Temporal Networks

Lecture 21

CSCI 4974/6971

21 Nov 2016

Today's Biz

1. **Reminders**
2. Review
3. Temporal Networks

Reminders

- Assignment 6: due date Dec 8th
- Final Project Presentation: December 8th
- Project Report: December 11th
- Office hours: Tuesday & Wednesday 14:00-16:00 Lally 317
  - Or email me for other availability

Today's Biz

1. Reminders
2. **Review**
3. Temporal Networks

Quick Review

Graph Sampling:

- Vertex sampling methods
  - Uniform random
  - Degree-biased
  - Centrality-biased (PageRank)
- Edge sampling methods
  - Uniform random
  - Vertex-edge (select vertex, then random edge)
  - Induced edge (select edge, include all edges of attached vertices)

Quick Review

Random Walks:

- Sample by exploring the graph
- Sampling methods
  - Uniform random
  - Random with restarts
  - Random with jumps
  - Biased

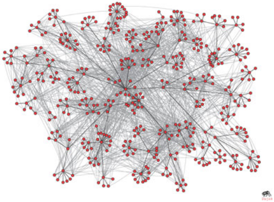
Today's Biz

1. Reminders
2. Review
3. **Dynamic Networks**

Temporal Graphs for Dynamic Network Analysis

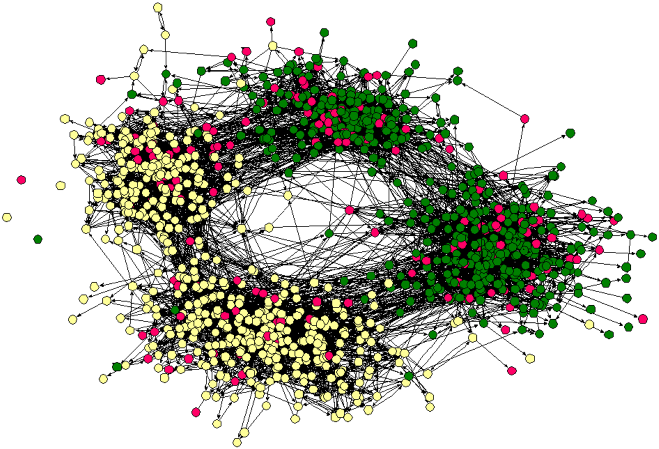
Mirco Musolesi, University of Birmingham

Temporal Graphs for Dynamic Network Analysis

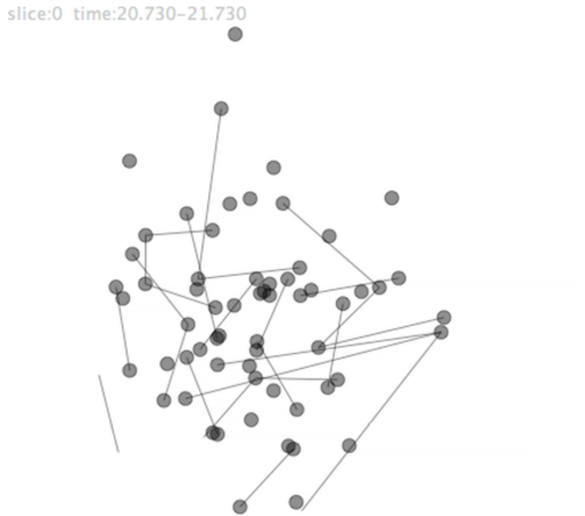


Mirco Musolesi
School of Computer Science
University of Birmingham

Joint work with Vito Latora, Cecilia Mascolo, Vincenzo Nicosia, Salvatore Scellato and John Tang

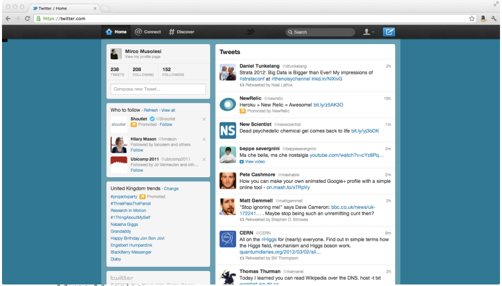


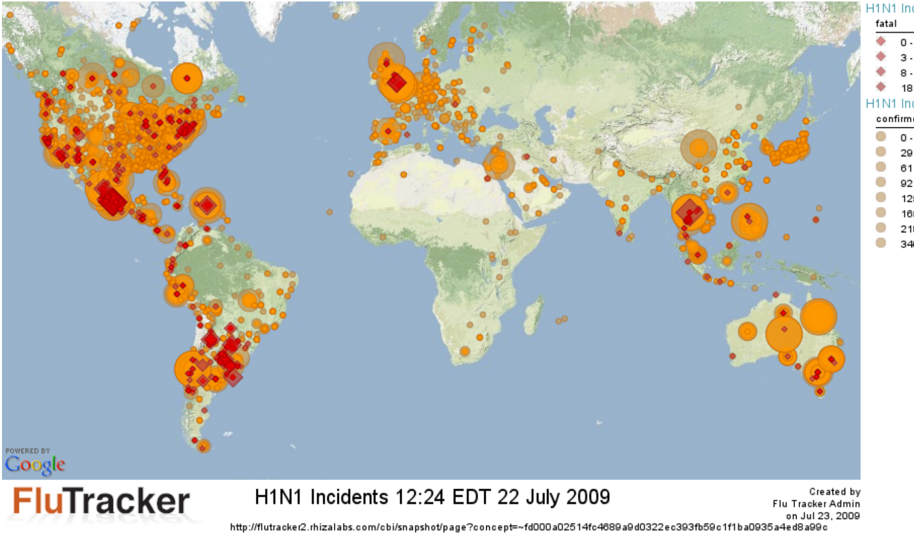
Credit: Mark Newman



Reality Mining Dataset


Facebook
https://www.facebook.com
facebook
Mirco Musolesi
Home
Update status
Add photo/video
Ask question
FAVOURITES
News Feed
Messages
Events
APPS
Links
Apps and Games
LISTS
Close friends
Family
Dartmouth College
University of Birmingham
Leamington Spa Area
GROUPS
The Computer Laboratory
Create group
Jamie Anderson
BBQSoc has only had 2 events and is already in the top 15 biggest societies.
Dan Winch likes this.
Dan Winch I'm joining!
Write a comment...
Gianmarco Walden Ferrari
We'll Do Better Next Time
but This Is a Good Start
Sigur Rós - Heima - Ágætis Byrjun - HD
www.youtube.com
Loris Marinetti likes this.
Samuele Cassani
mi sento lievemente disturbato
Salvatore Scellato
just loved Andrew Bird - Sifters
Matteo Rizzo's birthday is today
Create event
Sponsored
See all
College Careers Advisers
Netflix - 1 month FREE
Unlimited films and TV shows. 1 month FREE!
Farrow & Ball Showroom
Dress to Impress
Tiziana Degli Esposti
Go online to see who's online to chat.
Search


