Congestion and Centrality in Data Networks

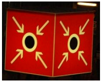
Raúl J. Mondragón C[1]
David K. Arrowsmith[2]

[1]Dept. of Electronic Engineering
[2]School of Mathematical Sciences
Queen Mary, University of London

20th December 2004

Outline


Transport for London
Congestion charging
Central ZONE
Mon - Fri
7 am - 6.30 pm



Introduction

- In a "simple" data/packet network the onset of congestion (a dynamical characteristic) depends on the average of all shortest path lengths (a topological characteristic).

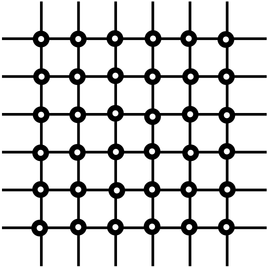
Ohira and Sawatari, 1998; Fukś and Lawniczak 1999; Solé and Valverde 2001; Woolf *et. al* 2002

Introduction

- In a “simple” data/packet network the onset of congestion (a dynamical characteristic) depends on the average of all shortest path lengths (a topological characteristic).
- Is the above result valid for all network’s topologies?
- Can this result help us when simulating very large networks?

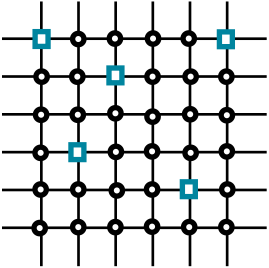
Ohira and Sawatari, 1998; Fukś and Lawniczak 1999; Solé and Valverde 2001; Woolf *et. al* 2002

Congestion in a Manhattan Network



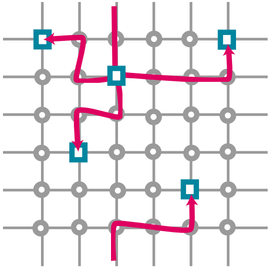
- Consider a Manhattan-toroidal network with S nodes
- Each node contains a queue where packets can be stored in transit (if the node is busy)

Congestion in a Manhattan Network



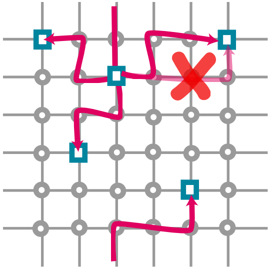
- Consider a Manhattan-toroidal network with S nodes
- Each node contains a queue where packets can be stored in transit (if the node is busy)
- The proportion of sources/sinks of traffic is $\rho \in (0, 1]$, i.e.$\#\text{sources} = \rho S$
- Each traffic source generates, on average, the same amount of traffic λ

Congestion in a Manhattan Network



- Consider a Manhattan-toroidal network with S nodes
- Each node contains a queue where packets can be stored in transit (if the node is busy)
- The proportion of sources/sinks of traffic is $\rho \in (0, 1]$, i.e.$\#\mathrm{sources} = \rho S$
- Each traffic source generates, on average, the same amount of traffic λ
- The packets are sent through the shortest and/or *less busy* route

Congestion in a Manhattan Network



Packet generation, hop movement, queue movement and updating of the routing table occurs at one time step.

- Consider a Manhattan-toroidal network with S nodes
- Each node contains a queue where packets can be stored in transit (if the node is busy)
- The proportion of sources/sinks of traffic is $\rho \in (0, 1]$, i.e.$\#\text{sources} = \rho S$
- Each traffic source generates, on average, the same amount of traffic λ
- The packets are sent through the shortest and/or *less busy* route
- If one node is busy (queue busy), then another route is chosen

Delay and Congestion

- τ_{sd} is the journey time from s to d

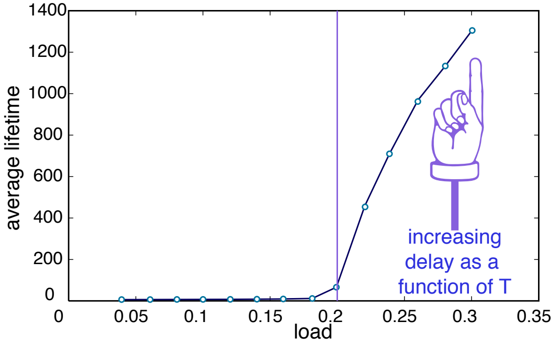
Delay and Congestion

- τ_{sd} is the journey time from s to d
- For low load, $\tau_{sd} \approx \ell_{sd}$
- For higher load, $\tau_{sd} \approx \ell_{sd}+$ delays due to the queuing.

