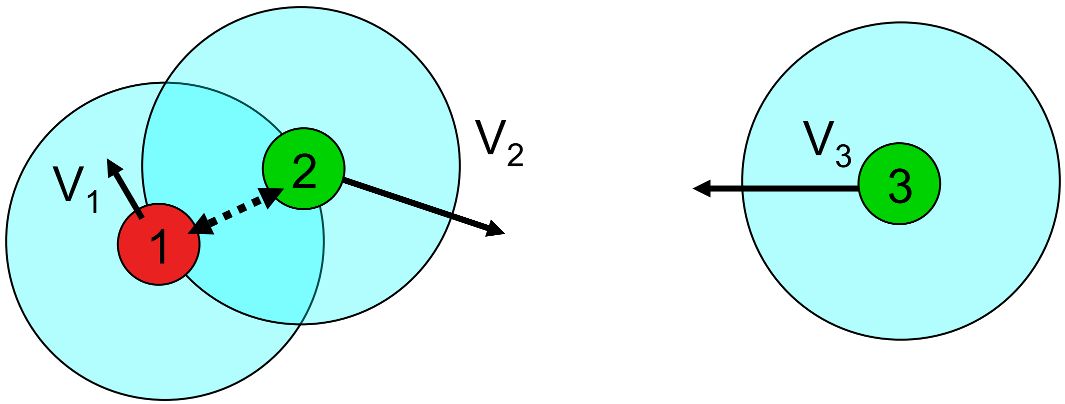
Optimization in Delay Tolerant Networks

Giovanni Neglia

EVALUATION SEMINAR

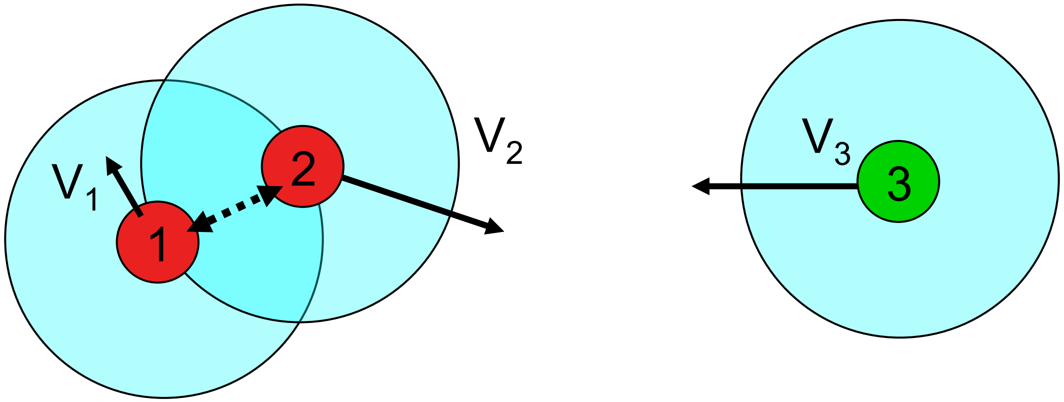
23 March 2012

Delay Tolerant Networks



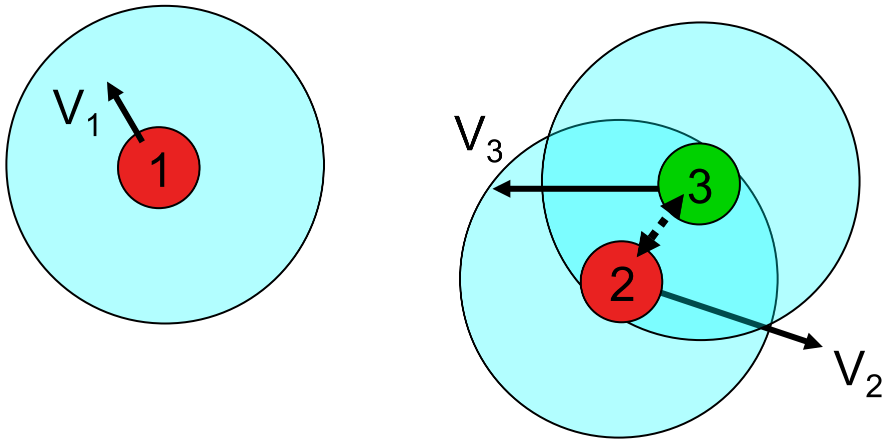
- no path at a given time instant between two nodes
- need to rely on nodes mobility to deliver packets (store, carry and forward)
- incomplete knowledge about future meetings
  - rely on multiple copies