Twitter / Home
https://twitter.com
Home
Connect
Discover
Search
Mirco Musolesi
View my profile page
238 TWEETS
208 FOLLOWING
152 FOLLOWERS
Compose new Tweet...
Who to follow · Refresh · View all
Shoutlet
Hilary Mason
Ubicomp 2011
United Kingdom trends · Change
#projectxparty
#ThreePassTheParcel
Research In Motion
#1ThingAboutMySelf
Natasha Giggs
Grandaddy
Happy Birthday Jon Bon Jovi
Engelbert Humperdink
BlackBerry Messenger
Diaby
Tweets
Daniel Tunkelang
Strata 2012: Big Data is Bigger than Ever! My impressions of #strataconf at #thenoisychannel lnkd.in/NiXivG
NewRelic
Heroku + New Relic = Awesome! bit.ly/z5AK3O
New Scientist
Dead psychedelic chemical gel comes back to life bit.ly/yi3bOK
beppe severgnini
Ma che bella, ma che nostalgia youtube.com/watch?v=cYz8Pq...
Pete Cashmore
How you can make your own animated Google+ profile with a simple online tool - on.mash.to/xTRpVy
Matt Gemmell
"Stop ignoring me!" says Dave Cameron: bbc.co.uk/news/uk-172241... . Maybe stop being such an unremitting cunt then?
CERN
All on the #Higgs for (nearly) everyone. Find out in simple terms how the Higgs field, mechanism and Higgs boson work. quantumdiaries.org/2012/03/02/all...
Thomas Thurman
Today I learned you can read Wikipedia over the DNS. host -t txt

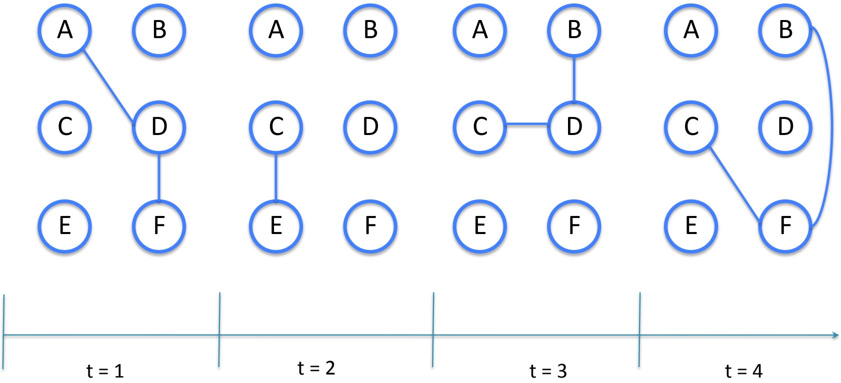


Credit: Flutracker.com

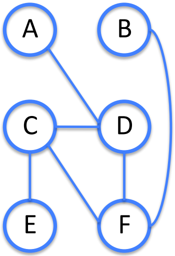
Problem: existing metrics do not capture the inherent dynamism of networks over time.

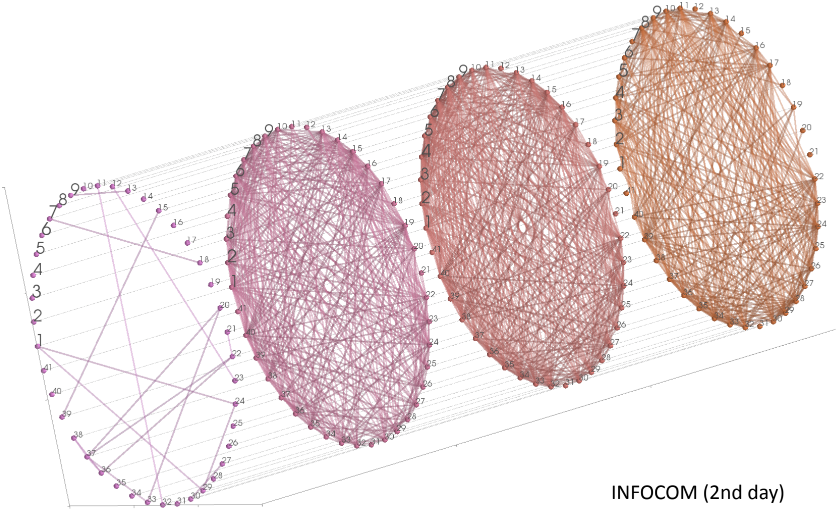
We need new **temporal metrics** defined over **temporal graphs** for studying dynamic processes over these networks.

An Example of Temporal Graph

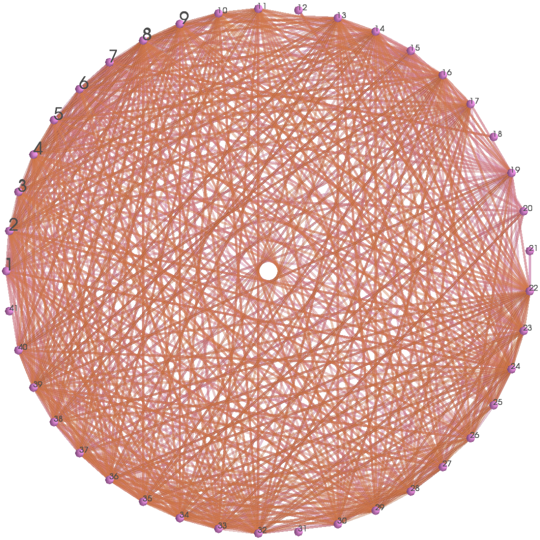


...and the Corresponding Static Graph

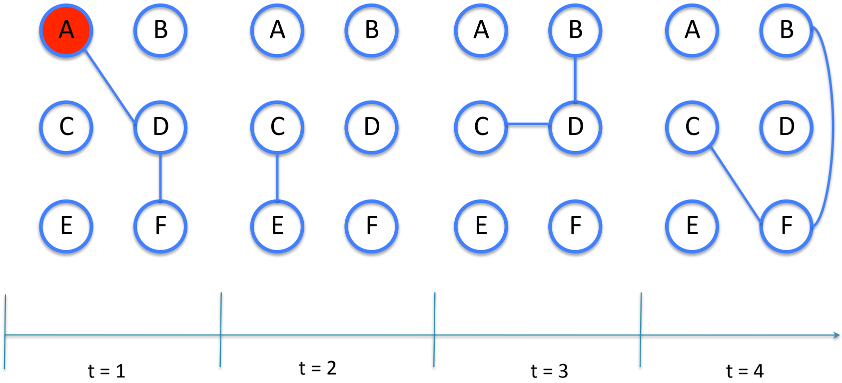




INFOCOM (2nd day)

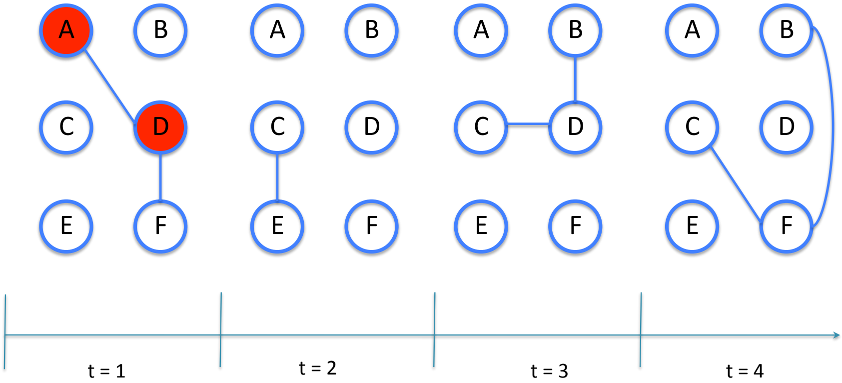

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41

Calculating the Temporal Distance

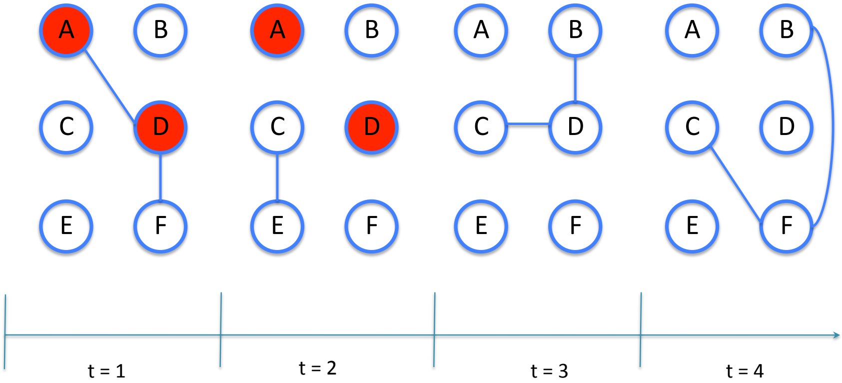


Calculating the Temporal Distance (t = 1)

