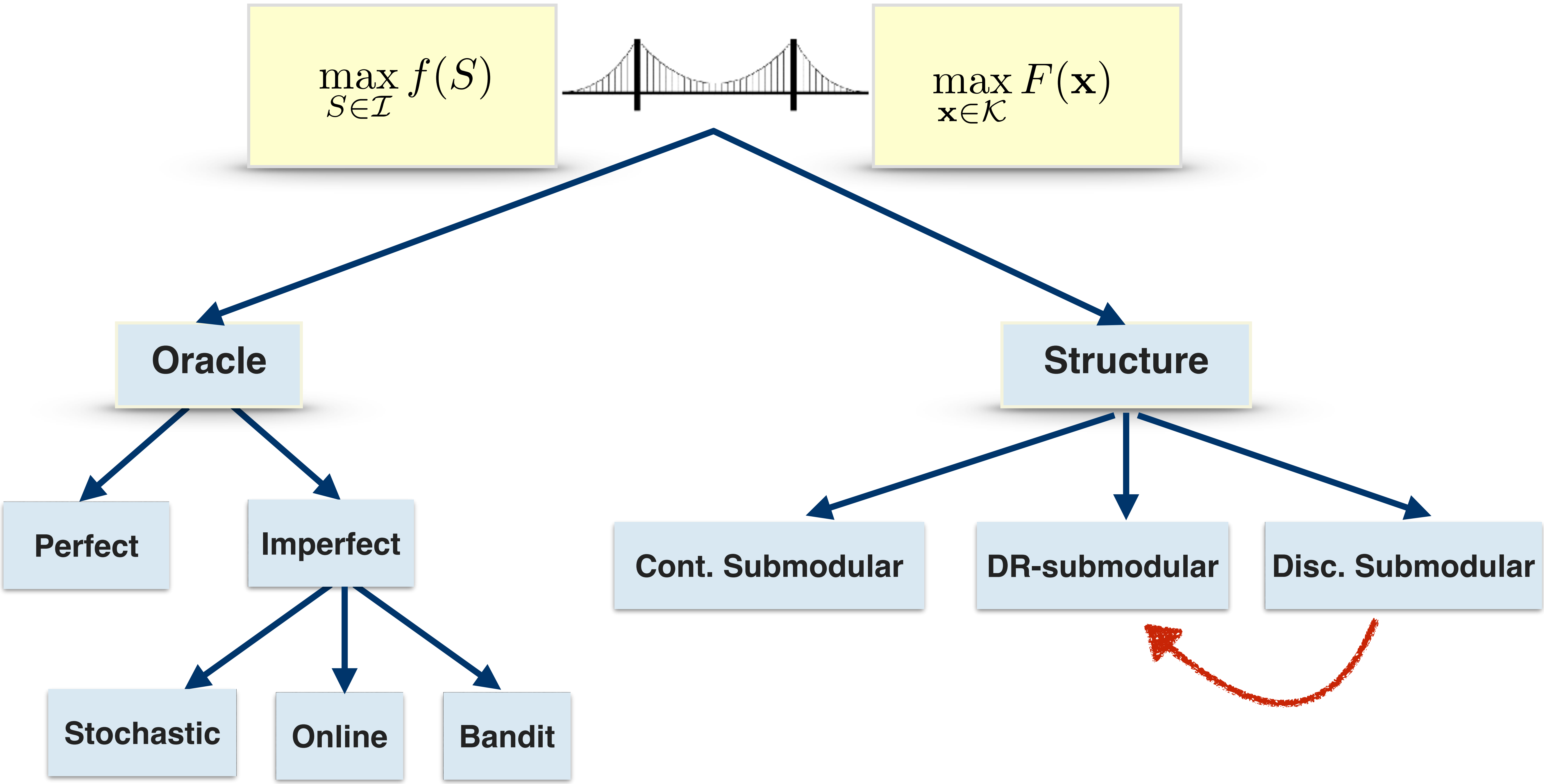
if $F(\mathbf{x}_1) \geq F(\mathbf{x}_2)$: $\mathbf{x} \leftarrow \mathbf{x}_1$

else: $\mathbf{x} \leftarrow \mathbf{x}_2$

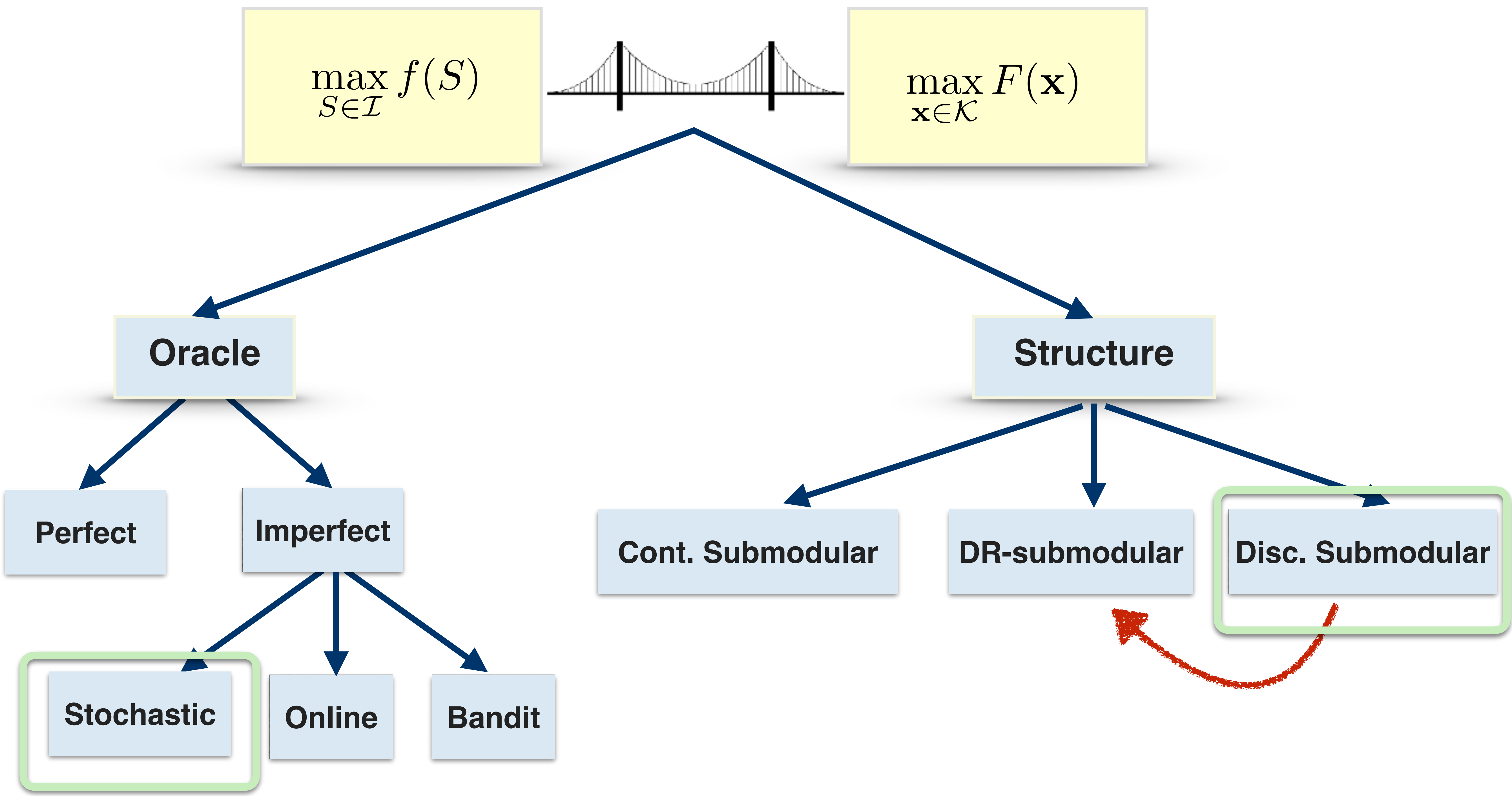
Submodular Maximization: Discrete and Continuous



Submodular Maximization: Discrete and Continuous



Submodular Maximization: Discrete and Continuous



Stochastic Submodular Maximization: Discrete

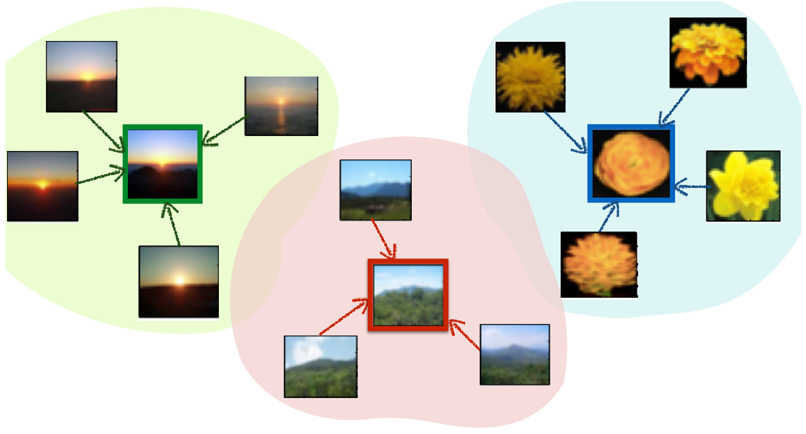
$$\max_{S \in \mathcal{I}} f(S) \doteq \max_{S \in \mathcal{I}} \mathop{\mathbb{E}}_{\theta \sim D} [f_\theta(S)]$$

Stochastic Submodular Maximization: Discrete

$$\max_{S\in\mathcal{I}} f(S) \doteq \max_{S\in\mathcal{I}} \mathop{\mathbb{E}}_{\theta\sim D}\left[f_\theta(S)\right]$$

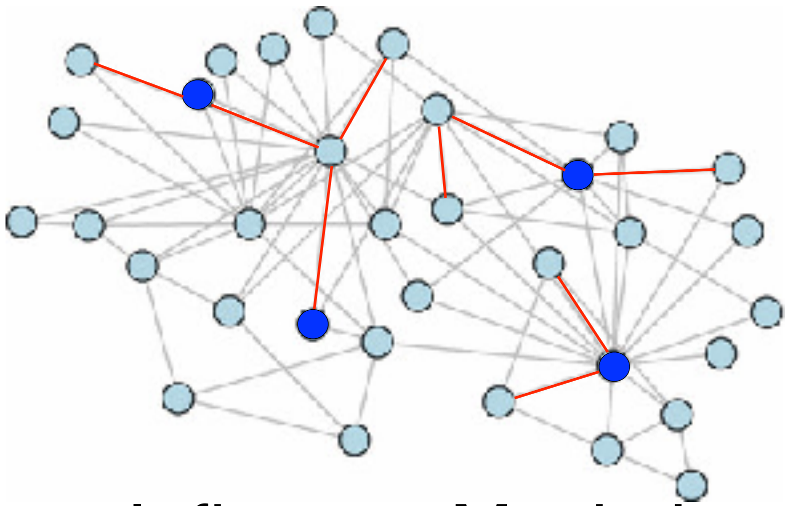
Stochastic Submodular Maximization: Discrete

$$\max_{S \in \mathcal{I}} f(S) \doteq \max_{S \in \mathcal{I}} \mathop{\mathbb{E}}_{\theta \sim D} [f_\theta(S)]$$



Data Summarization

$$f(S) = \frac{1}{|V|} \sum_{i \in V} f_i(S)$$



Influence Maximization

$$f(S) = \mathbb{E}_{H \sim D}[f_H(S)]$$

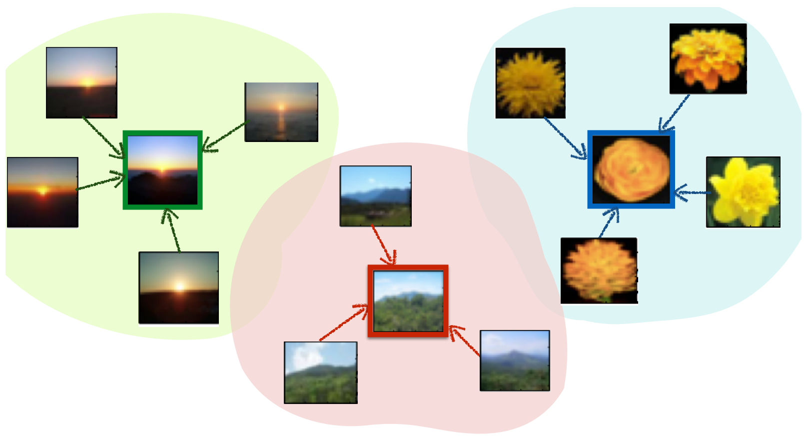
Yale

Penn

Stochastic Submodular Maximization: Discrete

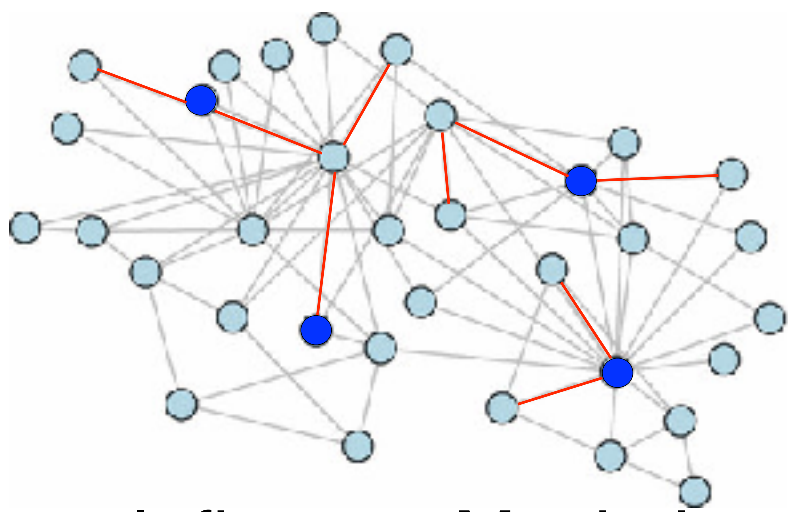
$$\max_{S \in \mathcal{I}} f(S) \doteq \max_{S \in \mathcal{I}} \mathop{\mathbb{E}}_{\theta \sim D} [f_\theta(S)]$$

- We see f through its samples, i.e., $f_{\theta_1}, f_{\theta_2}, \cdots$



Data Summarization

$$f(S) = \frac{1}{|V|} \sum_{i \in V} f_i(S)$$



Influence Maximization

$$f(S) = \mathbb{E}_{H \sim D}[f_H(S)]$$

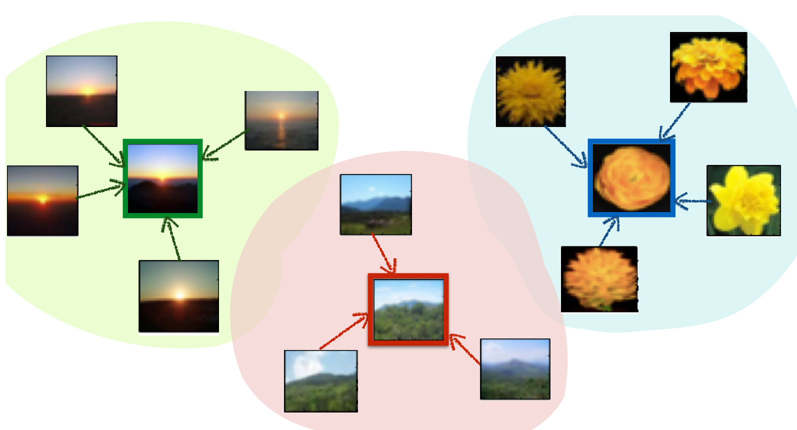
Stochastic Submodular Maximization: Discrete

$$\max_{S \in \mathcal{I}} f(S) \doteq \max_{S \in \mathcal{I}} \mathbb{E}_{\theta \sim D} [f_\theta(S)]$$

- We see f through its samples, i.e., $f_{\theta_1}, f_{\theta_2}, \cdots$

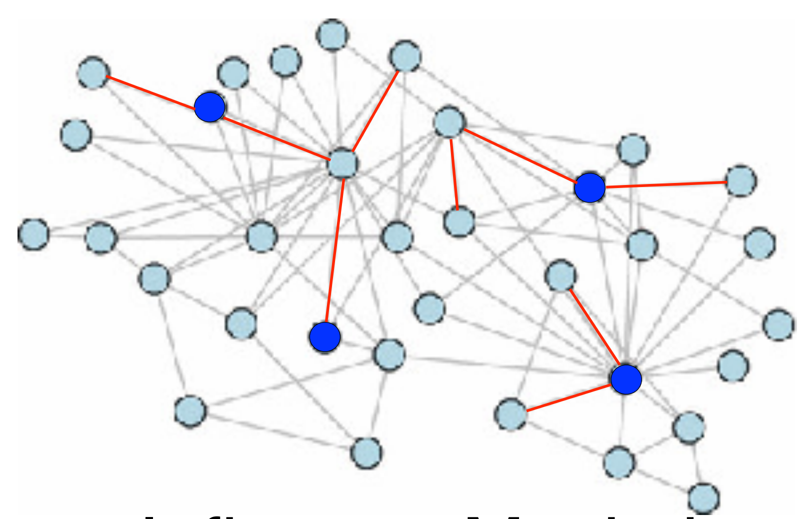
stochastic oracle

$S \uparrow \qquad \downarrow f_\theta(S)$



Data Summarization

$$f(S) = \frac{1}{|V|} \sum_{i \in V} f_i(S)$$



Influence Maximization

$$f(S) = \mathbb{E}_{H \sim D}[f_H(S)]$$

Yale Penn

Stochastic Submodular Maximization: Discrete

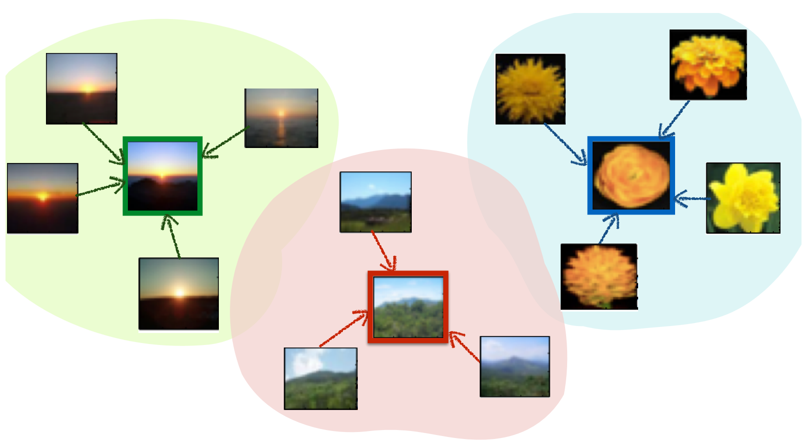
$$\max_{S \in \mathcal{I}} f(S) \doteq \max_{S \in \mathcal{I}} \mathop{\mathbb{E}}_{\theta \sim D} [f_\theta(S)]$$

