Decentralized Submodular Maximization: Bridging Discrete and Continuous Settings

Hamed Hassani, University of Pennsylvania

Joint work with: Aryan Mokhtari Amin Karbasi

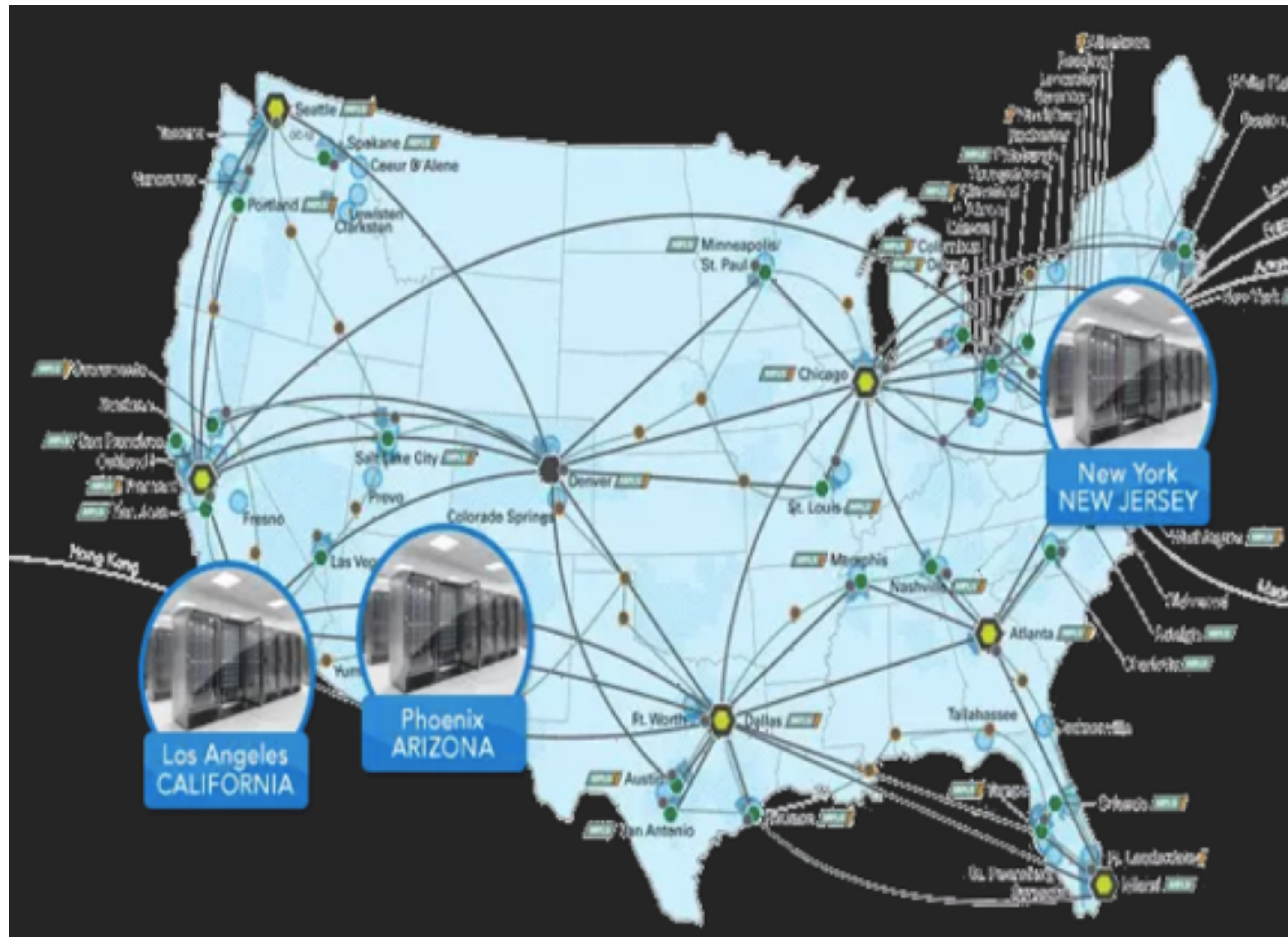
Workshop on Local Algorithms (WOLA)
June 15th, 2018

Penn
UNIVERSITY of PENNSYLVANIA

Yale

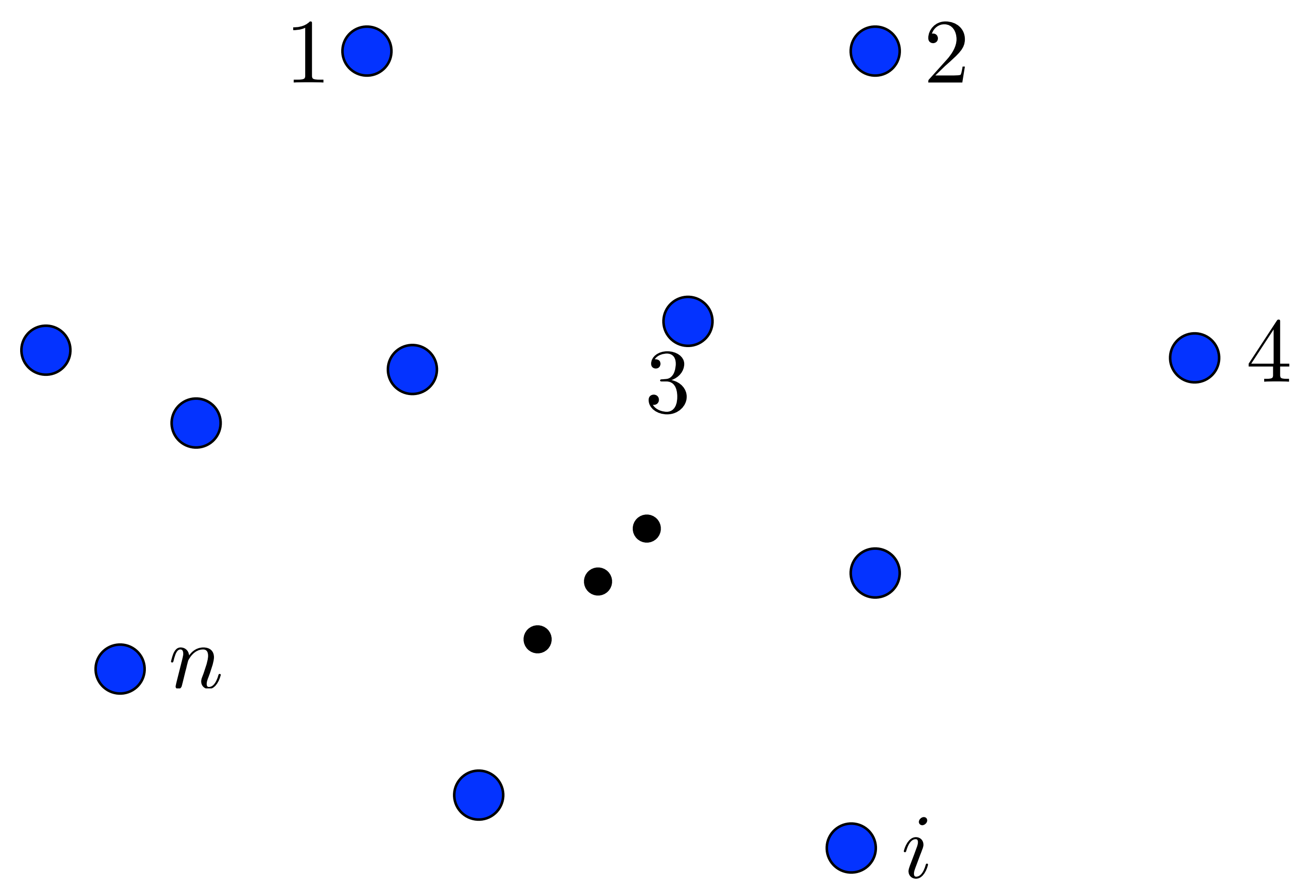
The Decentralized Setting

- Computing units distributed geographically
- Perform a global task (e.g. optimize a global function)
- Each unit has only access to a small portion of the problem (data)
- Data can’t be shared due to communication, storage, privacy constraints
- So the units have to cooperate/communicate with each other …

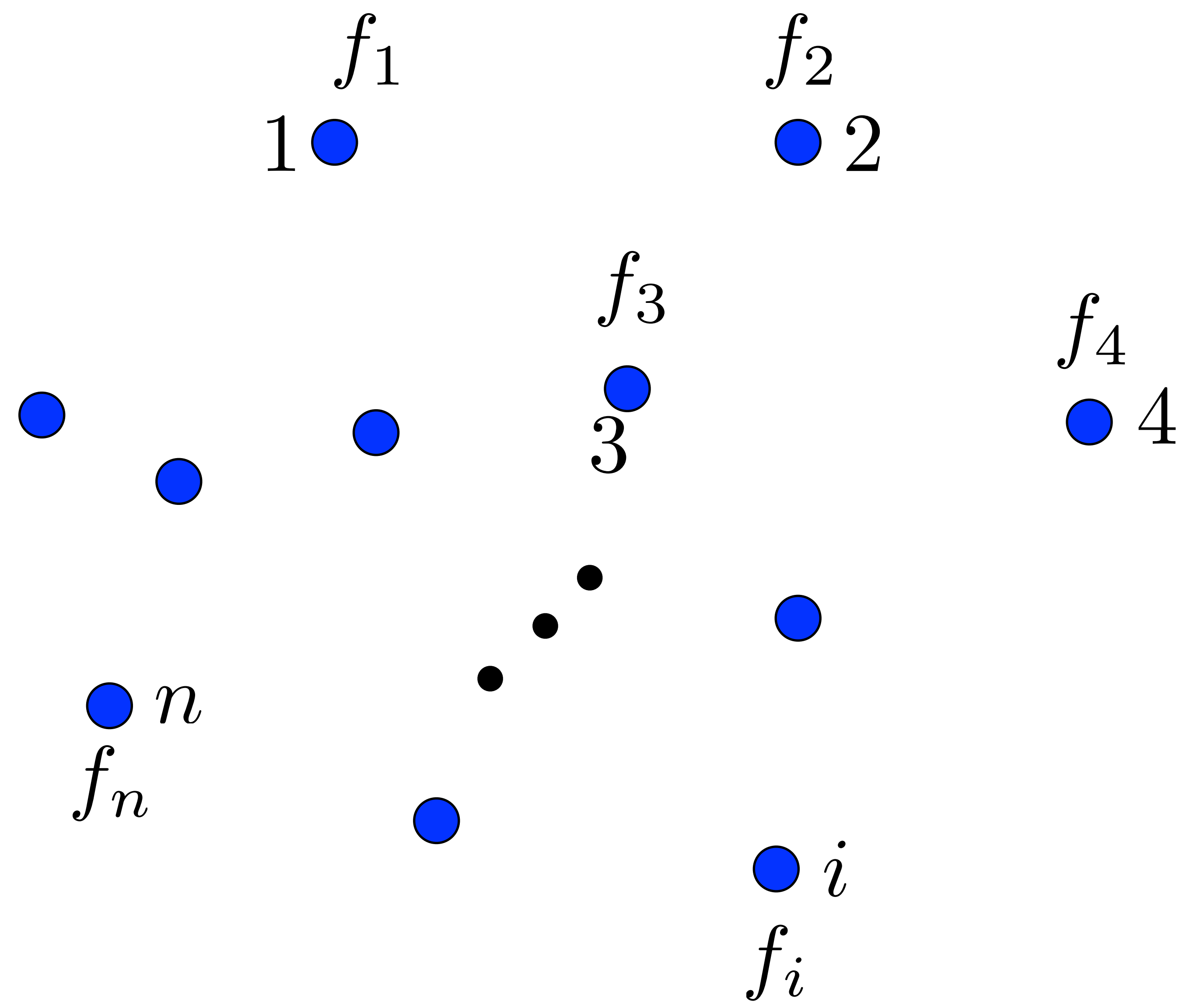


Decentralized Submodular Maximization

- n nodes (agents, computing units, etc)

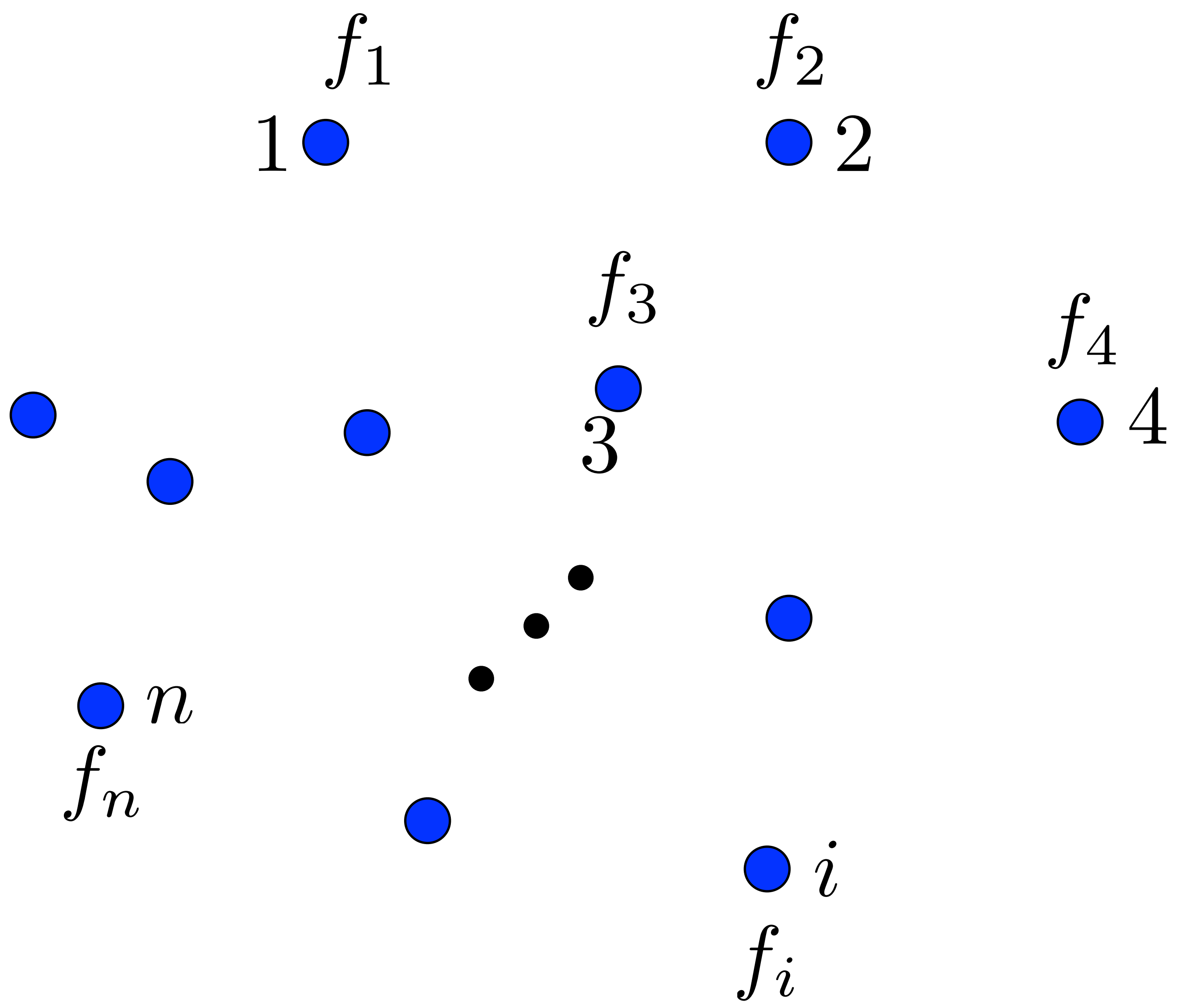


Decentralized Submodular Maximization



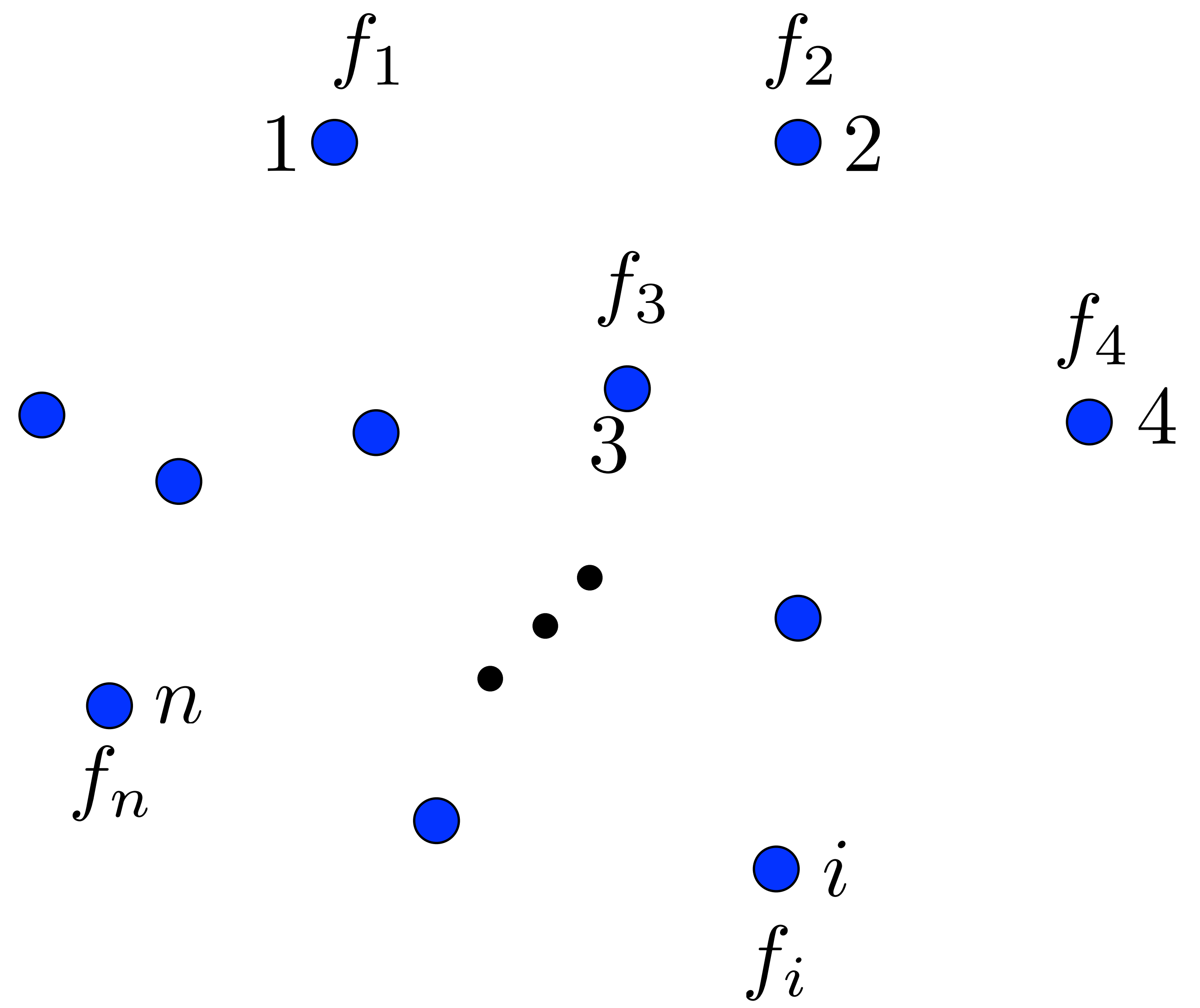
- n nodes (agents, computing units, etc)
- Each node has access to a local function f_i

Decentralized Submodular Maximization



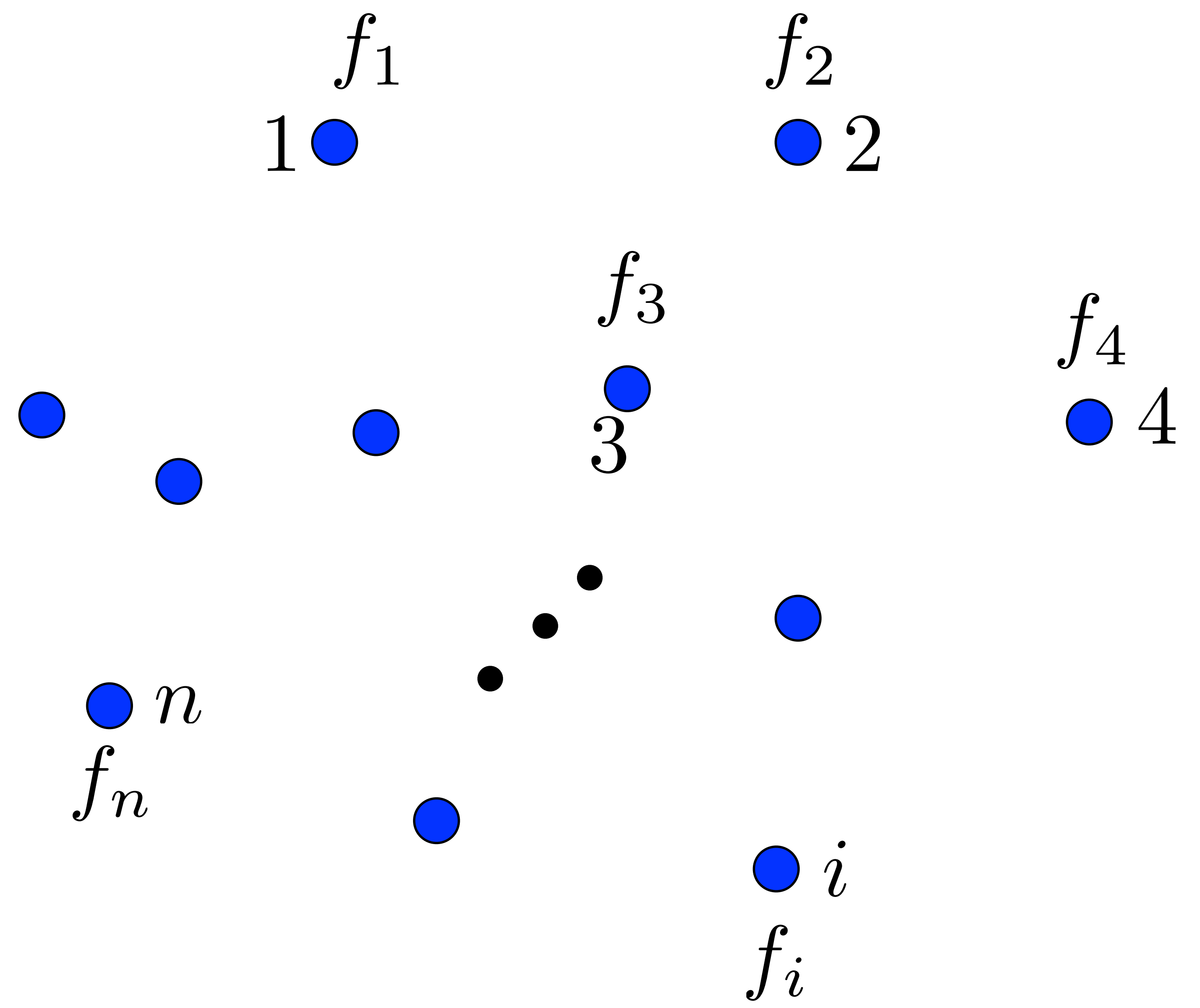
- n nodes (agents, computing units, etc)
- Each node has access to a local function f_i
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular

Decentralized Submodular Maximization



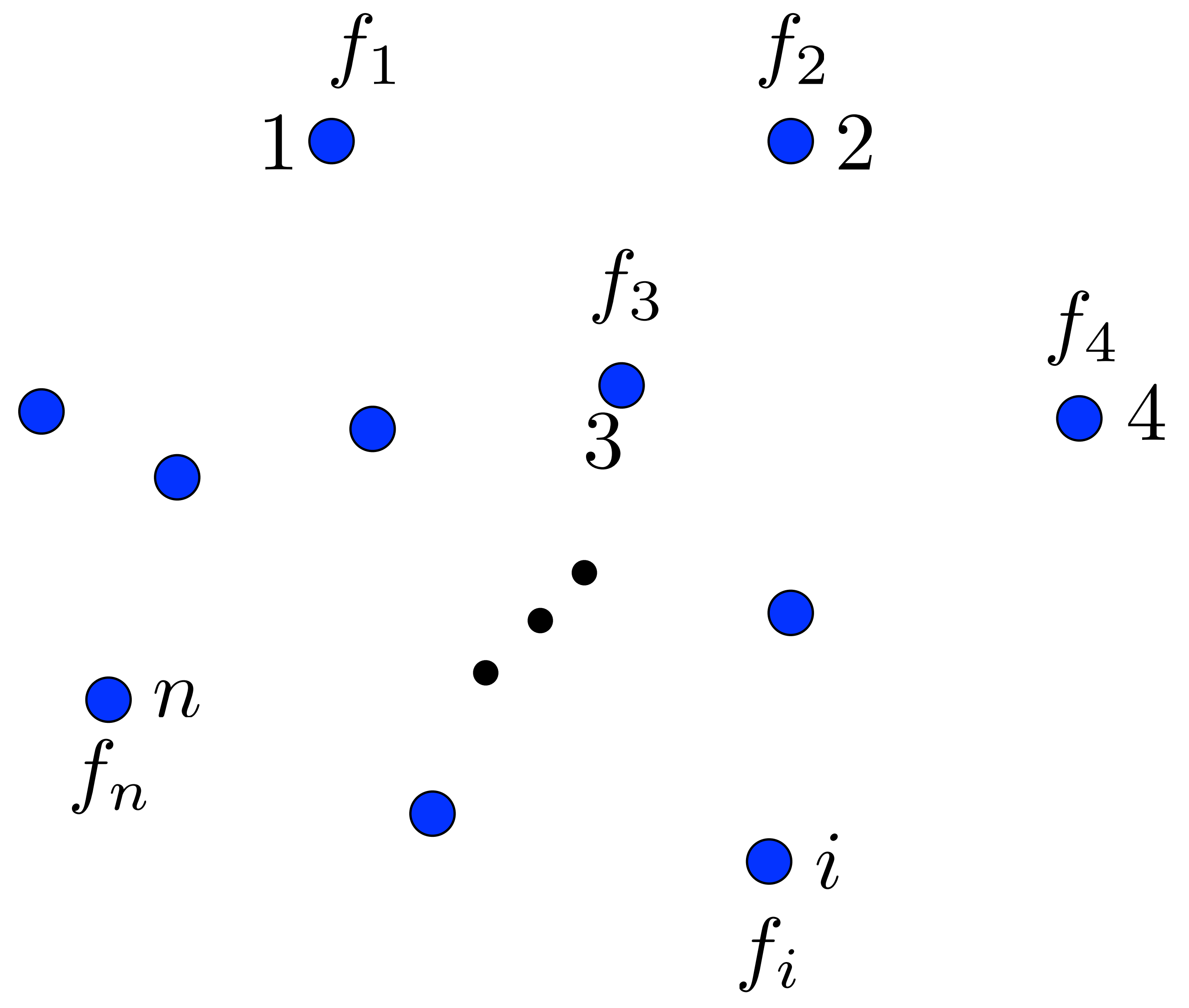
- n nodes (agents, computing units, etc)
- Each node has access to a local function f_i
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular
- Goal: maximize $\frac{1}{n}\sum_{i=1}^{n} f_i(S)$

Decentralized Submodular Maximization



- n nodes (agents, computing units, etc)
- Each node has access to a local function f_i
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular
- Goal: $\underset{|S| \leq k}{\text{maximize}} \quad \frac{1}{n} \sum_{i=1}^{n} f_i(S)$