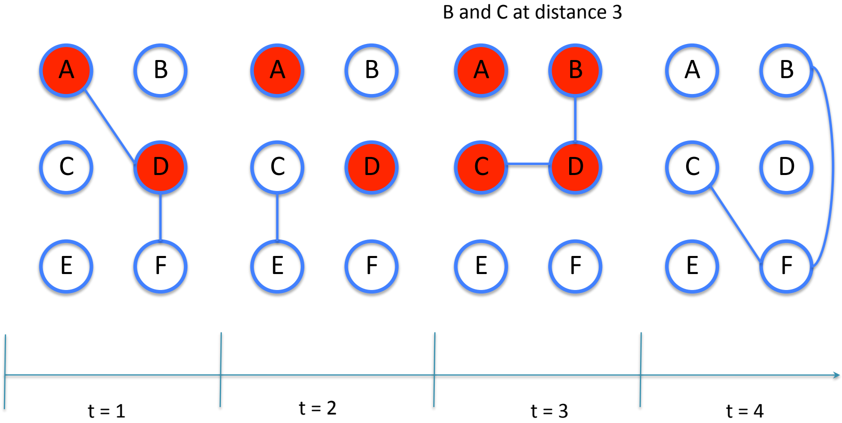
D at distance 1



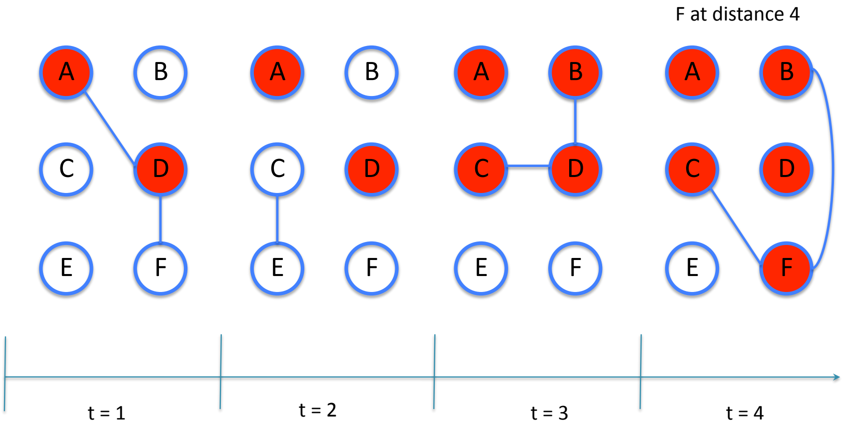
Calculating the Temporal Distance (t = 2)



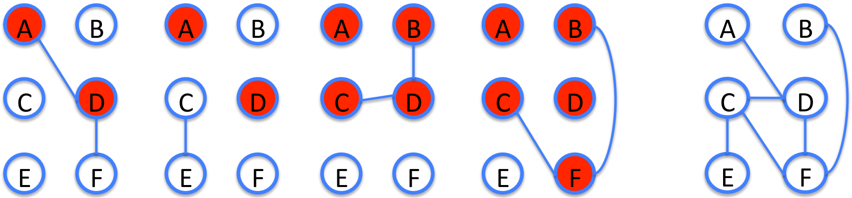
Calculating the Temporal Distance (t = 3)



Calculating the Temporal Distance (t = 4)



What about the Static Distance?

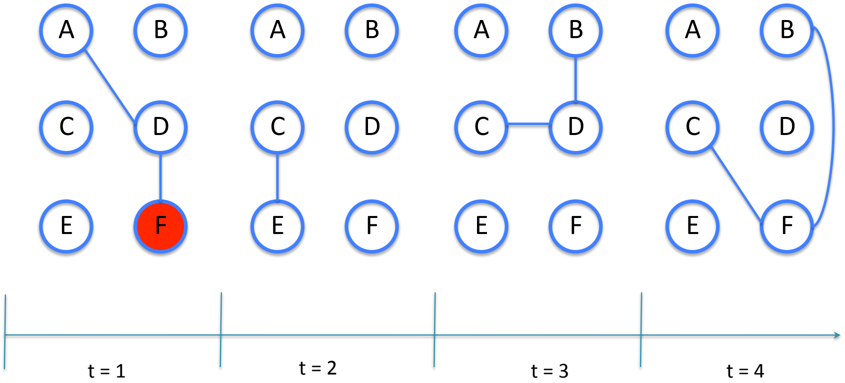


E is *statically* reachable but in reality it is not *dynamically* reachable!

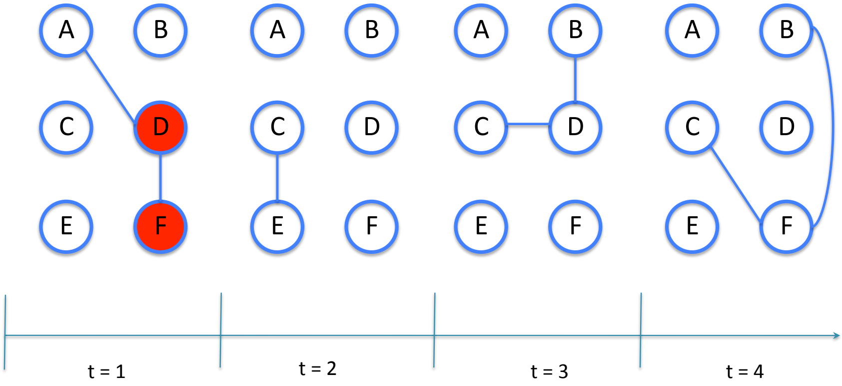
A-> F requires 2 transmissions (hops), but in reality it requires 3

No information about the duration of the process

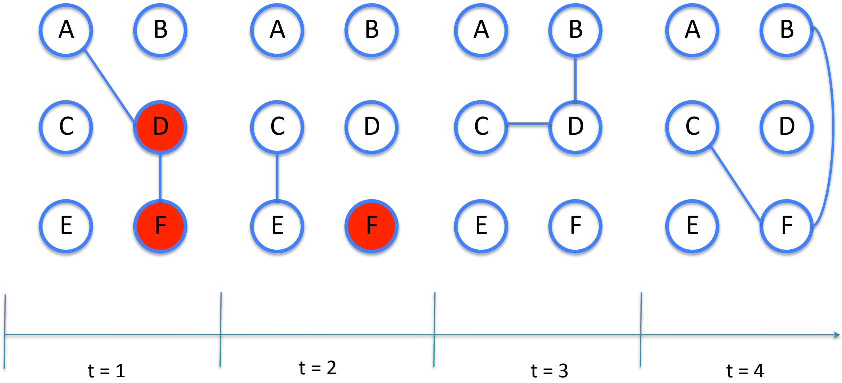
What about the Symmetric Distance (F to A)?