- We see f through its samples, i.e., $f_{\theta_1}, f_{\theta_2}, \cdots$

stochastic oracle

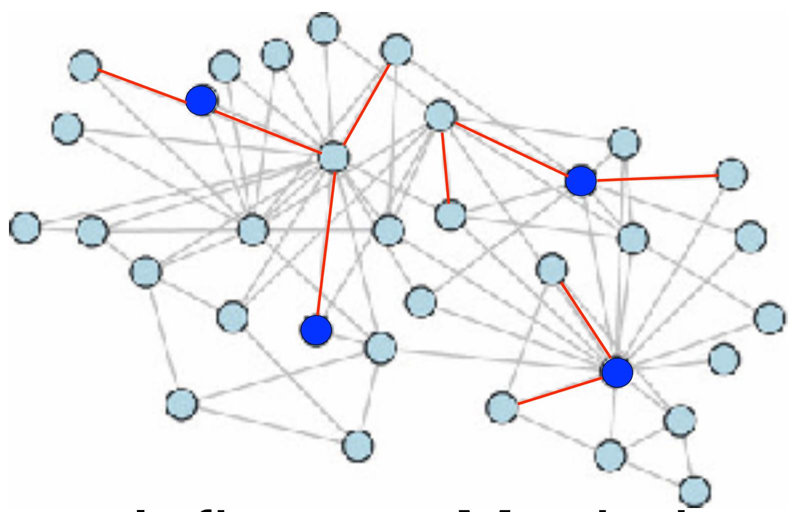
$S \uparrow \quad \downarrow f_\theta(S)$

$$\mathbb{E}_{\theta \sim D}[f_\theta(S)] = f(S)$$



Data Summarization

$$f(S) = \frac{1}{|V|} \sum_{i \in V} f_i(S)$$



Influence Maximization

$$f(S) = \mathbb{E}_{H \sim D}[f_H(S)]$$

Stochastic Submodular Maximization: Discrete and Continuous

$$f(S) = \mathbb{E}_{\theta\sim D}[f_\theta(S)] \xrightarrow{\text{multi-linear ext.}} F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[F_\theta(\mathbf{x})]$$

Stochastic Submodular Maximization: Discrete and Continuous

$$f(S) = \mathbb{E}_{\theta\sim D}[f_\theta(S)] \xrightarrow{\text{multi-linear ext.}} F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[F_\theta(\mathbf{x})]$$

Stochastic Submodular Maximization: Discrete and Continuous

$$f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \xrightarrow{\text{multi-linear ext.}} F(\mathbf{x}) = \mathbb{E}_{\theta \sim D}[F_\theta(\mathbf{x})]$$

$$\nabla F(\mathbf{x}) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(\mathbf{x})]$$

Stochastic Submodular Maximization: Discrete and Continuous

$$f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \xrightarrow{\text{multi-linear ext.}} F(\mathbf{x}) = \mathbb{E}_{\theta \sim D}[F_\theta(\mathbf{x})]$$

$$\nabla F(\mathbf{x}) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(\mathbf{x})]$$

$$\frac{\partial F(\mathbf{x})}{\partial x_i} = \mathbb{E}_{\theta \sim D, S \sim \mathbf{x}}\left[f_\theta(S \cup \{i\}) - f_\theta(S \setminus \{i\})\right]$$

Stochastic Submodular Maximization: Discrete and Continuous

$$f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \xrightarrow{\text{multi-linear ext.}} F(\mathbf{x}) = \mathbb{E}_{\theta \sim D}[F_\theta(\mathbf{x})]$$

$$\nabla F(\mathbf{x}) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(\mathbf{x})]$$

$$\frac{\partial F(\mathbf{x})}{\partial x_i} = \mathbb{E}_{\theta \sim D, S \sim \mathbf{x}}\Big[\overbrace{f_\theta(S \cup \{i\}) - f_\theta(S \setminus \{i\})}^{\text{unbiased estimate } \mathbf{g}_i}\Big]$$

Yale Penn

Stochastic Submodular Maximization: Discrete and Continuous

$$f(S) = \mathbb{E}_{\theta\sim D}[f_\theta(S)] \xrightarrow{\text{multi-linear ext.}} F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[F_\theta(\mathbf{x})]$$

$$\nabla F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[\nabla F_\theta(\mathbf{x})]$$

$$\frac{\partial F(\mathbf{x})}{\partial x_i} = \mathbb{E}_{\theta\sim D, S\sim\mathbf{x}}\Big[\overbrace{f_\theta(S\cup\{i\}) - f_\theta(S\setminus\{i\})}^{\text{unbiased estimate } \mathbf{g}_i}\Big]$$

stochastic oracle

$S \uparrow$ $\quad\downarrow f_\theta(S)$

$[f_\theta(S)] = f(S)$

Stochastic Submodular Maximization: Discrete and Continuous

$$f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \xrightarrow{\text{multi-linear ext.}} F(\mathbf{x}) = \mathbb{E}_{\theta \sim D}[F_\theta(\mathbf{x})]$$

$$\nabla F(\mathbf{x}) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(\mathbf{x})]$$

$$\frac{\partial F(\mathbf{x})}{\partial x_i} = \mathbb{E}_{\theta \sim D, S \sim \mathbf{x}}\Big[\overbrace{f_\theta(S \cup \{i\}) - f_\theta(S \setminus \{i\})}^{\text{unbiased estimate } \mathbf{g}_i}\Big]$$

stochastic oracle $\longleftrightarrow$ stochastic first order oracle

$[f_\theta(S)] = f(S)$ $\quad S \uparrow \quad \downarrow f_\theta(S)$

$\mathbf{x} \uparrow \quad \downarrow \mathbf{g} \quad \mathbb{E}[\mathbf{g}] = \nabla F(\mathbf{x})$

Stochastic Submodular Maximization: Discrete and Continuous

$$f(S) = \mathbb{E}_{\theta\sim D}[f_\theta(S)] \xrightarrow{\text{multi-linear ext.}} F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[F_\theta(\mathbf{x})]$$

$$\nabla F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[\nabla F_\theta(\mathbf{x})]$$

$$\frac{\partial F(\mathbf{x})}{\partial x_i} = \mathbb{E}_{\theta\sim D, S\sim\mathbf{x}}\Big[\overbrace{f_\theta(S\cup\{i\}) - f_\theta(S\setminus\{i\})}^{\text{unbiased estimate } \mathbf{g}_i}\Big]$$

stochastic oracle $\longleftrightarrow$ stochastic first order oracle

$[f_\theta(S)] = f(S)$ $\quad S \uparrow \quad \downarrow f_\theta(S)$

$$\max_{S\in\mathcal{I}} f(S)$$

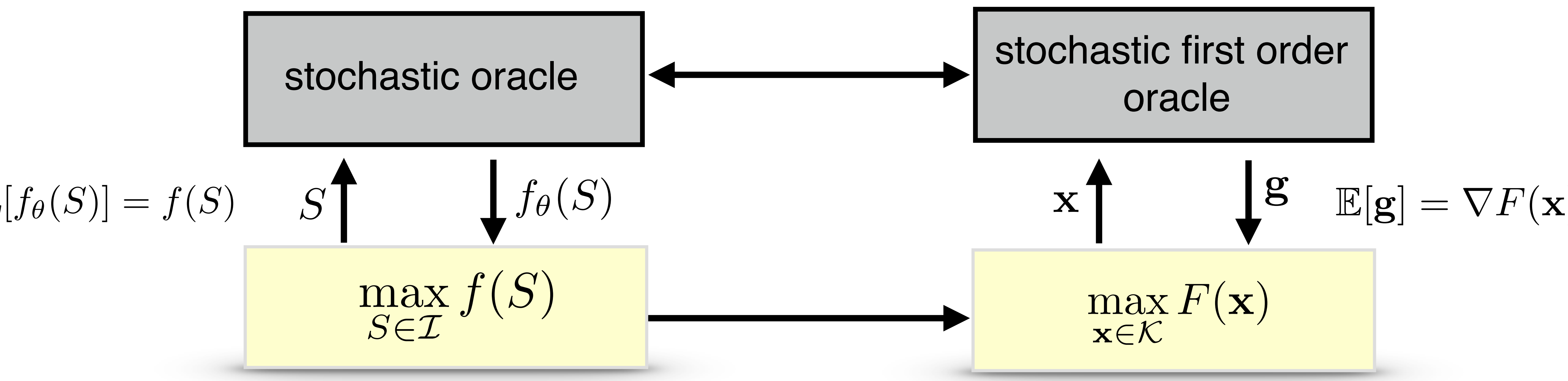
$\mathbf{x} \uparrow \quad \downarrow \mathbf{g} \quad \mathbb{E}[\mathbf{g}] = \nabla F(\mathbf{x})$

Stochastic Submodular Maximization: Discrete and Continuous

$$f(S) = \mathbb{E}_{\theta\sim D}[f_\theta(S)] \xrightarrow{\text{multi-linear ext.}} F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[F_\theta(\mathbf{x})]$$

$$\nabla F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[\nabla F_\theta(\mathbf{x})]$$

$$\frac{\partial F(\mathbf{x})}{\partial x_i} = \mathbb{E}_{\theta\sim D, S\sim\mathbf{x}}\big[\overbrace{f_\theta(S\cup\{i\}) - f_\theta(S\setminus\{i\})}^{\text{unbiased estimate } \mathbf{g}_i}\big]$$

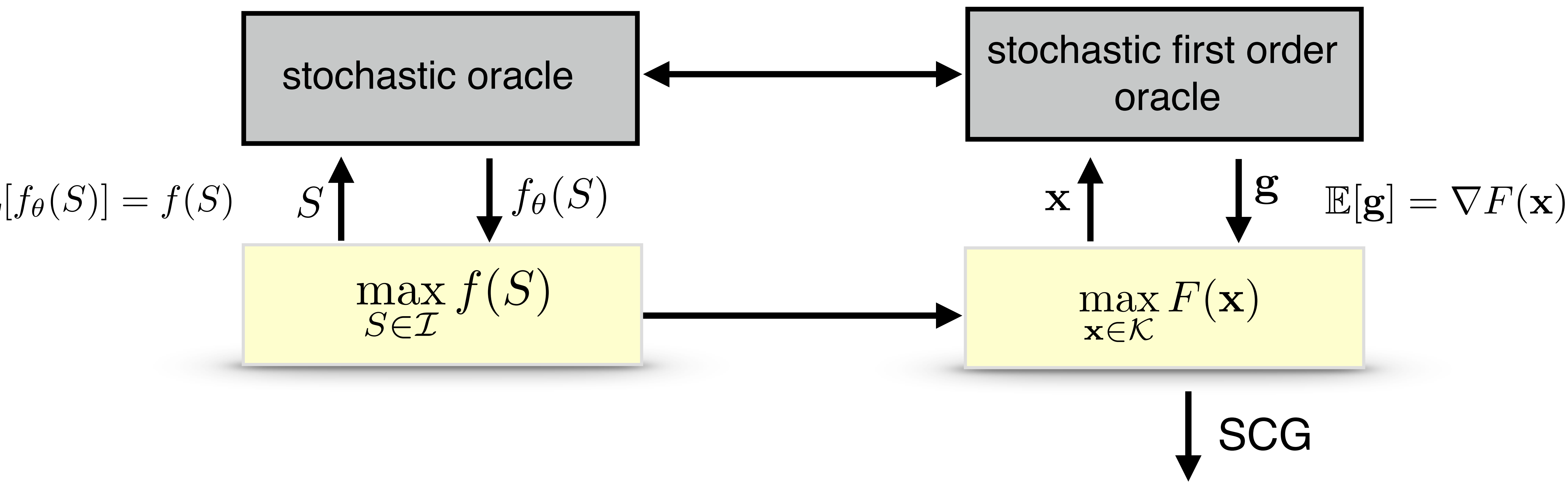


Stochastic Submodular Maximization: Discrete and Continuous

$$f(S) = \mathbb{E}_{\theta\sim D}[f_\theta(S)] \xrightarrow{\text{multi-linear ext.}} F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[F_\theta(\mathbf{x})]$$

$$\nabla F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[\nabla F_\theta(\mathbf{x})]$$

$$\frac{\partial F(\mathbf{x})}{\partial x_i} = \mathbb{E}_{\theta\sim D, S\sim\mathbf{x}}\Big[\underbrace{f_\theta(S\cup\{i\}) - f_\theta(S\setminus\{i\})}_{\text{unbiased estimate } \mathbf{g}_i}\Big]$$



$$F(\mathbf{x}_{\mathrm{sol}}) \geq (1 - \frac{1}{e})\mathrm{OPT} - \epsilon$$

Stochastic Submodular Maximization: Discrete and Continuous

$$f(S) = \mathbb{E}_{\theta\sim D}[f_\theta(S)] \xrightarrow{\text{multi-linear ext.}} F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[F_\theta(\mathbf{x})]$$

$$\nabla F(\mathbf{x}) = \mathbb{E}_{\theta\sim D}[\nabla F_\theta(\mathbf{x})]$$

$$\frac{\partial F(\mathbf{x})}{\partial x_i} = \mathbb{E}_{\theta\sim D, S\sim\mathbf{x}}\Big[\overbrace{f_\theta(S\cup\{i\}) - f_\theta(S\setminus\{i\})}^{\text{unbiased estimate } \mathbf{g}_i}\Big]$$

stochastic oracle $\longleftrightarrow$ stochastic first order oracle

$[f_\theta(S)] = f(S)$ $S \uparrow$ $\downarrow f_\theta(S)$ $\mathbf{x} \uparrow$ $\downarrow \mathbf{g}$ $\mathbb{E}[\mathbf{g}] = \nabla F(\mathbf{x})$

$$\max_{S\in\mathcal{I}} f(S) \longrightarrow \max_{\mathbf{x}\in\mathcal{K}} F(\mathbf{x})$$

SCG

$$f(S_{\text{sol}}) \geq (1-\frac{1}{e})\text{OPT} - \epsilon \xleftarrow{\text{rounding}} F(\mathbf{x}_{\text{sol}}) \geq (1-\frac{1}{e})\text{OPT} - \epsilon$$

Yale   Penn

Discrete vs. Continuous

- Discrete submodular optimization with matroid constraints:

$$O(\frac{n^8}{\epsilon^4}) \longrightarrow O(\frac{n^2}{\epsilon^4}) \longrightarrow O(\frac{n^2}{\epsilon^2})$$

Calinescu et al, 2011 **Badanidiyuru, Vondrak, 2013** **SCG++, 2019**

"Maximizing a monotone submodular function subject to a matroid constraint", Calinescu, Chekuri, Pal, Vondrák, 2011

"Fast algorithms for maximizing submodular functions", Badanidiyuru, Vondrak, 2013

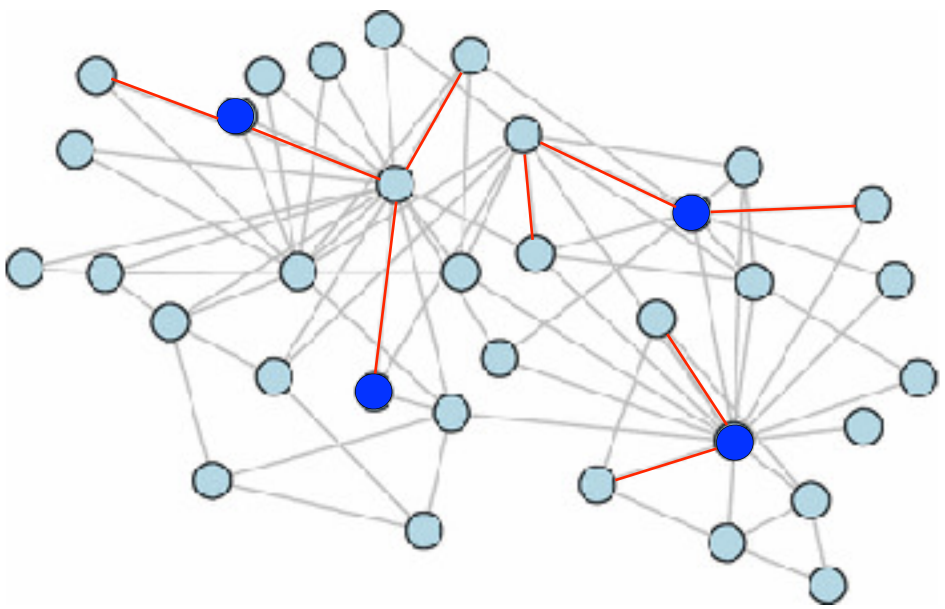
"Stochastic Continuous Greedy++: When Upper and Lower Bounds Match", Hassani, Karbasi, Mokhtari, Shen, 2019

Discrete vs. Continuous

- Discrete submodular optimization with matroid constraints:

$$O(\frac{n^8}{\epsilon^4}) \longrightarrow O(\frac{n^2}{\epsilon^4}) \longrightarrow O(\frac{n^2}{\epsilon^2})$$

- Epinions social network (79k nodes, 580k edges)