Delay Tolerant Networks



- no path at a given time instant between two nodes
- need to rely on nodes mobility to deliver packets (store, carry and forward)
- incomplete knowledge about future meetings
  - rely on multiple copies

Delay Tolerant Networks



- no path at a given time instant between two nodes
- need to rely on nodes mobility to deliver packets (store, carry and forward)
- incomplete knowledge about future meetings
  - rely on multiple copies

Resources vs Performance Trade-off

- More copies (C) => smaller delivery time (T_d)
  - Different ways to curb the infection
    - Gossip, K-hop, Spray&Wait
- Optimization problems:
  - Determine policy parameters/routing in order to
    - minimize $E[T_d+\gamma C]$
    - minimize $E[T_d]$ given that $C \leq C_{max}$
    - maximize $P(T_d<T_{max})$ given that $C \leq C_{max}$
  - …for different routing schemes

Contributions: Centralized Approaches

The controller knows infection state/topology

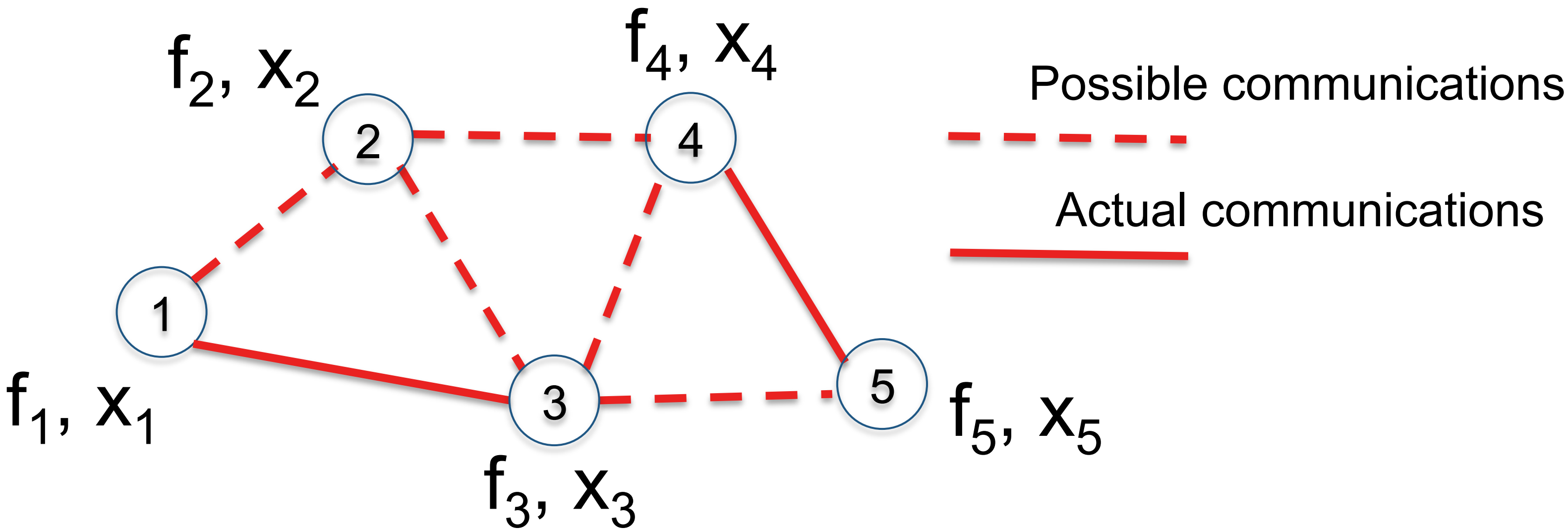
- Markov Decision Processes => threshold policies
  - minimize $E[T_d+\gamma C]$, [Chants06]
  - minimize $E[T_d]$ given $E[C] \leq C_{max}$, (with E. Altman)
    - [Infocom09] controller learns policy at runtime (stochastic approximation)
- Cast as a stochastic shortest path problem
  - maximize $P(T_d < T_{max})$, Frank's problem
    - Efficient heuristic for an urban bus network [IEEETVT12]