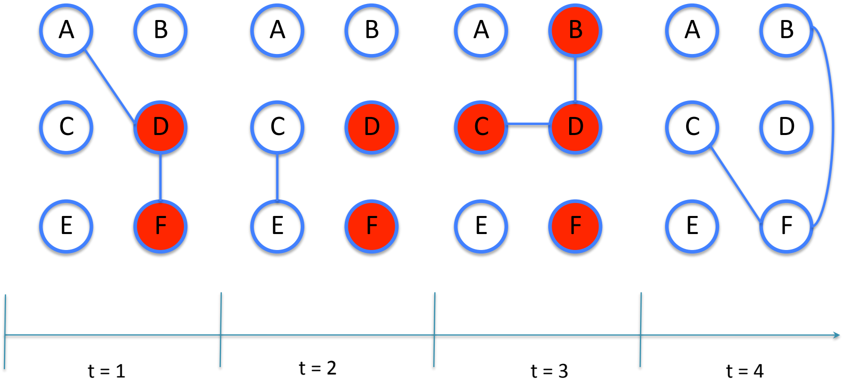
Calculating the Inverse Temporal Distance (t = 1)



Calculating the Inverse Temporal Distance (t = 2)

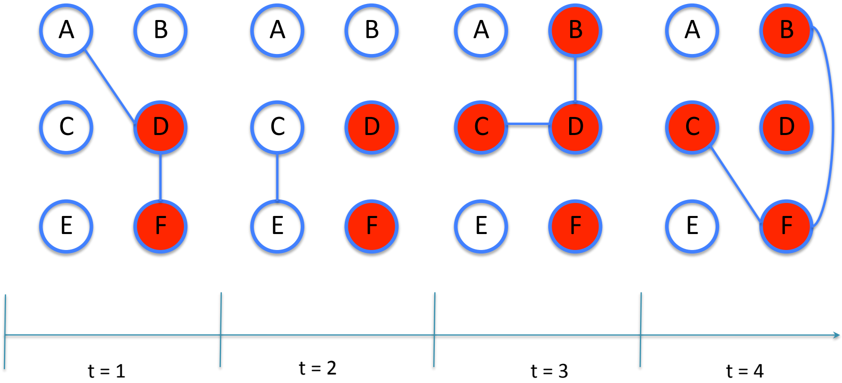


Calculating the Inverse Temporal Distance (t = 3)



Calculating the Inverse Temporal Distance (t = 4)

A is not reachable
[infinite distance]



Characteristic Temporal Path Length

- Characteristic temporal path length:

$$L^h(t_{min}, t_{max}) = \frac{1}{N(N-1)} \sum_{ij} d_{ij}^h(t_{min}, t_{max})$$

- Defined considering the horizon of the infection
- Possible problem related to the potential divergence due to pairs of nodes that are not temporally connected

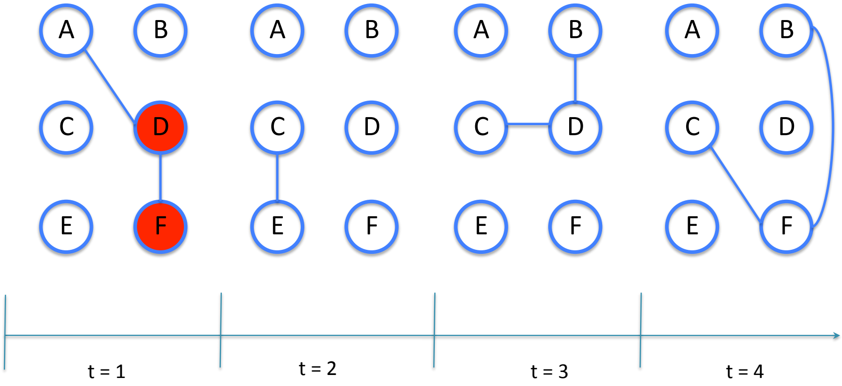
Characteristic Temporal Path Length

- Characteristic temporal path length:

$$L^h(t_{min}, t_{max}) = \frac{1}{N(N-1)} \sum_{ij} d_{ij}^h(t_{min}, t_{max})$$

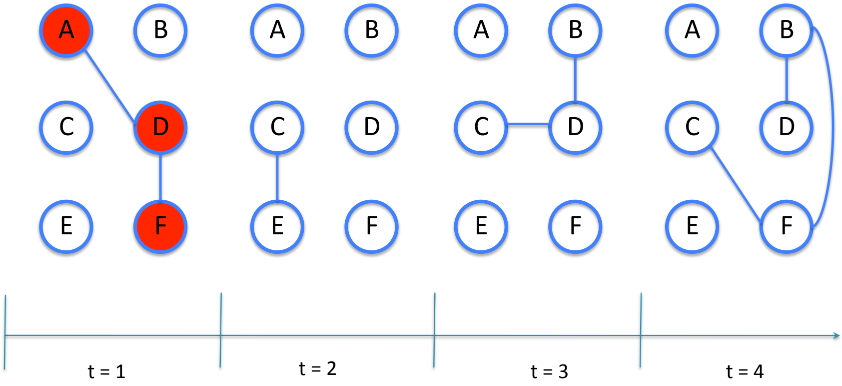
- **Defined considering the horizon of the infection**
- Possible problem due to the potential divergence due to pairs of nodes that are not temporally connected

Impact of the Horizon Parameter (F -> A, h = 1)



Impact of the Horizon Parameter (F -> A, h = 2)

A was not reachable at all with h = 1 (in 4 time windows), but with h = 2 it is a distance 1!



Characteristic Temporal Path Length

- Characteristic temporal path length:

$$L^h(t_{min}, t_{max}) = \frac{1}{N(N-1)} \sum_{ij} d_{ij}^h(t_{min}, t_{max})$$

- Defined considering the horizon of the infection
- **Possible problem related to the potential divergence due to pairs of nodes that are not temporally connected**

Temporal Efficiency

- Solution: definition of temporal efficiency:

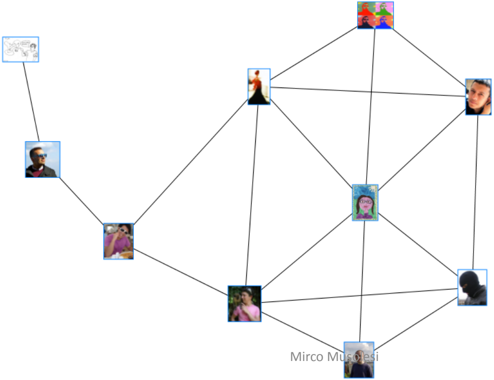
$$E^h_{T_{ij}}(t_{min}, t_{max}) \quad = \quad \frac{1}{d^h_{ij}(t_{min}, t_{max})}$$

$$E^h_{glob}(t_{min}, t_{max}) = \frac{1}{N(N-1)} \sum_{ij} E^h_{T_{ij}}(t_{min}, t_{max})$$

- High value of E (low value of L) means that the nodes of the graphs can communicate efficiently

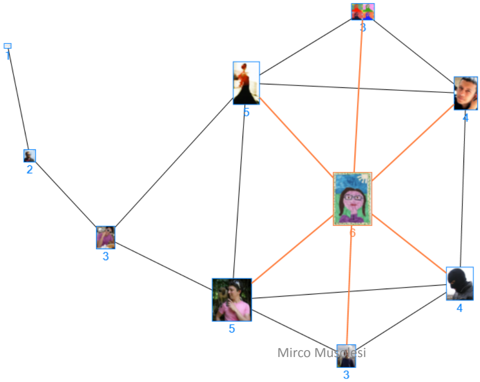
Centrality Metrics

- Most number of friends
- Quickly spread information to many people
- Mediates between the most information flows