Influence Maximization

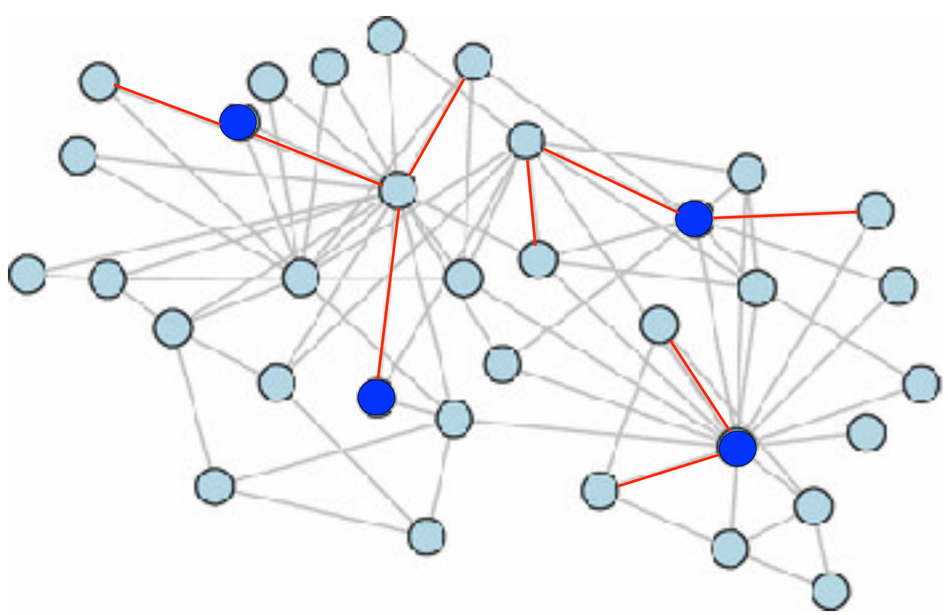
$$f(S) = \mathbb{E}_{H\sim D}[f_H(S)]$$

Discrete vs. Continuous

- Discrete submodular optimization with matroid constraints:

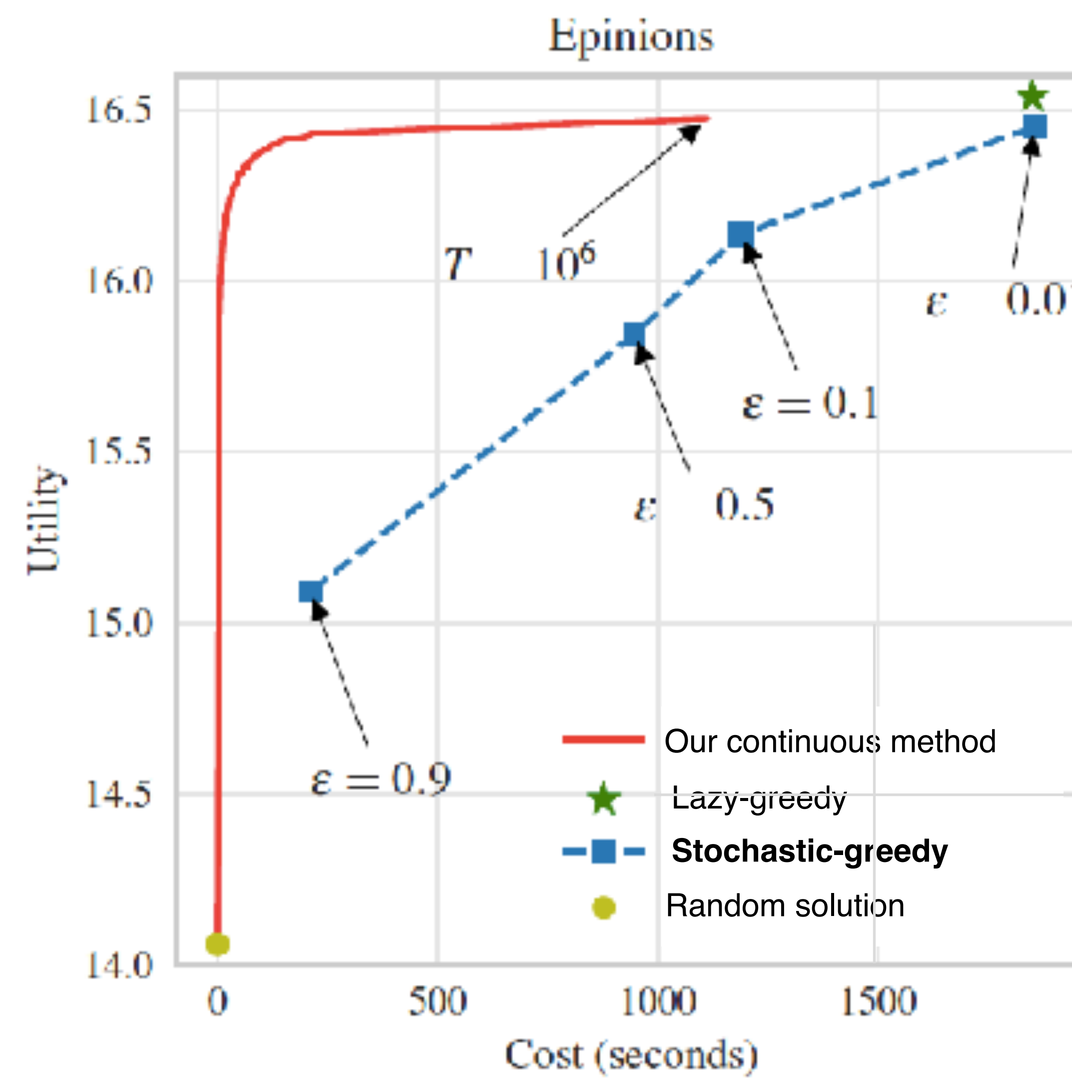
$$O(\frac{n^8}{\epsilon^4}) \longrightarrow O(\frac{n^2}{\epsilon^4}) \longrightarrow O(\frac{n^2}{\epsilon^2})$$

- Epinions social network (79k nodes, 580k edges)



Influence Maximization

$$f(S) = \mathbb{E}_{H\sim D}[f_H(S)]$$



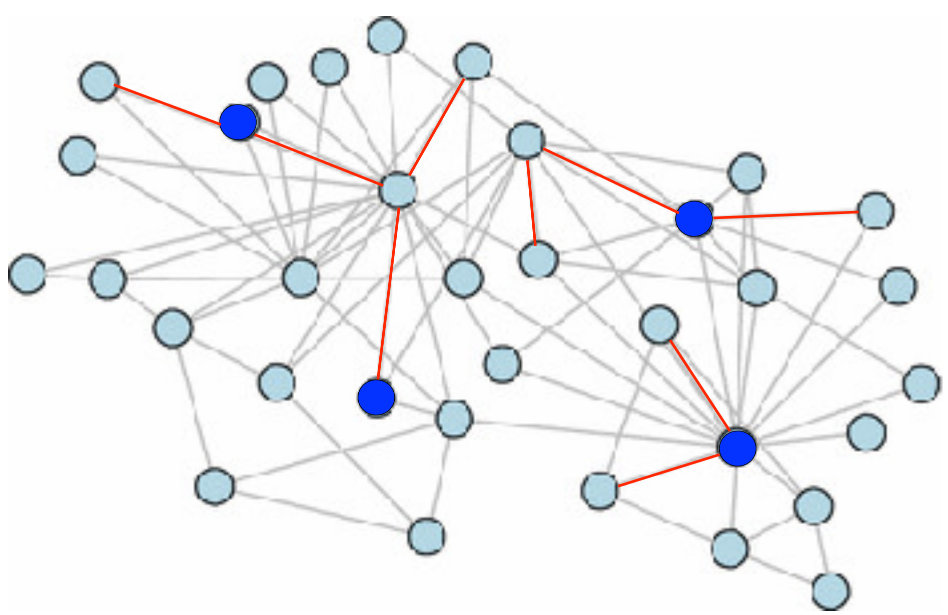
"Stochastic Submodular Maximization: The Case of Coverage Functions", Karimi et al.

Discrete vs. Continuous

- Discrete submodular optimization with matroid constraints:

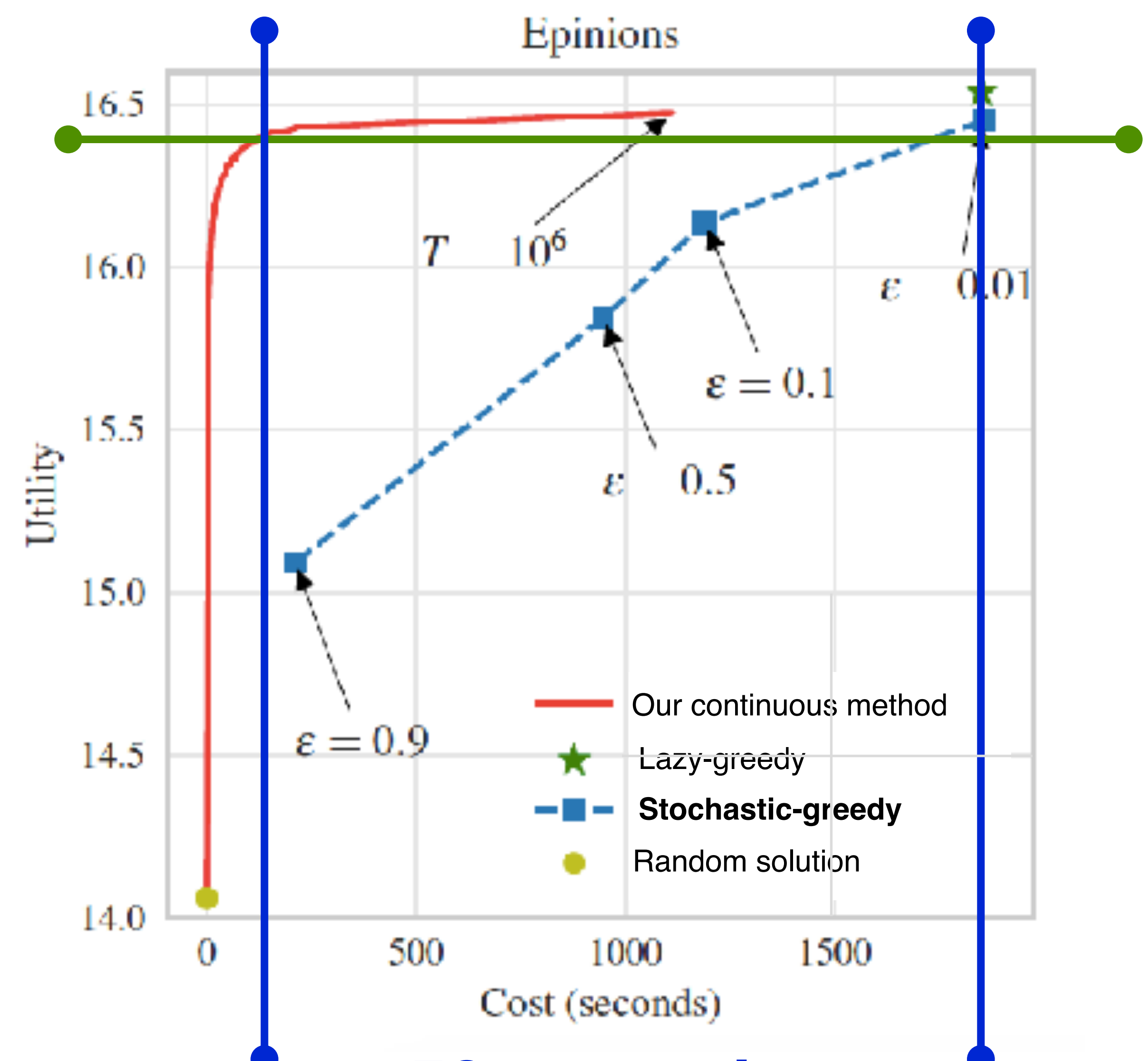
$$O(\frac{n^8}{\epsilon^4}) \longrightarrow O(\frac{n^2}{\epsilon^4}) \longrightarrow O(\frac{n^2}{\epsilon^2})$$

- Epinions social network (79k nodes, 580k edges)



Influence Maximization

$$f(S) = \mathbb{E}_{H\sim D}[f_H(S)]$$

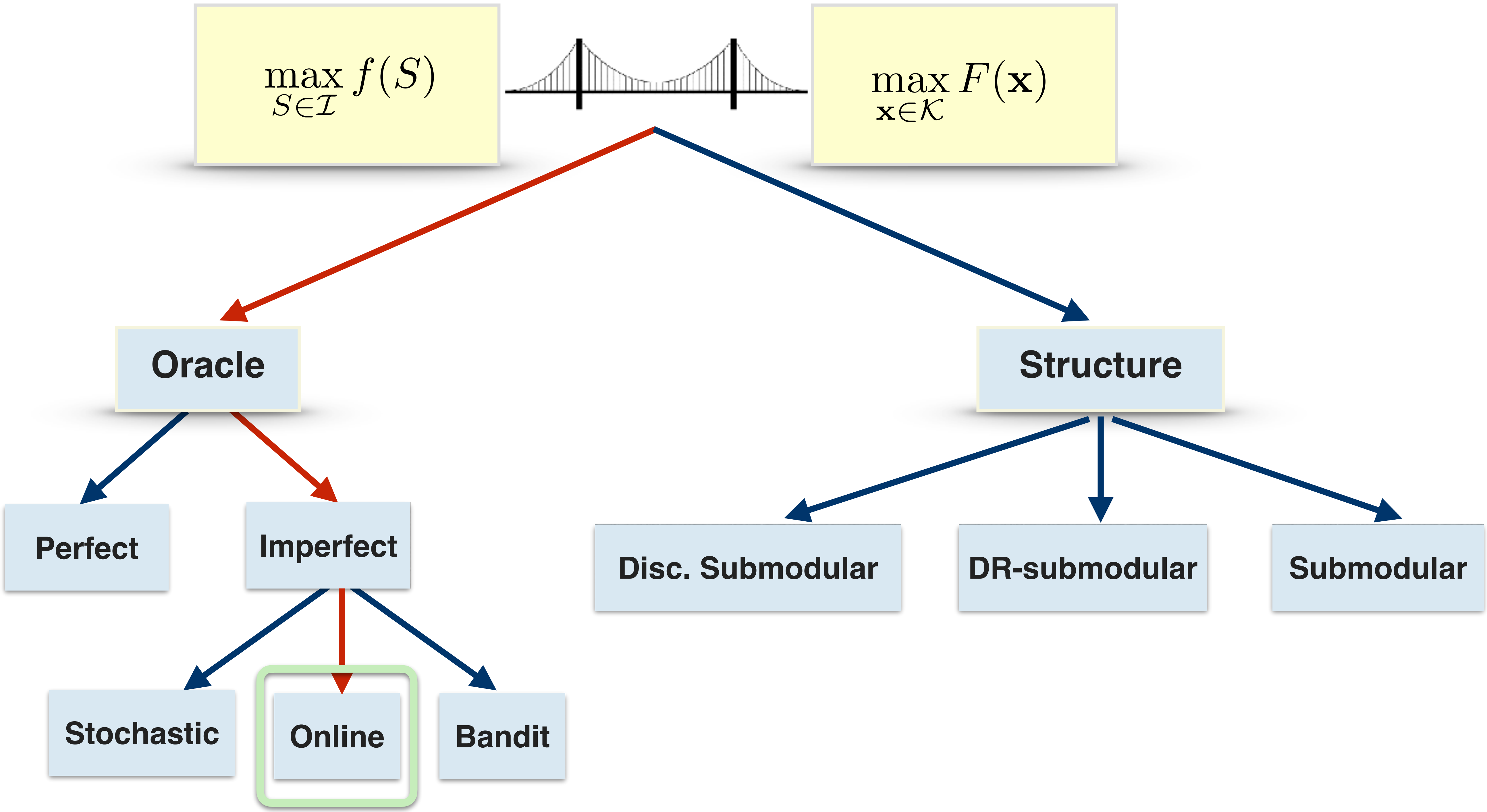


Yale

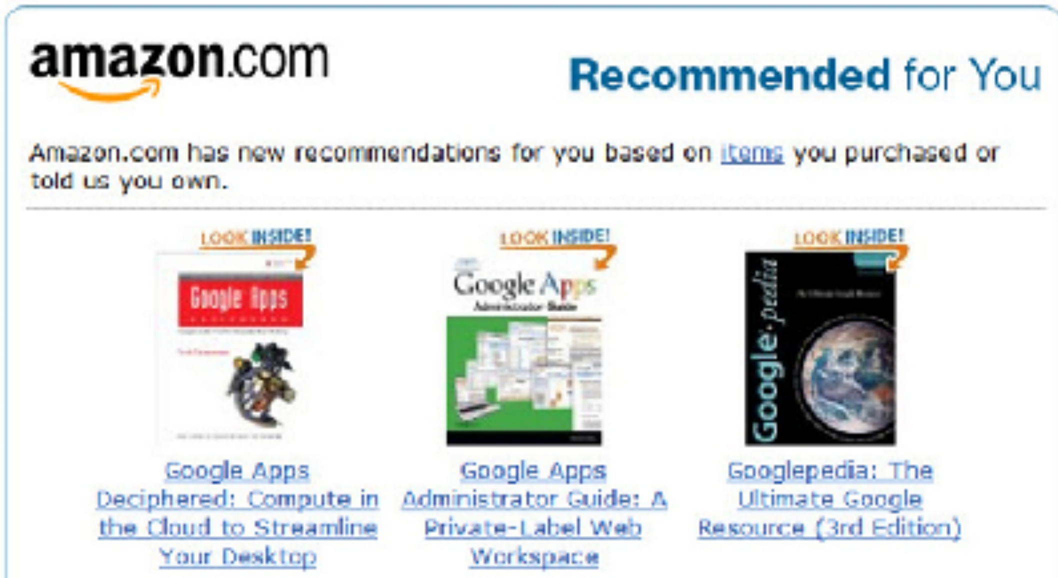
"Stochastic Submodular Maximization: The Case of Coverage Functions", Karimi et al.