Decentralized Submodular Maximization

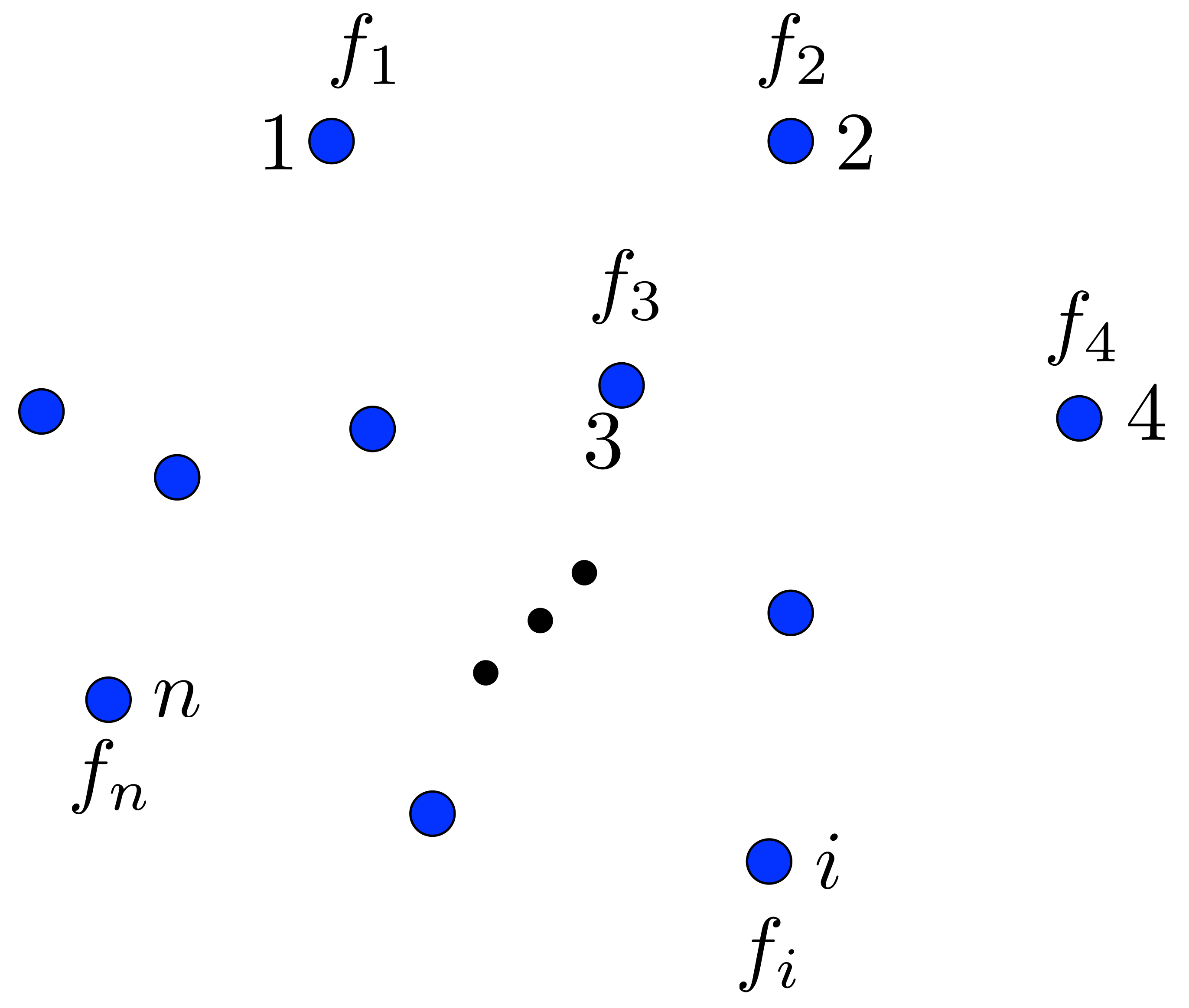


- n nodes (agents, computing units, etc)
- Each node has access to a local function f_i
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular
- Goal: maximize $\frac{1}{n}\sum_{i=1}^{n} f_i(S)$ subject to $|S| \leq k$

global objective

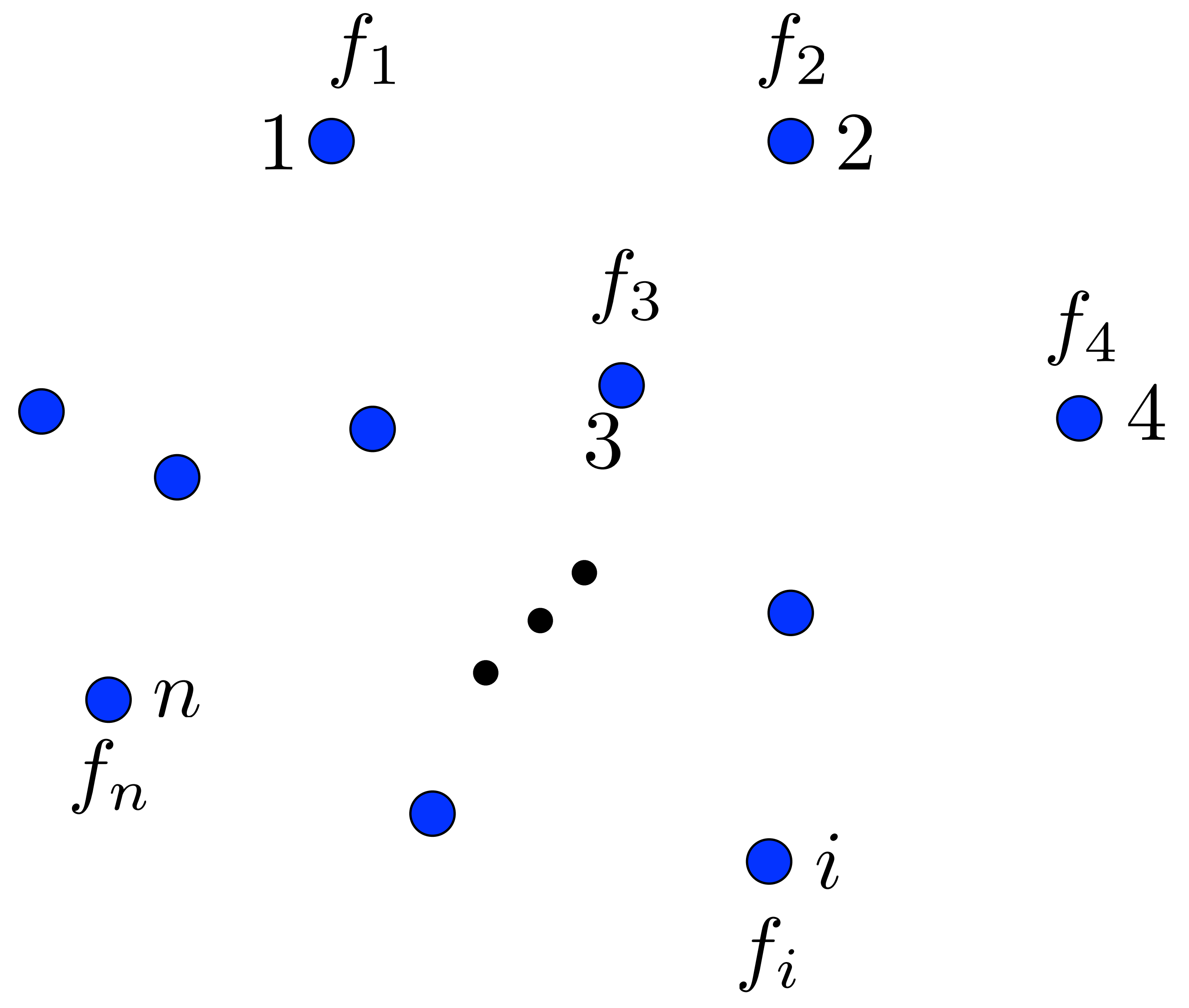
k-cardinality constraint

Decentralized Submodular Maximization



- n nodes (agents, computing units, etc)
- Each node has access to a local function f_i
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular
- Goal: maximize $\frac{1}{n}\sum_{i=1}^{n} f_i(S)$ subject to $|S| \leq k$
  - $\frac{1}{n}\sum_{i=1}^{n} f_i(S)$ → global objective
  - $|S| \leq k$ → k-cardinality constraint
- Challenge: Each node only knows its local f_i

Decentralized Submodular Maximization



- n nodes (agents, computing units, etc)
- Each node has access to a local function f_i
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular
- Goal:

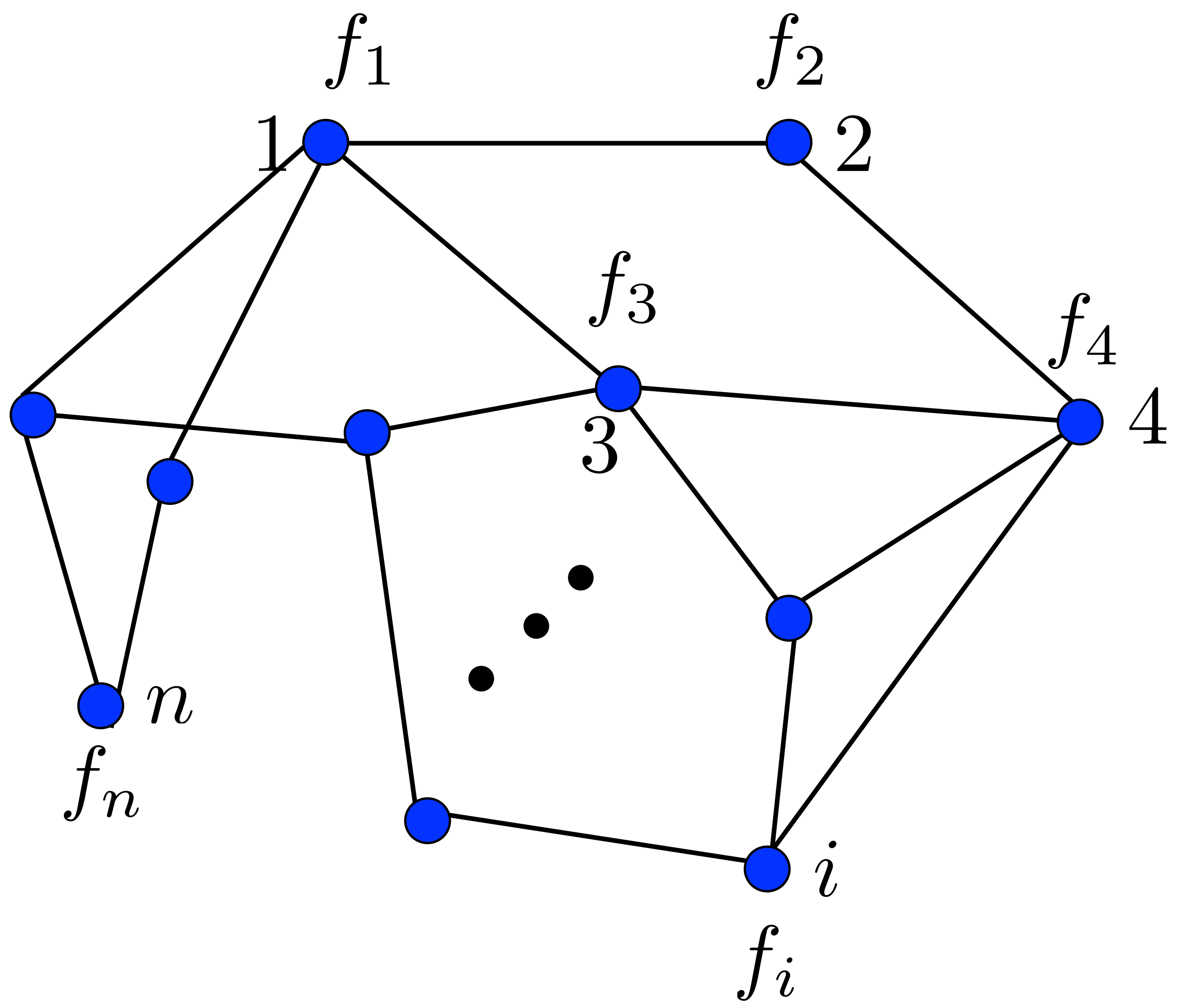
$$\text{maximize}_{|S| \leq k} \; \frac{1}{n} \sum_{i=1}^{n} f_i(S)$$

global objective

k-cardinality constraint

- Challenge: Each node only knows its local f_i
- To maximize the global objective, the nodes have to cooperate/communicate

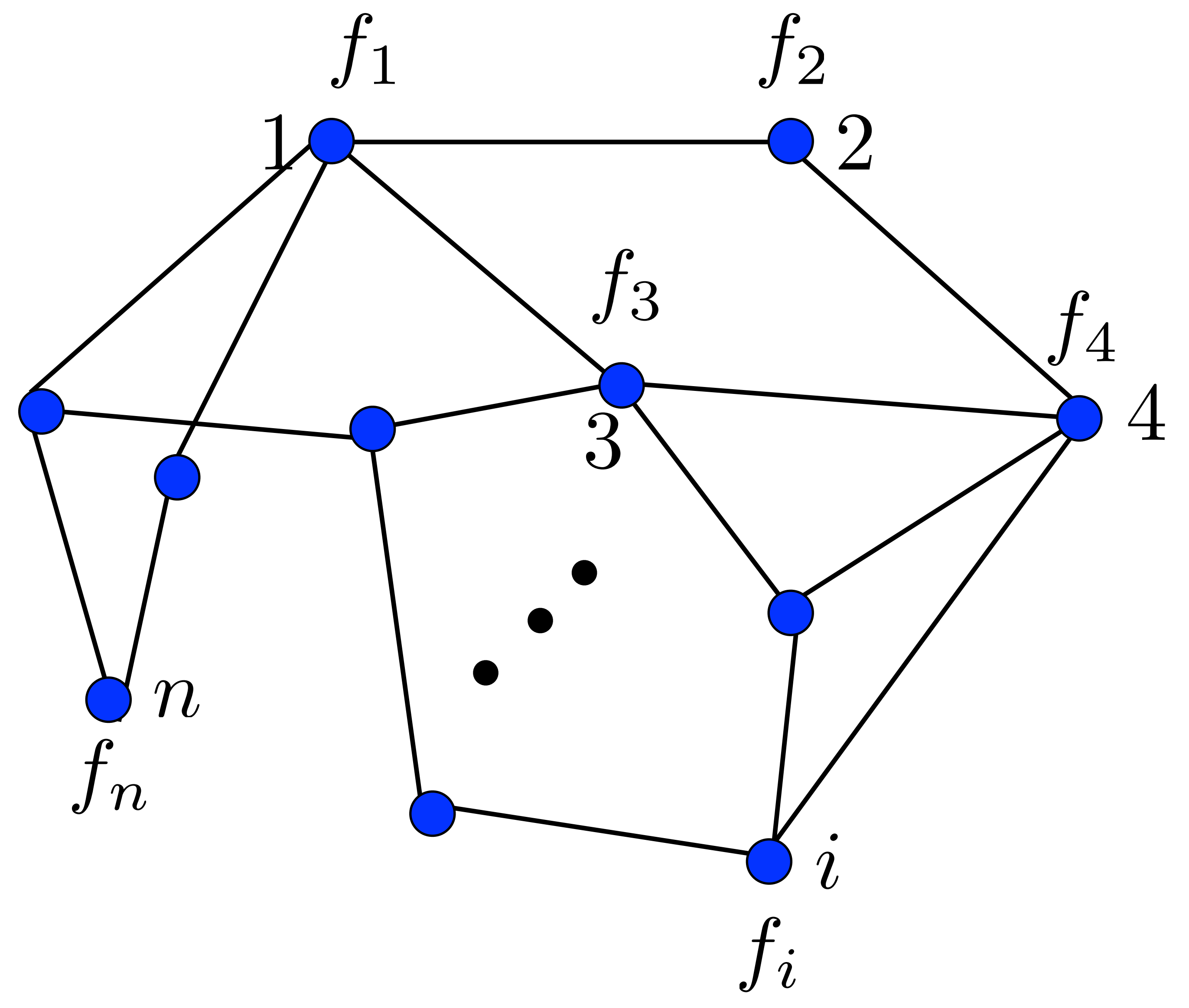
Decentralized Submodular Maximization



$G = (N, E)$

- n nodes (agents, computing units, etc)
- Each node has access to a local function f_i
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular
- Goal: maximize $\frac{1}{n} \sum_{i=1}^{n} f_i(S)$ (global objective), $|S| \leq k$ (k-cardinality constraint)
- Challenge: Each node only knows its local f_i
- To maximize the global objective, the nodes have to cooperate/communicate

Decentralized Submodular Maximization



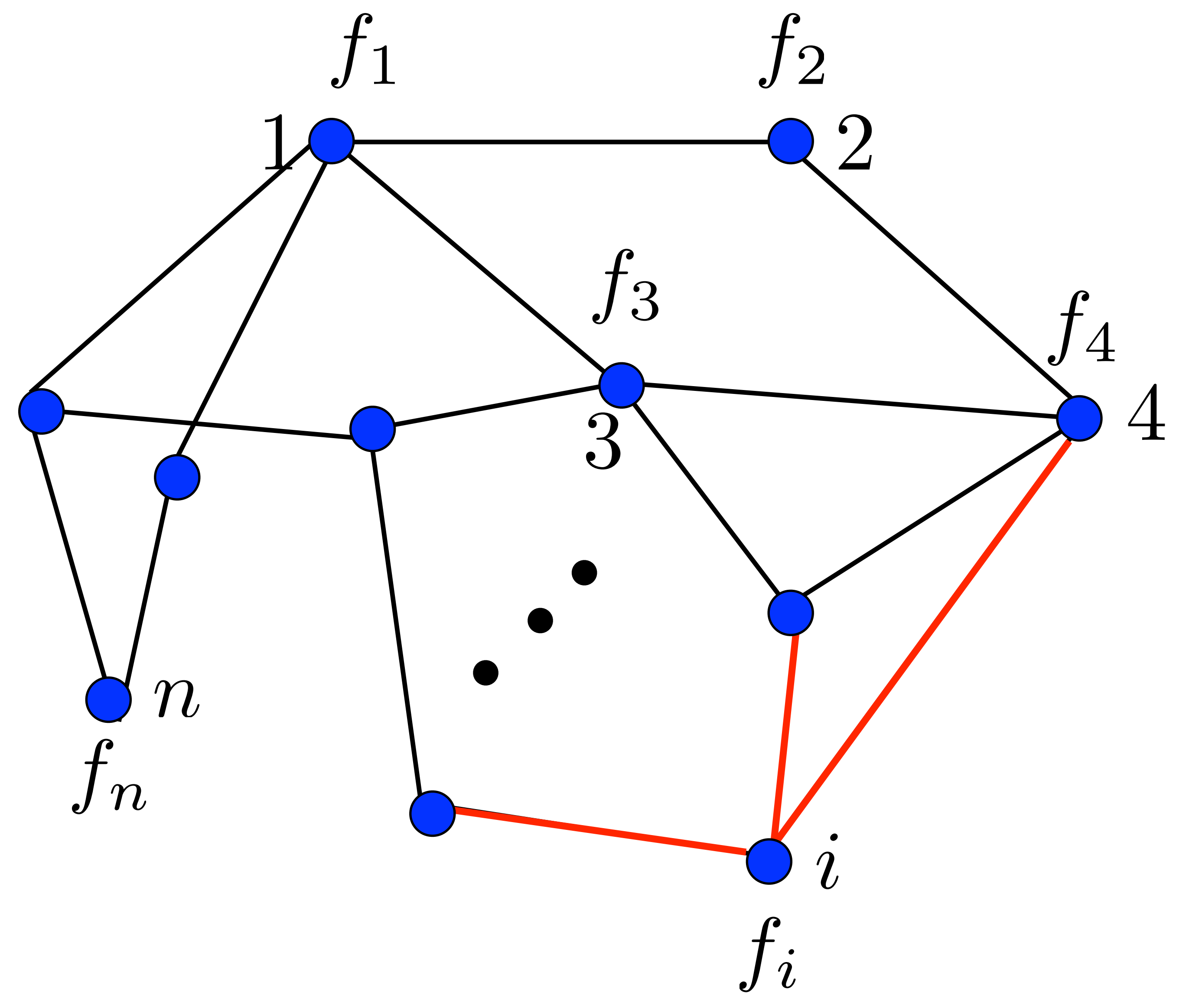
$G = (N, E)$

- n nodes (agents, computing units, etc)
- Each node has access to a local function f_i
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular
- Goal: maximize$_{|S| \leq k}$ $\frac{1}{n} \sum_{i=1}^{n} f_i(S)$

  $\frac{1}{n} \sum_{i=1}^{n} f_i(S)$ → global objective

  $|S| \leq k$ → k-cardinality constraint
- Challenge: Each node only knows its local f_i
- To maximize the global objective, the nodes have to cooperate/communicate
- Each node can communicate to its neighbours within G

Decentralized Submodular Maximization



$G = (N, E)$

- n nodes (agents, computing units, etc)
- Each node has access to a local function f_i
- $f_i : 2^V \rightarrow \mathbb{R}$ is monotone and submodular
- Goal: maximize$_{|S| \leq k}$ $\frac{1}{n}\sum_{i=1}^{n} f_i(S)$

  $|S| \leq k$ → k-cardinality constraint

  $\frac{1}{n}\sum_{i=1}^{n} f_i(S)$ → global objective
- Challenge: Each node only knows its local f_i
- To maximize the global objective, the nodes have to cooperate/communicate
- Each node can communicate to its neighbours within G

Submodular Functions

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

Submodular Functions

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

$A \subseteq V \Rightarrow f(A)$

Submodular Functions

$f : 2^V \to \mathbb{R}$ $\quad\quad$ $A \subseteq V$ ➡ $f(A)$

- The "diminishing returns" property:

$A \subseteq B$ ➡ $f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$

Submodular Functions

$f : 2^V \rightarrow \mathbb{R}$ $\qquad A \subseteq V \Rightarrow f(A)$

- The “diminishing returns” property:

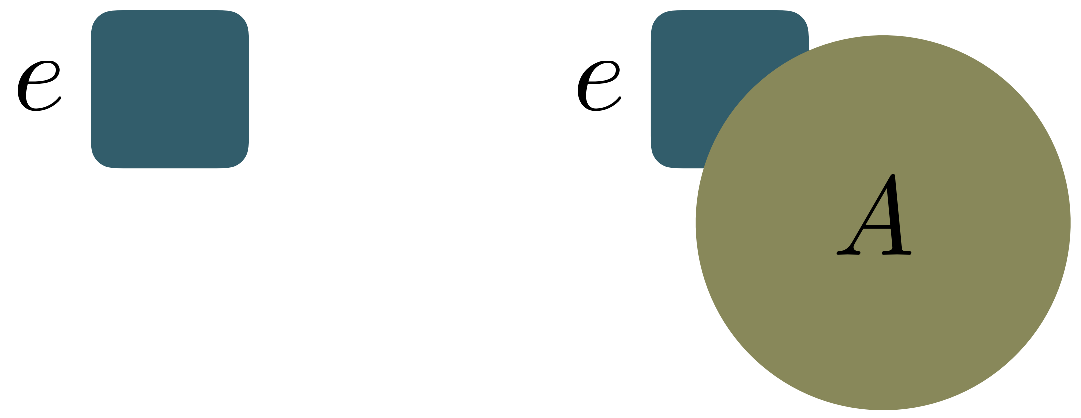
$$A \subseteq B \Rightarrow f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$$

Submodular Functions

$$f : 2^V \to \mathbb{R} \qquad A \subseteq V \Rightarrow f(A)$$

- The "diminishing returns" property:

$$A \subseteq B \Rightarrow f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$$

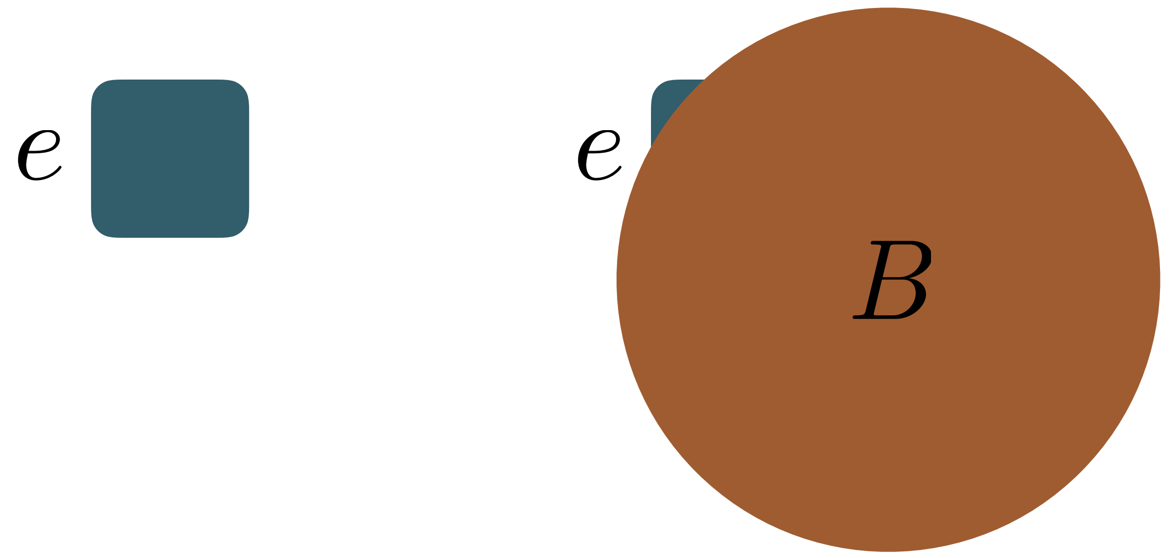


Submodular Functions

$$f : 2^V \to \mathbb{R} \qquad A \subseteq V \Rightarrow f(A)$$

- The "diminishing returns" property:

$$A \subseteq B \Rightarrow f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$$

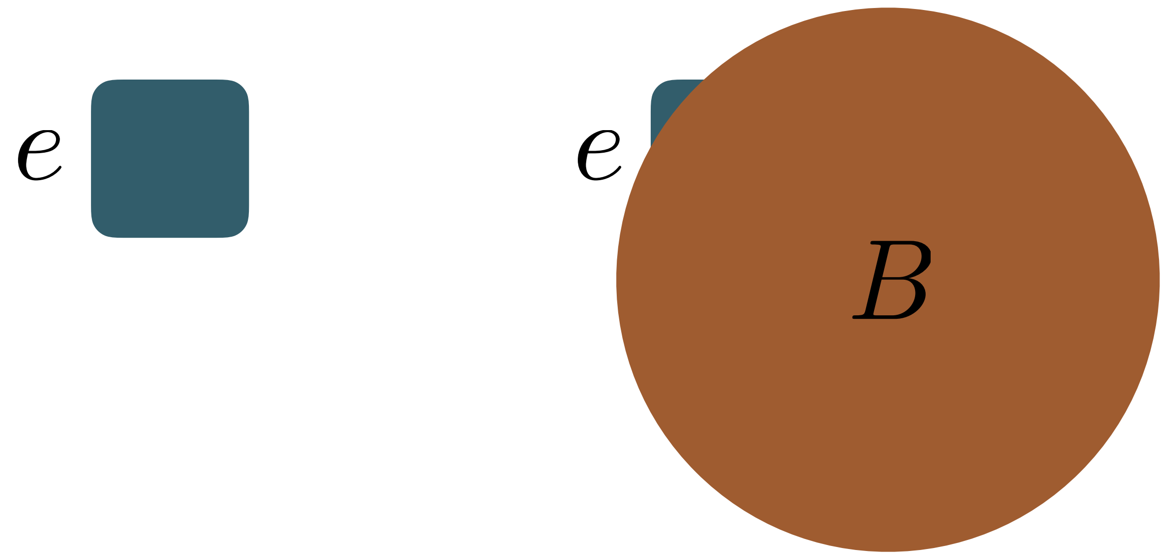


Submodular Functions

$f : 2^V \to \mathbb{R}$ $\qquad$ $A \subseteq V \Rightarrow f(A)$

- The "diminishing returns" property:

$$A \subseteq B \Rightarrow f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$$