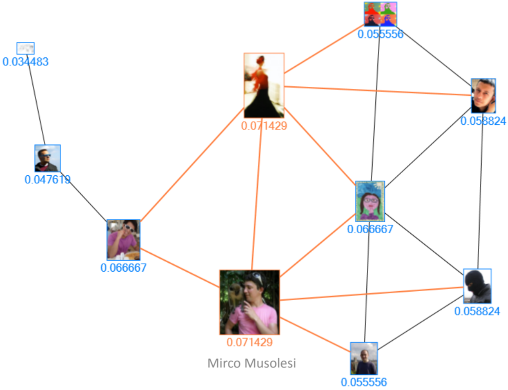
Degree

- $C_i^{deg} = \text{number of links to i}$
- Popular nodes



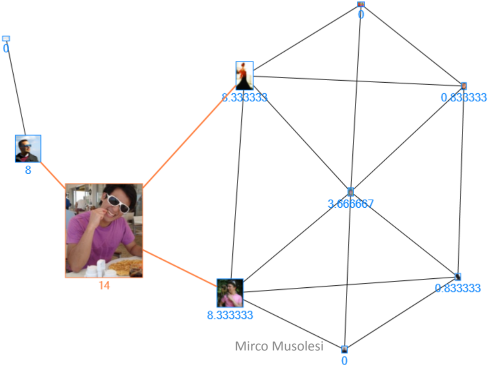
Static Closeness Centrality

- $C_i = \sum_{i \neq j} d_{ij}$
- Average shortest path length to all other nodes



Static Betweenness Centrality

- $C_i^{bet} = \sum_{i \neq s \neq t} \frac{\delta_s t(i)}{\delta_{st}}$ where δ_{st} is # shortest paths from s to t, $\delta_{st}(i)$ is # shortest paths passing through i
- Fraction of shortest paths which pass through node i



Temporal Centrality Measures

- Static Closeness and Betweenness based on *static shortest paths*
- Definition of *closeness* and *betweenness* with temporal paths:
  - Duration
  - Time Order
  - Frequency

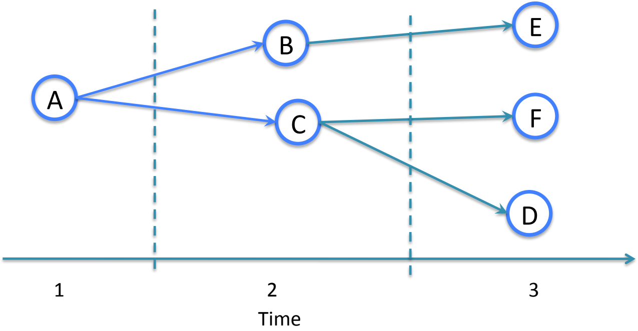
Temporal Closeness

Average over shortest *temporal* paths to all other nodes:

$$C_i = \frac{1}{W(N-1)} \sum_{j \neq i \in V} d_{i,j}$$

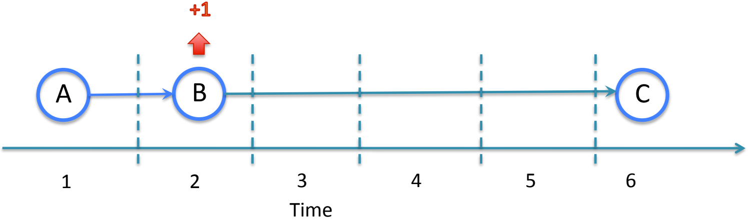
Temporal Closeness

$$C_A = \frac{(2+2)+(3+3+3)}{(3*(6-1))} = 0.867$$



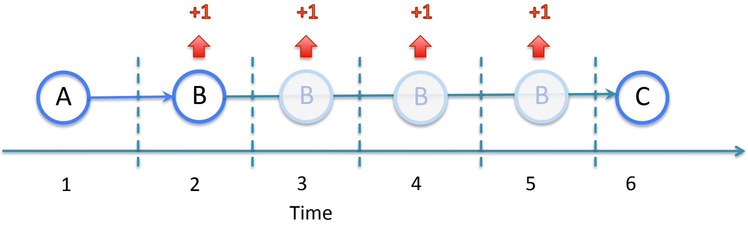
Temporal Betweenness

- Using temporal path length



Temporal Betweenness

- Take into account **duration**



Temporal Betweenness

$$C_i^B(t) = \frac{1}{(N-1)(N-2)} \sum_{\substack{j \in V \\ j \neq i}} \sum_{\substack{k \in V \\ k \neq i \\ k \neq j}} \frac{U(i,t,j,k)}{|S_{jk}^h|}$$

Where:

- $U(i,t,j,k)$ number of shortest paths from j to k, where node i is holding a message at time window t
- $|S_{jk}^h|$ number of shortest temporal paths between j and k

Temporal Betweenness

Sum over all time windows for each node:

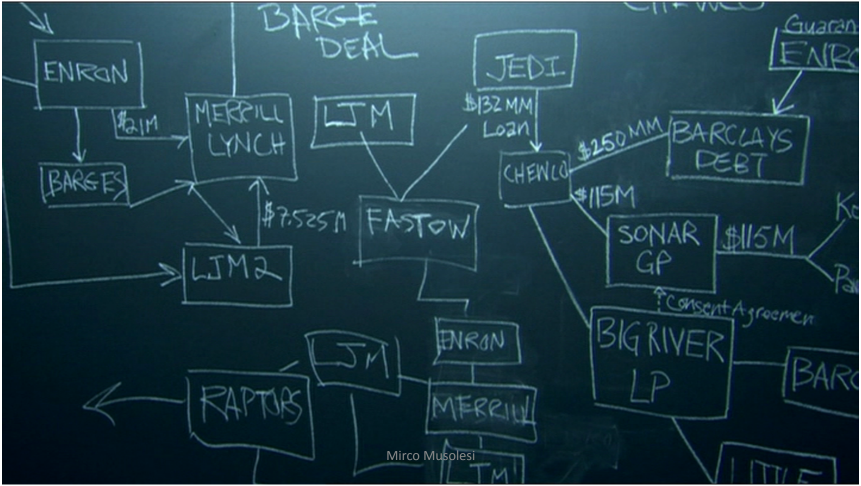
$$C_i^B = \frac{1}{W} \sum_{t=1}^{W} C_i^B((t \times w) + t_{min})$$

Evaluating Centrality

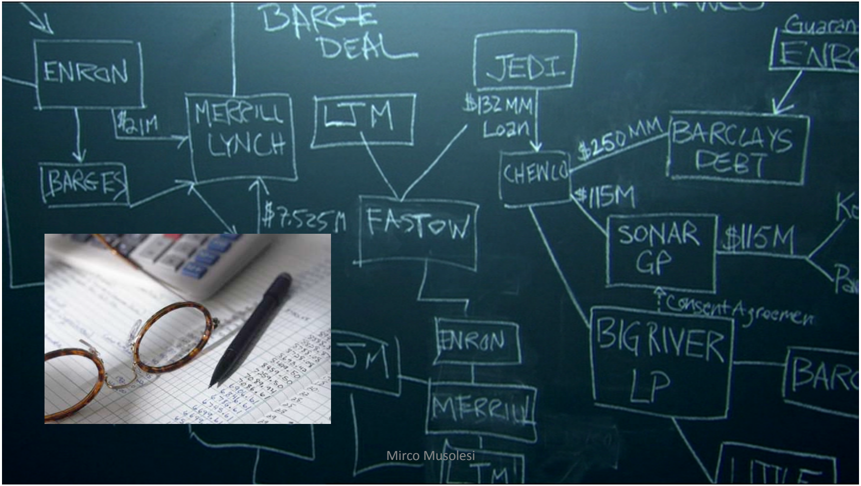
- Corporate Email Dataset
- Two perspectives:
  - Semantic: roles of each node
  - Dynamic Processes: simulate communication
    - Information Dissemination
    - Information Mediation