Penn

Submodular Maximization: Discrete and Continuous

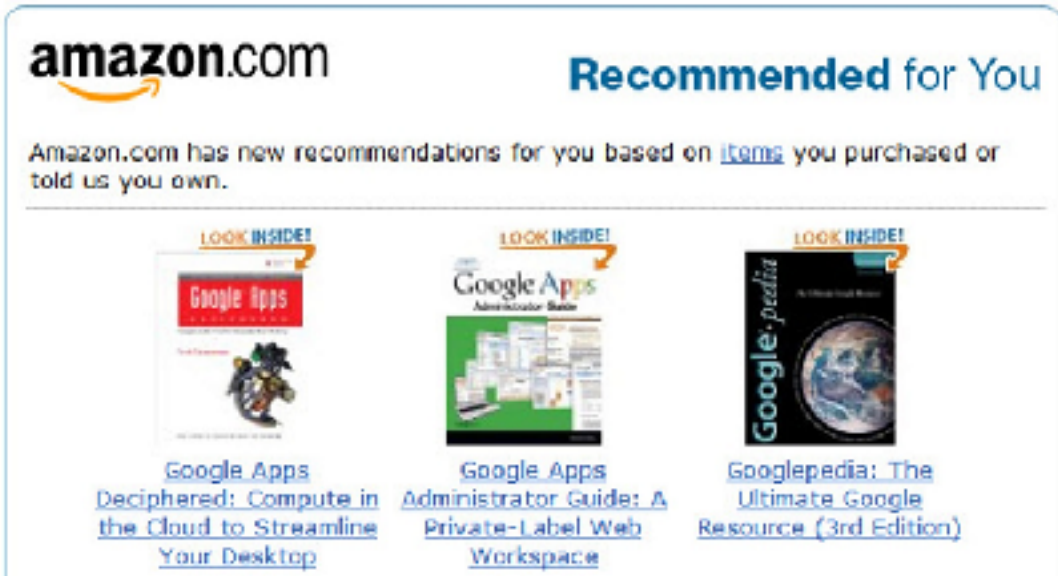


Online Submodular Maximization



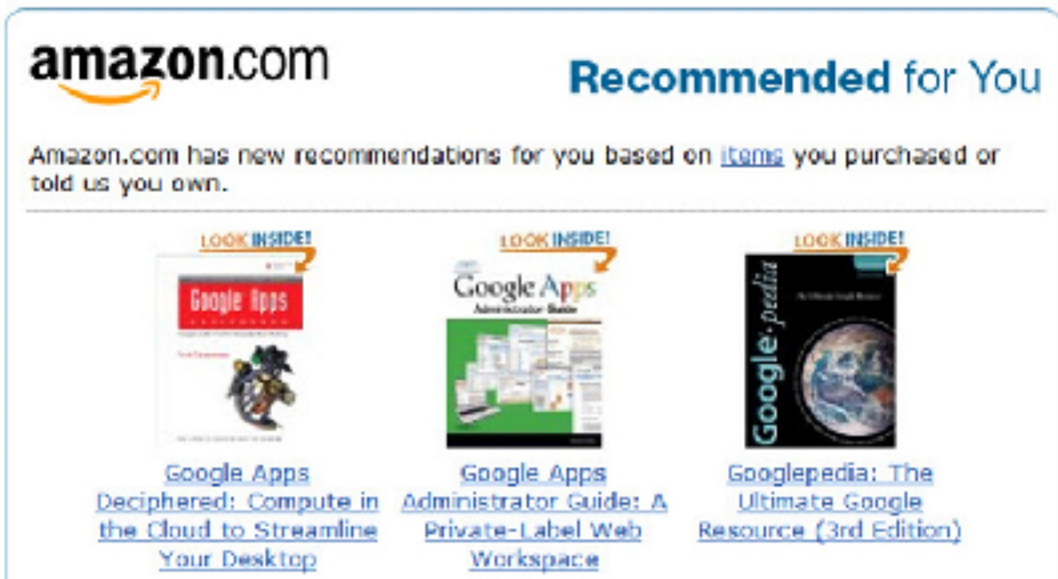
Online recommendation, advertising, …

Online Submodular Maximization



Online recommendation, advertising, …

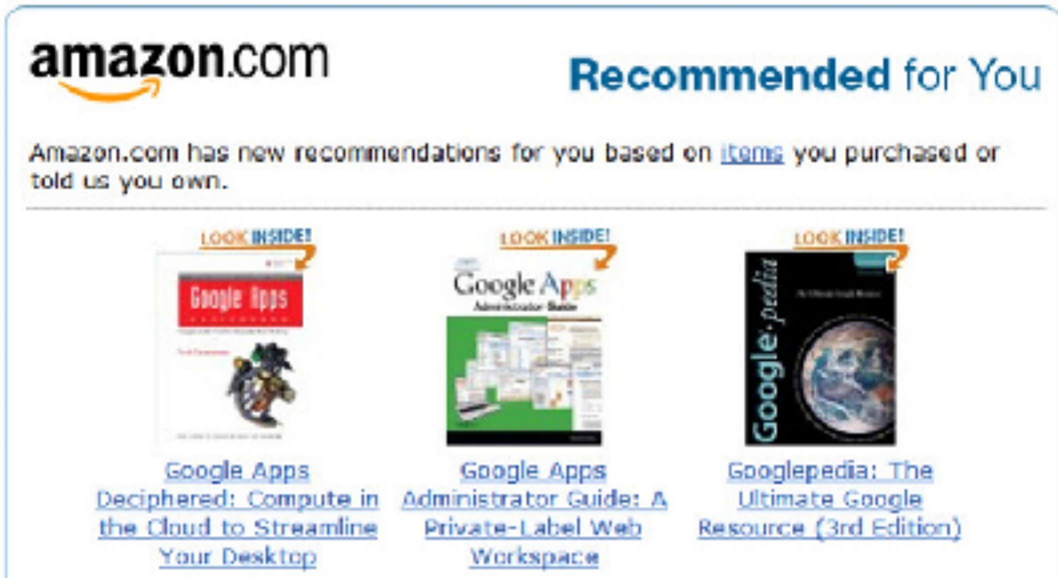
Online Submodular Maximization



Online recommendation, advertising, …

additional dimension:

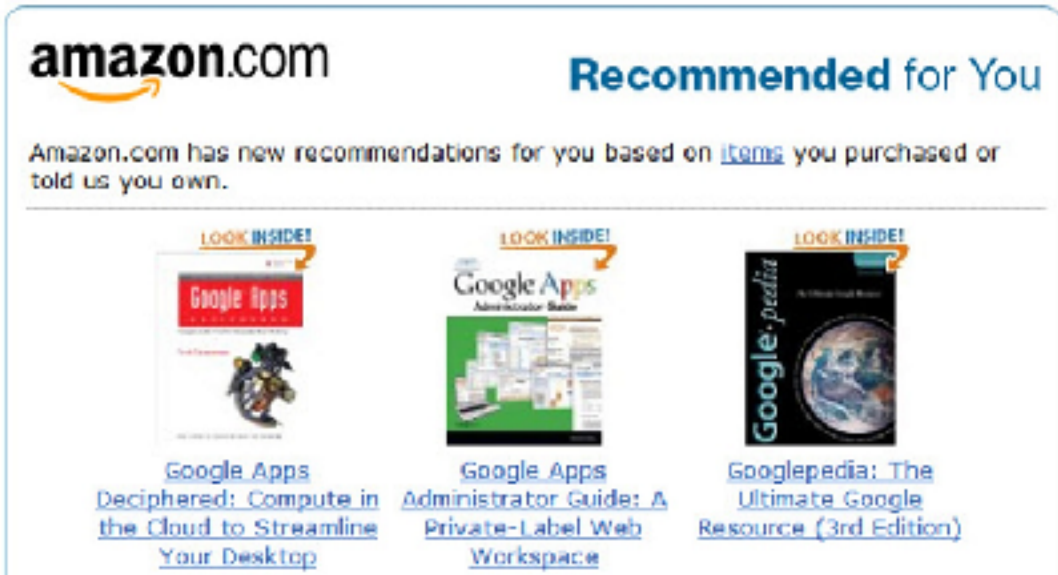
Online Submodular Maximization



Online recommendation, advertising, …

additional dimension: ⟶ time

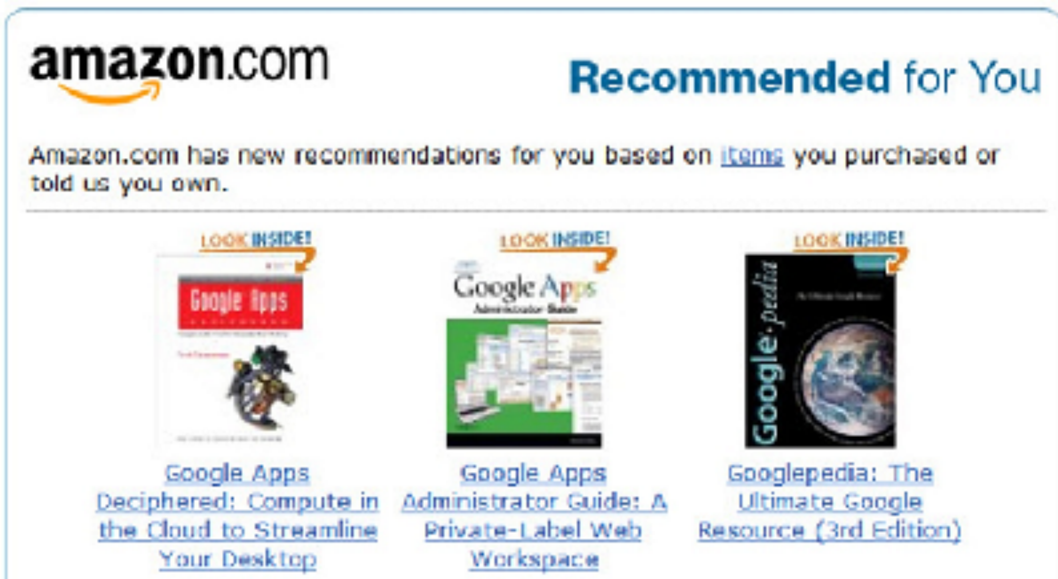
Online Submodular Maximization



Online recommendation, advertising, …

additional dimension: $\xrightarrow{f_1, f_2, f_3, \cdots, f_T}$ time

Online Submodular Maximization

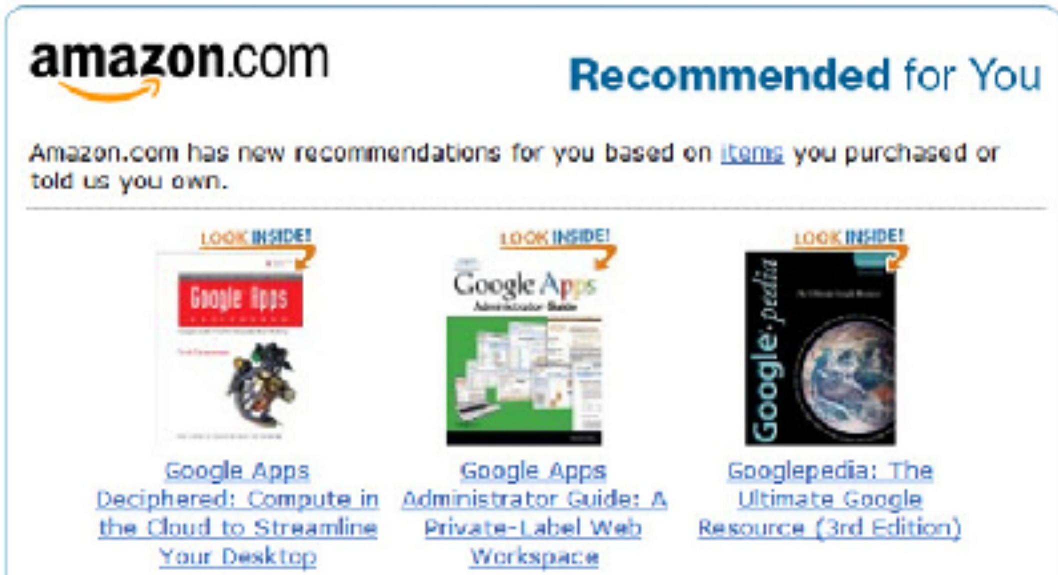


Online recommendation, advertising, …

additional dimension: $\xrightarrow{f_1, f_2, f_3, \cdots, f_T}$ time

- at the beginning of each round t, a set $S_t \in \mathcal{I}$ is selected

Online Submodular Maximization

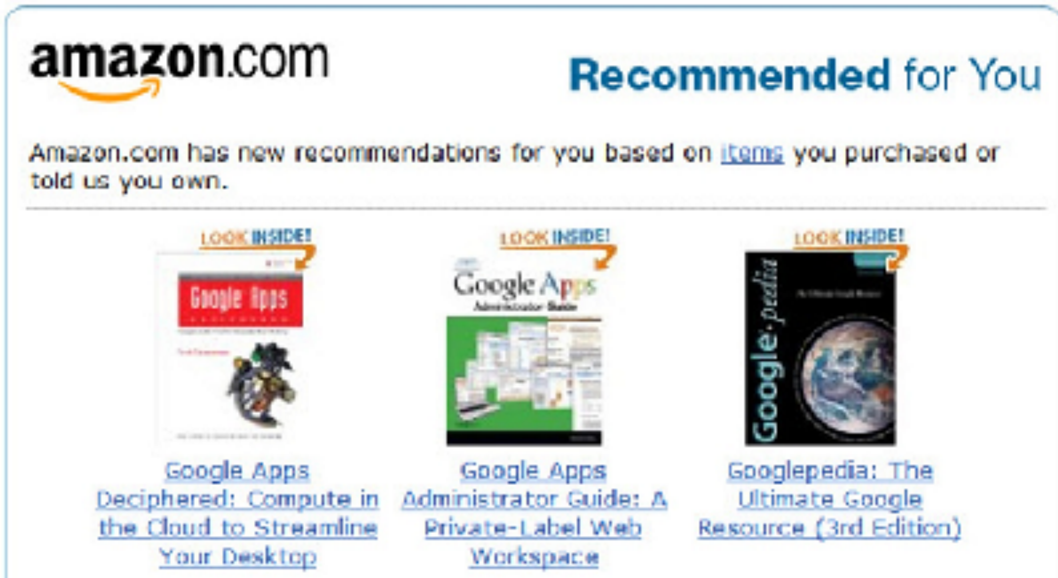


Online recommendation, advertising, …

additional dimension: $\xrightarrow{f_1, f_2, f_3, \cdots, f_T}$ time

- at the beginning of each round t, a set $S_t \in \mathcal{I}$ is selected
- at the end of the round, a reward function f_t is revealed

Online Submodular Maximization

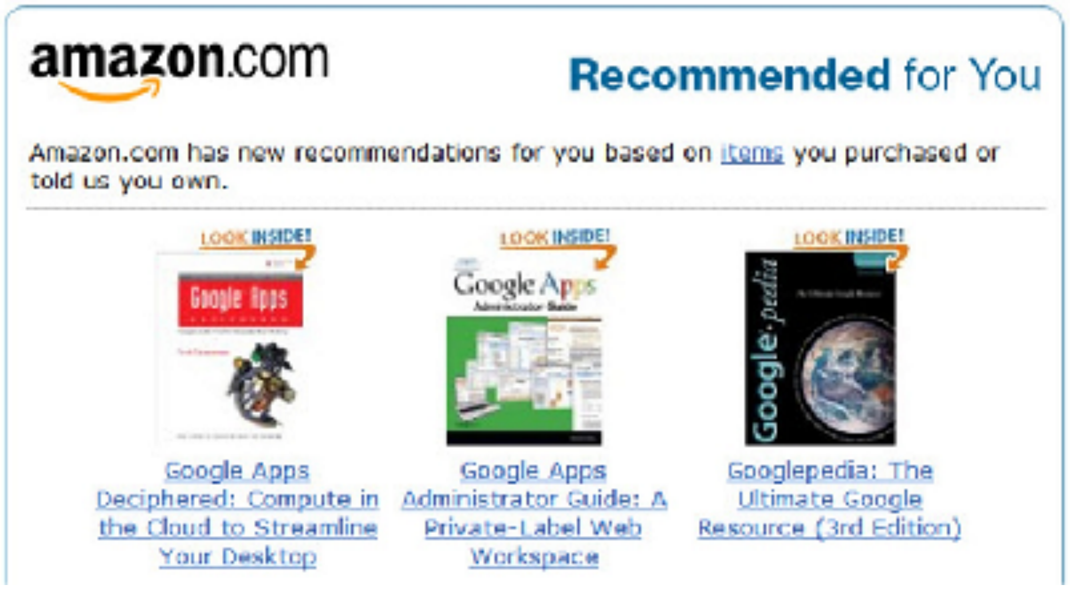


Online recommendation, advertising, …

additional dimension: $\xrightarrow{f_1, f_2, f_3, \cdots, f_T}$ time

- at the beginning of each round t, a set $S_t \in \mathcal{I}$ is selected
- at the end of the round, a reward function f_t is revealed
- $f_t : 2^V \to \mathbb{R}$; the reward of round t is $f_t(S_t)$

Online Submodular Maximization



Online recommendation, advertising, …

additional dimension: $f_1, f_2, f_3, \cdots, f_T$ → time

- at the beginning of each round t, a set $S_t \in \mathcal{I}$ is selected
- at the end of the round, a reward function f_t is revealed
- $f_t : 2^V \to \mathbb{R}$; the reward of round t is $f_t(S_t)$

time-varying oracle

action S_t ↑ reward ↓ $f_t(S_t)$ feedback ↓ $f_t(\cdot)$

Yale Penn

Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

$$\text{regret} := \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t)$$

online oracle

action: $S_t \in \mathcal{I}$

reward: $f_t(S_t)$

feedback: $f_t(\cdot)$

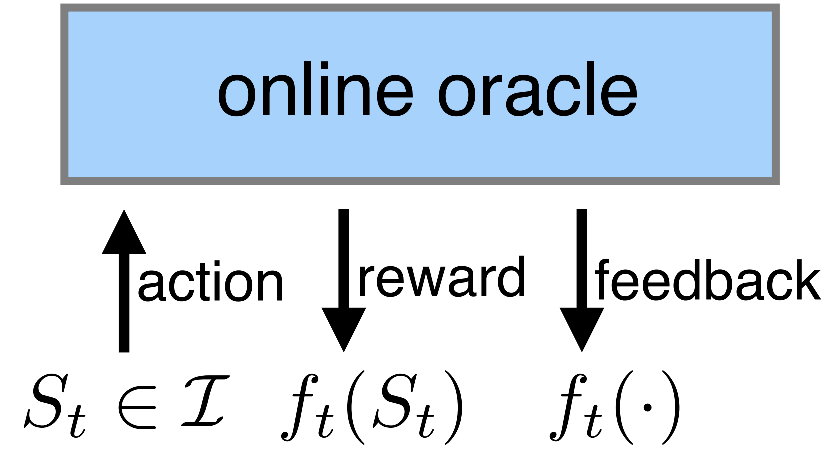
"An online algorithm for maximizing submodular functions", Streeter, Golovin, 2008

Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

$$\text{regret} := \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t)$$



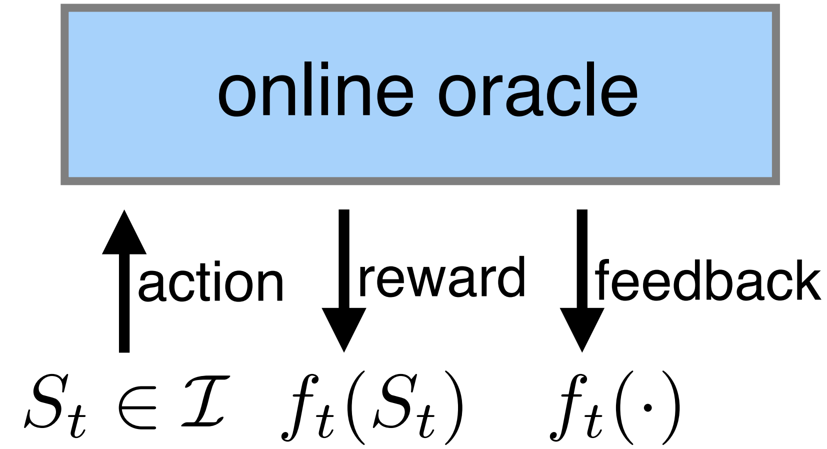
"An online algorithm for maximizing submodular functions", Streeter, Golovin, 2008

Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

$$\text{regret} := \underbrace{\max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S)}_{\text{best action in hindsight}} - \sum_{t=1}^{T} f_t(S_t)$$



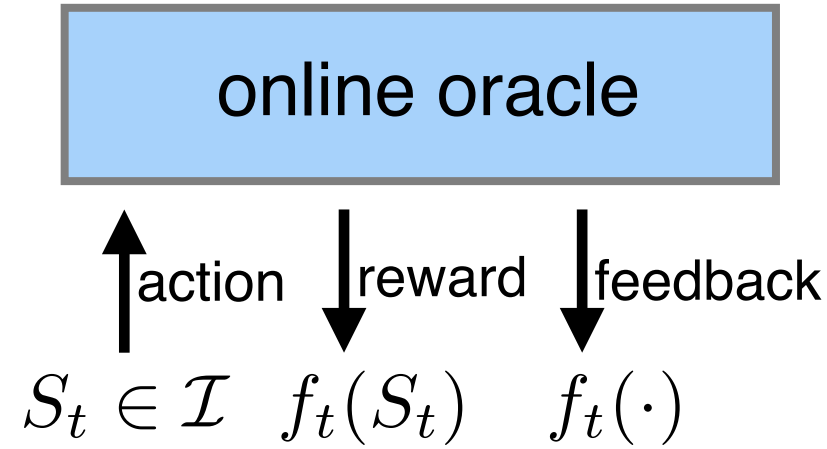
"An online algorithm for maximizing submodular functions", Streeter, Golovin, 2008

Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

$$\text{regret} := \underbrace{\max_{S\in\mathcal{I}}\sum_{t=1}^{T} f_t(S)}_{\text{best action in hindsight}} - \underbrace{\sum_{t=1}^{T} f_t(S_t)}_{\text{algorithm}}$$



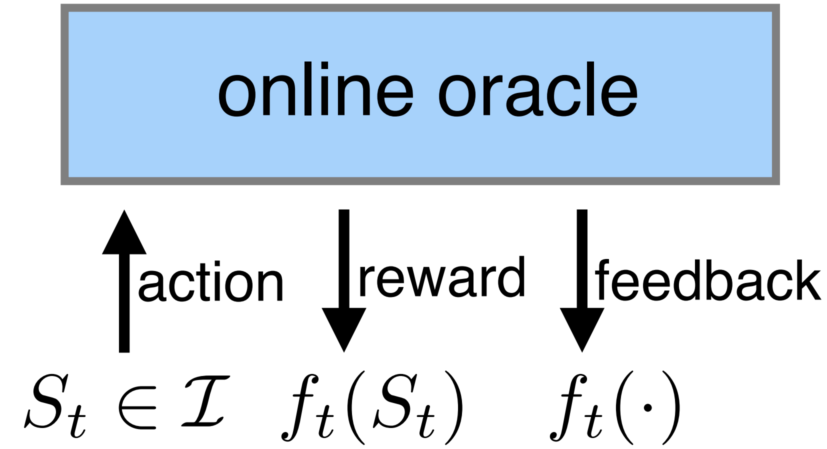
"An online algorithm for maximizing submodular functions", Streeter, Golovin, 2008

Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

$$\color{red}{\alpha} - \text{regret} := \color{red}{\alpha} \underbrace{\max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S)}_{\text{best action in hindsight}} - \underbrace{\sum_{t=1}^{T} f_t(S_t)}_{\text{algorithm}}$$



"An online algorithm for maximizing submodular functions", Streeter, Golovin, 2008

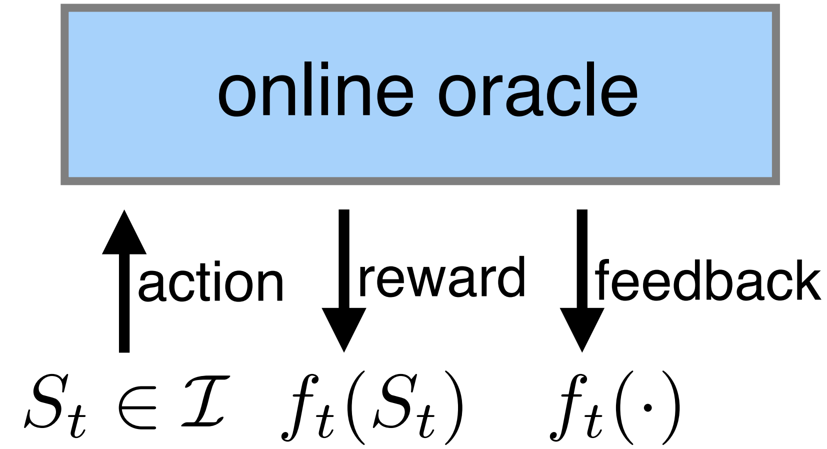
Yale Penn

Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

$$\alpha-\mathrm{regret} := \alpha \underbrace{\max_{S\in\mathcal{I}} \sum_{t=1}^{T} f_t(S)}_{\text{best action in hindsight}} - \underbrace{\sum_{t=1}^{T} f_t(S_t)}_{\text{algorithm}}$$



Continuous:

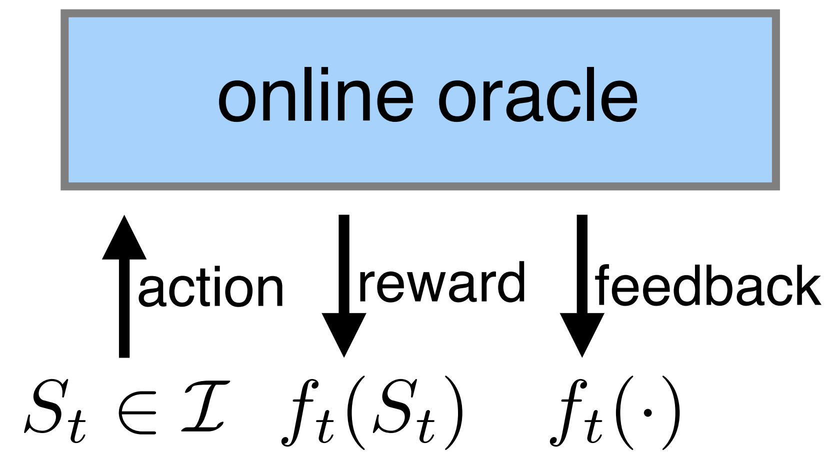
"Online Continuous Submodular Maximization", Chen, Hassani, Karbasi, 2018

Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

$$\alpha - \text{regret} := \alpha \underbrace{\max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S)}_{\text{best action in hindsight}} - \underbrace{\sum_{t=1}^{T} f_t(S_t)}_{\text{algorithm}}$$



Continuous:

- $F_1, F_2, \cdots, F_T \,;\, F_i : \mathcal{X} \to \mathbb{R}$

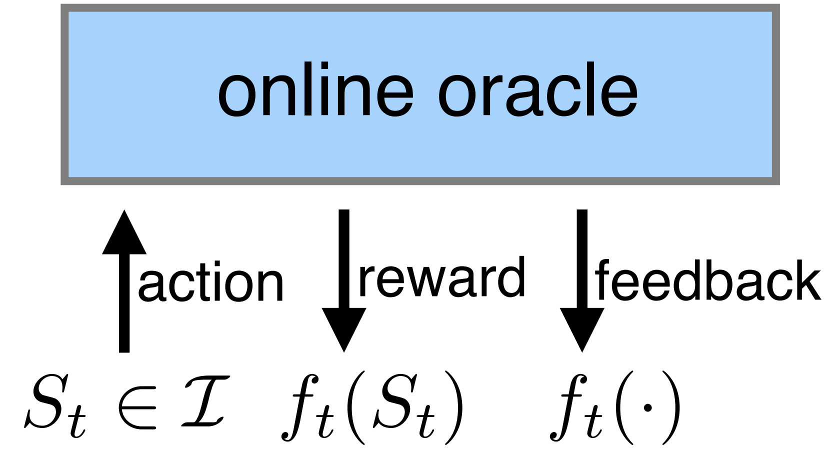
"Online Continuous Submodular Maximization", Chen, Hassani, Karbasi, 2018

Online Submodular Maximization: Discrete and Continuous

Discrete:

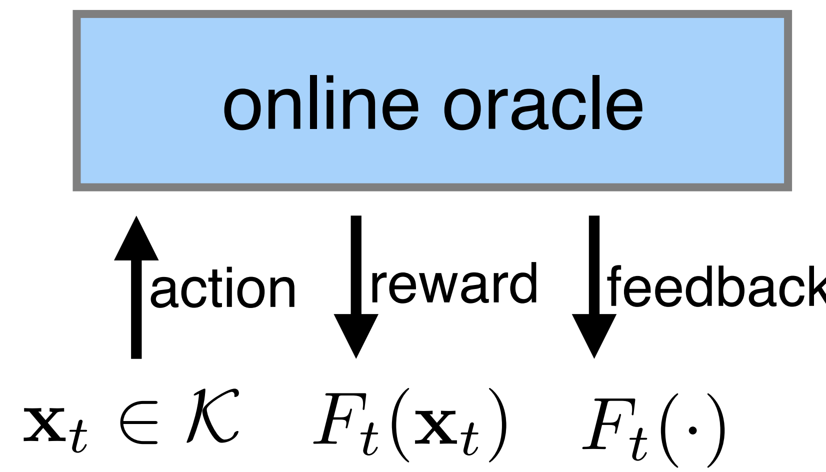
- performance metric:

$$\alpha - \text{regret} := \alpha \underbrace{\max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S)}_{\text{best action in hindsight}} - \underbrace{\sum_{t=1}^{T} f_t(S_t)}_{\text{algorithm}}$$



Continuous:

- $F_1, F_2, \cdots, F_T \,;\, F_i : \mathcal{X} \to \mathbb{R}$



Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

$$\color{red}{\alpha} - \text{regret} := \color{red}{\alpha} \max_{S \in \mathcal{I}} \underbrace{\sum_{t=1}^{T} f_t(S)}_{\text{best action in hindsight}} - \underbrace{\sum_{t=1}^{T} f_t(S_t)}_{\text{algorithm}}$$

online oracle

action $\uparrow$ $S_t \in \mathcal{I}$ reward $\downarrow$ $f_t(S_t)$ feedback $\downarrow$ $f_t(\cdot)$

Continuous:

- $F_1, F_2, \cdots, F_T$; $F_i : \mathcal{X} \to \mathbb{R}$

- performance metric:

$$\alpha - \text{regret} := \alpha \max_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^{T} F_t(\mathbf{x}) - \sum_{t=1}^{T} F_t(\mathbf{x}_t)$$

online oracle

action $\uparrow$ $\mathbf{x}_t \in \mathcal{K}$ reward $\downarrow$ $F_t(\mathbf{x}_t)$ feedback $\downarrow$ $F_t(\cdot)$

“Online Continuous Submodular Maximization”, Chen, Hassani, Karbasi, 2018

Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

$$\alpha - \text{regret} := \alpha \underbrace{\max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S)}_{\text{best action in hindsight}} - \underbrace{\sum_{t=1}^{T} f_t(S_t)}_{\text{algorithm}}$$

online oracle

action: $S_t \in \mathcal{I}$ reward: $f_t(S_t)$ feedback: $f_t(\cdot)$

Continuous:

- $F_1, F_2, \cdots, F_T$; $F_i : \mathcal{X} \to \mathbb{R}$

online oracle

action: $\mathbf{x}_t \in \mathcal{K}$ reward: $F_t(\mathbf{x}_t)$ feedback: $F_t(\cdot)$

Devise algorithms for the online setting with α-regret being sub-linear in T

best value of α? lowest regret?

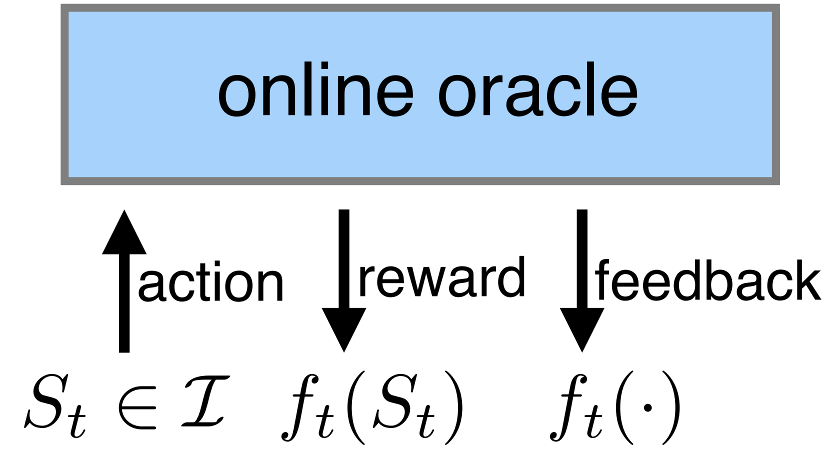
Yale Penn

Online Submodular Maximization: Discrete and Continuous

Discrete:

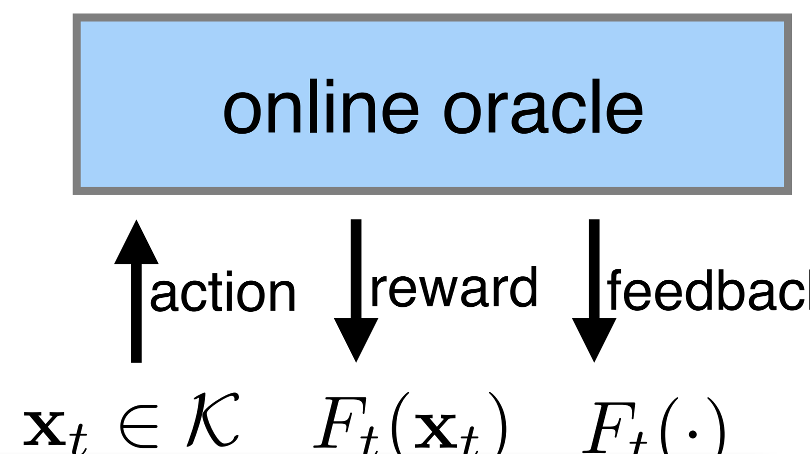
- performance metric:

$$\alpha - \text{regret} := \alpha \underbrace{\max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S)}_{\text{best action in hindsight}} - \underbrace{\sum_{t=1}^{T} f_t(S_t)}_{\text{algorithm}}$$



Continuous:

- $F_1, F_2, \cdots, F_T$; $F_i : \mathcal{X} \to \mathbb{R}$



Devise algorithms for the online setting with α-regret being sub-linear in T

best value of α?　　lowest regret?