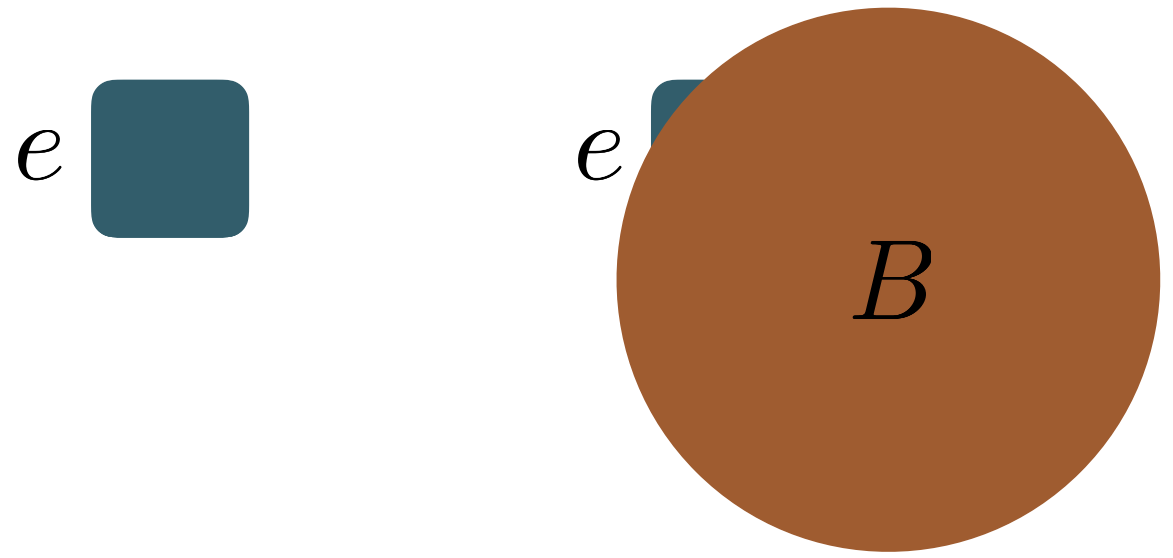
- Monotonicity: $A \subseteq B \Rightarrow f(A) \leq f(B)$

Submodular Functions

$f : 2^V \to \mathbb{R}$ $\quad$ $A \subseteq V \Rightarrow f(A)$

- The "diminishing returns" property:

$$A \subseteq B \Rightarrow f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$$



- Monotonicity: $A \subseteq B \Rightarrow f(A) \leq f(B)$

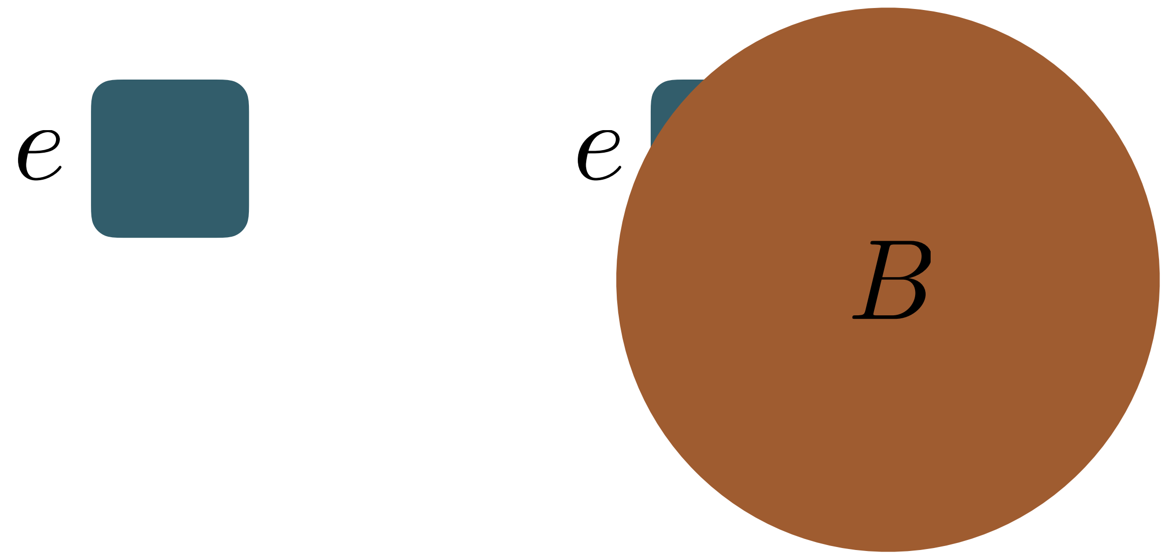
- Submodular maximization: $\max_{S:|S|\leq k} f(S)$

Submodular Functions

$f : 2^V \to \mathbb{R}$ $\qquad A \subseteq V \rightarrow f(A)$

- The "diminishing returns" property:

$$A \subseteq B \Rightarrow f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$$



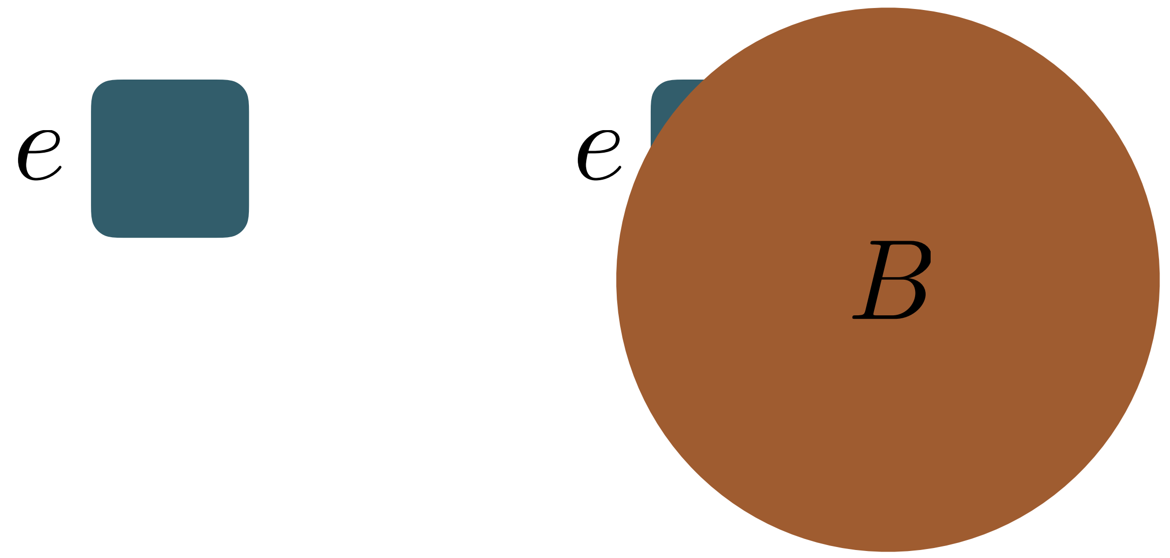
- Monotonicity: $A \subseteq B \Rightarrow f(A) \leq f(B)$
- Submodular maximization: $\max_{S:|S|\leq k} f(S)$
- Greedy algorithm: $(1 - 1/e)$-approximation

Submodular Functions

$f : 2^V \to \mathbb{R}$ $A \subseteq V \rightarrow f(A)$

- The "diminishing returns" property:

$$A \subseteq B \rightarrow f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$$



- Monotonicity: $A \subseteq B \rightarrow f(A) \leq f(B)$

- Submodular maximization: $\max_{S:|S|\leq k} f(S)$

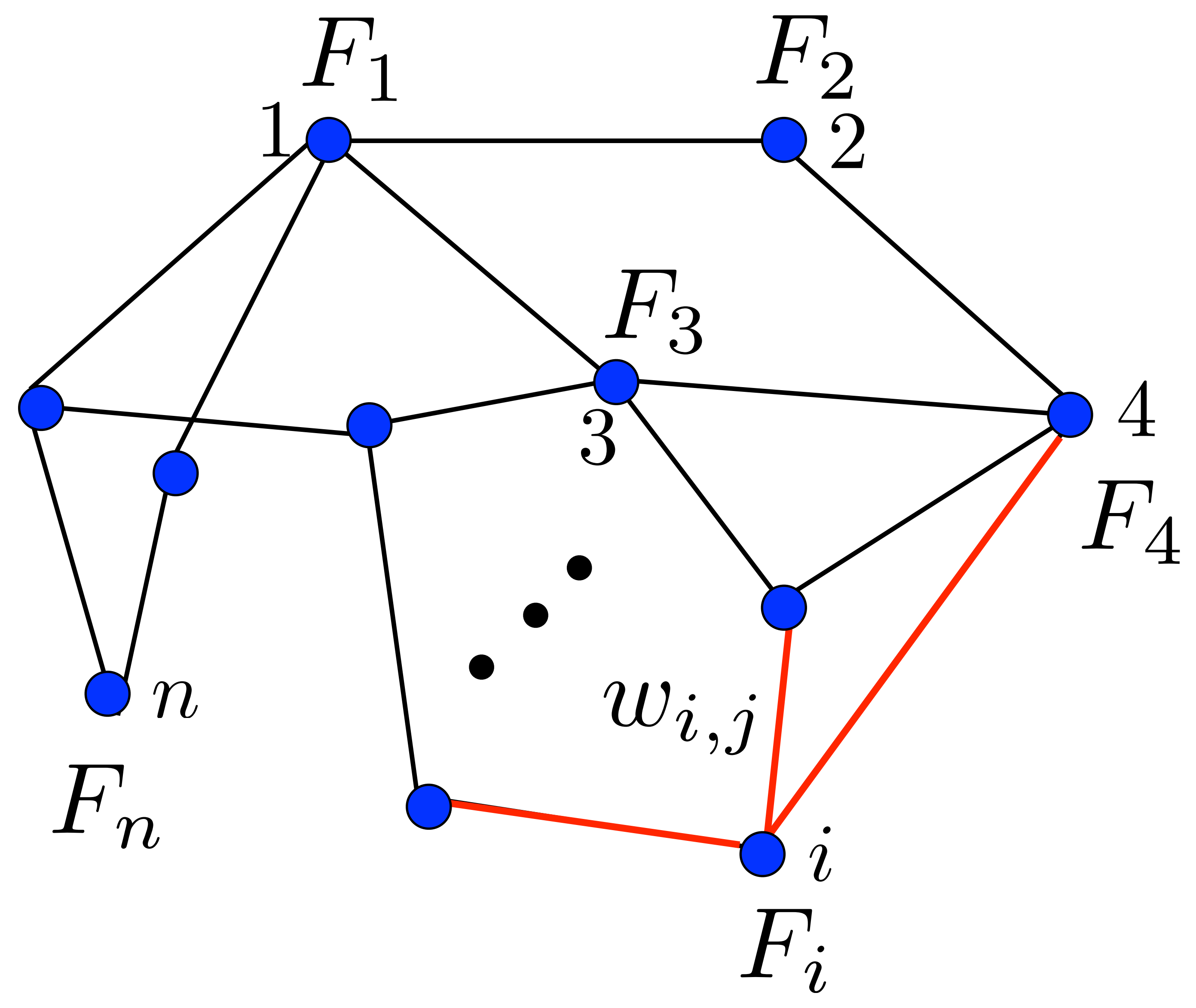
- Greedy algorithm: $(1 - 1/e)$-approximation

- Graph cut
- Coverage (vertex or set cover)
- Entropy
- Feature selection
- Experimental design
- Social Influence Maximization

Decentralized Submodular Maximization

- Outline for the rest of the talk:
    - Decentralized convex optimization
    - Continuous extensions of submodular functions
    - Centralized maximization of the continuous extension
    - The Decentralized Continuous Greedy (DCG) algorithm
    - Experiments

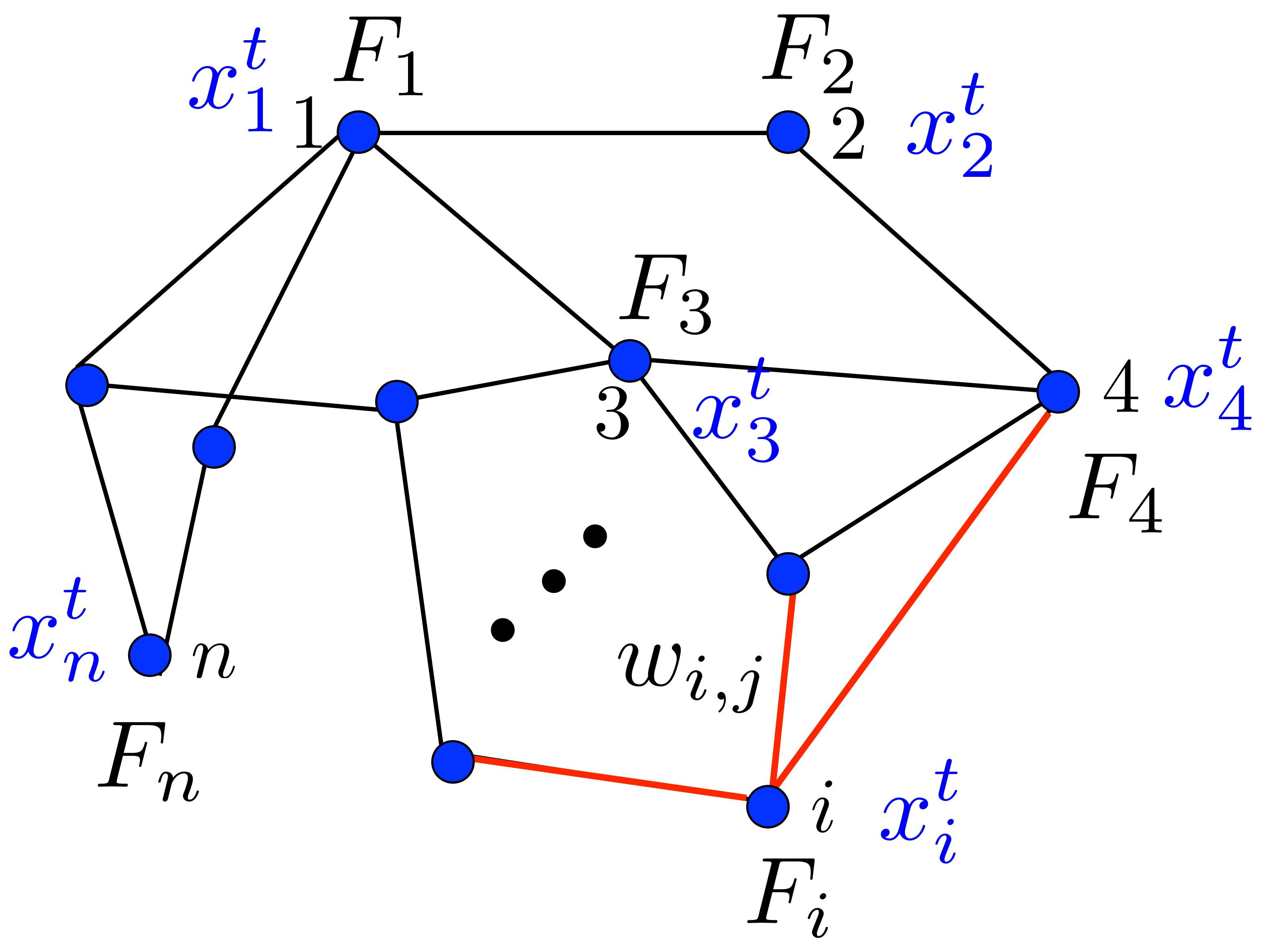
Decentralized Convex Minimization



$G = (N, E)$

- **Goal**: $\underset{x \in \mathcal{C}}{\text{minimize}} \; \frac{1}{n}\sum_{i=1}^{n} F_i(x)$ (F_i's are convex)

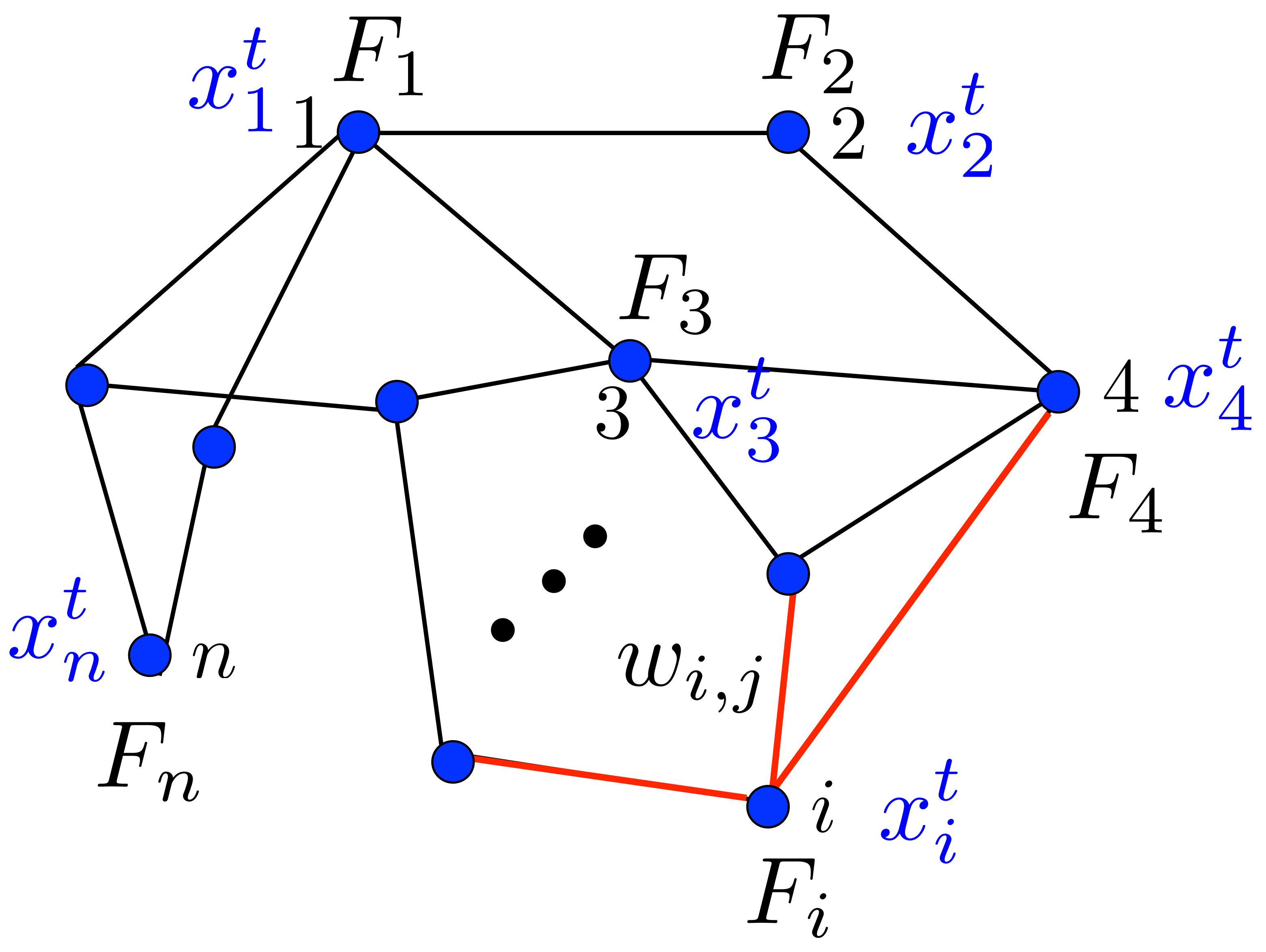
Decentralized Convex Minimization



$G = (N, E)$

- **Goal**: $\underset{x \in \mathcal{C}}{\text{minimize}} \ \frac{1}{n} \sum_{i=1}^{n} F_i(x)$ (F_i's are convex)

Decentralized Convex Minimization



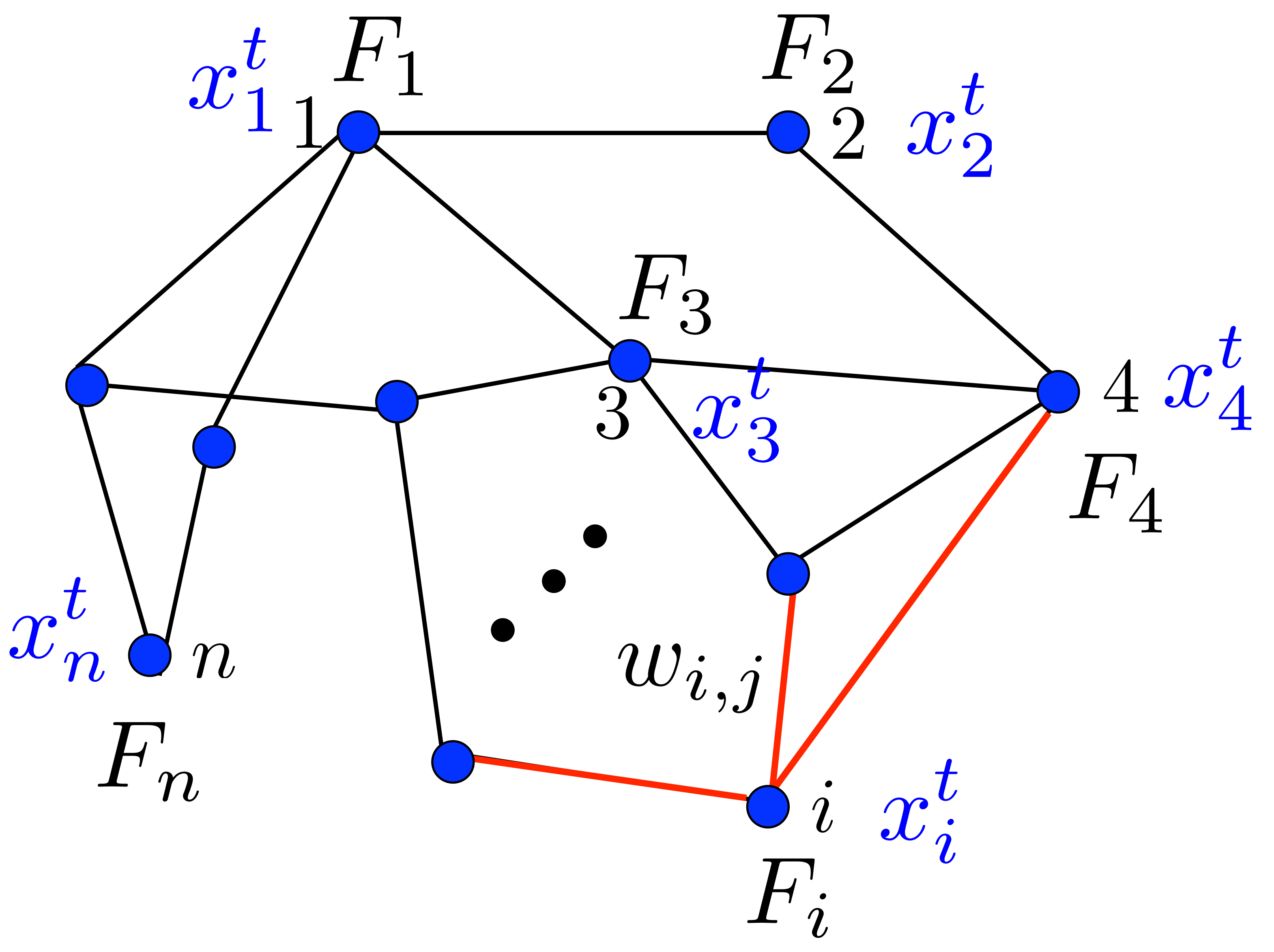
$G = (N, E)$

- **Goal**: $\underset{x \in \mathcal{C}}{\text{minimize}} \ \frac{1}{n}\sum_{i=1}^{n} F_i(x)$ (F_i's are convex)

[Nedic, Ozdaglar '09]

- **Basic idea**: $x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t - \eta_t \nabla F_i(x_i^t)$

Decentralized Convex Minimization



$G = (N, E)$

- **Goal:** $\underset{x \in \mathcal{C}}{\text{minimize}} \; \frac{1}{n} \sum_{i=1}^{n} F_i(x)$ (F_i's are convex)

[Nedic, Ozdaglar '09]

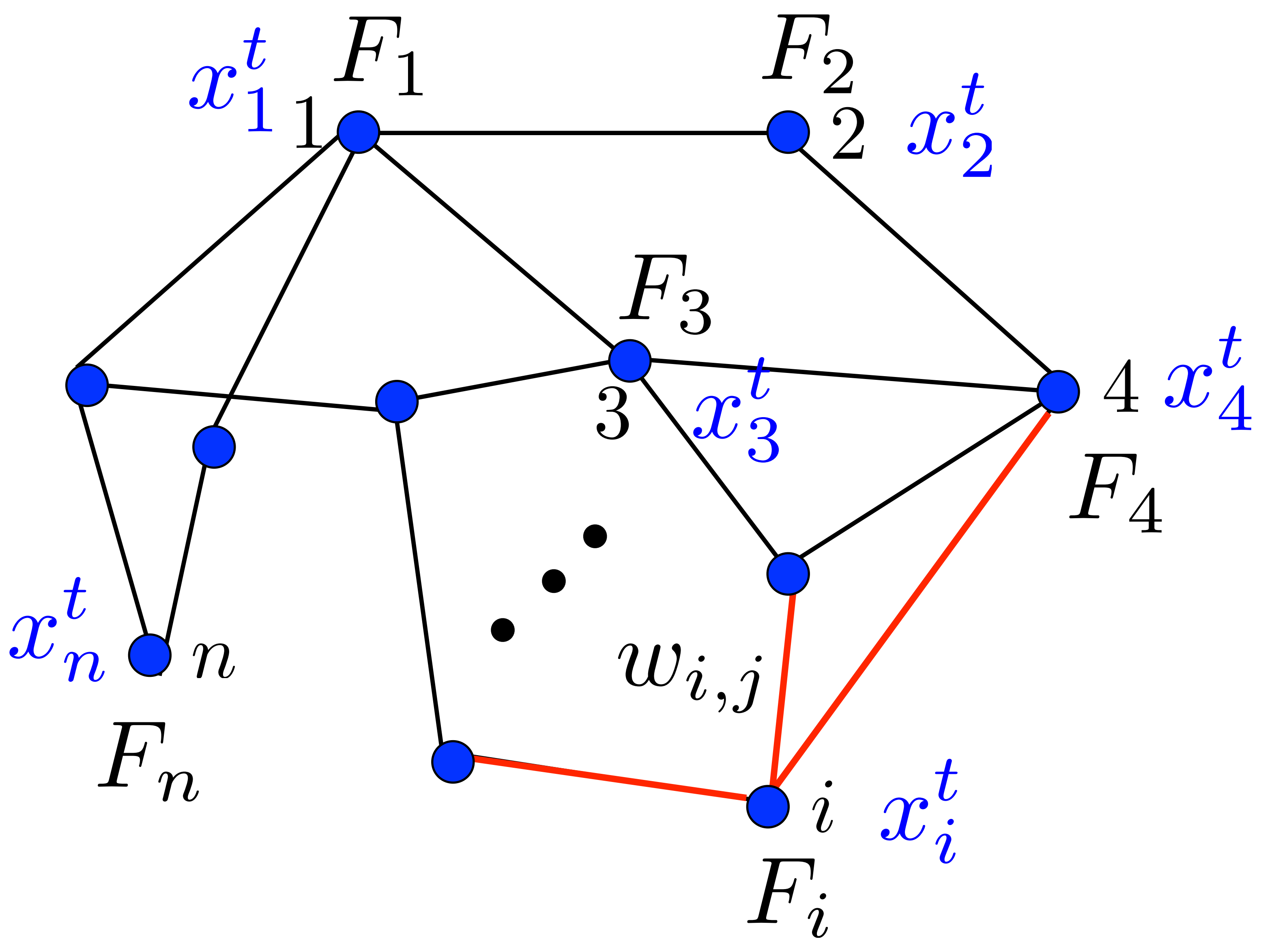
- **Basic idea:** $x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t - \eta_t \nabla F_i(x_i^t)$

$$\sum_{j \in N_i} w_{i,j} = 1$$

$$w_{i,j} = w_{j,i} \geq 0$$

forcing consensus

Decentralized Convex Minimization



$G = (N, E)$

- **Goal:** minimize $\frac{1}{n}\sum_{i=1}^{n} F_i(x)$ over $x \in \mathcal{C}$ (F_i's are convex)