Evaluating Centrality:
Enron in the News

Scandals


BARGE DEAL
ENRON
$21M
MERRILL LYNCH
LJM
JEDI
$132 MM Loan
CHEWCO
$250 MM
BARCLAYS DEBT
BARGES
$7.525M
FASTOW
$115M
SONAR GP
$115M
LJM2
Consent Agreemen
BIG RIVER LP
ENRON
LJM
RAPTORS
MERRILL

Scandals


BARGE DEAL
ENRON
JEDI
MERRILL LYNCH
LJM
$132 MM Loan
$21M
BARGES
CHEWCO
$250 MM
BARCLAYS DEBT
$115M
$7.525M
FASTOW
SONAR GP
$115M
Consent Agreemen
BIG RIVER LP
ENRON
MERRILL

Scandals

BARGE DEAL
ENRON
$12M
MERRILL LYNCH
LJM
BARGES
$7.525M
LIFE
NEWS
CELEBRITY
TRAVEL
ANIMALS
SPORTS
Return to Search Results
Civil Rights: Women in the Struggle
Taliban: Oppression and Resistance
Disasters at Sea
Troops Honor Fallen Comrades
EMAIL
RSS
RATE
PRINT
LINK TO
LICENSE
DIGG IT
TWEET
SHARE
California Experiences Second Week of Energy Crisis
384758 04: FILE PHOTO: Steam rises from the Etiwanda Generating Plant behind power lines near Rancho Cucamonga, CA, January 24, 2001, as the statewide energy crisis continues. The power plant is owned and operated by Edison O&M Service Division. Blackout warnings were issued for the first time by the California Independent System Operator June 17, reporting that blackouts could occure Monday and Tuesday afternoons between noon and eight in the evening.
Photo: David McNew/Getty Images
Jun 18, 2001

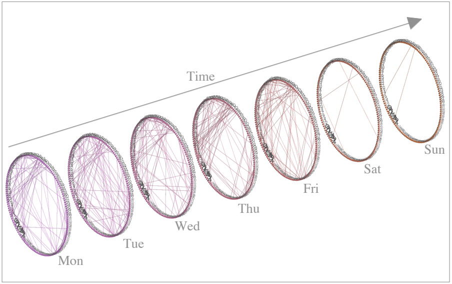
Public Investigation

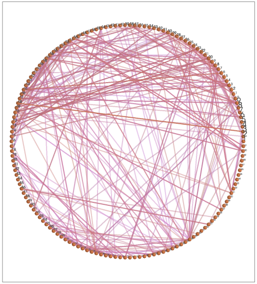
- Telephone logs
- Documents
- Financials
- Emails

  - 151 user mailboxes
  - May 1999 to Jun 2002
  - 250,000 emails
  - NOT anonymised

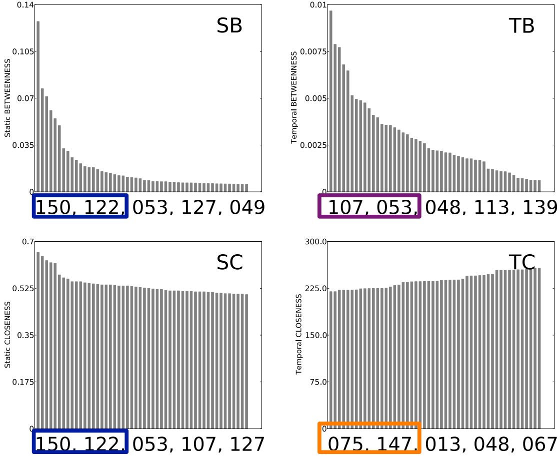
Email exchanges to Temporal Graph

- Core 151 users
- Window size= 1 business day
- 1137 days





Semantics



ID	Role
009	(Unknown)
013	Legal
017	Manager
048	Executive
053	Trader
054	President
067	Vice President
073	Trader
075	Director of Trading
107	Trader
122	Managing Director
127	Manager
139	Director
147	Trader
150	Secretary

Semantics

ID	Name	Role
9	Stephanie Panus	(Unknown)
13	Marie Heard	Legal
17	Mike Grigsby	Manager
48	Tana Jones	Executive
53	John Lavorato	Trader
54	Greg Whalley	President
67	Sara Shackleton	Vice President
73	Jeff Dasovich	Trader
75	Gerald Nemec	Director of Trading
107	Louise Kitchen	Trader
122	Sally Beck	Managing Director
127	Kenneth Lay	Manager
139	Mary Hain	Director
147	Carol Clair	Trader
150	Liz Taylor	Secretary

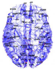
CNN.com/LAWCENTER

Top bonuses awarded

John Lavorato: $5 million
Louise Kitchen: $2 million
Jeffrey McMahon: $1.5 million
James Fallon: $1.5 million
Raymond Bowen Jr.: $750,000
Mark Haedicke: $750,000
Gary Hickerson: $700,000
Wesley Colwell: $600,000
Richard Dimichele: $600,000

- Big bonuses linked with information mediators

Small-world Behaviour in Time-Varying Networks



Brain network

Bluetooth (INFOCOM'06)

Facebook London Network

	C	C^{rand}	L	L^{rand}	E	E^{rand}
α	0.44	0.18	3.9 (100%)	4.2 (98%)	0.50	0.48
β	0.40	0.17	6.0 (94%)	3.6 (92%)	0.41	0.45
γ	0.48	0.13	12.2 (86%)	8.7 (89%)	0.39	0.37
δ	0.44	0.17	2.2 (100%)	2.4 (92%)	0.57	0.56
d1	0.80	0.44	8.84 (61%)	6.00 (65%)	0.192	0.209
d2	0.78	0.35	5.04 (87%)	4.01 (88%)	0.293	0.298
d3	0.81	0.38	9.06 (57%)	6.76 (59%)	0.134	0.141
d4	0.83	0.39	21.42 (15%)	15.55(22%)	0.019	0.028
Mar	0.044	0.007	456	451	0.000183	0.000210
Jun	0.046	0.006	380	361	0.000047	0.000057
Sep	0.046	0.006	414	415	0.000058	0.000074
Dec	0.049	0.006	403	395	0.000047	0.000059

What's Next?

- Analysis vs Prediction
- Multi-dimensional networks
- Integration of the geographic aspects: spatio-temporal analysis
- Application to security problems
- System issues: design of scalable systems for real-time data processing

Introduction to Dynamic Networks Models, Algorithms, and Analysis

Rajmohan Rajaraman, Northeastern University

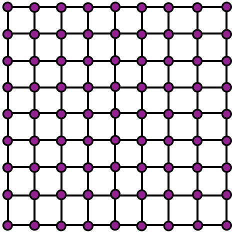
Introduction to Dynamic Networks
Models, Algorithms, and Analysis

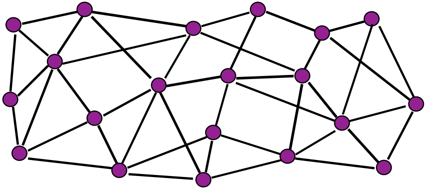
Rajmohan Rajaraman, *Northeastern U.*
www.ccs.neu.edu/home/rraj/Talks/DynamicNetworks/DYNAMO/
June 2006

Many Thanks to...