Online Gradient Ascent

- At each time t :

  play $\mathbf{x}_t$ and receive reward $F_t(\mathbf{x}_t)$

$$\mathbf{x}_{t+1} = \mathrm{Proj}_{\mathcal{K}}\{\mathbf{x}_t + \eta_t \nabla F_t(\mathbf{x}_t)\}$$

Online Gradient Ascent

- At each time t :

  play $\mathbf{x}_t$ and receive reward $F_t(\mathbf{x}_t)$

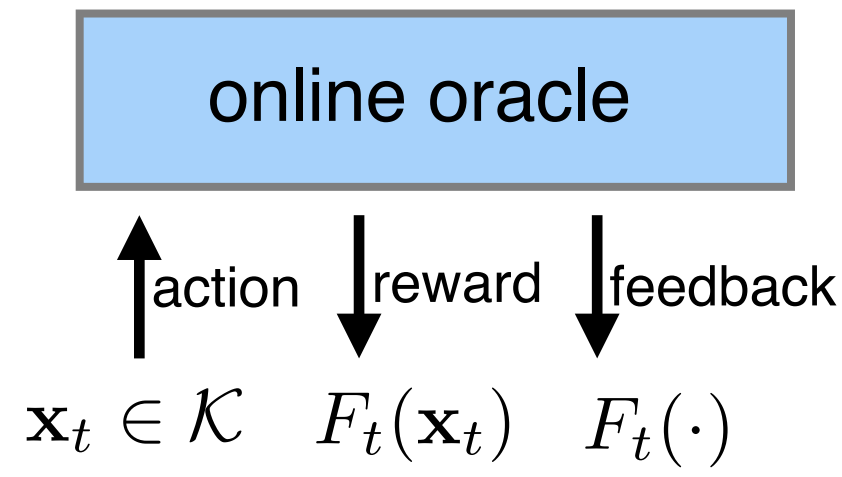
  $$\mathbf{x}_{t+1} = \mathrm{Proj}_{\mathcal{K}}\{\mathbf{x}_t + \eta_t \nabla F_t(\mathbf{x}_t)\}$$

Online Gradient Ascent

- At each time t :

  play $\mathbf{x}_t$ and receive reward $F_t(\mathbf{x}_t)$

  $$\mathbf{x}_{t+1} = \mathrm{Proj}_{\mathcal{K}}\{\mathbf{x}_t + \eta_t \nabla F_t(\mathbf{x}_t)\}$$

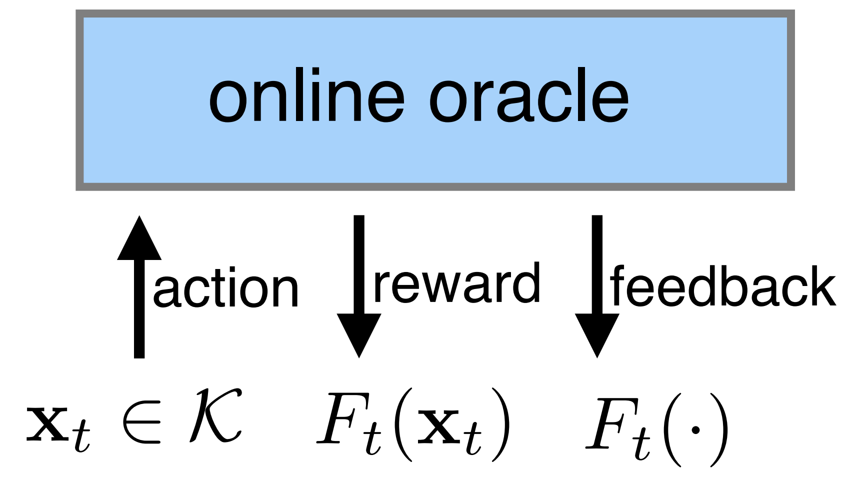


Online Gradient Ascent

- At each time t :

  play $\mathbf{x}_t$ and receive reward $F_t(\mathbf{x}_t)$

$$\mathbf{x}_{t+1} = \mathrm{Proj}_{\mathcal{K}}\{\mathbf{x}_t + \eta_t \nabla F_t(\mathbf{x}_t)\}$$



[Chen, Hassani, Karbasi]

If F_t's are monotone and DR-submodular, then for the online gradient ascent algorithm we have

$$\frac{1}{2} - \mathrm{regret} = O(\sqrt{T})$$

"Online Continuous Submodular Maximization", AISTATS '18

Online Gradient Ascent

- At each time t :

  play $\mathbf{x}_t$ and receive reward $F_t(\mathbf{x}_t)$

$$\mathbf{x}_{t+1} = \mathrm{Proj}_{\mathcal{K}}\{\mathbf{x}_t + \eta_t \nabla F_t(\mathbf{x}_t)\}$$

online oracle

action ↑ $\mathbf{x}_t \in \mathcal{K}$

reward ↓ $F_t(\mathbf{x}_t)$

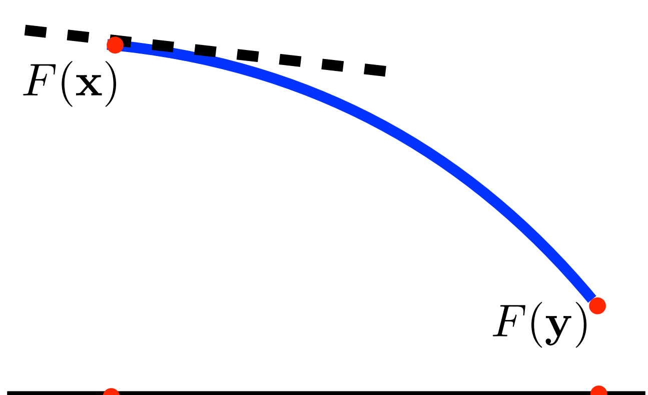
feedback ↓ $F_t(\cdot)$

[Chen, Hassani, Karbasi]

If F_t's are monotone and DR-submodular, then for the online gradient ascent algorithm we have

$$\frac{1}{2} - \text{regret} = O(\sqrt{T})$$

"Online Continuous Submodular Maximization", AISTATS '18



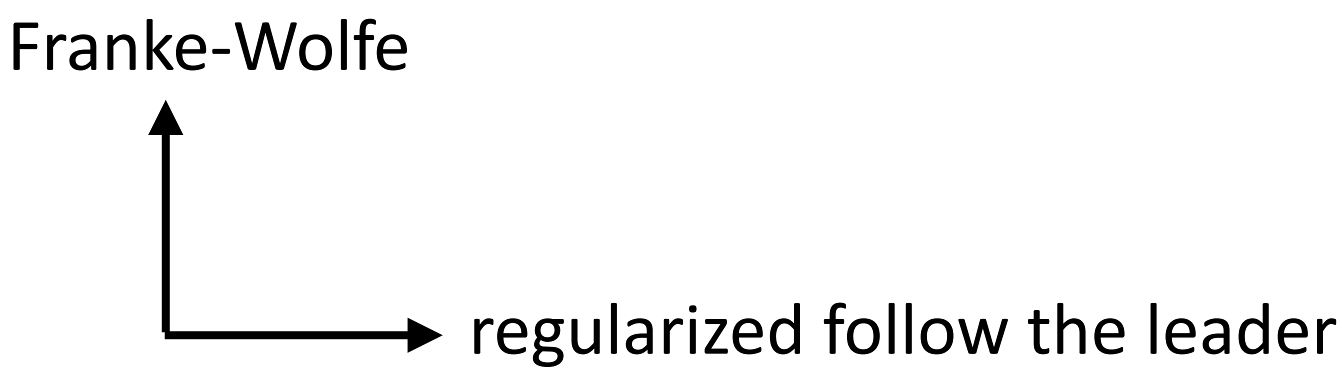
concave: $F(\mathbf{y}) - F(\mathbf{x}) \leq \langle \nabla F(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$

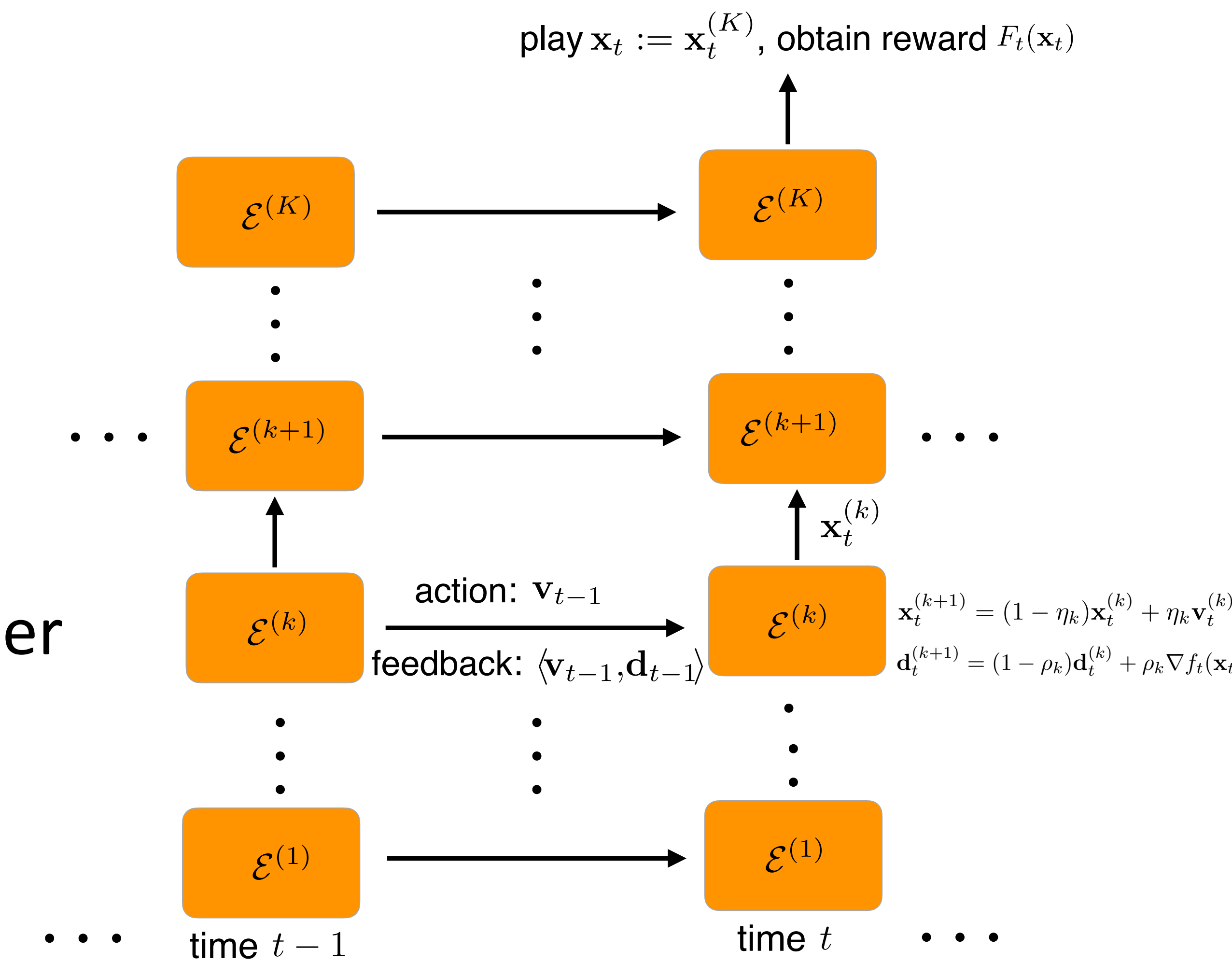
monotone DR-submodular: $F(\mathbf{y}) - 2F(\mathbf{x}) \leq \langle \nabla F(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$

Yale Penn

Meta Franke-Wolfe

- The Meta-FW Framework:





"An online algorithm for maximizing submodular functions", Streeter, Golovin, 2008

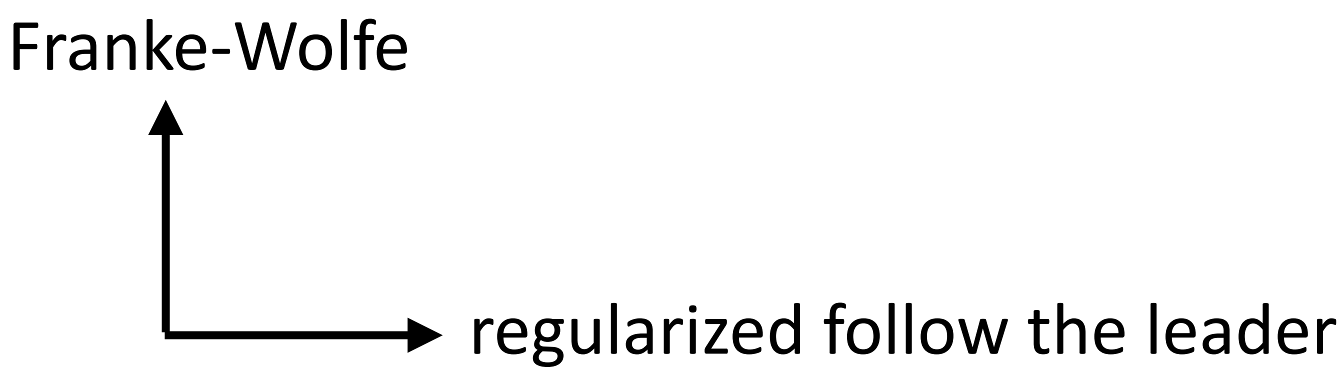
"Online Submodular Maximization under a Matroid Constraint with Application to Learning Assignments", Golovin, Krause, Streeter, 2014

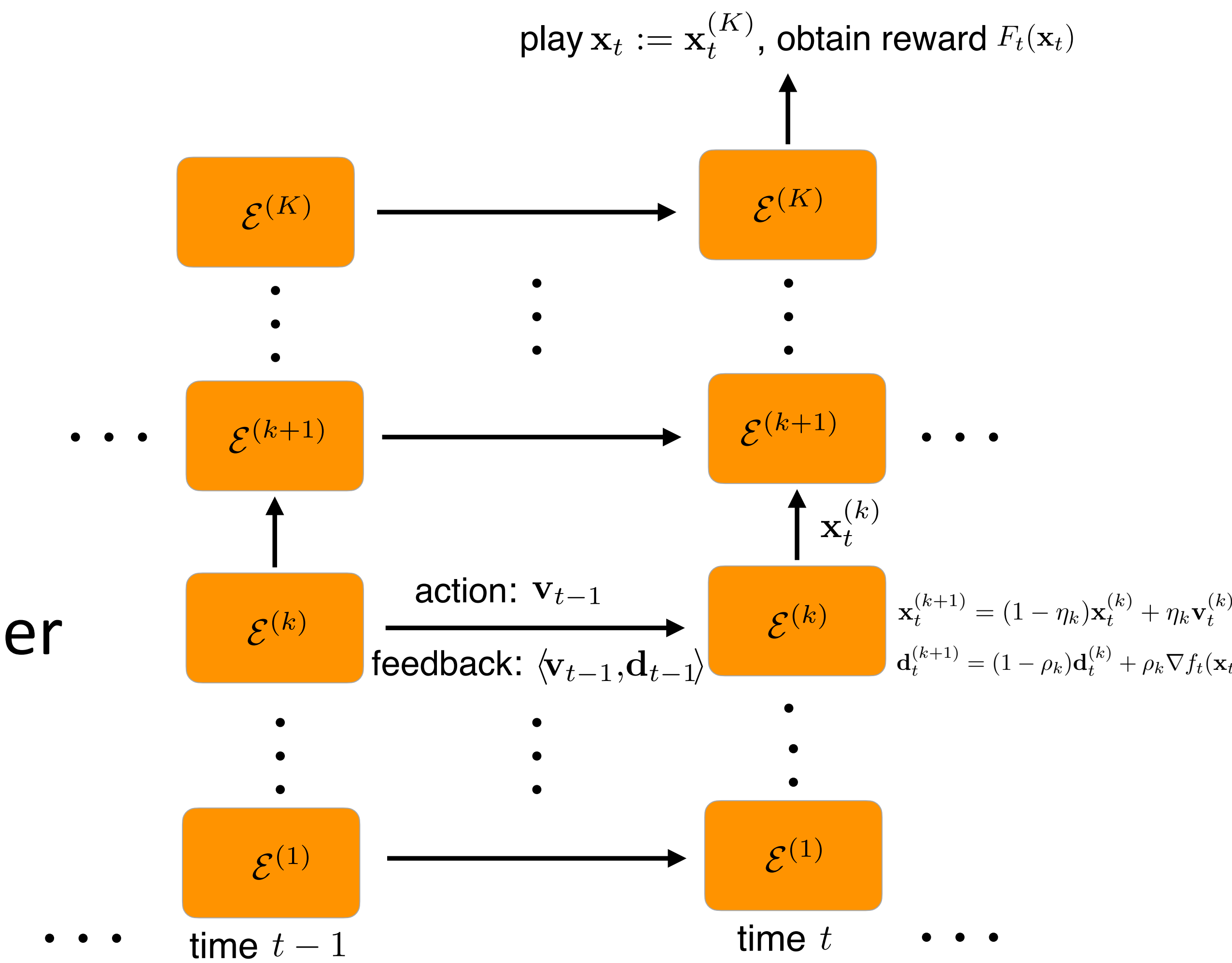
"Online Continuous Submodular Maximization", Chen, Hassani, Karbasi, 2018

"Projection-Free Online Optimization with Stochastic Gradient: From Convexity to Submodularity", Chen, Harshaw, Hassani, Karbasi, 2018

Meta Franke-Wolfe

- The Meta-FW Framework:





"An online algorithm for maximizing submodular functions", Streeter, Golovin, 2008

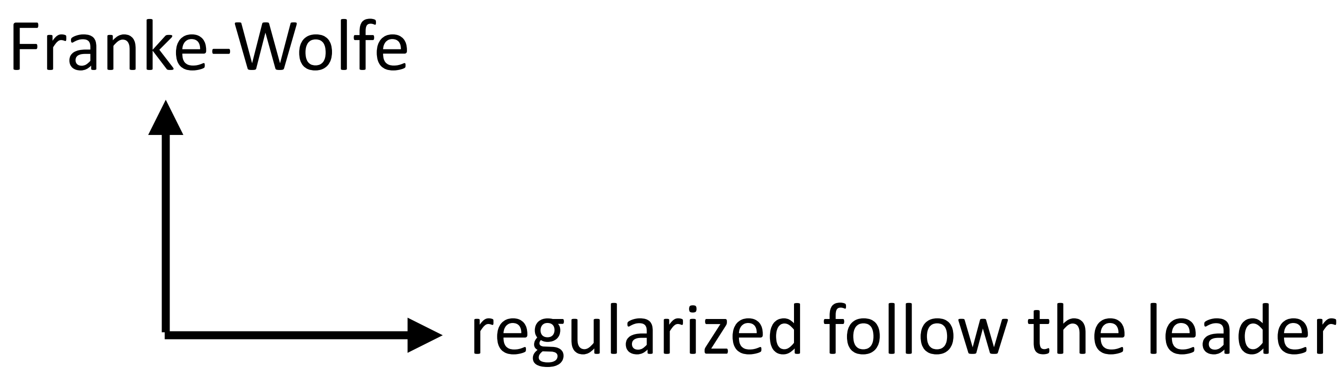
"Online Submodular Maximization under a Matroid Constraint with Application to Learning Assignments", Golovin, Krause, Streeter, 2014