[Nedic, Ozdaglar '09]

- **Basic idea:** $$x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t - \eta_t \nabla F_i(x_i^t)$$

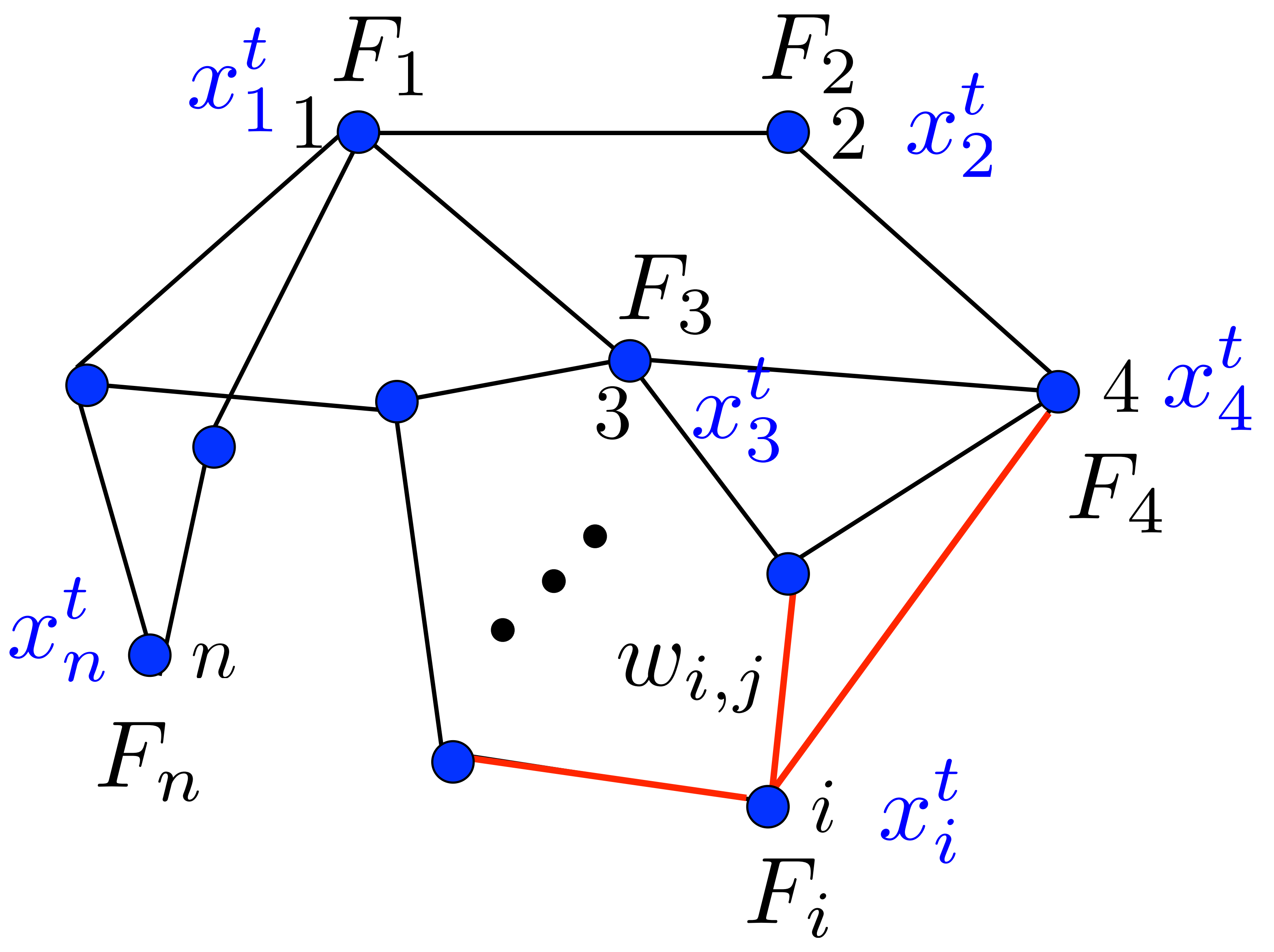
$$\sum_{j \in N_i} w_{i,j} = 1$$

$$w_{i,j} = w_{j,i} \geq 0$$

forcing consensus

descent direction

Decentralized Convex Minimization



$$G = (N, E)$$

- **Goal:** $\underset{x \in \mathcal{C}}{\text{minimize}} \ \frac{1}{n} \sum_{i=1}^{n} F_i(x)$ (F_i's are convex)

[Nedic, Ozdaglar '09]

- **Basic idea:** $x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t - \eta_t \nabla F_i(x_i^t)$

$$\sum_{j \in N_i} w_{i,j} = 1$$

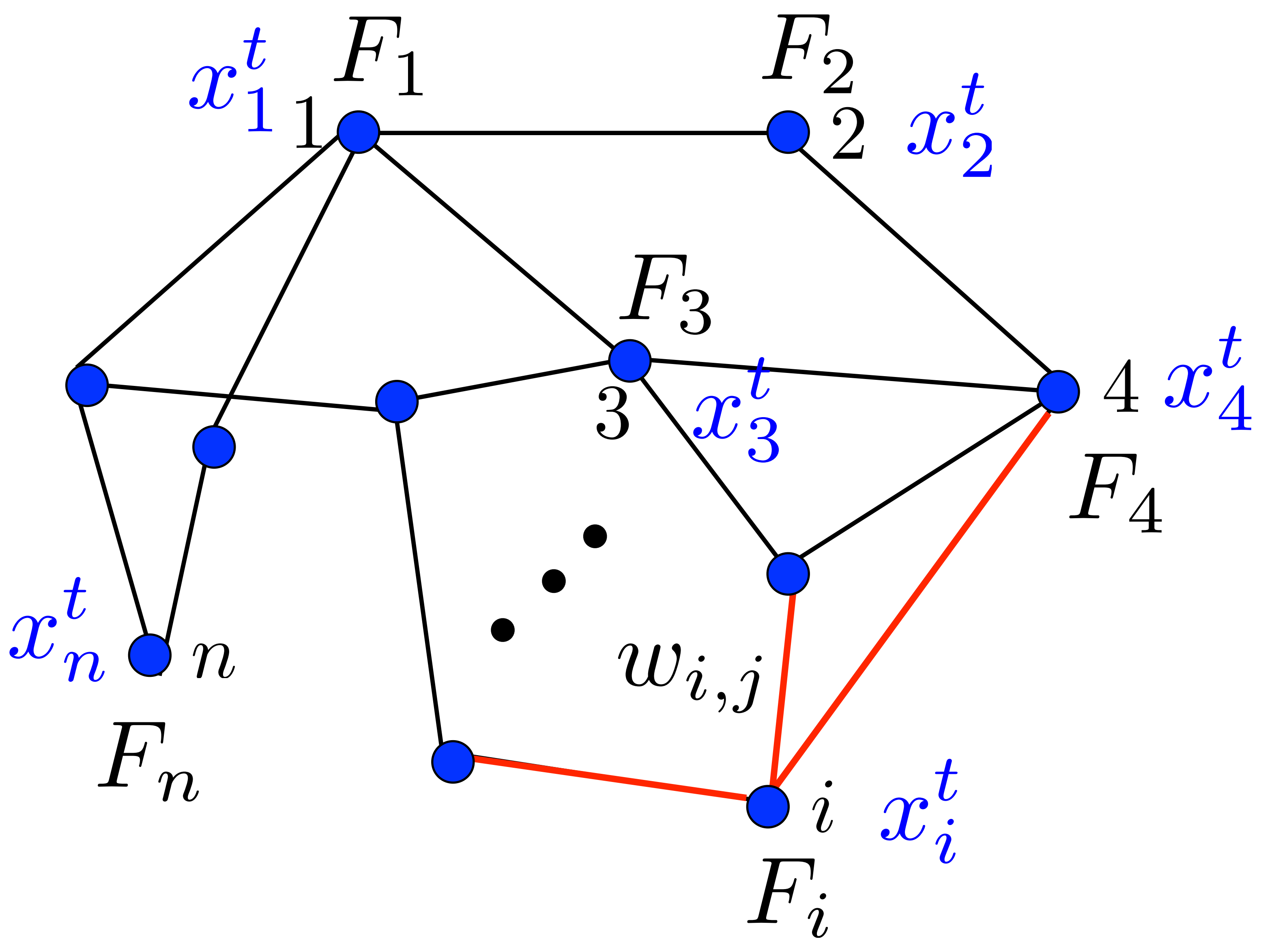
descent direction

$$w_{i,j} = w_{j,i} \geq 0$$

forcing consensus

- **Analysis:** $x_i^t \xrightarrow{t \to \infty} \bar{x}^t$

Decentralized Convex Minimization



$G = (N, E)$

- **Goal:** $\underset{x \in \mathcal{C}}{\text{minimize}} \ \frac{1}{n} \sum_{i=1}^{n} F_i(x)$ (F_i's are convex)

[Nedic, Ozdaglar '09]

- **Basic idea:** $x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t - \eta_t \nabla F_i(x_i^t)$

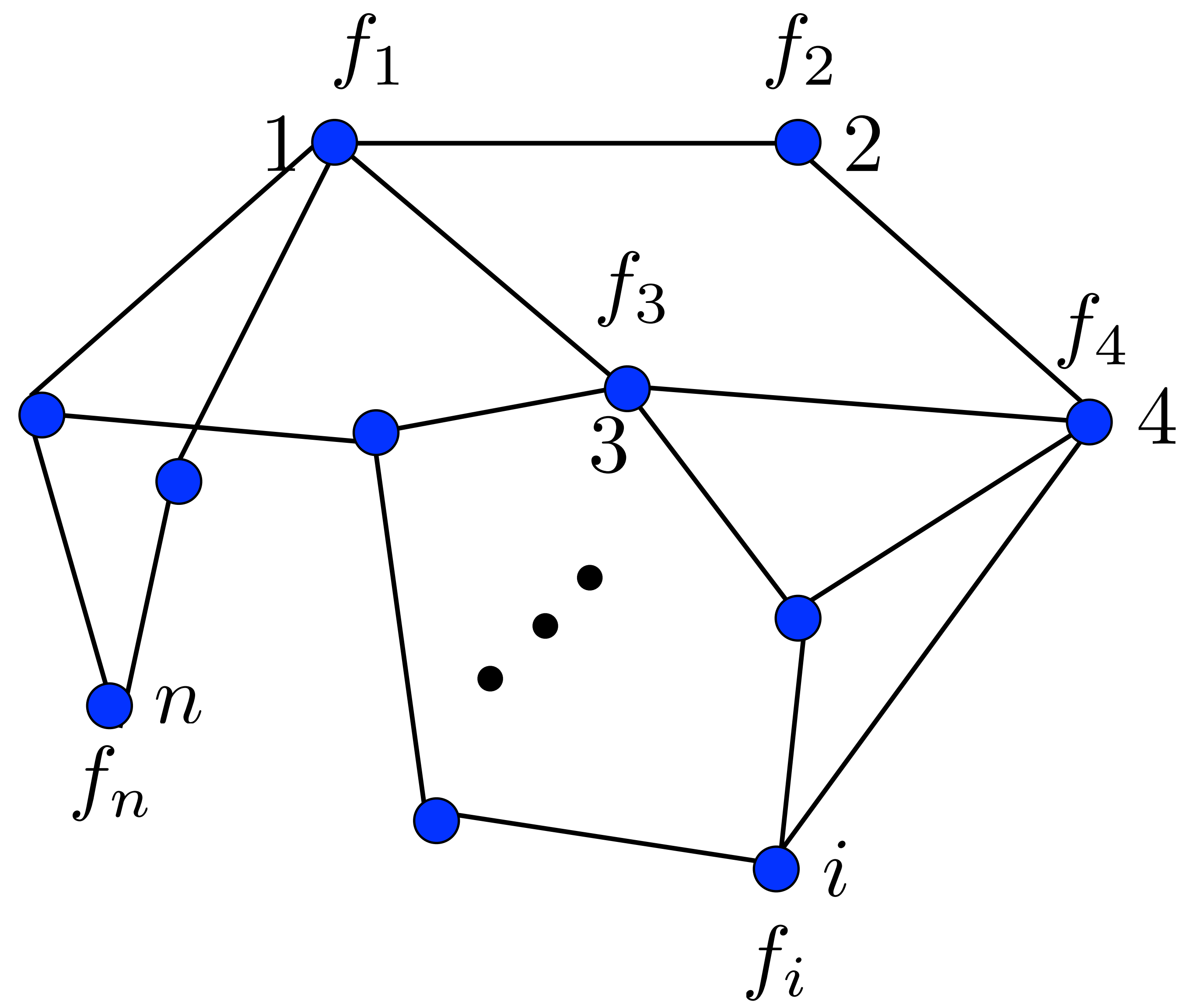
$\sum_{j \in N_i} w_{i,j} = 1$ descent direction

$w_{i,j} = w_{j,i} \geq 0$

forcing consensus

- **Analysis:** $x_i^t \xrightarrow{t \to \infty} \bar{x}^t \xrightarrow{t \to \infty} x^*$

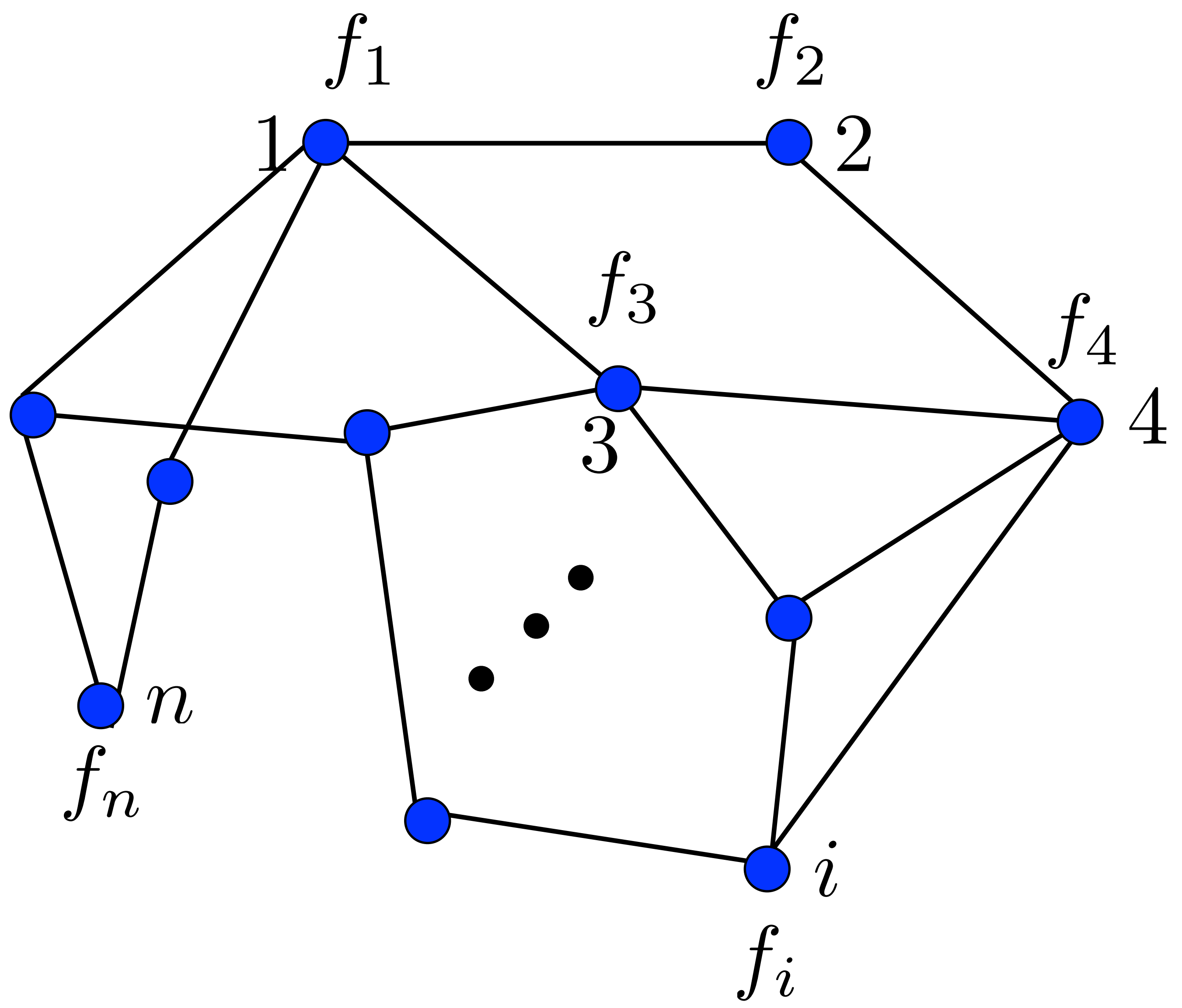
Decentralized Submodular Maximization



$G = (N, E)$

- Goal: $\underset{|S| \leq k}{\text{maximize}} \quad \frac{1}{n} \sum_{i=1}^{n} f_i(S)$
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular

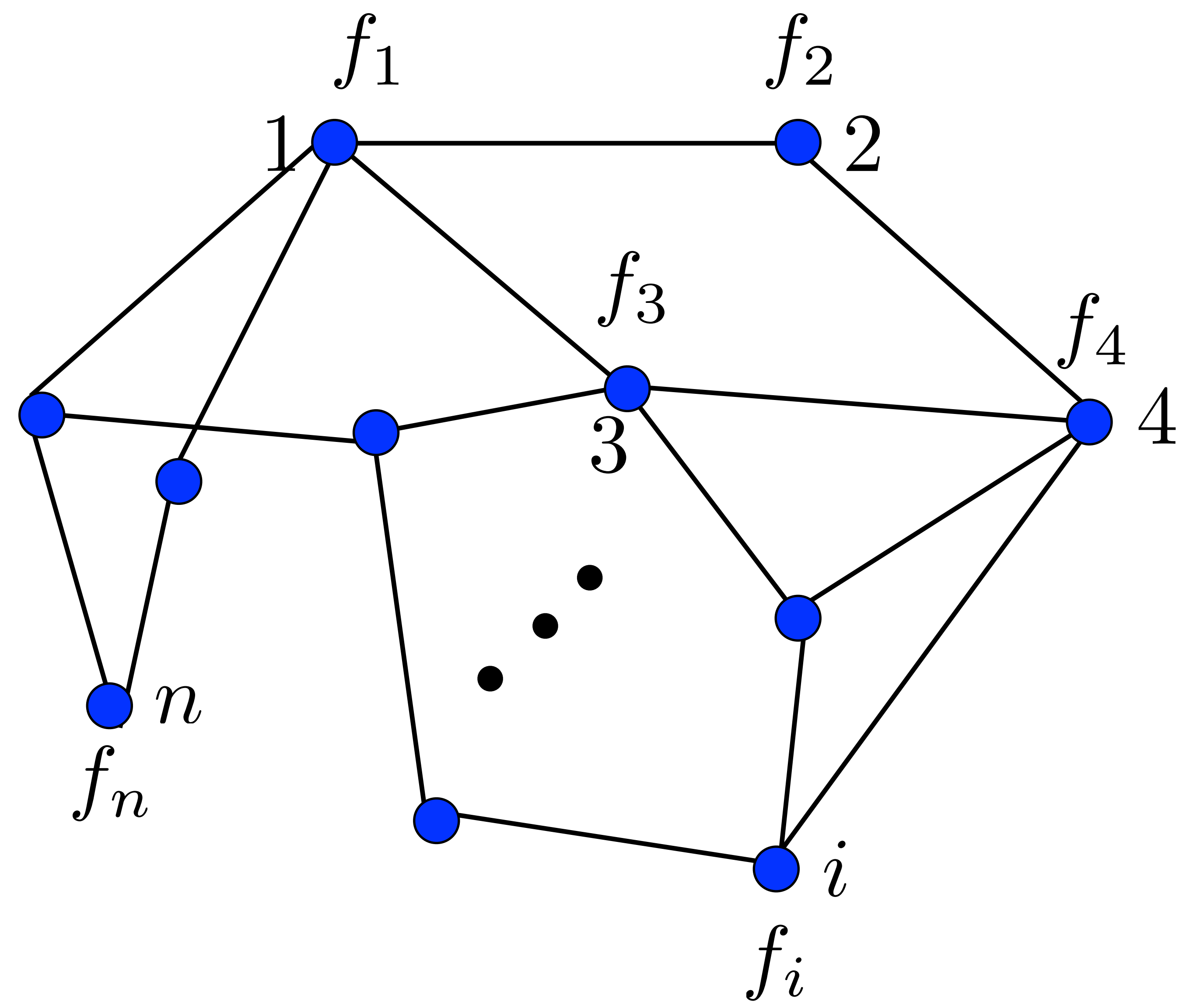
Decentralized Submodular Maximization



$G = (N, E)$

- **Goal**: $\underset{|S| \leq k}{\text{maximize}} \quad \frac{1}{n} \sum_{i=1}^{n} f_i(S)$
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular
- **Basic idea:** Exchange sets?

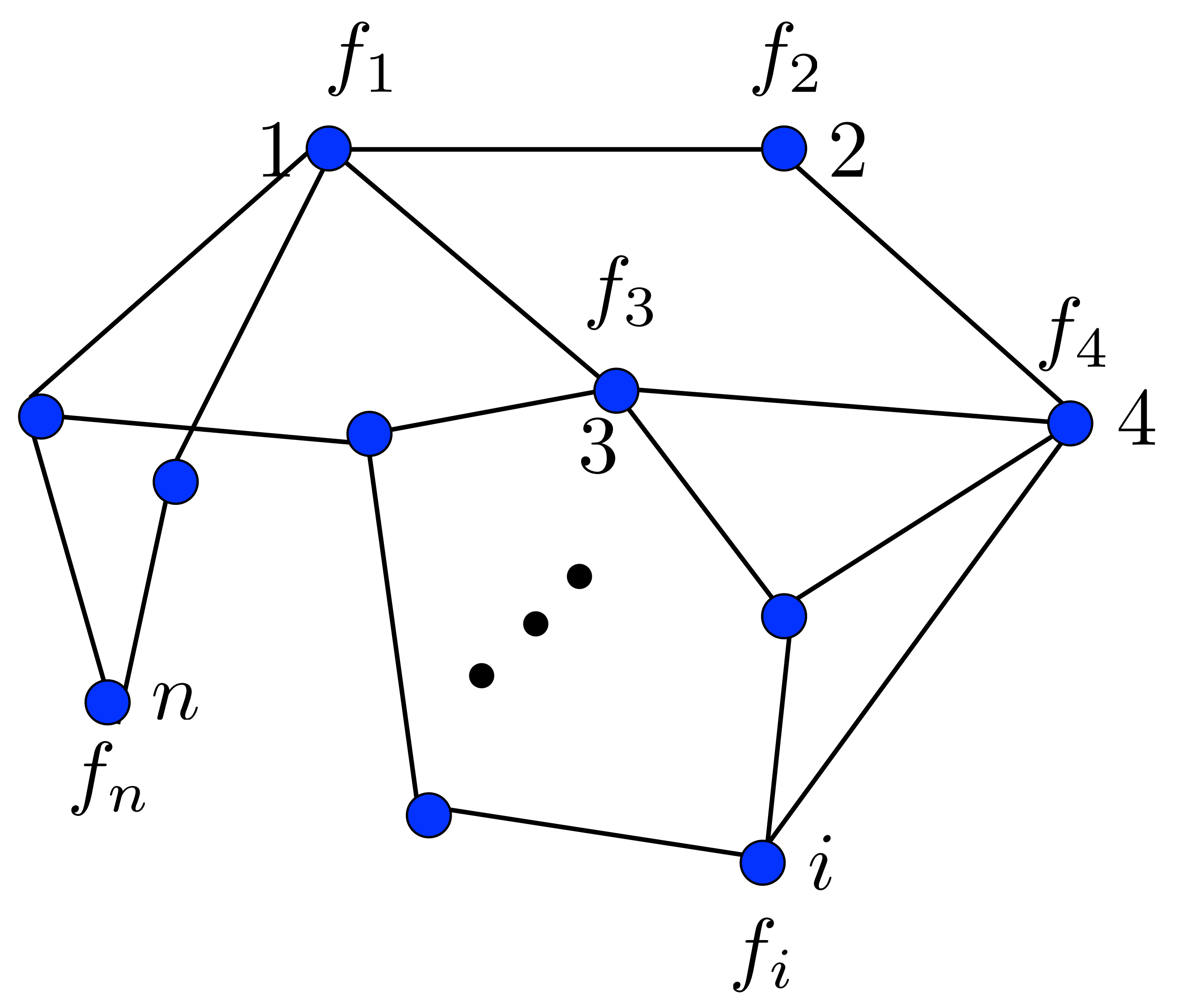
Decentralized Submodular Maximization



$G = (N, E)$

- Goal: $\underset{|S| \leq k}{\text{maximize}} \quad \frac{1}{n} \sum_{i=1}^{n} f_i(S)$
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular
- Basic idea: Exchange ~~sets~~ beliefs?