Contributions: Decentralized Approaches

Goal: nodes learn their optimal policy under partial information

- Explore the solution space at runtime with genetic algorithm-like techniques (with S. Alouf)
  - EU-project BIONETS (BIOlogically-inspired autonomic NETworks and Services) [ElsevierComNet10]
- Use distributed (sub-)gradient techniques

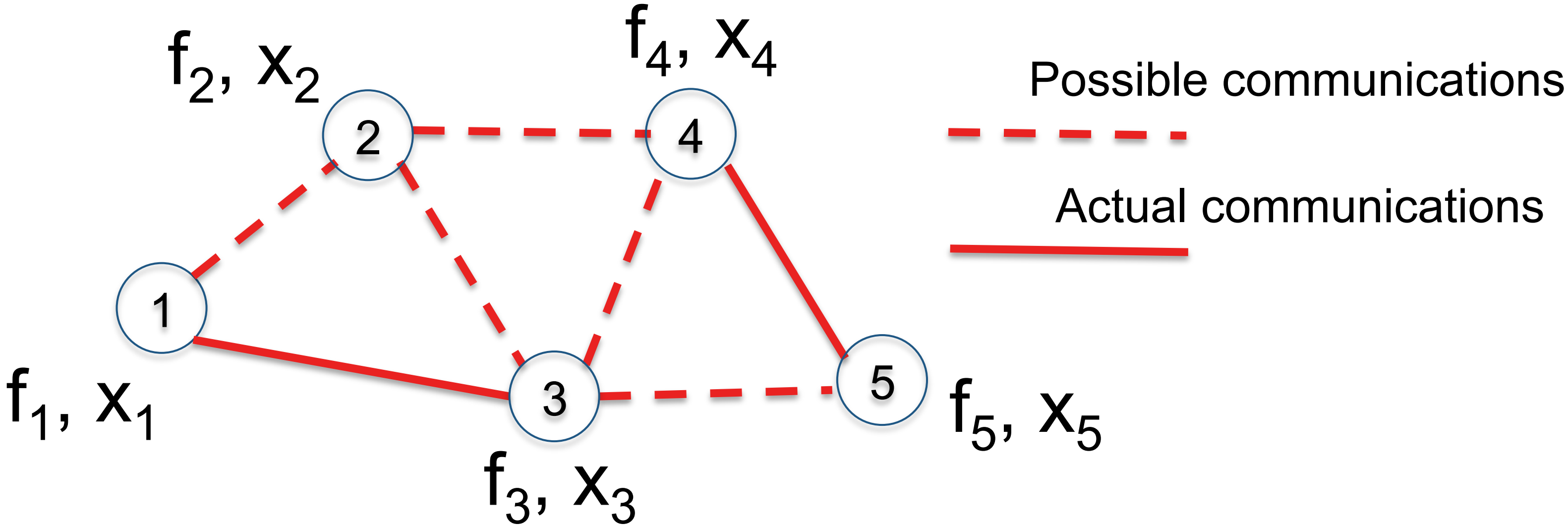
Distributed Subgradients Methods



The goal:

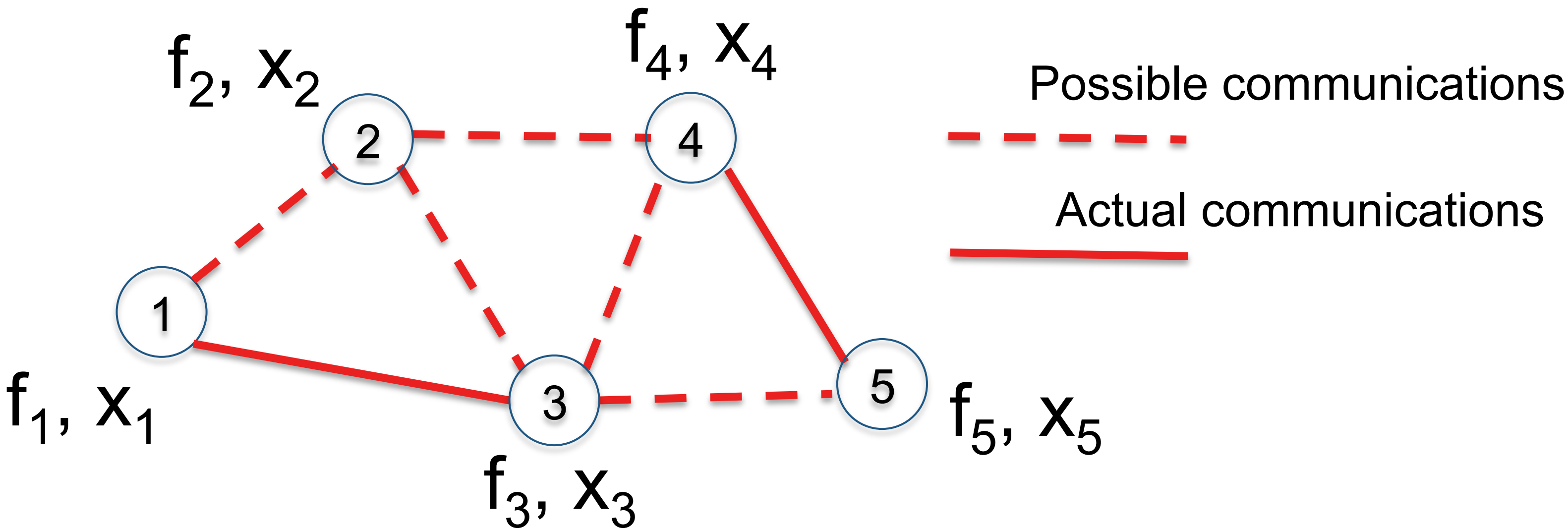
- $\underset{\mathbf{x}}{\operatorname{argmin}} F(\mathbf{x}) = \sum_{i=1}^{N} f_i(\mathbf{x})$, f_i convex
- each node knows only f_i and can communicate with its neighbours