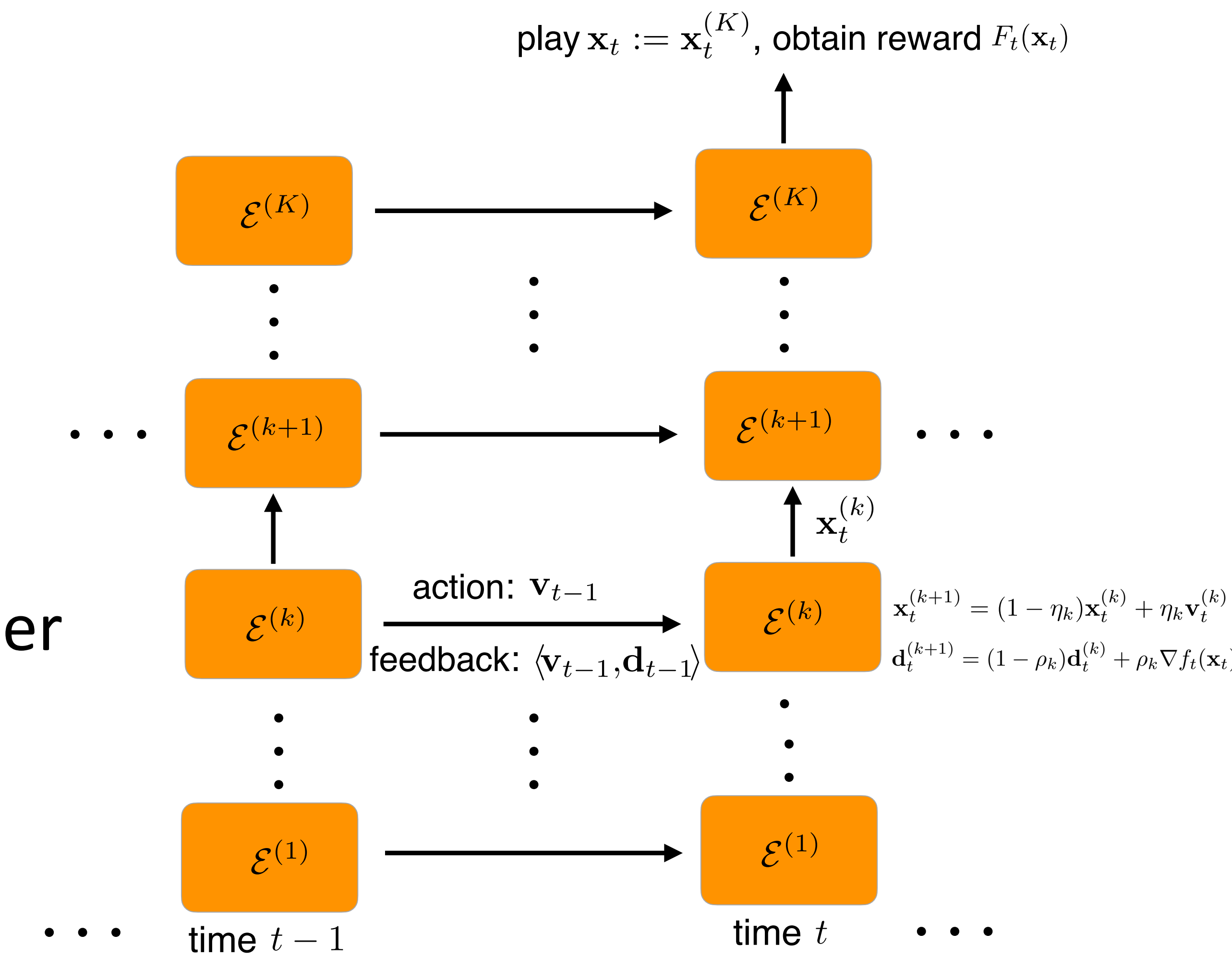
"Online Continuous Submodular Maximization", Chen, Hassani, Karbasi, 2018

"Projection-Free Online Optimization with Stochastic Gradient: From Convexity to Submodularity", Chen, Harshaw, Hassani, Karbasi, 2018

Meta Franke-Wolfe

- The Meta-FW Framework:





[Chen, Hassani, Karbasi]

If F_t's are monotone and DR-submodular, then for the Meta-Frank-Wolfe algorithm we have

$$(1-\frac{1}{e}) - \text{regret} = O(\sqrt{T})$$

"Online Continuous Submodular Maximization", AISTATS '18

Yale Penn

The Online Setting

- Many open questions:
  continuous submodular, non-monotone, other constraints
  simpler algorithms, better regret bounds, etc

- Related work/ Recent Progress:

The Online Setting

- Many open questions:
  continuous submodular, non-monotone, other constraints
  simpler algorithms, better regret bounds, etc

- Related work/ Recent Progress:

  "Designing smoothing functions for improved worst-case competitive ratio in online optimization", Eghbali, Fazel, 2016

  "Online Non-Monotone DR-submodular Maximization", Thang, Srivastav, 2019

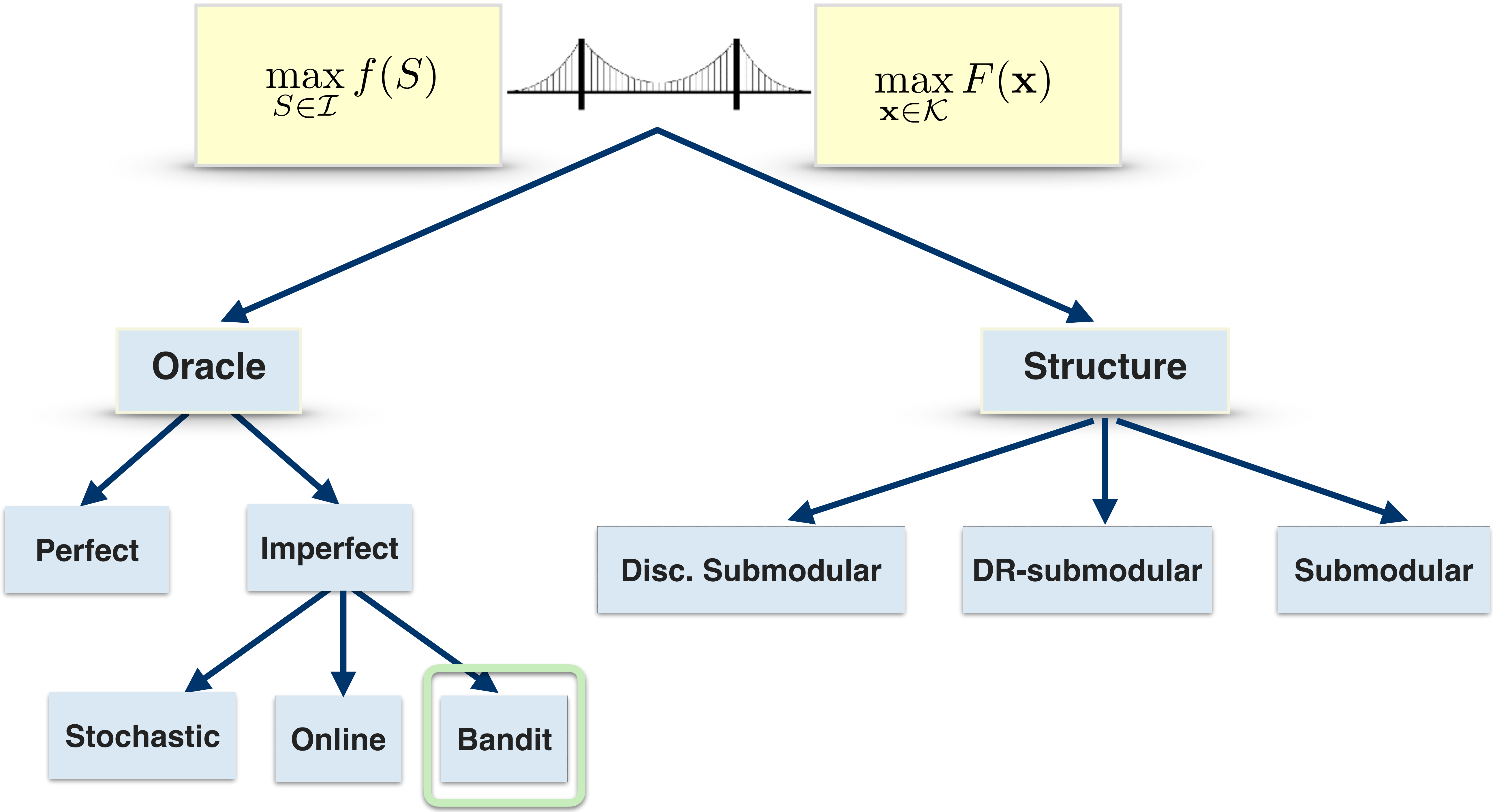
  "No-regret Algoriths for Online k-submodular optimization", Soma, 2019

  "Consistent Online Optimization: Convex and Submodular", Karimi, Krause, Lattanzi, Vassilvtiskii, 2019

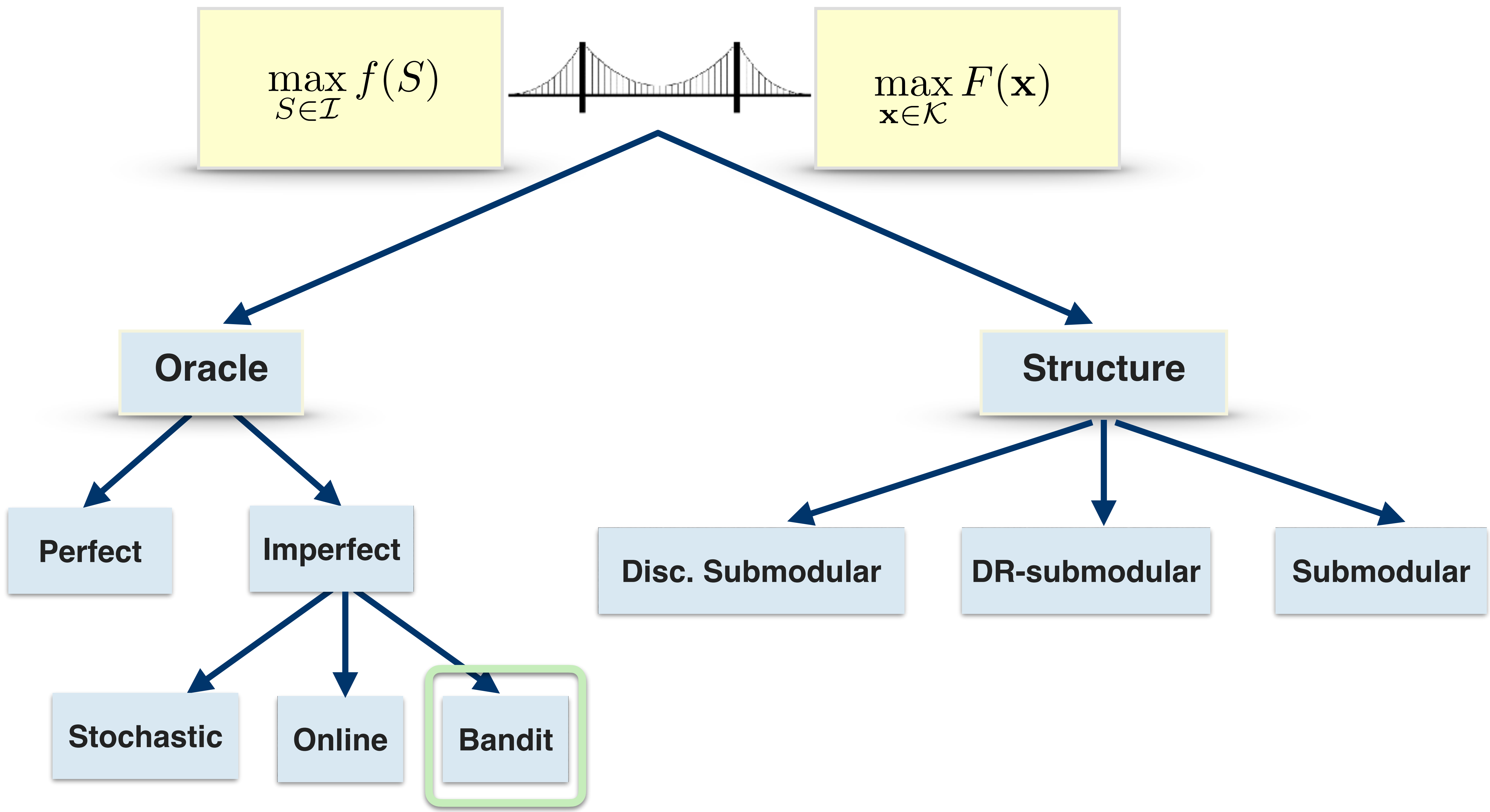
  "Online DR-Submodular Maximization with Stochastic Cumulative Constraints", Raut, Sadeghi, Fazel, 2020

  "Online continuous DR-submodular maximization with long-term budget constraints", Sadeghi, Fazel, 2020

Submodular Maximization: Discrete and Continuous



Submodular Maximization: Discrete and Continuous



Bandit Submodular Maximization

Bandit oracle

action $\uparrow$ reward & feedback $\downarrow$

$S_t \in \mathcal{I}$ $\quad$ $f_t(S_t)$

Bandit oracle

action $\uparrow$ reward & feedback $\downarrow$

$\mathbf{x}_t \in \mathcal{K}$ $\quad$ $F_t(\mathbf{x}_t)$

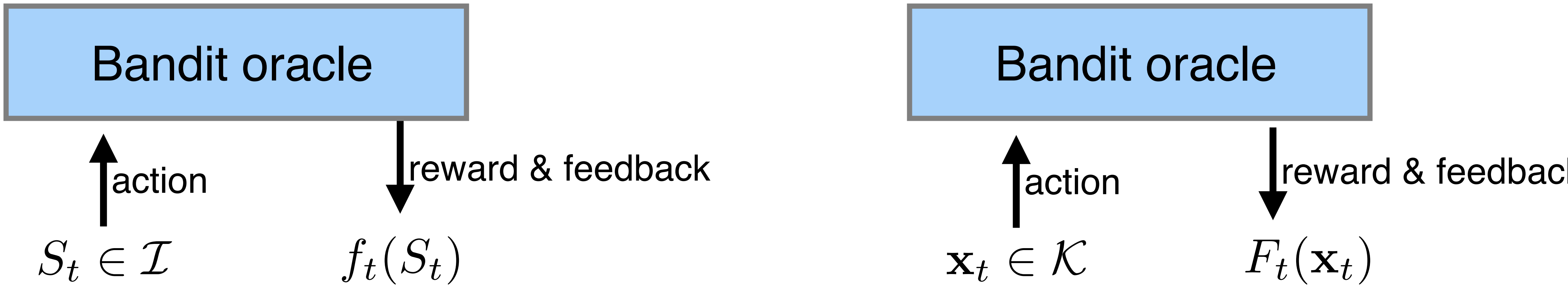
"Online Continuous Submodular Maximization: From Full-Information to Bandit Feedback, Zhang, chen, Hassani, Karbasi, 2019

"Online Submodular Maximization under a Matroid Constraint with Application to Learning Assignments", Golovin, Krause, Streeter, 2014

"Linear submodular bandits and their application to diversified retrieval", Yue, Guestrin, 2012

"An online algorithm for maximizing submodular functions", Streeter, Golovin, 2008

Bandit Submodular Maximization



[Zhang, Chen, Hassani, Karbasi]

If F_t's are monotone and DR-submodular, then for the Bandit Frank-Wolfe algorithm we have

$$(1-\frac{1}{e}) - \text{regret} = O(T^{\frac{8}{9}})$$

"Online Continuous Submodular Maximization: From Full-Information to Bandit Feedback", NEURIPS '19

"Online Continuous Submodular Maximization: From Full-Information to Bandit Feedback, Zhang, chen, Hassani, Karbasi, 2019

"Online Submodular Maximization under a Matroid Constraint with Application to Learning Assignments", Golovin, Krause, Streeter, 2014

"Linear submodular bandits and their application to diversified retrieval", Yue, Guestrin, 2012

"An online algorithm for maximizing submodular functions", Streeter, Golovin, 2008

Yale Penn

Bandit Submodular Maximization

Bandit oracle		Bandit oracle	
↑ action	↓ reward & feedback	↑ action	↓ reward & feedback
$S_t \in \mathcal{I}$	$f_t(S_t)$	$\mathbf{x}_t \in \mathcal{K}$	$F_t(\mathbf{x}_t)$

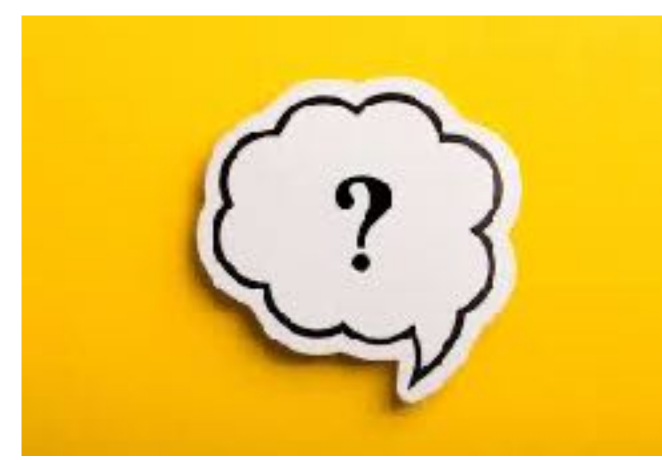
[Zhang, Chen, Hassani, Karbasi]

If F_t's are monotone and DR-submodular, then for the Bandit Frank-Wolfe algorithm we have