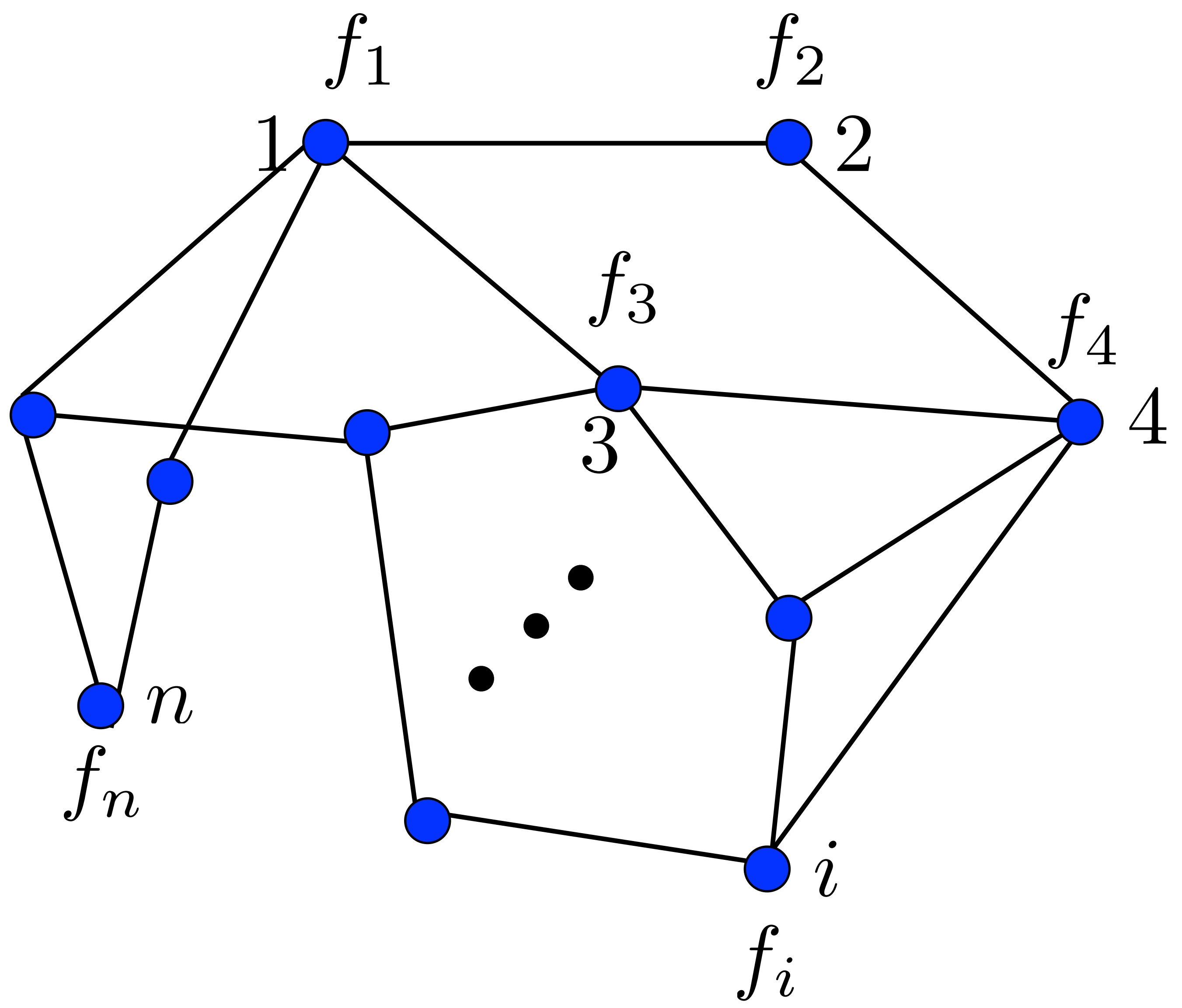
Decentralized Submodular Maximization



$G = (N, E)$

- **Goal**: $\underset{|S| \le k}{\text{maximize}} \ \frac{1}{n} \sum_{i=1}^{n} f_i(S)$
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular
- **Basic idea**: Exchange ~~sets~~? beliefs set functions?

Decentralized Submodular Maximization



$G = (N, E)$

- Goal: $\underset{|S| \leq k}{\text{maximize}} \; \frac{1}{n} \sum_{i=1}^{n} f_i(S)$
- $f_i : 2^V \to \mathbb{R}$ is monotone and submodular
- Basic idea: Exchange ~~sets~~? beliefs   ~~set functions~~? continuous extensions

Multilinear Extension

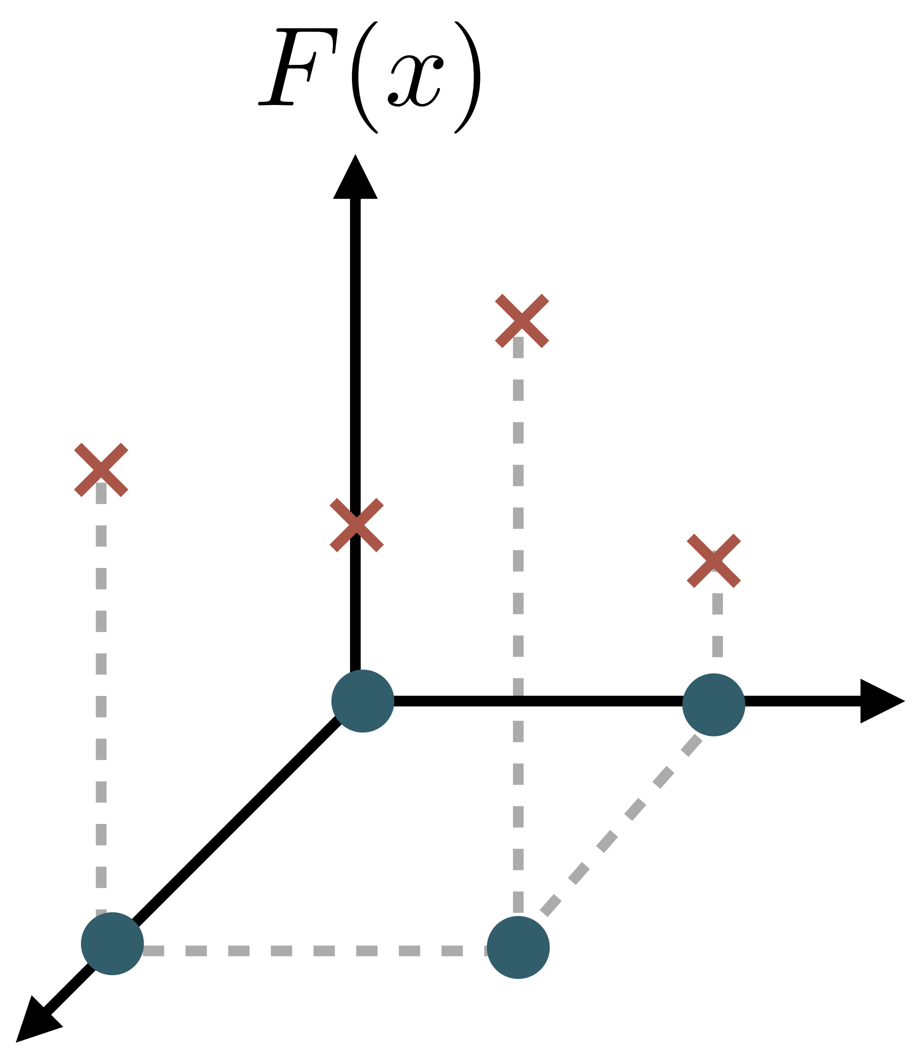
$f : 2^V \to \mathbb{R}$ ➡ $F : [0,1]^m \to \mathbb{R}$

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

Multilinear Extension

$f : 2^V \to \mathbb{R}$ → $F : [0,1]^m \to \mathbb{R}$

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



$f(S)$ →

$S \subseteq \{1, 2, \cdots, m\}$

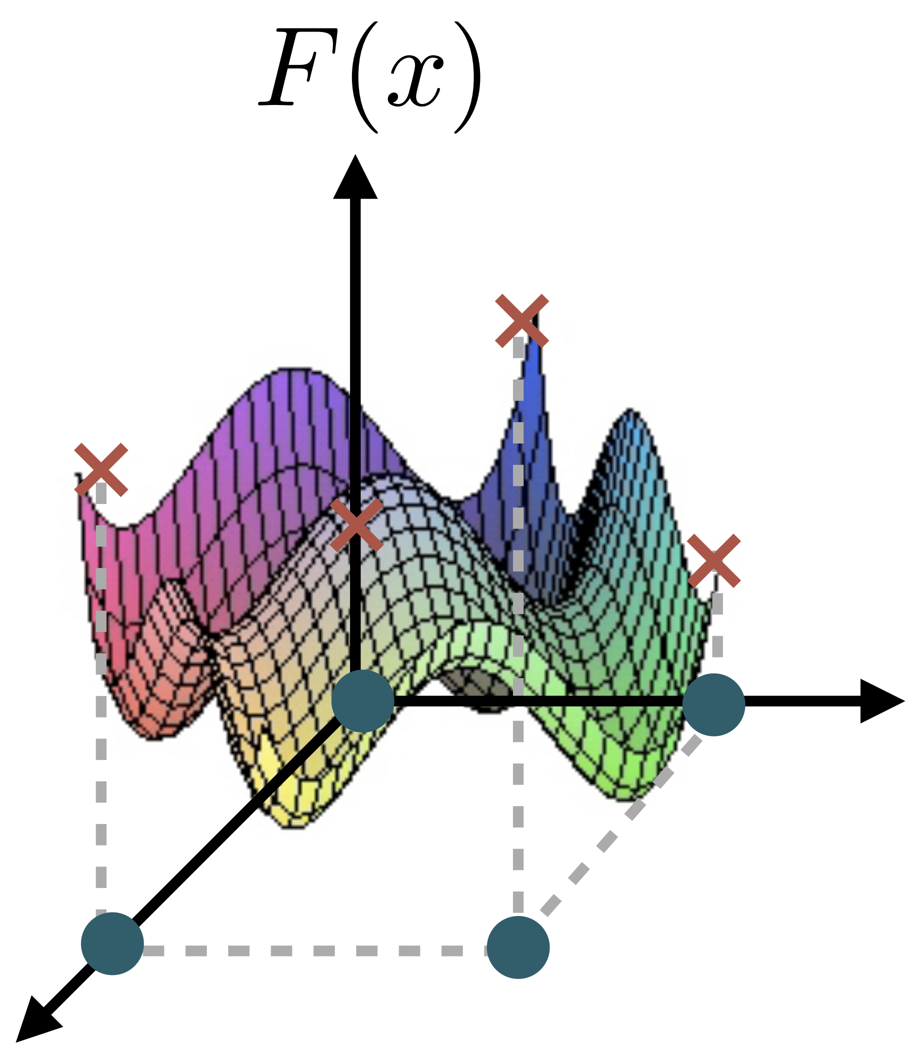
Multilinear Extension

$f : 2^V \to \mathbb{R}$ ➡ $F : [0,1]^m \to \mathbb{R}$

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

$f(S)$ ➡

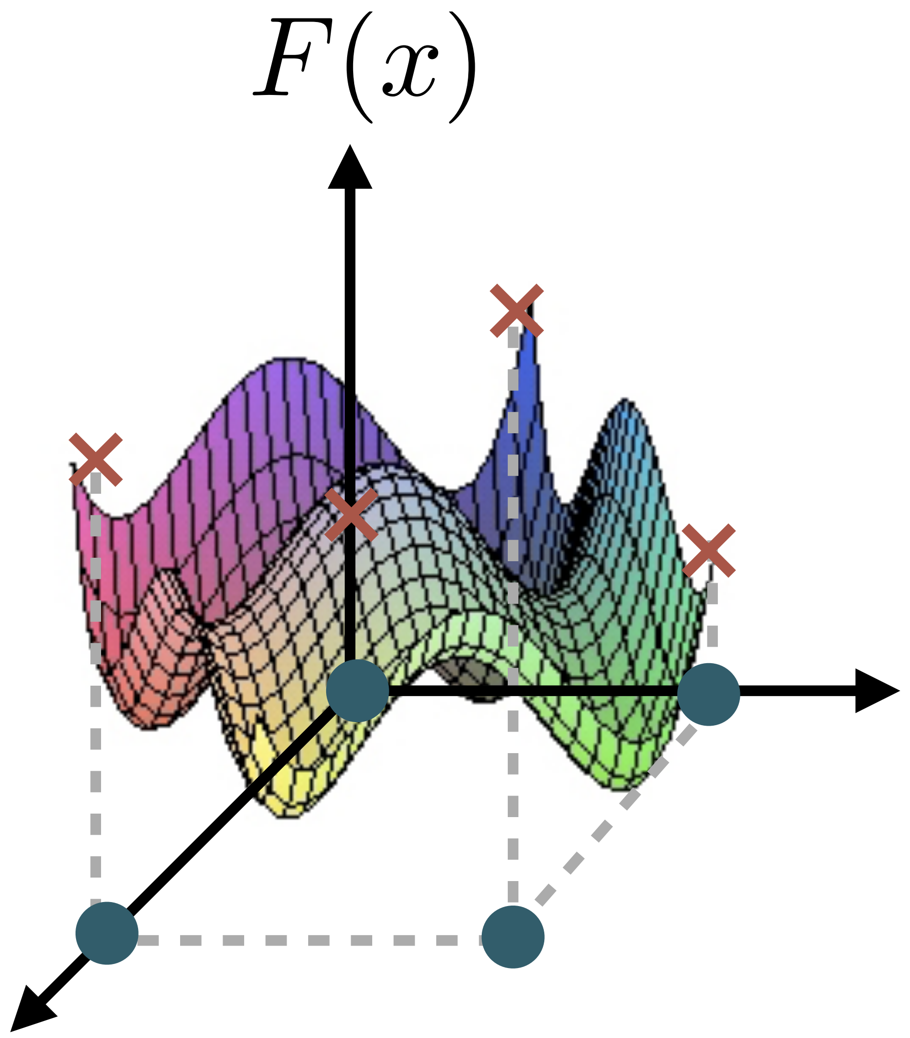
$S \subseteq \{1, 2, \cdots, m\}$



Multilinear Extension

$f : 2^V \to \mathbb{R}$ ➡ $F : [0,1]^m \to \mathbb{R}$

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



$f(S)$ ➡

$S \subseteq \{1, 2, \cdots, m\}$

$x = (x_1, x_2 \cdots, x_m) \in [0,1]^m$

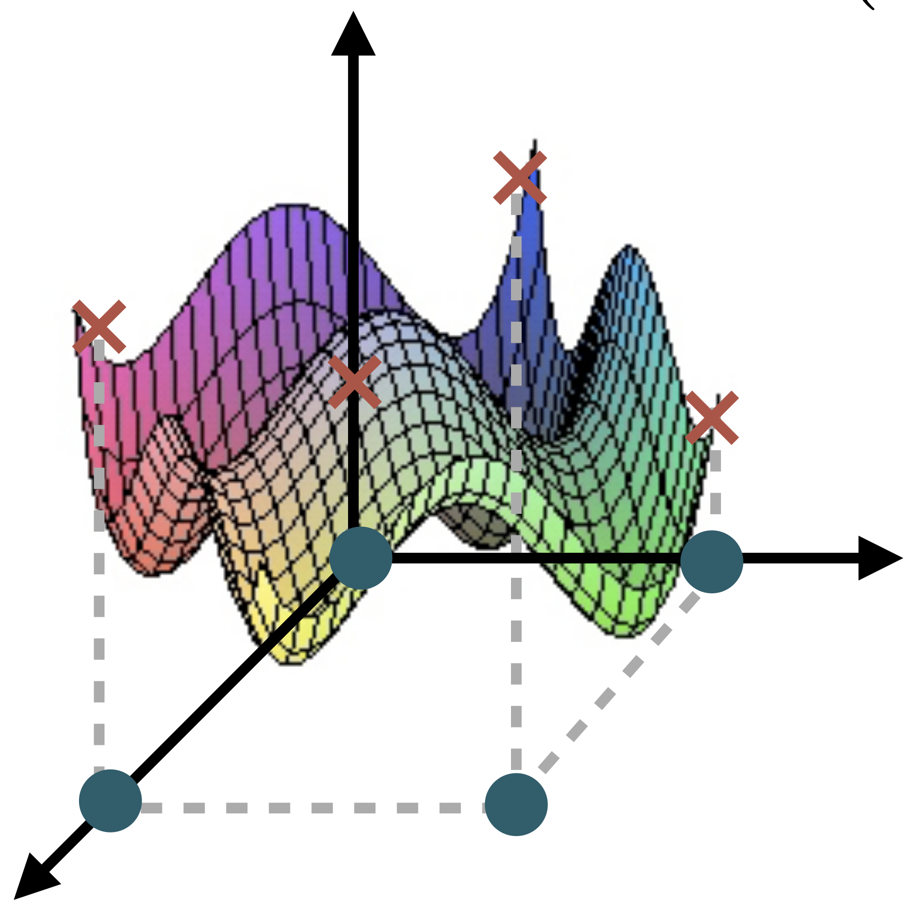
$$F(x) = \sum_{S \in V} f(S) \prod_{a \in S} x_a \prod_{b \notin S} (1 - x_b) = \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$

Multilinear Extension

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

$f(S)$

$S \subseteq \{1, \cdots, n\}$



$$\max_{|S| \leq k} f(S) \quad = \quad \max_{x \in \mathcal{C}} F(x)$$

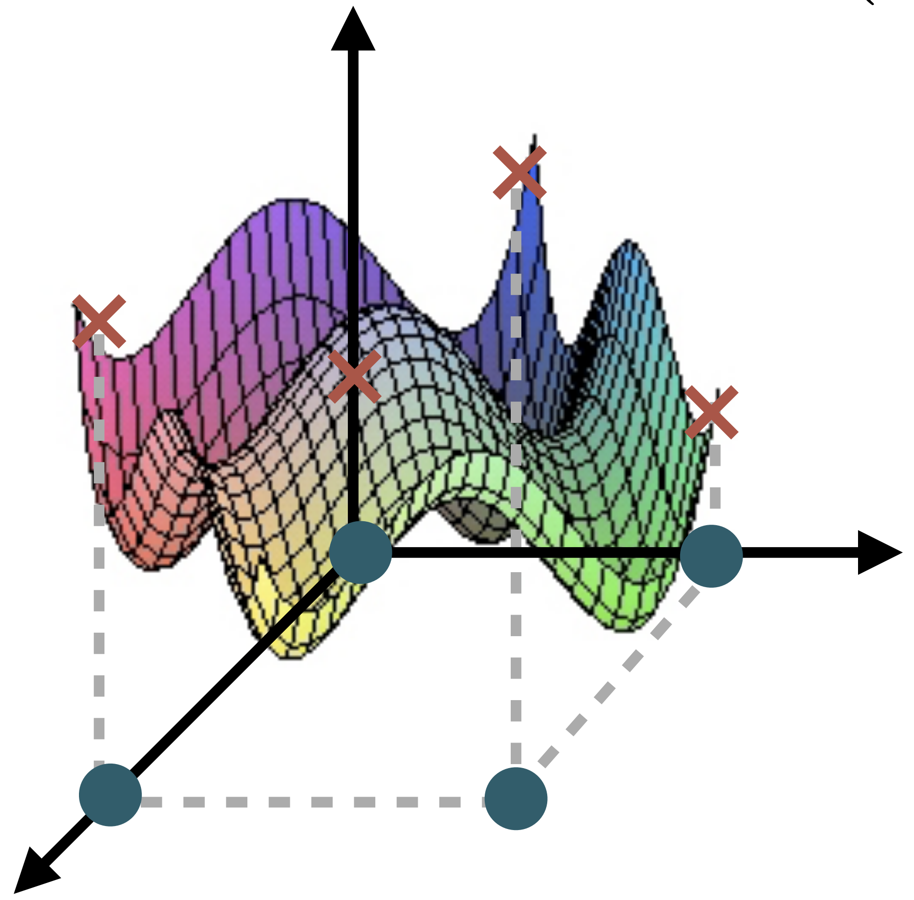
[Calinescu, Chekuri, Vondrak 2011]

Multilinear Extension

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

$f(S)$

$S \subseteq \{1, \cdots, n\}$



$$\max_{|S| \leq k} f(S) \quad = \quad \max_{x \in \mathcal{C}} F(x)$$

[Calinescu, Chekuri, Vondrak 2011]

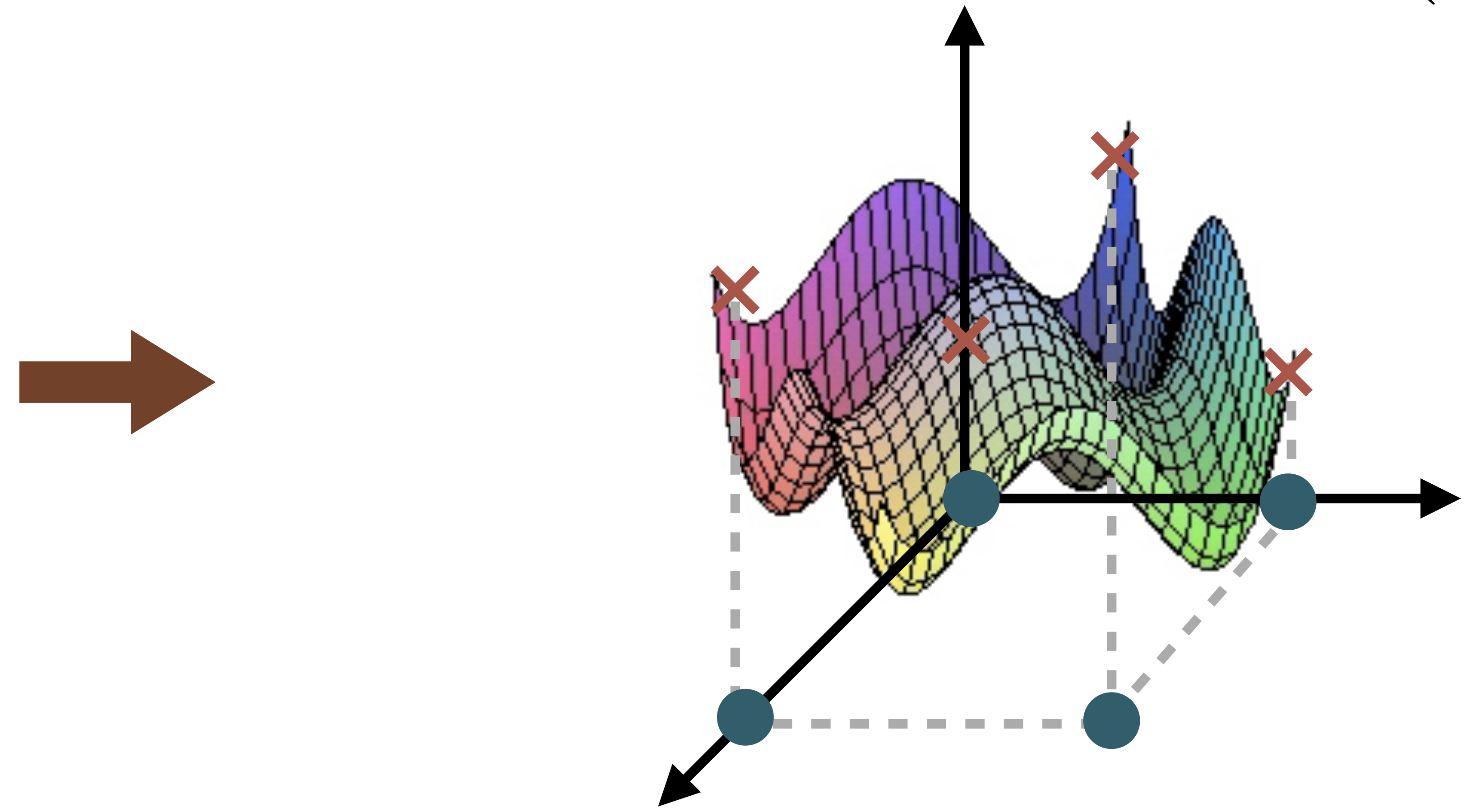
Greedy algorithm provides a (1 - 1/e)-OPT solution

Multilinear Extension

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

$$f(S)$$

$$S \subseteq \{1, \cdots, n\}$$



$$\max_{|S| \leq k} f(S) \quad = \quad \max_{x \in \mathcal{C}} F(x)$$

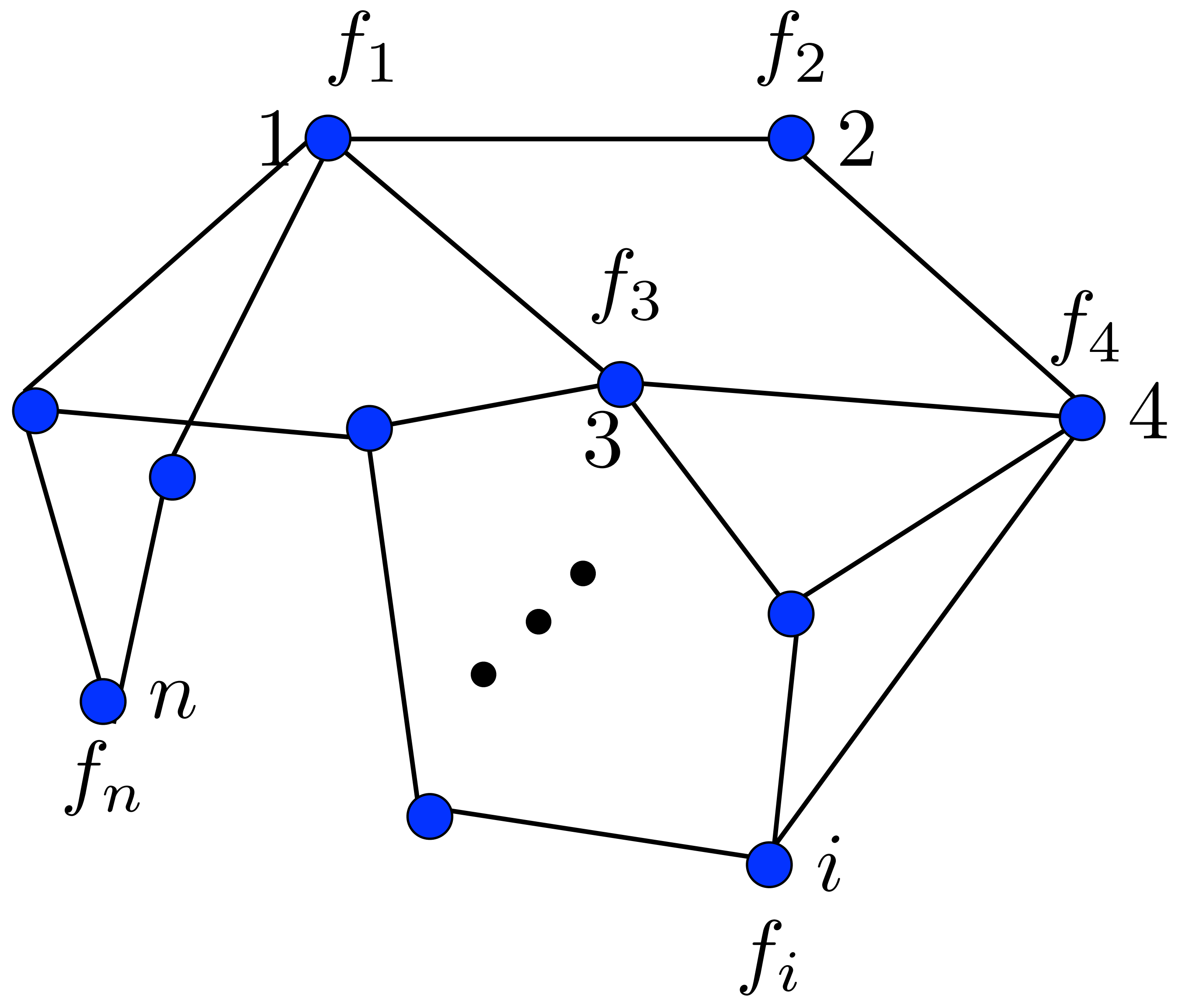
[Calinescu, Chekuri, Vondrak 2011]