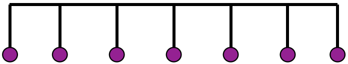
- Filipe Araujo, Pierre Fraigniaud, Luis Rodrigues, Roger Wattenhofer, and organizers of the summer school
- All the researchers whose contributions will be discussed in this tutorial

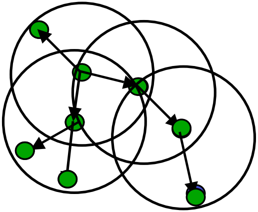
What is a Network?





General undirected or directed graph





Classification of Networks

- **Synchronous**:
  - Messages delivered within one time unit
  - Nodes have access to a common clock
- **Asynchronous**:
  - Message delays are arbitrary
  - No common clock
- **Static**:
  - Nodes never crash
  - Edges maintain operational status forever
- **Dynamic**:
  - Nodes may come and go
  - Edges may crash and recover

Dynamic Networks: What?

- Network dynamics:
  - The network topology changes over times
  - Nodes and/or edges may come and go
  - Captures faults and reliability issues
- Input dynamics:
  - Load on network changes over time
  - Packets to be routed come and go
  - Objects in an application are added and deleted

Dynamic Networks: How?

- Duration:
  - Transient: The dynamics occur for a short period, after which the system is static for an extended time period
  - Continuous: Changes are constantly occurring and the system has to constantly adapt to them
- Control:
  - Adversarial
  - Stochastic
  - Game-theoretic

Dynamic Networks are Everywhere

- Internet
  - The network, traffic, applications are all dynamically changing
- Local-area networks
  - Users, and hence traffic, are dynamic
- Mobile ad hoc wireless networks
  - Moving nodes
  - Changing environmental conditions
- Communication networks, social networks, Web, transportation networks, other infrastructure

Adversarial Models

- Dynamics are controlled by an adversary
  - Adversary decides when and where changes occur
  - Edge crashes and recoveries, node arrivals and departures
  - Packet arrival rates, sources, and destinations
- For meaningful analysis, need to constrain adversary
  - Maintain some level of connectivity
  - Keep packet arrivals below a certain rate

Stochastic Models

- Dynamics are described by a probabilistic process
  - Neighbors of new nodes randomly selected
  - Edge failure/recovery events drawn from some probability distribution
  - Packet arrivals and lengths drawn from some probability distribution
- Process parameters are constrained
  - Mean rate of packet arrivals and service time distribution moments
  - Maintain some level of connectivity in network

Game-Theoretic Models

- Implicit assumptions in previous two models:
  - All network nodes are under one administration
  - Dynamics through external influence
- Here, each node is a potentially independent agent
  - Own utility function, and rationally behaved
  - Responds to actions of other agents
  - Dynamics through their interactions
- Notion of stability:
  - Nash equilibrium

Design & Analysis Considerations

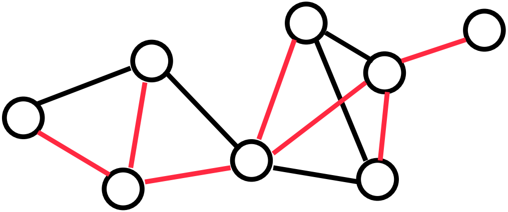
- Distributed computing:
  - For static networks, can do pre-processing
  - For dynamic networks (even with transient dynamics), need distributed algorithms
- Stability:
  - Transient dynamics: Self-stabilization
  - Continuous dynamics: Resources bounded at all times
  - Game-theoretic: Nash equilibrium
- Convergence time
- Properties of stable states:
  - How much resource is consumed?
  - How well is the network connected?
  - How far is equilibrium from socially optimal?

Five Illustrative Problem Domains

- Spanning trees
  - Transient dynamics, self-stabilization
- Load balancing
  - Continuous dynamics, adversarial input
- Packet routing
  - Transient & continuous dynamics, adversarial
- Queuing systems
  - Adversarial input
- Network evolution
  - Stochastic & game-theoretic

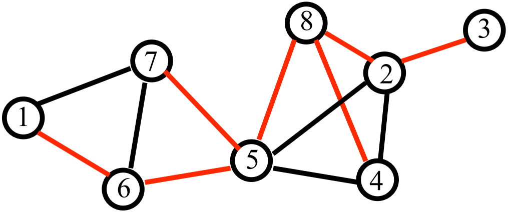
Spanning Trees

Spanning Trees



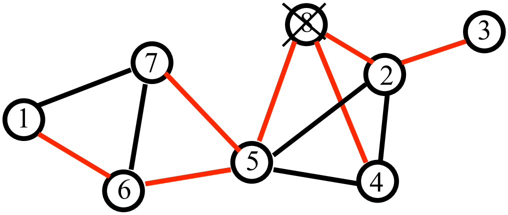
- One of the most fundamental network structures
- Often the basis for several distributed system operations including leader election, clustering, routing, and multicast
- Variants: any tree, BFS, DFS, minimum spanning trees

Spanning Tree in a Static Network



- Assumption: Every node has a unique identifier
- The largest id node will become the root
- Each node v maintains distance d(v) and next-hop h(v) to largest id node r(v) it is aware of:
  - Node v propagates (d(v),r(v)) to neighbors
  - If message (d,r) from u with r > r(v), then store (d+1,r,u)
  - If message (d,r) from p(v), then store (d+1,r,p(v))

Spanning Tree in a Dynamic Network



- Suppose node 8 crashes
- Nodes 2, 4, and 5 detect the crash
- Each separately discards its own triple, but believes it can reach 8 through one of the other two nodes
  - Can result in an infinite loop
- How do we design a self-stabilizing algorithm?

Exercise

- Consider the following spanning tree algorithm in a synchronous network
- Each node v maintains distance d(v) and next-hop h(v) to largest id node r(v) it is aware of
- In each step, node v propagates (d(v),r(v)) to neighbors
- On receipt of a message:
  - If message (d,r) from u with r > r(v), then store (d+1,r,u)
  - If message (d,r) from p(v), then store (d+1,r,p(v))
- Show that there exists a scenario in which a node fails, after which the algorithm never stabilizes

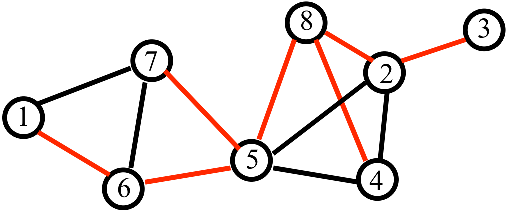
Self-Stabilization

- Introduced by Dijkstra [Dij74]
  - Motivated by fault-tolerance issues [Sch93]
  - Hundreds of studies since early 90s
- A system S is self-stabilizing with respect to predicate P
  - Once P is established, P remains true under no dynamics
  - From an arbitrary state, S reaches a state satisfying P within finite number of steps
- Applies to transient dynamics
- Super-stabilization notion introduced for continuous dynamics [DH97]

Self-Stabilizing ST Algorithms

- Dozens of self-stabilizing algorithms for finding spanning trees under various models [Gär03]
  - Uniform vs non-uniform networks
  - Fixed root vs non-fixed root
  - Known bound on the number of nodes
  - Network remains connected
- Basic idea:
  - Some variant of distance vector approach to build a BFS
  - Symmetry-breaking
    - Use distinguished root or distinct ids
  - Cycle-breaking
    - Use known upper bound on number of nodes
    - Local detection paradigm