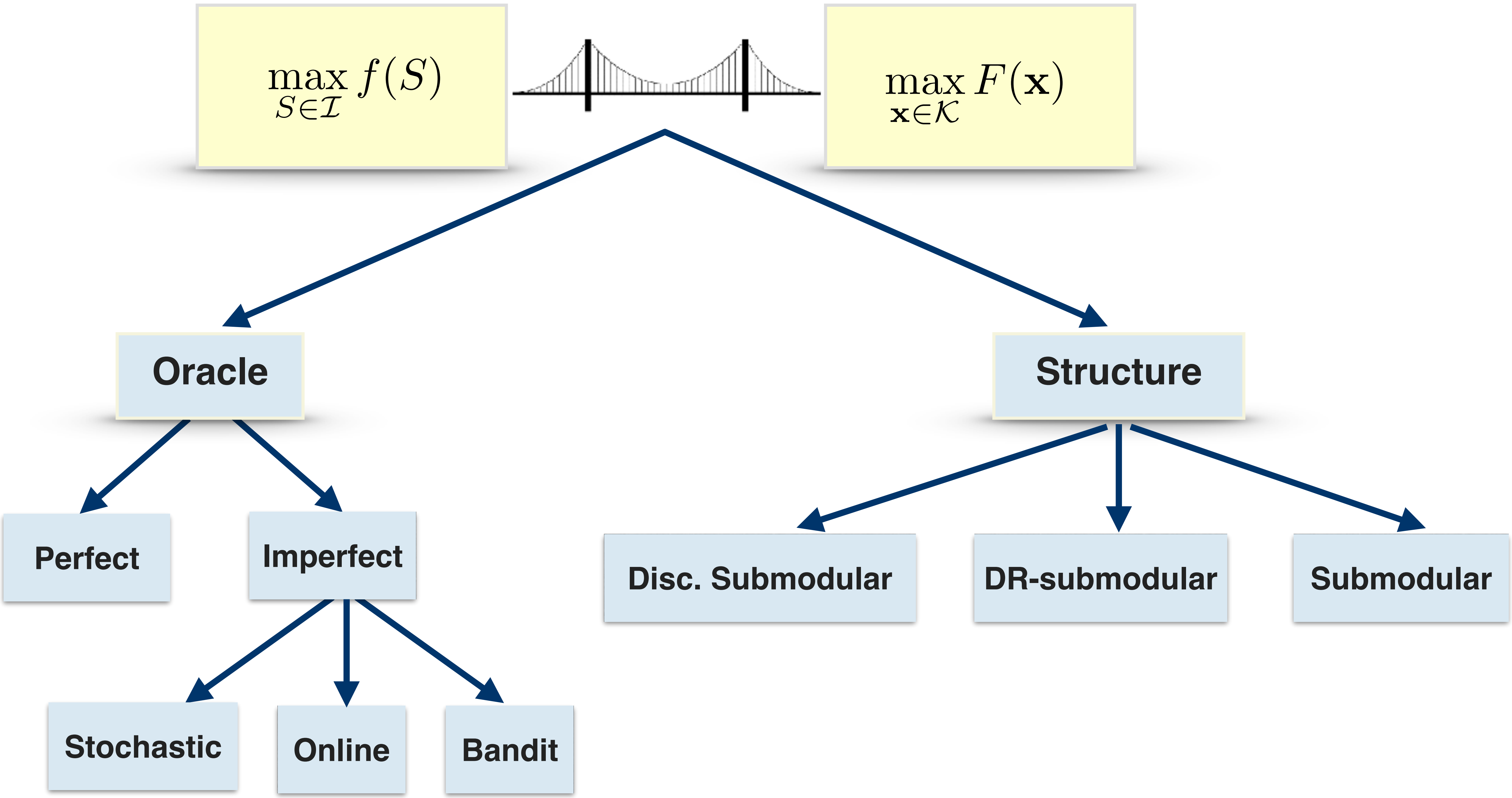
$$(1 - \frac{1}{e}) - \text{regret} = O(T^{\frac{8}{9}})$$

"Online Continuous Submodular Maximization: From Full-Information to Bandit Feedback", NEURIPS '19

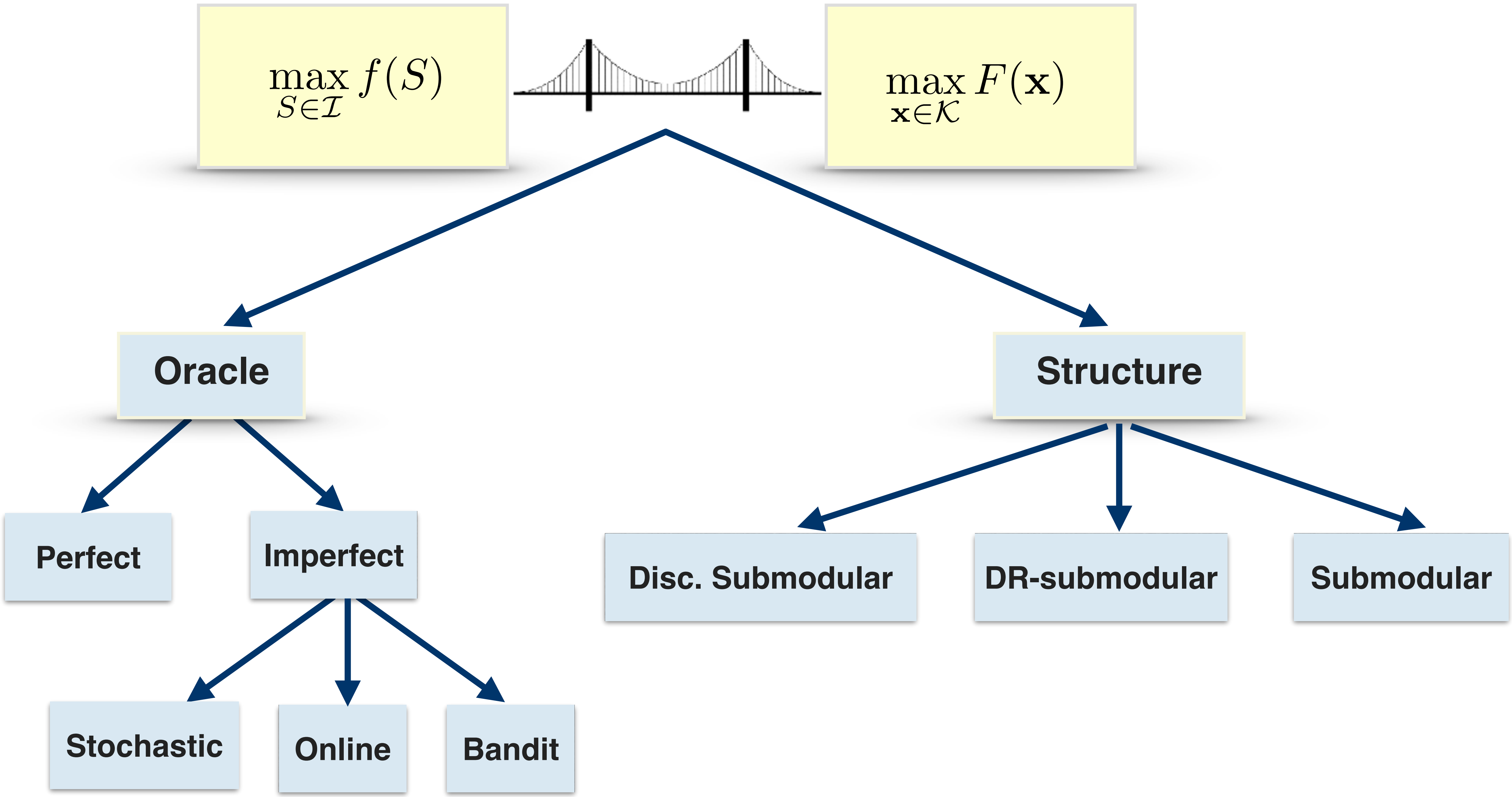
- Many open questions:
  continuous submodular, non-monotone, other constraints
  simpler algorithms, better regret bounds, etc



Final Remarks



Final Remarks



Final Remarks

- Minimization of continuous submodular functions is another story, see:

"Submodular Functions: from Discrete to Continuous Domains", Bach, 2016

Final Remarks

- Minimization of continuous submodular functions is another story, see:

"Submodular Functions: from Discrete to Continuous Domains", Bach, 2016

Final Remarks

- Minimization of continuous submodular functions is another story, see:

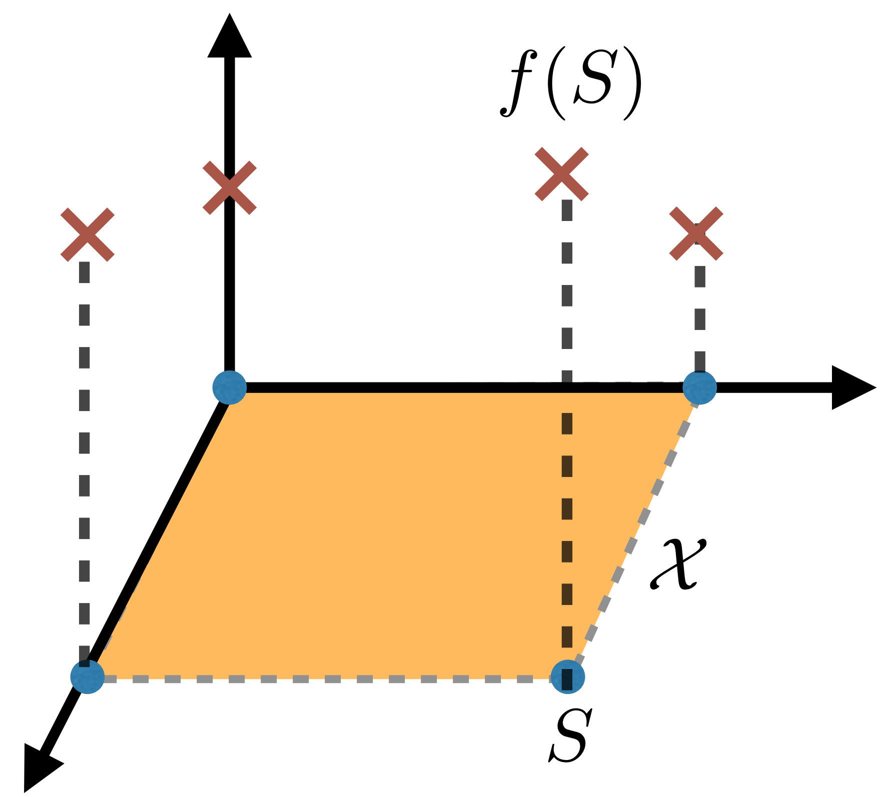
  "Submodular Functions: from Discrete to Continuous Domains", Bach, 2016

- Other continuous extensions of submodular set functions:

  F^+ : concave extension

  F^-: Lovasz (convex) extension

  F : Multi-linear extension



"Submodularity in Combinatorial Optimization", Vondrak, 2006

Final Remarks

- Minimization of continuous submodular functions is another story, see:

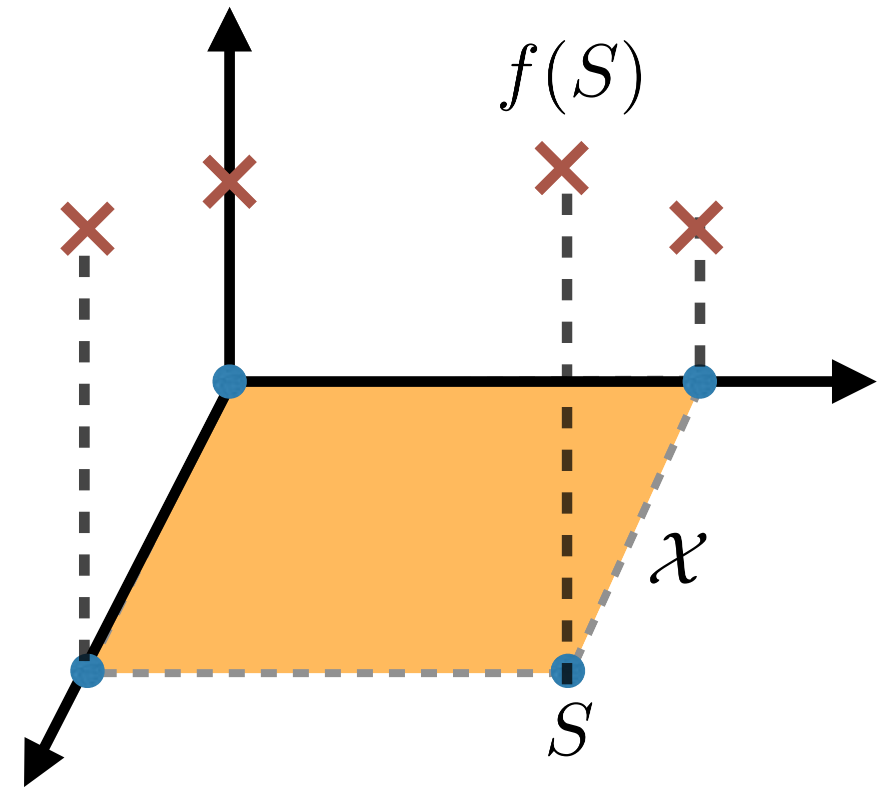
"Submodular Functions: from Discrete to Continuous Domains", Bach, 2016

- Other continuous extensions of submodular set functions:

F^+ : concave extension

F^-: Lovasz (convex) extension

F : Multi-linear extension



$$F^-(\mathbf{x}) \leq F(\mathbf{x}) \leq F^+(\mathbf{x}) \leq (1-\frac{1}{e})^{-1}F(\mathbf{x})$$

"Submodularity in Combinatorial Optimization", Vondrak, 2006

Final Remarks

- Minimization of continuous submodular functions is another story, see:

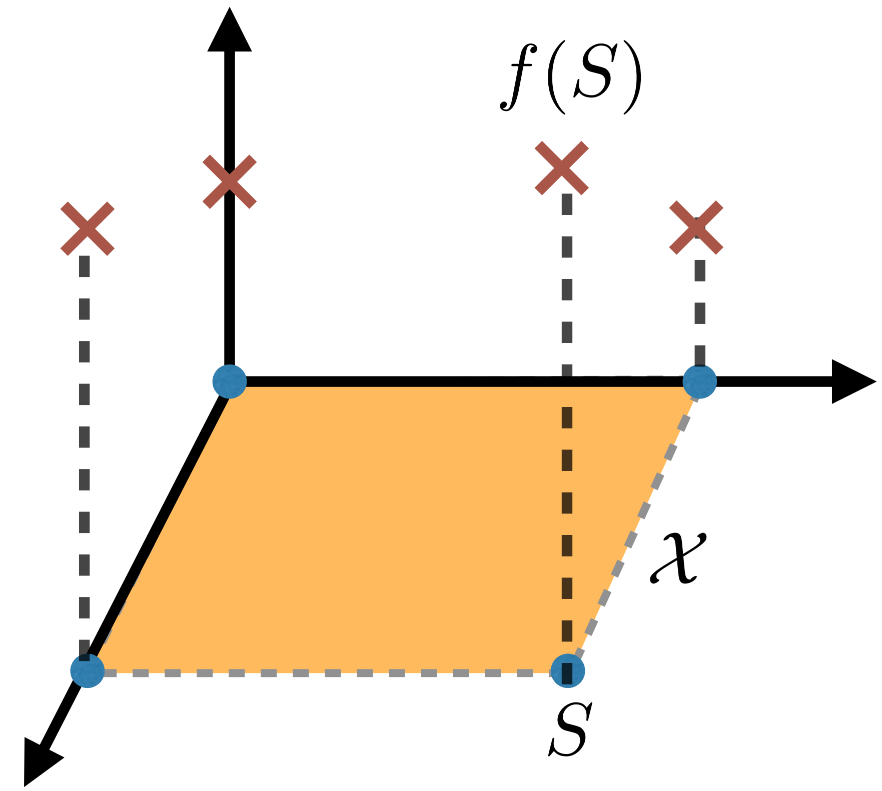
  "Submodular Functions: from Discrete to Continuous Domains", Bach, 2016

- Other continuous extensions of submodular set functions:

  F^{+} : concave extension

  F^{-}: Lovasz (convex) extension

  F : Multi-linear extension



- For coverage and deep submodular functions, tight concave extensions are possible which are also efficiently computable

"Stochastic Submodular Maximization: The Case of coverage Functions", Karimi, Lucic, Hassani, Krause, 2019

"Deep Submodular Functions", Bilmes, Bai, 2017

Other Resources on Submodularity

- To watch:

  (tutorial) S. Jegelka and A. Krause: Submodularity: Theory and Applications, Simons Institute, 2017 (available on YouTube)

  (course) J. Bilmes: Submodular Functions, Optimization, & Applications to Machine Learning, University of Washington, 2014 (available on YouTube)

- To read (Surveys & Monographs):

  A. Krause, D. Golovin: Submodular Function Maximization, 2012

  F. Bach: Learning with Submodular Functions: A Convex Optimization Perspective, 2013

  N. Buchbinder, M. Feldman: Submodular Functions Maximization Problems, 2018

  New: H. Hassani, A. Karbasi: Submodular Optimization: From Discrete to Continuous and Back (in progress, will appear later this year)

Other Resources on Submodularity

- To watch:

  (tutorial) S. Jegelka and A. Krause: Submodularity: Theory and Applications, Simons Institute, 2017 (available on YouTube)

  (course) J. Bilmes: Submodular Functions, Optimization, & Applications to Machine Learning, University of Washington, 2014 (available on YouTube)

- To read (Surveys & Monographs):

  A. Krause, D. Golovin: Submodular Function Maximization, 2012

  F. Bach: Learning with Submodular Functions: A Convex Optimization Perspective, 2013

  N. Buchbinder, M. Feldman: Submodular Functions Maximization Problems, 2018

  New: H. Hassani, A. Karbasi: Submodular Optimization: From Discrete to Continuous and Back (in progress, will appear later this year)

Thank you!

Submodular Optimization: From Discrete to Continuous and Back

Hamed Hassani

Amin Karbasi

Penn

Yale

Slides + references: http://iid.yale.edu/icml/icml-20.md/