Greedy algorithm provides a (1 - 1/e)-OPT solution

Continuous Greedy algorithm provides a (1 - 1/e)-OPT solution

Decentralized Submodular Maximization

Replace by the multilinear extension

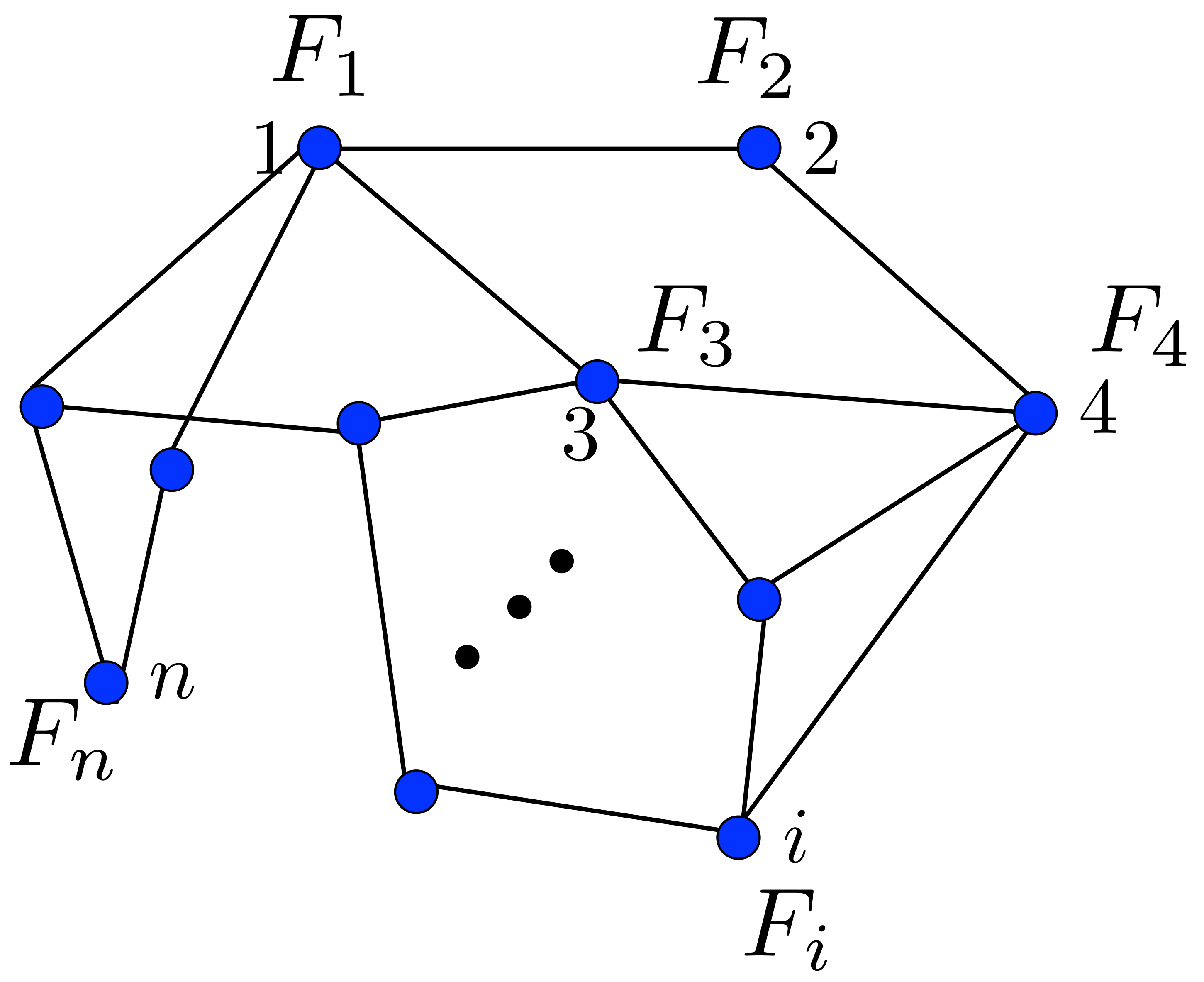


$G = (N, E)$

- Goal: $\underset{|S| \leq k}{\text{maximize}} \quad \frac{1}{n} \sum_{i=1}^{n} f_i(S)$

Decentralized Submodular Maximization

Replace by the multilinear extension

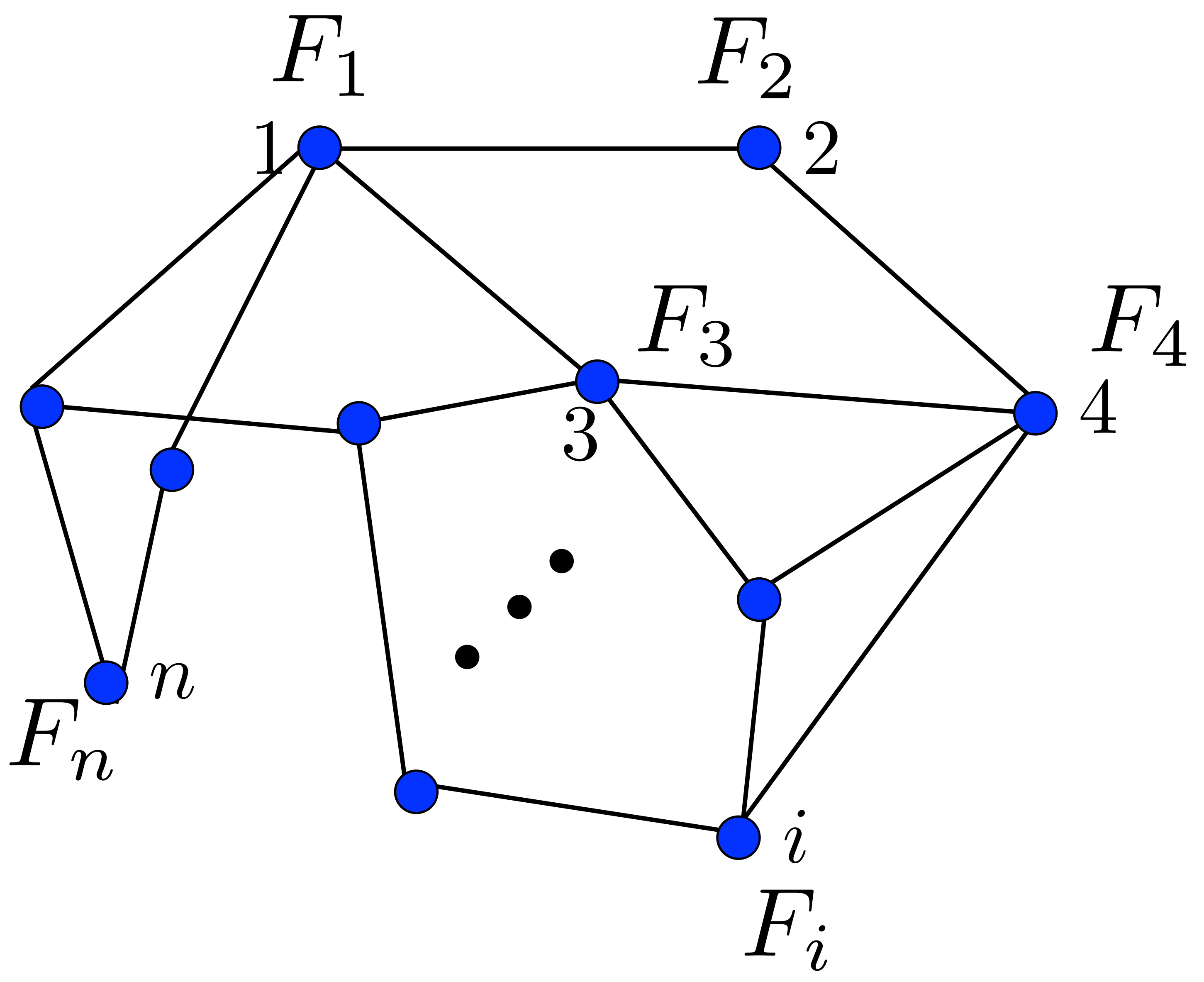


$G = (N, E)$

- Goal: $\underset{|S| \leq k}{\text{maximize}} \quad \frac{1}{n} \sum_{i=1}^{n} f_i(S)$

Decentralized Submodular Maximization

Replace by the multilinear extension

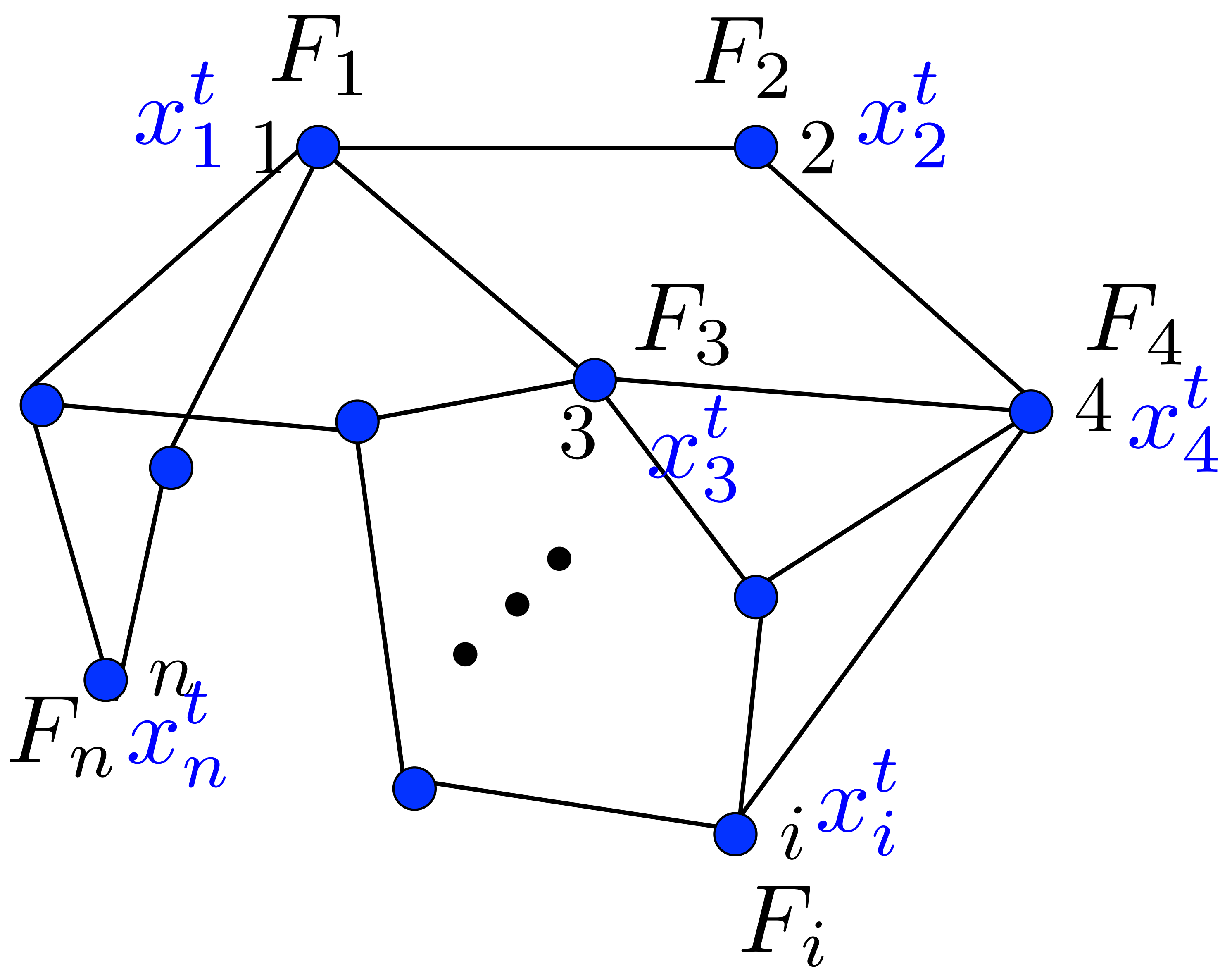


$G = (N, E)$

- Goal: $\underset{x \in \mathcal{C}}{\text{maximize}} \ \frac{1}{n} \sum_{i=1}^{n} F_i(x)$

Decentralized Submodular Maximization

Replace by the multilinear extension

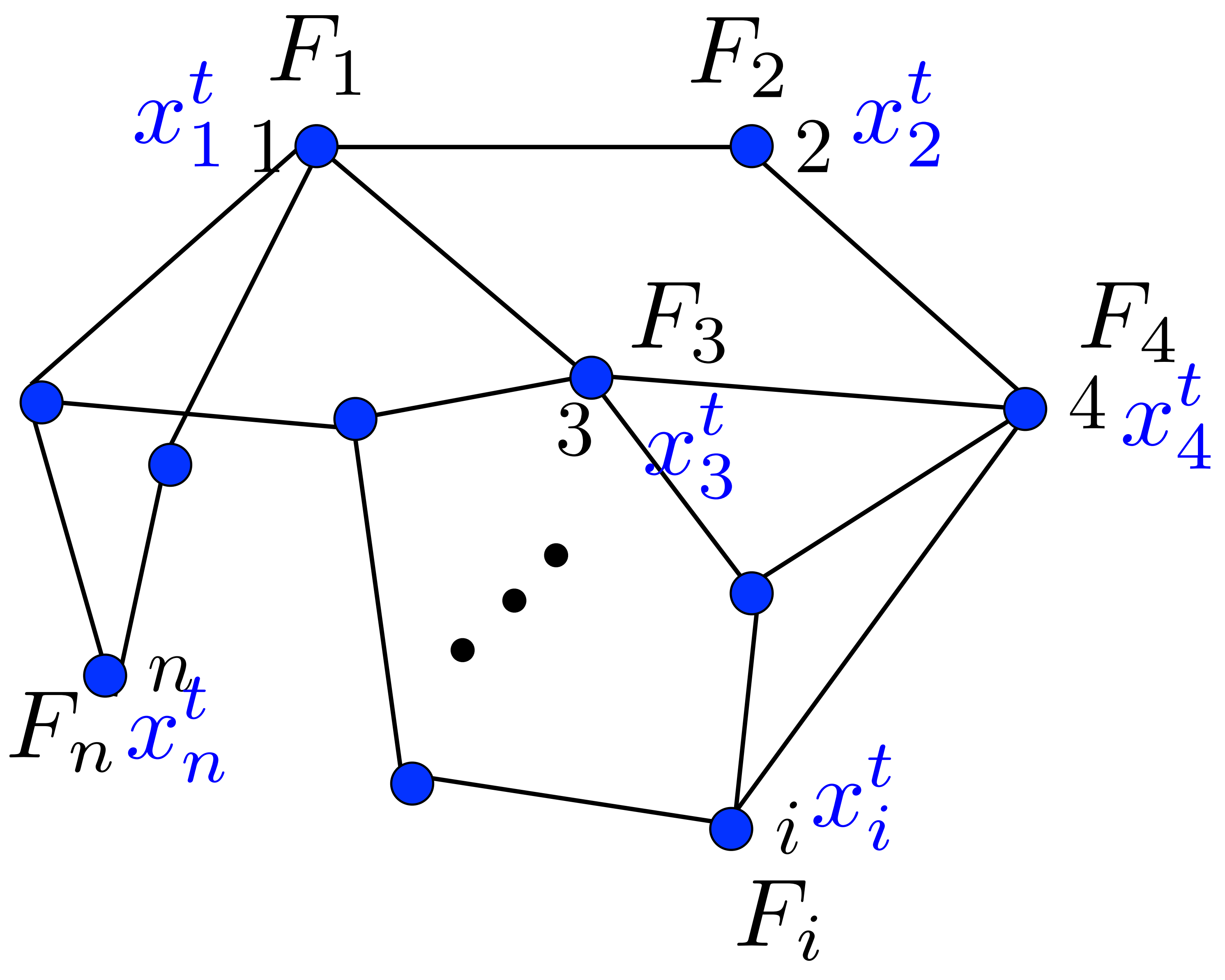


$G = (N, E)$

- Goal: $\underset{x \in \mathcal{C}}{\text{maximize}} \; \frac{1}{n} \sum_{i=1}^{n} F_i(x)$

Decentralized Submodular Maximization

Replace by the multilinear extension



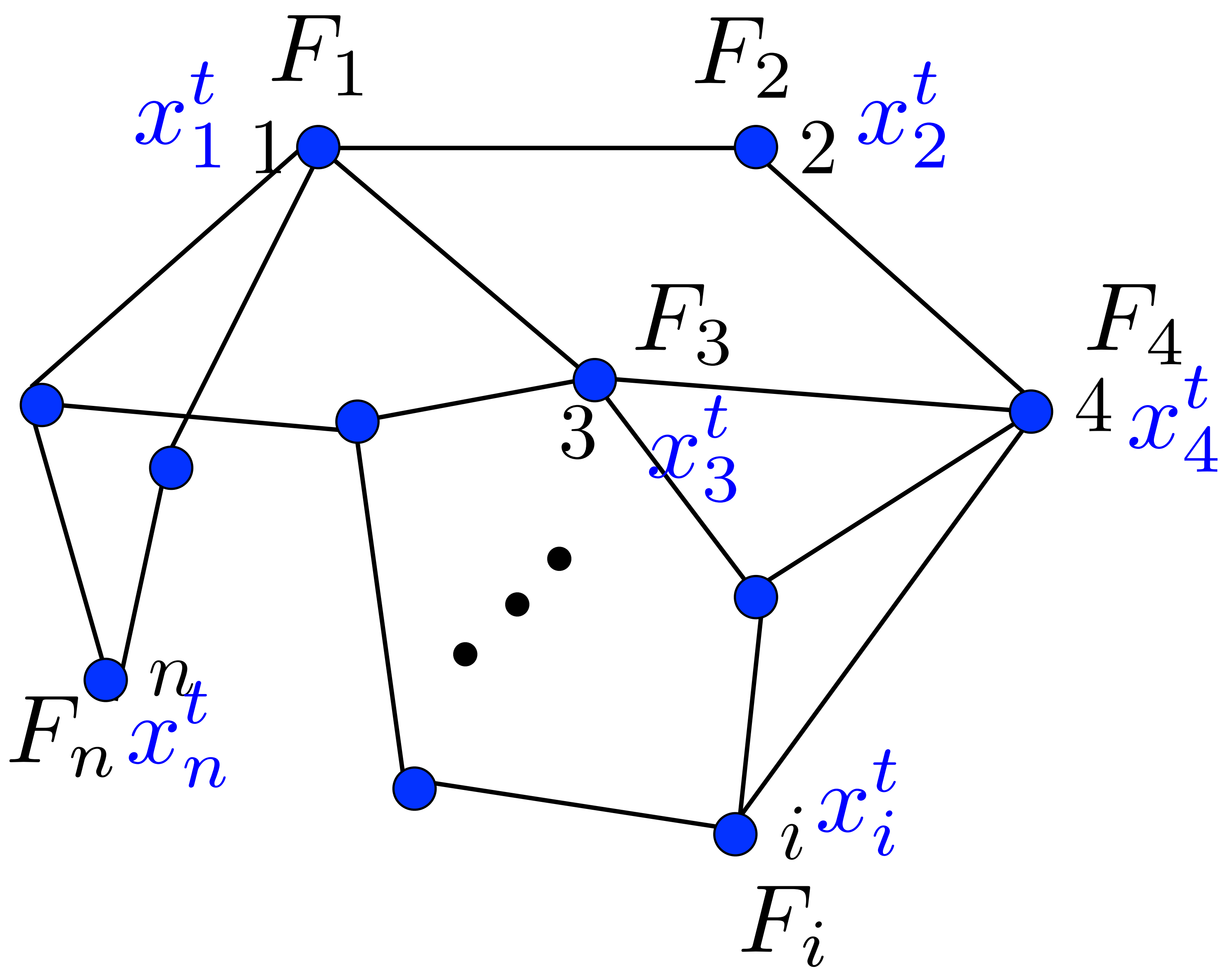
$G = (N, E)$

- **Goal**: $\underset{x \in \mathcal{C}}{\text{maximize}} \ \frac{1}{n} \sum_{i=1}^{n} F_i(x)$

- **Algorithm**: $x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t + \eta_t \nabla F_i(x_i^t)$

Decentralized Submodular Maximization

Replace by the multilinear extension



$G = (N, E)$

- **Goal**: $\underset{x \in \mathcal{C}}{\text{maximize}} \ \frac{1}{n} \sum_{i=1}^{n} F_i(x)$

- **Algorithm**: $x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t + \eta_t \nabla F_i(x_i^t)$

Does not work as F_i 's are non-convex!

Multilinear Extension

$F : [0,1]^m \to \mathbb{R}$

$$F(x) = \sum_{S \in V} f(S) \prod_{a \in S} x_a \prod_{b \notin S} (1 - x_b) = \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$

Multilinear Extension

$F : [0,1]^m \to \mathbb{R}$

$$F(x) = \sum_{S \in V} f(S) \prod_{a \in S} x_a \prod_{b \notin S} (1 - x_b) = \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$

F is non-convex but: $\dfrac{\partial^2 F(x)}{\partial x_i \partial x_j} \leq 0$ (all the elements of the Hessian are non-positive)

Multilinear Extension

$F : [0,1]^m \to \mathbb{R}$

$$F(x) = \sum_{S \in V} f(S) \prod_{a \in S} x_a \prod_{b \notin S} (1 - x_b) = \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$

F is non-convex but: $\dfrac{\partial^2 F(x)}{\partial x_i \partial x_j} \leq 0$ (all the elements of the Hessian are non-positive)

Continuous Greedy Algorithm:

maximize $F(x)$
$x \in \mathcal{C}$

Multilinear Extension

$F : [0,1]^m \to \mathbb{R}$

$$F(x) = \sum_{S \in V} f(S) \prod_{a \in S} x_a \prod_{b \notin S} (1 - x_b) = \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$

F is non-convex but: $\dfrac{\partial^2 F(x)}{\partial x_i \partial x_j} \leq 0$ (all the elements of the Hessian are non-positive)

Continuous Greedy Algorithm:

maximize $F(x)$
$x \in \mathcal{C}$

$$x_{t+1} = x_t + \frac{1}{T} v_t$$

Multilinear Extension

$F : [0,1]^m \to \mathbb{R}$

$$F(x) = \sum_{S \in V} f(S) \prod_{a \in S} x_a \prod_{b \notin S} (1 - x_b) = \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$

F is non-convex but: $\dfrac{\partial^2 F(x)}{\partial x_i \partial x_j} \leq 0$ (all the elements of the Hessian are non-positive)

Continuous Greedy Algorithm:

maximize $F(x)$
$x \in \mathcal{C}$

$$x_{t+1} = x_t + \frac{1}{T} v_t$$

$$v_t = \arg\max_{v \in \mathcal{C}} \langle \nabla F(x_t), v \rangle$$

Multilinear Extension

$F : [0,1]^m \to \mathbb{R}$

$$F(x) = \sum_{S \in V} f(S) \prod_{a \in S} x_a \prod_{b \notin S} (1 - x_b) = \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$

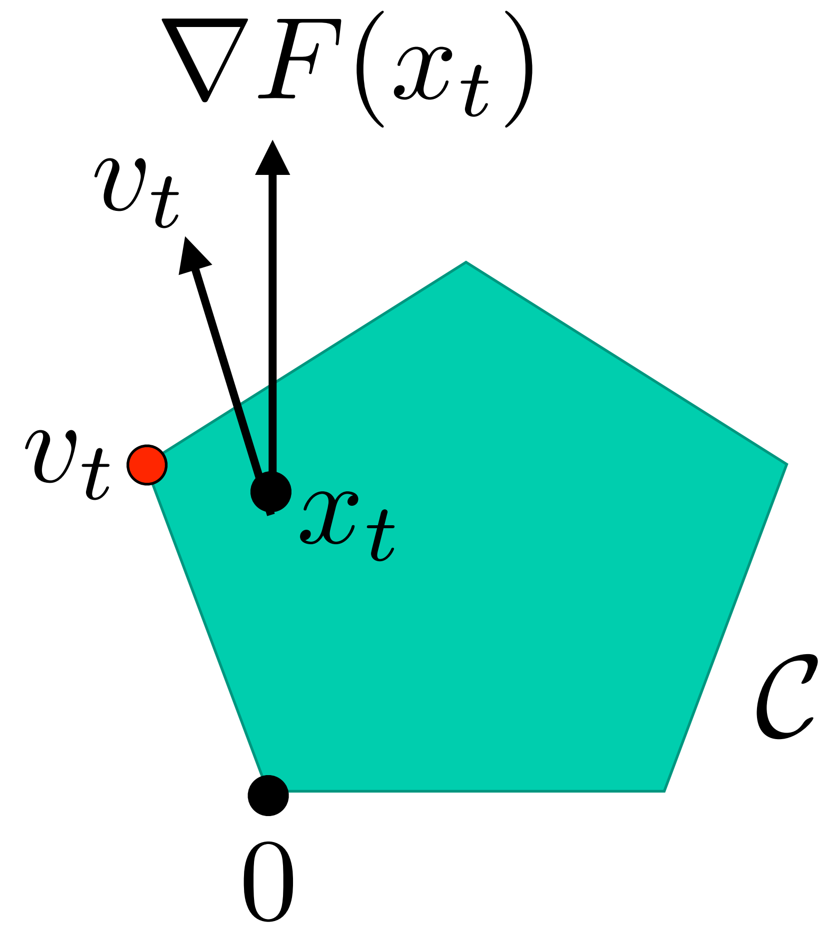
F is non-convex but: $\frac{\partial^2 F(x)}{\partial x_i \partial x_j} \leq 0$ (all the elements of the Hessian are non-positive)

Continuous Greedy Algorithm:

maximize $F(x)$
$x \in \mathcal{C}$

$$x_{t+1} = x_t + \frac{1}{T} v_t$$

$$v_t = \arg\max_{v \in \mathcal{C}} \langle \nabla F(x_t), v \rangle$$



Multilinear Extension

$F : [0,1]^m \to \mathbb{R}$

$$F(x) = \sum_{S \in V} f(S) \prod_{a \in S} x_a \prod_{b \notin S} (1 - x_b) = \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)]$$

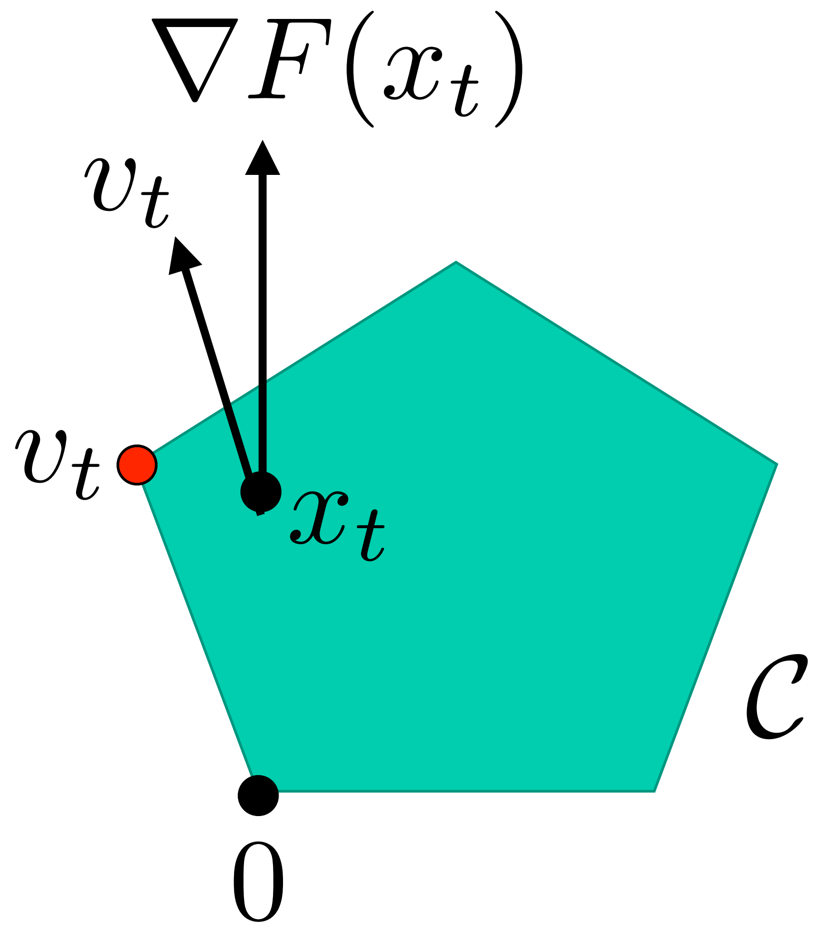
F is non-convex but: $\dfrac{\partial^2 F(x)}{\partial x_i \partial x_j} \leq 0$ (all the elements of the Hessian are non-positive)