Distributed Subgradients Methods



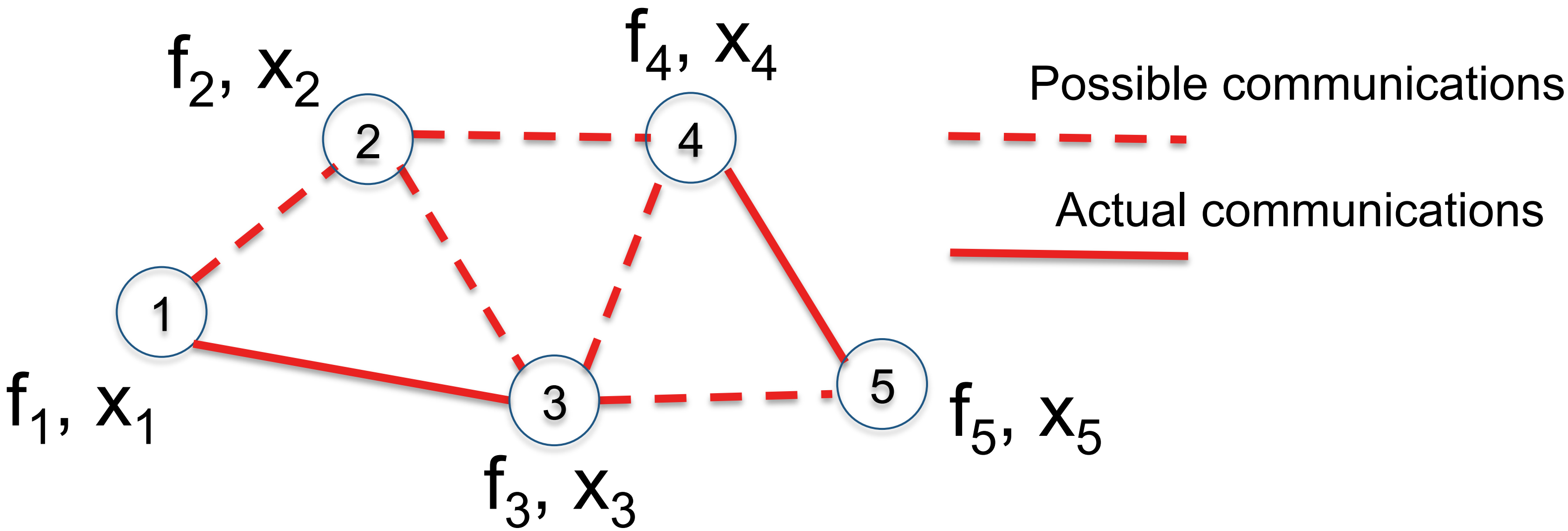
- $$\hat{\mathbf{x}}^{(i)}(k+1) = \sum_{j \in N_i}^{N} w_{ij} \hat{\mathbf{x}}^{(j)}(k) \ + \ \gamma(k) \nabla f_i(\hat{\mathbf{x}}^{(i)}(k))$$

Distributed Subgradients Methods



- $$\underbrace{\hat{\mathbf{x}}^{(i)}(k+1) = \sum_{j \in N_i}^{N} w_{ij}\hat{\mathbf{x}}^{(j)}(k)}_{\textbf{Consensus}} + \underbrace{\gamma(k)\nabla f_i(\hat{\mathbf{x}}^{(i)}(k))}_{\textbf{Gradient}}$$

Distributed Subgradients Methods



- $$\hat{\mathbf{x}}^{(i)}(k+1) = \sum_{j \in N_i}^{N} w_{ij} \hat{\mathbf{x}}^{(j)}(k) \;+\; \gamma(k) \nabla f_i(\hat{\mathbf{x}}^{(i)}(k))$$
- For dynamic networks w/ intercommunication times
  - Deterministic upper bounded
  - Independent ON/OFF

Our Research Directions

What is missing for application in DTNs?

- Link dynamics originated from nodes' mobility
  - Shown that convergence holds when contacts derive from a generic Markovian process [Infocom11]
- Asynchronous operation
  - $$\hat{\mathbf{x}}^{(i)}(k+1) = \sum_{j \in N_i}^{N} w_{ij} \hat{\mathbf{x}}^{(j)}(k) \;+\; \gamma_i(k) \nabla f_i(\hat{\mathbf{x}}^{(i)}(k))$$
  - ongoing
- Convergence speedup

Our Research Directions

Convergence speed (w/ K. Avrachenkov)

- $$\hat{\mathbf{x}}^{(i)}(k+1) = \sum_{j \in N_i}^{N} w_{ij} \hat{\mathbf{x}}^{(j)}(k) \; + \; \gamma_i(k) \nabla f_i(\hat{\mathbf{x}}^{(i)}(k))$$
  - two components: consensus and gradient
- How to set the weights in a **distributed way**?
  - maximize the spectral gap of weight matrix *W*
  - our proposal: minimize $\mathrm{tr}(W^{2n})$
    - [submitted CDC]

Our Research Directions

- How to set the weights in a **distributed way**?
  - maximize the spectral gap of weight matrix *W*
    - spectral gap = 1-max{$|\lambda_i|$, i=2,…N}
  - our proposal: minimize trace(W^{2n})
    - trace(W^{2n}) = $1+\sum_{i>1} \lambda_i^{2n}$
    - Distributed algorithm
    - Tradeoff

      approx. quality vs communication via n

Thank you