Self-Stabilizing Spanning Tree



- Suppose upper bound N known on number of nodes [AG90]
- Each node v maintains distance d(v) and parent h(v) to largest id node r(v) it is aware of:
  - Node v propagates (d(v),r(v)) to neighbors
  - If message (d,r) from u with r > r(v), then store (d+1,r,u)
  - If message (d,r) from p(v), then store (d+1,r,p(v))
- If d(v) exceeds N, then store (0,v,v): breaks cycles

Self-Stabilizing Spanning Tree

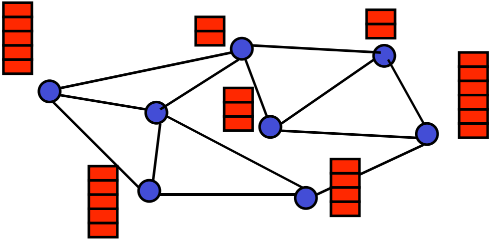
- Suppose upper bound N not known [AKY90]
- Maintain triple (d(v),r(v),p(v)) as before
  - If v > r(u) of all of its neighbors, then store (0,v,v)
  - If message (d,r) received from u with r > r(v), then v "joins" this tree
    - Sends a join request to the root r
    - On receiving a grant, v stores (d+1,r,u)
  - Other local consistency checks to ensure that cycles and fake root identifiers are eventually detected and removed

Spanning Trees: Summary

- Model:
  - Transient adversarial network dynamics
- Algorithmic techniques:
  - Symmetry-breaking through ids and/or a distinguished root
  - Cycle-breaking through sequence numbers or local detection
- Analysis techniques:
  - Self-stabilization paradigm
- Other network structures:
  - Hierarchical clustering
  - Spanners (related to metric embeddings)

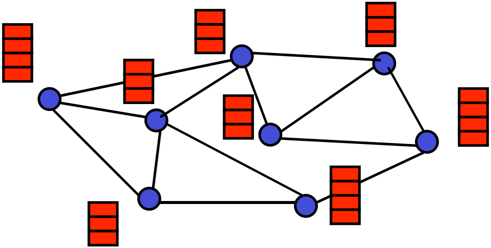
Load Balancing

Load Balancing



- Each node v has w(v) tokens
- Goal: To balance the tokens among the nodes
- Imbalance: $\max_{u,v} |w(u) - w_{avg}|$
- In each step, each node can send at most one token to each of its neighbors

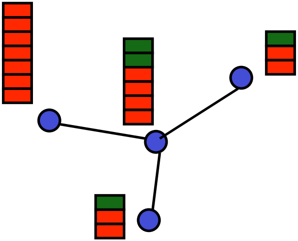
Load Balancing



- In a truly balanced configuration, we have $|w(u) - w(v)| \leq 1$
- Our goal is to achieve fast approximate balancing
- Preprocessing step in a parallel computation
- Related to routing and counting networks [PU89, AHS91]

Local Balancing

- Each node compares its number of tokens with its neighbors
- In each step, for each edge (u,v):
  - If w(u) > w(v) + 2d, then u sends a token to v
  - Here, d is maximum degree of the network
- Purely local operation

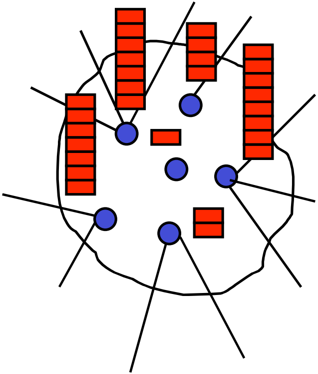


Convergence to Stable State

- How long does it take local balancing to converge?
- What does it mean to converge?
  - Imbalance is "constant" and remains so
- What do we mean by "how long"?
  - The number of time steps it takes to achieve the above imbalance
  - Clearly depends on the topology of the network and the imbalance of the original token distribution

Expansion of a Network

- Edge expansion α:
  - Minimum, over all sets S of size $\leq n/2$, of the term
    $$|E(S)|/|S|$$
- Lower bound on convergence time:
  $$(w(S) - |S| \cdot w_{avg})/E(S)$$
  $$= (w(S)/|S| - w_{avg})/ \alpha$$



Expansion = 12/6 = 2
$w_{avg} = 3$
Lower bound = (29 - 18)/12

Properties of Local Balancing

- For any network G with expansion α, any token distribution with imbalance Δ converges to a distribution with imbalance $O(d \cdot \log(n)/\alpha)$ in $O(\Delta/\alpha)$ steps [AAMR93, GLM+99]
- Analysis technique:
  - Associate a potential with every node v, which is a function of the w(v)
    - Example: $(w(v) - avg)^2$, $c^{w(v)-avg}$
    - Potential of balanced configuration is small
  - Argue that in every step, the potential decreases by a desired amount (or fraction)
  - Potential decrease rate yields the convergence time
- There exist distributions with imbalance Δ that would take $\Omega(\Delta/\alpha)$ steps

Exercise

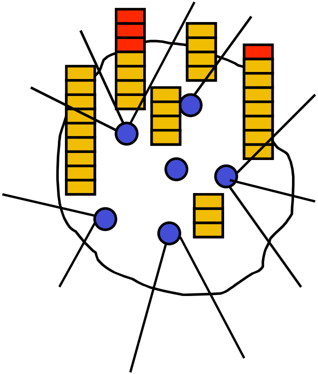
- For any graph G with edge expansion α, show that there is an initial distribution with imbalance Δ such that the time taken to reduce the imbalance by even half is $\Omega(\Delta/ \alpha)$ steps

Local Balancing in Dynamic Networks

- The "purely local" nature of the algorithm useful for dynamic networks
- Challenge:
  - May not "know" the correct load on neighbors since links are going up and down
- Key ideas:
  - Maintain an estimate of the neighbors' load, and update it whenever the link is live
  - Be more conservative in sending tokens
- Result:
  - Essentially same as for static networks, with a slightly higher final imbalance, under the assumption that the the set of live edges form a network with edge expansion α at each step

Adversarial Load Balancing

- Dynamic load [MR02]
  - Adversary inserts and/or deletes tokens
- In each step:
  - Balancing
  - Token insertion/deletion
- For any set S, let $d_t(S)$ be the change in number of tokens at step t
- Adversary is constrained in how much imbalance can be increased in a step
- Local balancing is stable against rate 1 adversaries [AKK02]



$$d_t(S) - (avg_{t+1} - avg_t)|S| \leq r \cdot e(S)$$

Stochastic Adversarial Input

- Studied under a different model [AKU05]
  - Any number of tokens can be exchanged per step, with one neighbor
- Local balancing in this model [GM96]
  - Select a random matching
  - Perform balancing across the edges in matching
- Load consumed by nodes
  - One token per step
- Load placed by adversary under statistical constraints
  - Expected injected load within window of w steps is at most rnw
  - The pth moment of total injected load is bounded, $p > 2$
- Local balancing is stable if $r < 1$

Load Balancing: Summary

- Algorithmic technique:
  - Local balancing
- Design technique:
  - Obtain a purely distributed solution for static network, emphasizing local operations
  - Extend it to dynamic networks by maintaining estimates
- Analysis technique:
  - Potential function method
  - Martingales

Today: In class work

- Implement framework for dynamic graph analysis
- Implement temporal BFS
- (Maybe) implement temporal load balancing

Blank code and data available on website (Lecture 21)

www.cs.rpi.edu/~slotag/classes/FA16/index.html