Continuous Greedy Algorithm:

maximize $F(x)$
$x \in \mathcal{C}$

$$x_{t+1} = x_t + \frac{1}{T} v_t$$

$$v_t = \arg\max_{v \in \mathcal{C}} \langle \nabla F(x_t), v \rangle$$

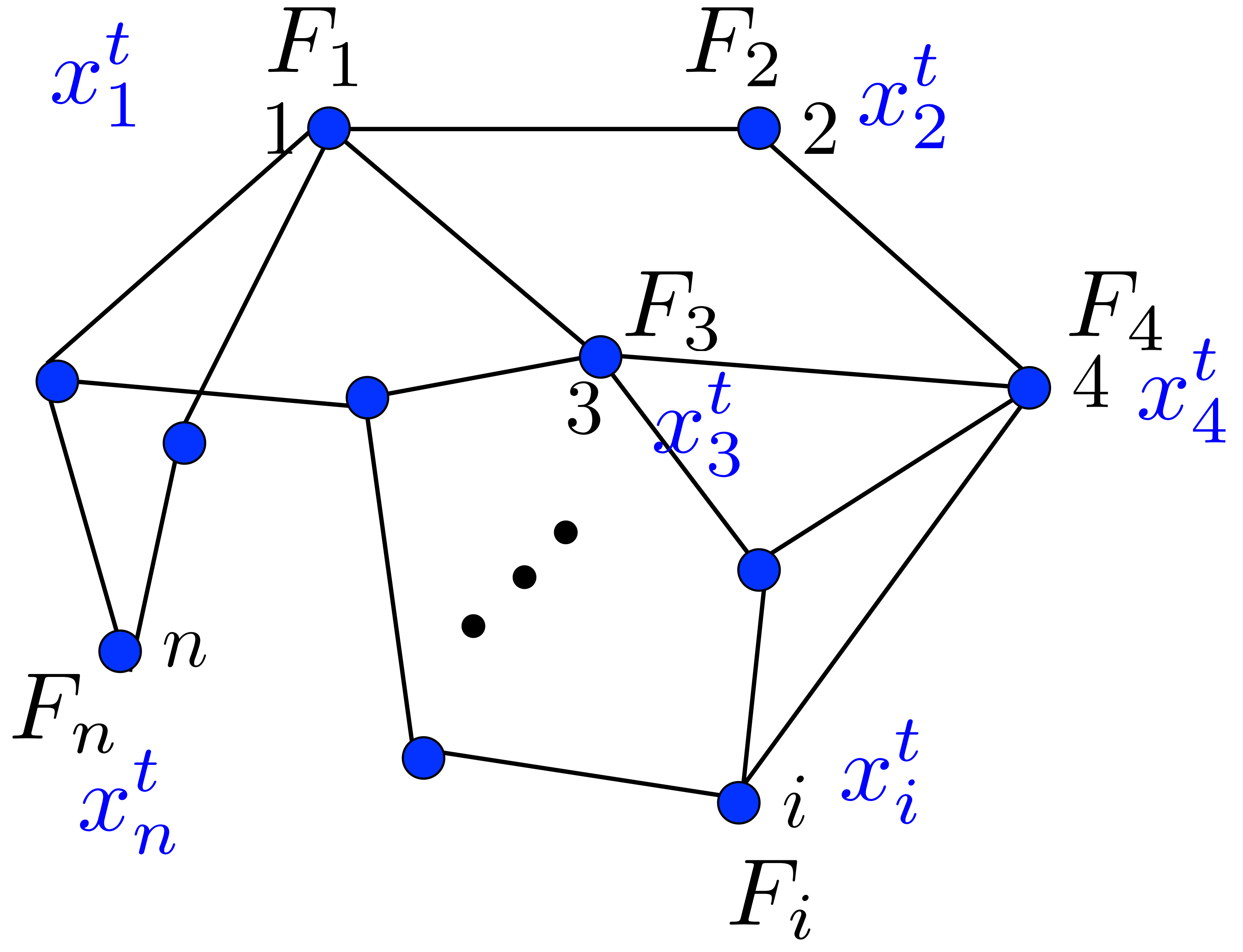


The Continuous Greedy Algorithm provides a tight (1-1/e)-optimum solution

[Calinescu, Chekuri, Vondrak 2011]

Decentralized Continuous Greedy (DCG)

Replace by the multilinear extension



$G = (N, E)$

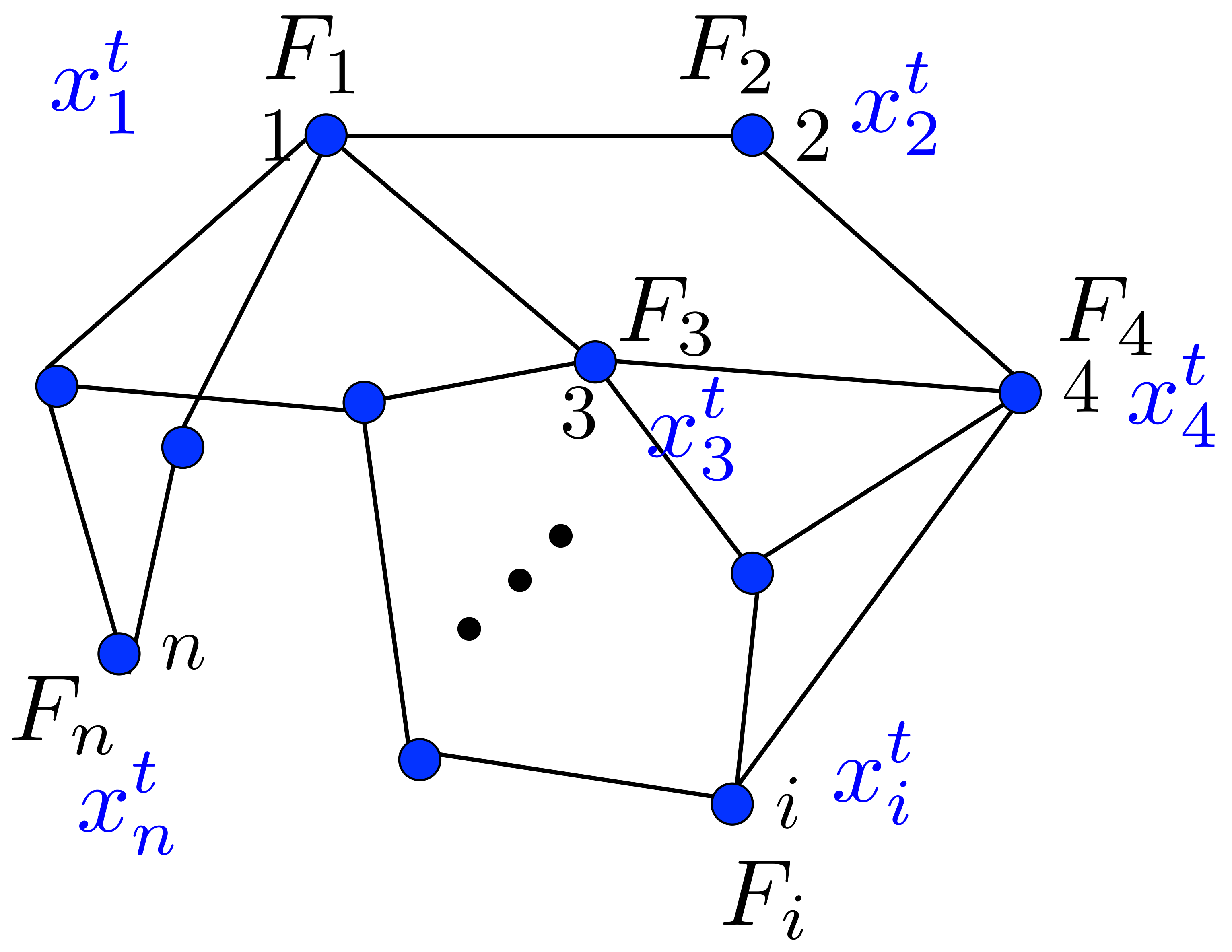
- Goal: $\underset{x \in \mathcal{C}}{\text{maximize}} \ \frac{1}{n}\sum_{i=1}^{n} F_i(x)$

- Algorithm: $x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t + \eta_t \nabla F_i(x_i^t)$

Does not work as F_i ‘s are non-convex!

Decentralized Continuous Greedy (DCG)

Replace by the multilinear extension

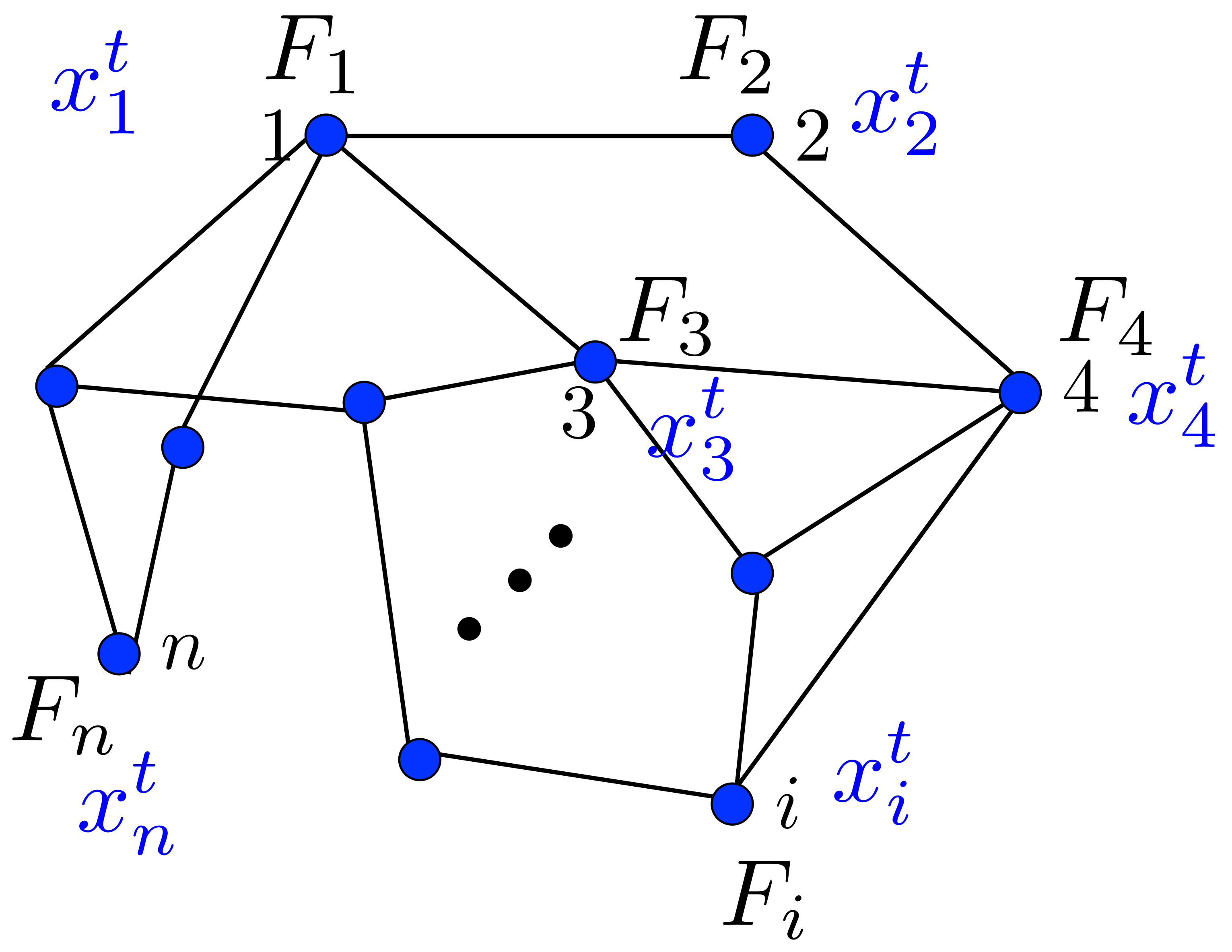


$G = (N, E)$

- Goal: $\underset{x \in \mathcal{C}}{\text{maximize}} \; \frac{1}{n} \sum_{i=1}^{n} F_i(x)$

Decentralized Continuous Greedy (DCG)

Replace by the multilinear extension



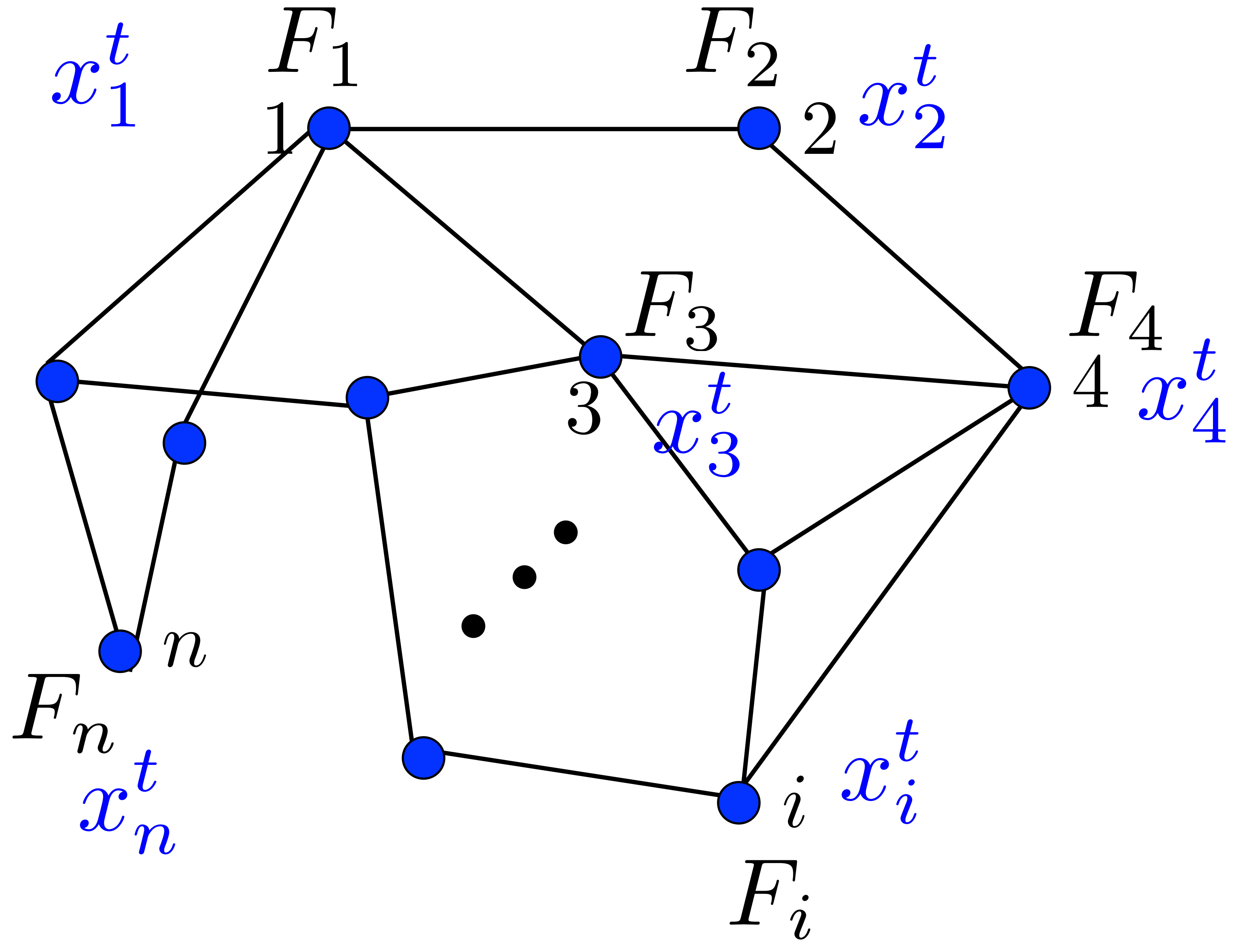
$G = (N, E)$

- **Goal**: $\underset{x \in \mathcal{C}}{\text{maximize}} \ \frac{1}{n} \sum_{i=1}^{n} F_i(x)$

- **Algorithm**: $x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t + \frac{1}{T} v_i^t$

Decentralized Continuous Greedy (DCG)

Replace by the multilinear extension



$G = (N, E)$

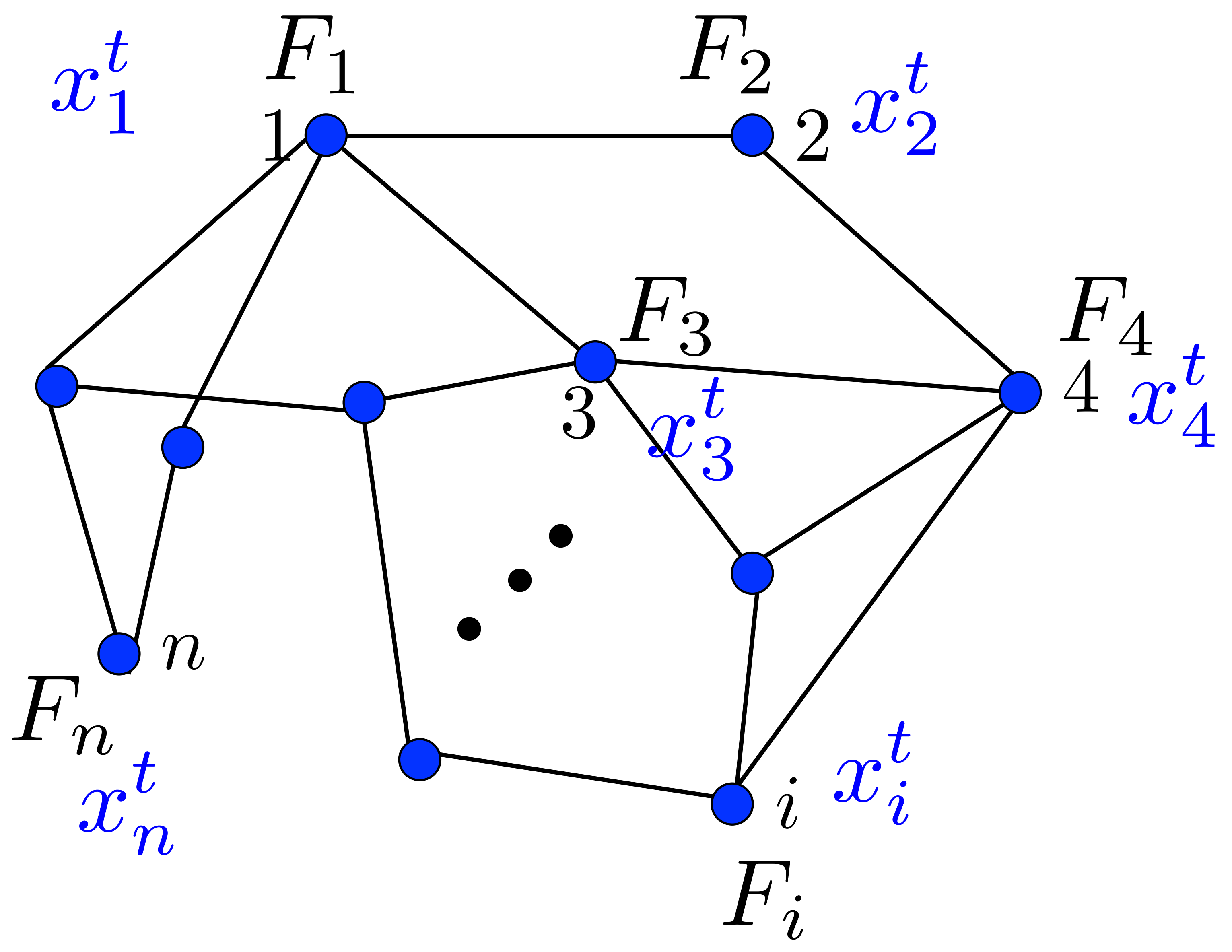
- Goal: $\underset{x \in \mathcal{C}}{\text{maximize}} \; \frac{1}{n} \sum_{i=1}^{n} F_i(x)$

- Algorithm: $x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t + \frac{1}{T} v_i^t$

$$v_i^t = \arg\max_{v \in C} \langle \nabla F(x_i^t), v \rangle$$

Decentralized Continuous Greedy (DCG)

Replace by the multilinear extension



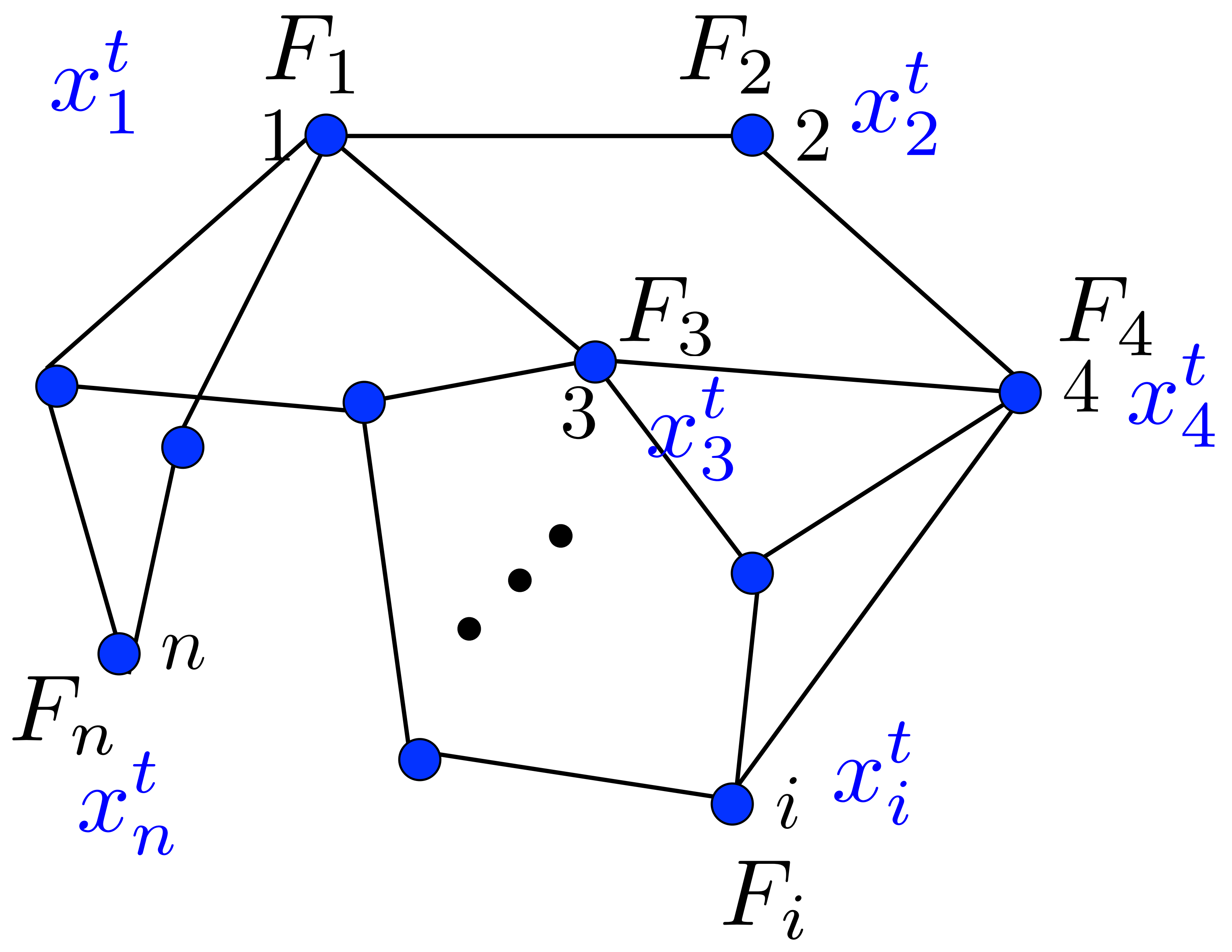
$G = (N, E)$

- Goal: $\underset{x \in \mathcal{C}}{\text{maximize}} \; \frac{1}{n} \sum_{i=1}^{n} F_i(x)$

- Algorithm: $x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t + \frac{1}{T} v_i^t$

Decentralized Continuous Greedy (DCG)

Replace by the multilinear extension



$G = (N, E)$

- Goal: $\text{maximize}_{x \in \mathcal{C}} \; \frac{1}{n} \sum_{i=1}^{n} F_i(x)$

- Algorithm:

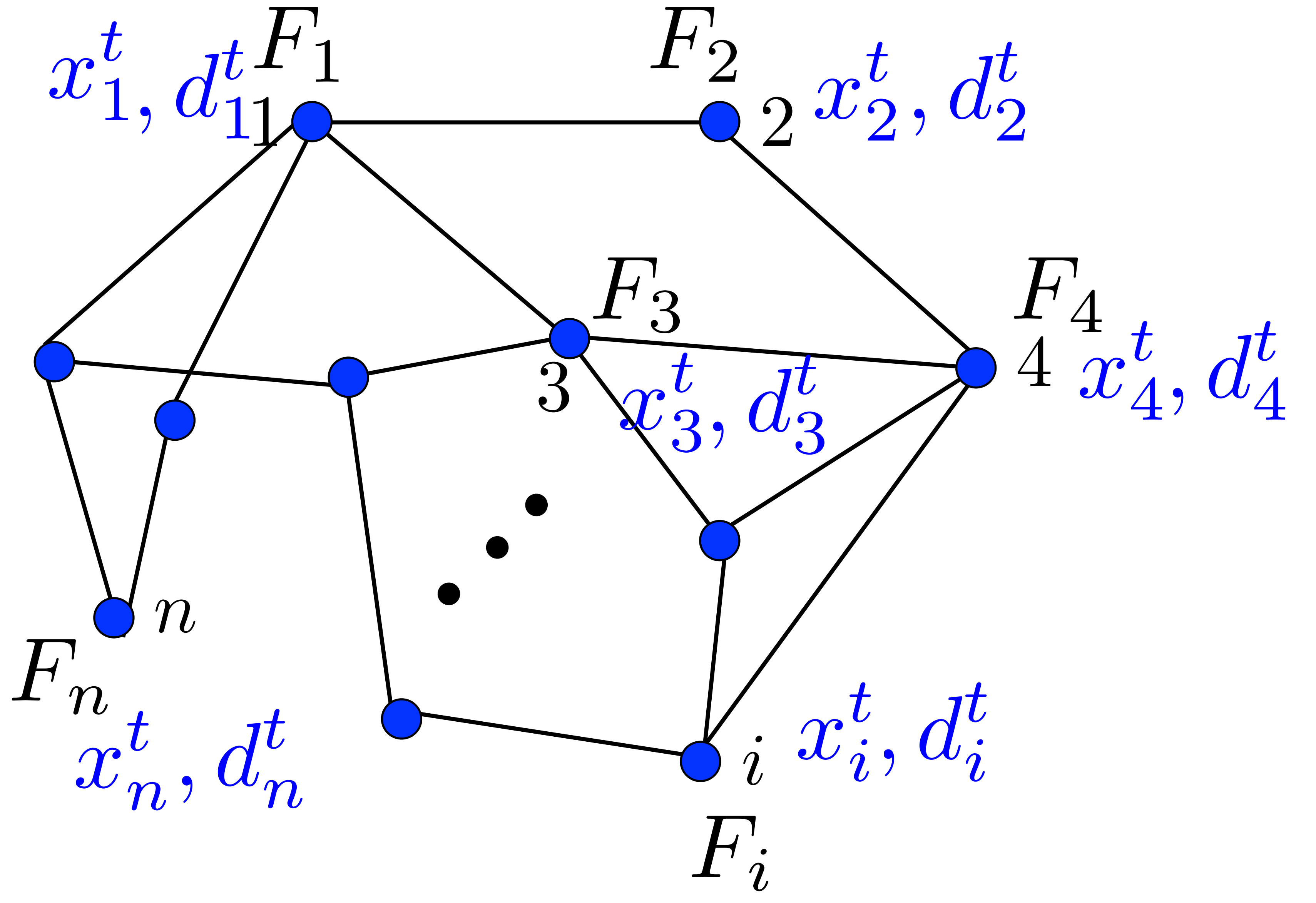
$$x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t + \frac{1}{T} v_i^t$$

$$v_i^t = \arg\max_{v \in C} \langle d_i^t, v \rangle$$

$$d_i^{t+1} = (1 - \alpha) \sum_{j \in N_i} w_{i,j} d_j^t + \alpha \nabla F_i(x_i^t)$$

Decentralized Continuous Greedy (DCG)

Replace by the multilinear extension



$$G = (N, E)$$

- Goal: $$\underset{x \in \mathcal{C}}{\text{maximize}} \ \frac{1}{n}\sum_{i=1}^{n} F_i(x)$$

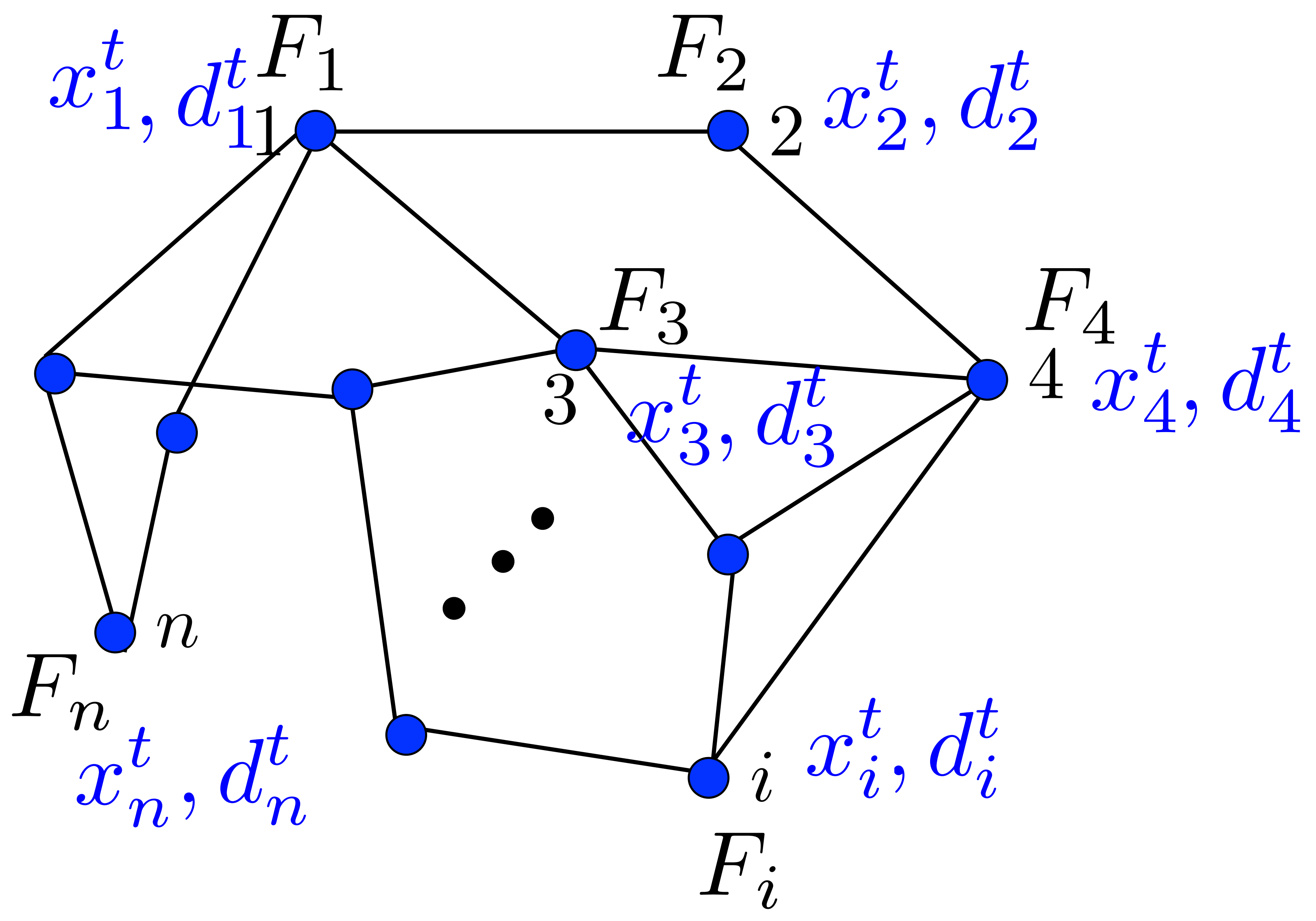
- Algorithm: $$x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t + \frac{1}{T} v_i^t$$

$$v_i^t = \arg\max_{v \in C} \langle d_i^t, v \rangle$$

$$d_i^{t+1} = (1-\alpha) \sum_{j \in N_i} w_{i,j} d_j^t + \alpha \nabla F_i(x_i^t)$$

Decentralized Continuous Greedy (DCG)

Replace by the multilinear extension



$$G = (N, E)$$

- Goal: $\underset{x \in \mathcal{C}}{\text{maximize}} \; \frac{1}{n} \sum_{i=1}^{n} F_i(x)$

consensus on beliefs

- Algorithm: $x_i^{t+1} = \sum_{j \in N_i} w_{i,j} x_j^t + \frac{1}{T} v_i^t$

$$v_i^t = \arg\max_{v \in C} \langle d_i^t, v \rangle$$

$$d_i^{t+1} = (1-\alpha) \sum_{j \in N_i} w_{i,j} d_j^t + \alpha \nabla F_i(x_i^t)$$

consensus on gradients

Decentralized Continuous Greedy (DCG)

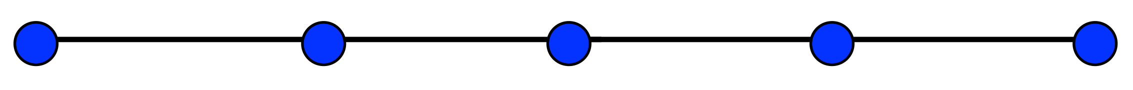
- Theorem: By choosing $\alpha = O(1/\sqrt{T})$, for any node j we have:

$$F(\mathbf{x}_j^T) \geq (1 - 1/e)F(\mathbf{x}^*) - \mathcal{O}\left(\frac{1}{(1-\beta)T^{1/2}}\right)$$

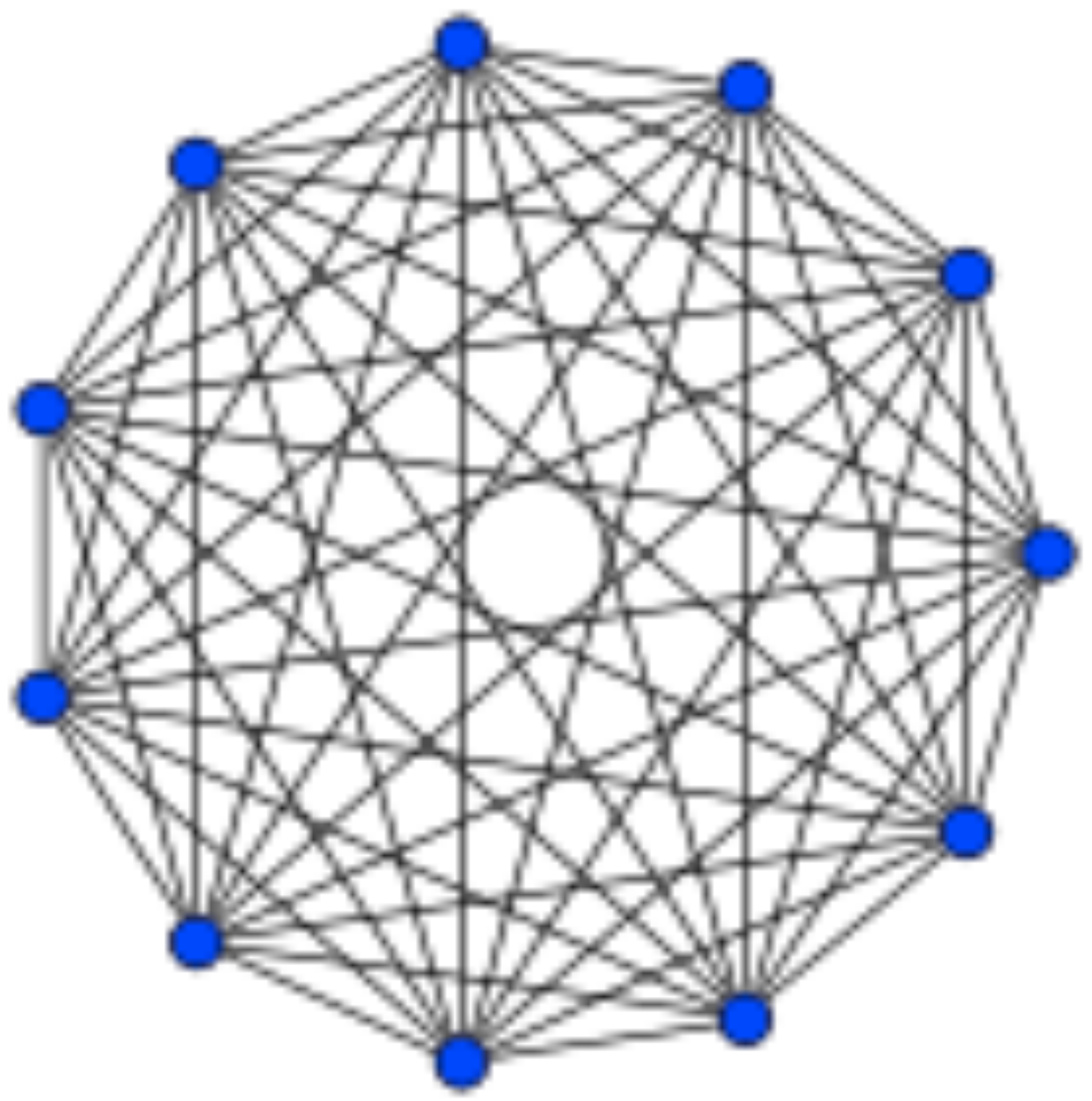
where β is the second largest magnitude of the eigenvalues of the weight matrix W.

- Note: There are many other details (computing gradients, assumptions, choice of the weights, etc)

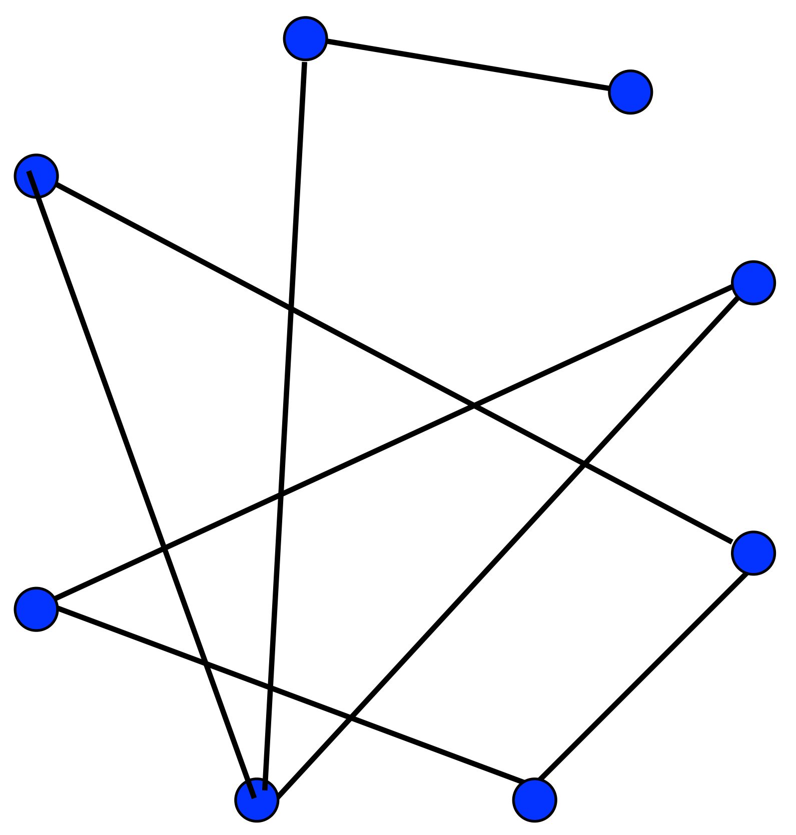
Experiments



Line graph



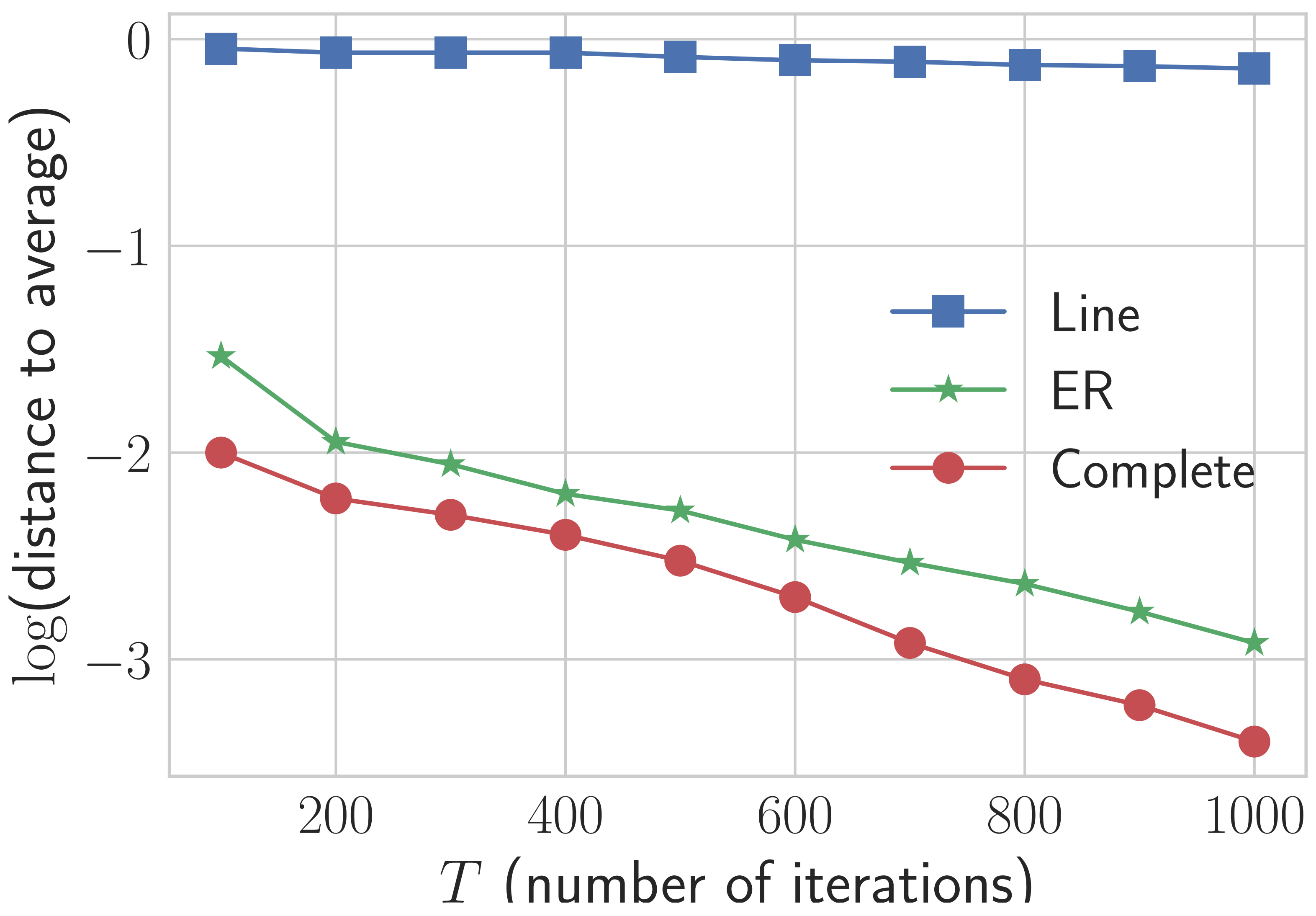
Complete graph

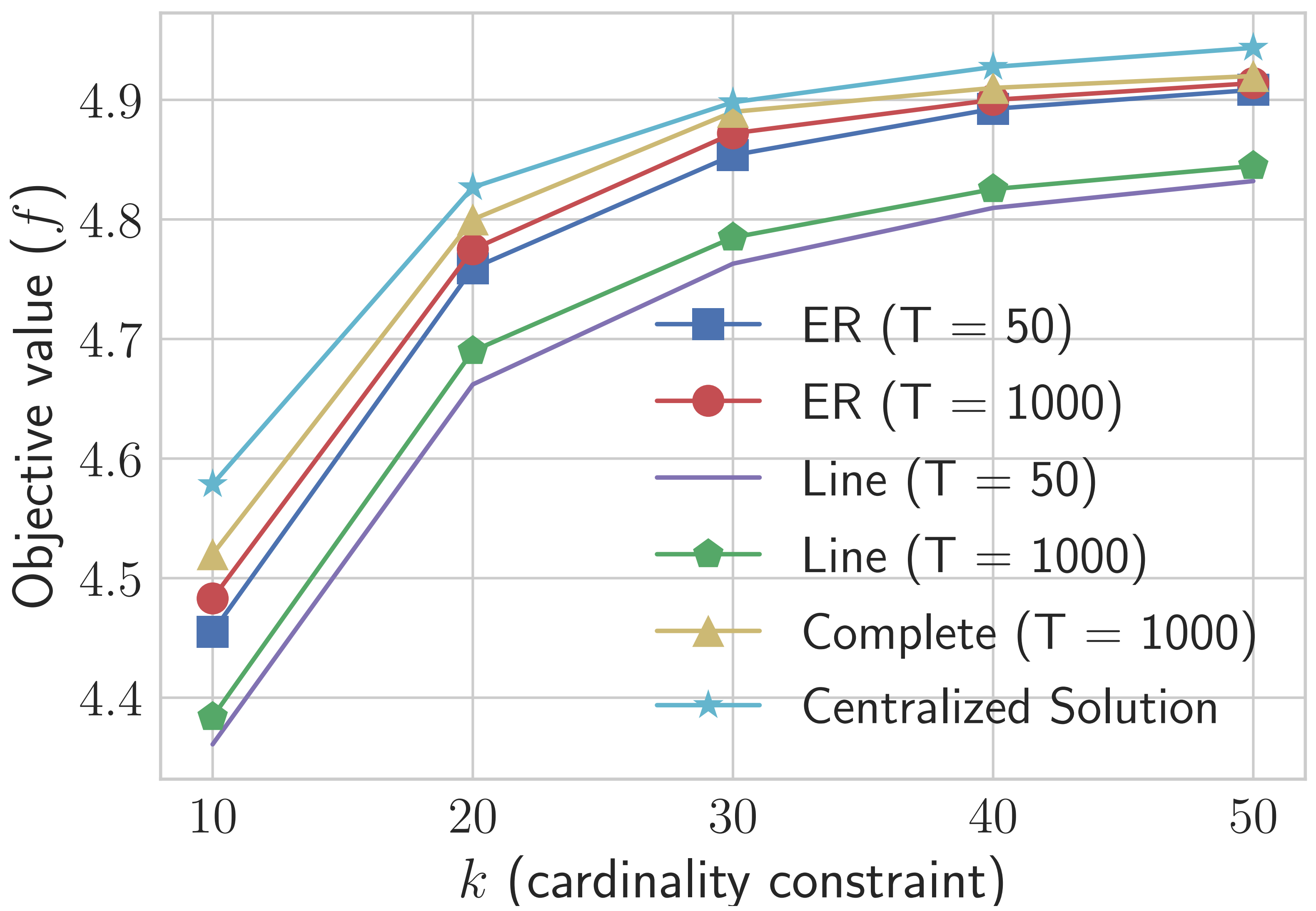


Erdos-Renyi Graph

Experiments

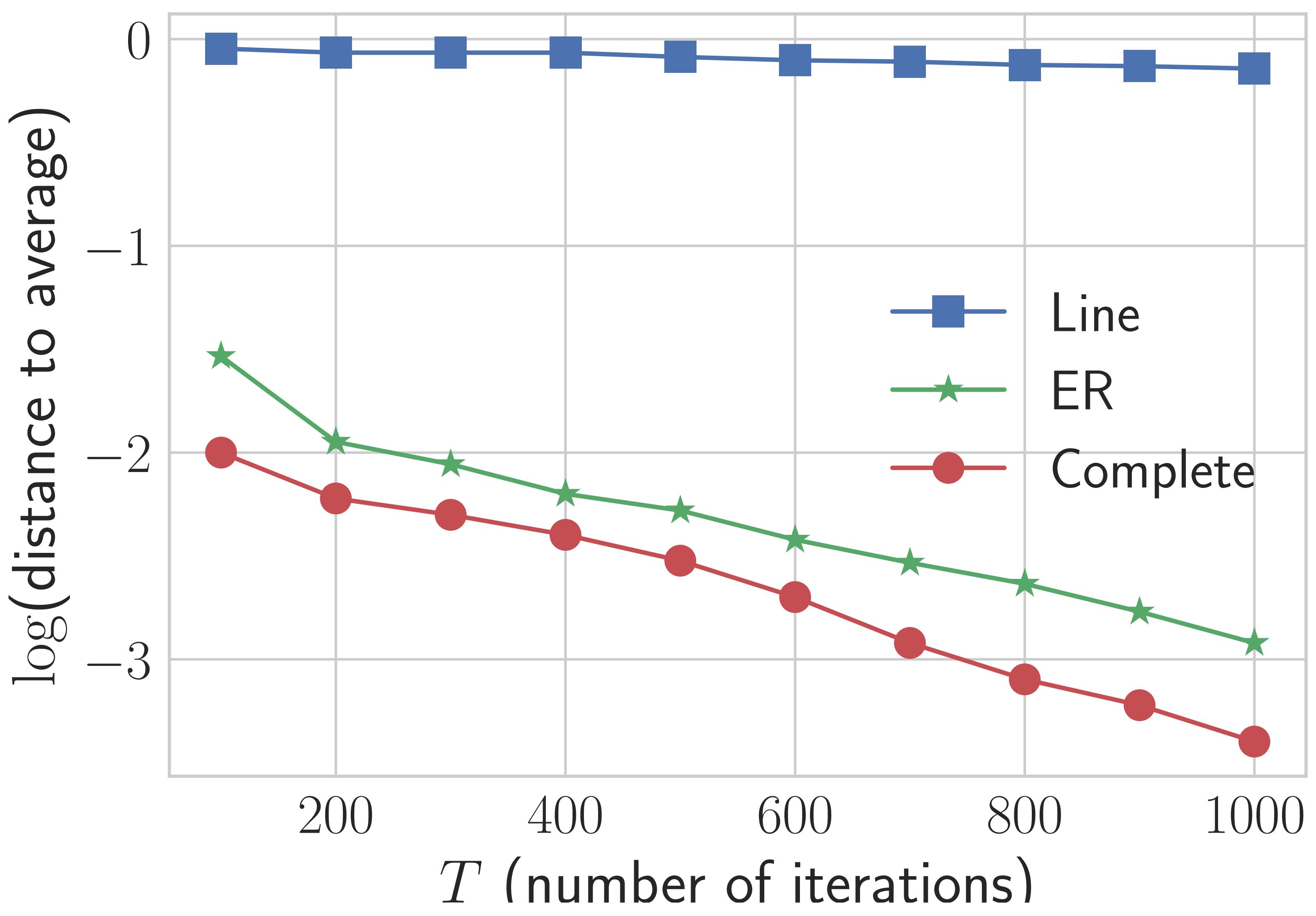
- MovieLens-1M data set, data distributed evenly between 100 nodes (units)
- Task: Find k movies that are most satisfactory
- Three types of networks: Line, Erdos-Renyi, Complete

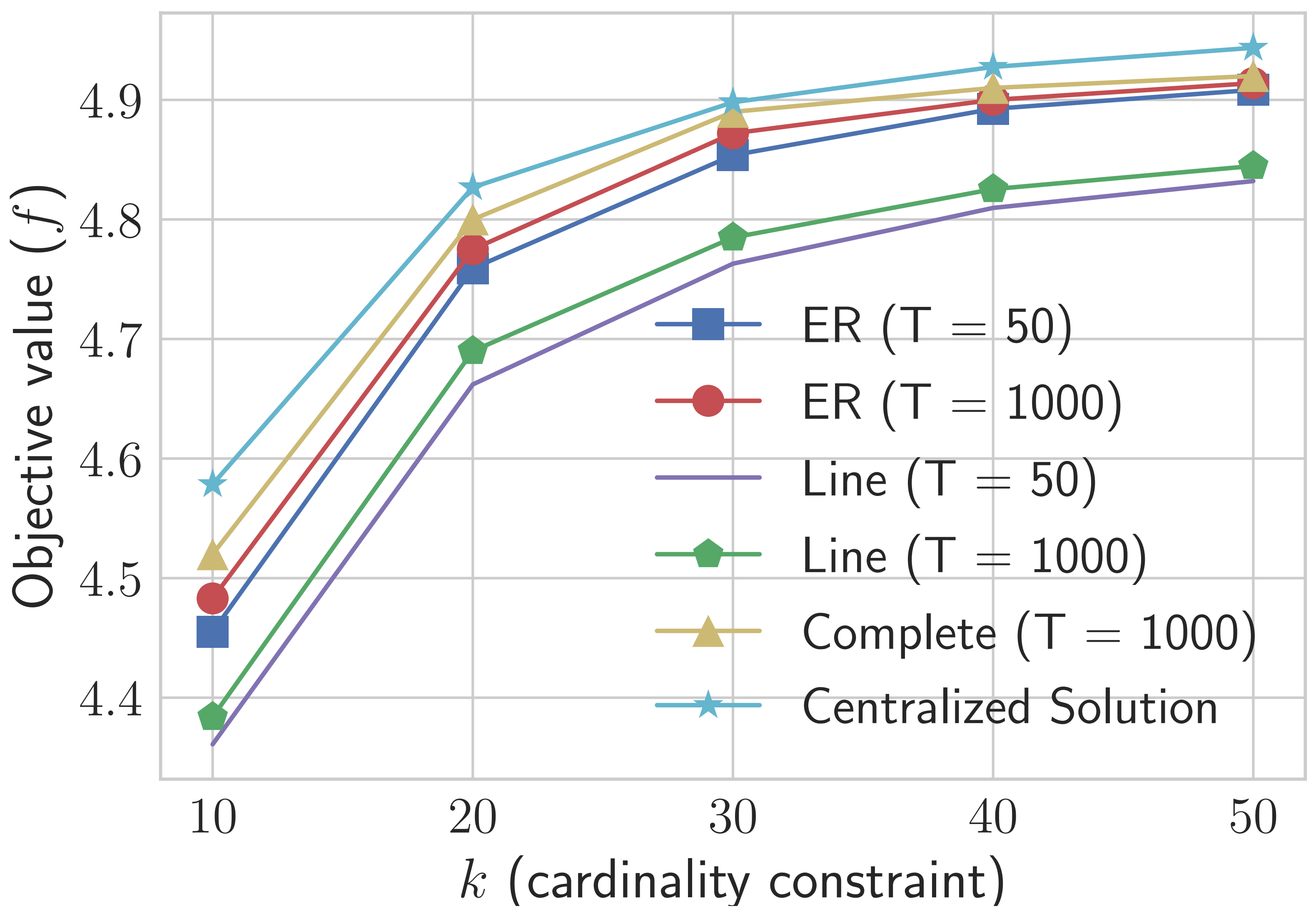




Experiments

- MovieLens-1M data set, data distributed evenly between 100 nodes (units)
- Task: Find k movies that are most satisfactory
- Three types of networks: Line, Erdos-Renyi, Complete





Thank you!