Delay and Congestion

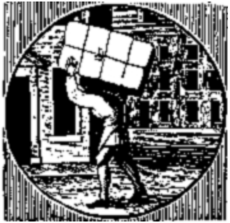
- τ_{sd} is the journey time from s to d
- For low load, $\tau_{sd} \approx \ell_{sd}$
- For higher load, $\tau_{sd} \approx \ell_{sd} +$ delays due to the queuing.
- If the traffic load increases even further, then at the critical load λ_c, the queues of some nodes will grow unbounded and the average delay time diverge.
- At this critical load, we consider that the network is congested.

Delay and Congestion



$T \Rightarrow$ running time of simulation

Total Number of Packets in the System and Congestion



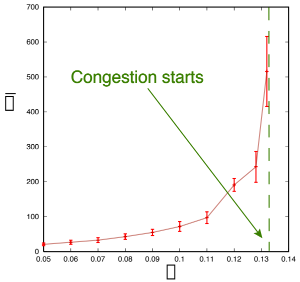
- Total number of packets:

$$N(t) = \sum_{i=1}^{S} Q_i(t)$$

$Q_i(t)$ is size of queue i.

- At the free flow state, $\bar{N} = \lim_{T-\infty} 1/T \sum_T N(T)$ is finite

Total Number of Packets in the System and Congestion



- Total number of packets:

$$N(t) = \sum_{i=1}^{S} Q_i(t)$$

$Q_i(t)$ is size of queue i.

- At the free flow state,
  $\bar{N} = \lim_{T-\infty} 1/T \sum_T N(T)$ is finite
- At the congestion point, the queues of the congested nodes start growing unbounded $\Rightarrow \bar{N} \to \infty$

Definitions and Assumptions

- The network is represented by the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of nodes (vertices) and $\mathcal{E}$ is the set of links (edges).
- The total number of nodes is denoted by S.
- The graph is undirected and connected

Definitions and Assumptions

- The network is represented by the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of nodes (vertices) and $\mathcal{E}$ is the set of links (edges).
- The total number of nodes is denoted by S.
- The graph is undirected and connected
- The minimum *distance* between vertices $s \in \mathcal{V}$ and $d \in \mathcal{V}$ is denoted by ℓ_{sd} (shortest path between s and d.)
- The characteristic path length

$$\bar{\ell} = \frac{1}{S(S-1)} \sum_{v \in \mathcal{V}} \sum_{d \in \mathcal{V} \setminus v} \ell_{sd}$$

(sometimes $\bar{\ell}$ is referred as the diameter of the network).

Mean Field Approximation

Little's Law

"The average number of customers in a queuing system is equal to the average arrival rate of customers to that system, times the average time spent in the system", Kleinrock 1975

Formulation

$$\frac{d\,N(t)}{d\,t} = \rho S \lambda - \frac{N(t)}{\tau(t)}.$$

Mean Field Approximation

Little's Law

"The average number of customers in a queuing system is equal to the average arrival rate of customers to that system, times the average time spent in the system", Kleinrock 1975

Formulation

$$\frac{d\,N(t)}{d\,t} = \rho S \lambda - \frac{N(t)}{\tau(t)}.$$

- $\rho S \lambda$ is the average arrival rate to the queues per unit of time,

Mean Field Approximation

Little's Law

"The average number of customers in a queuing system is equal to the average arrival rate of customers to that system, times the average time spent in the system", Kleinrock 1975

Formulation

$$\frac{d\,N(t)}{d\,t} = \rho S \lambda - \frac{N(t)}{\tau(t)}.$$

- $\rho S \lambda$ is the average arrival rate to the queues per unit of time,
- $\tau(t)$ is the average time spent in the system, and

Mean Field Approximation

Little's Law

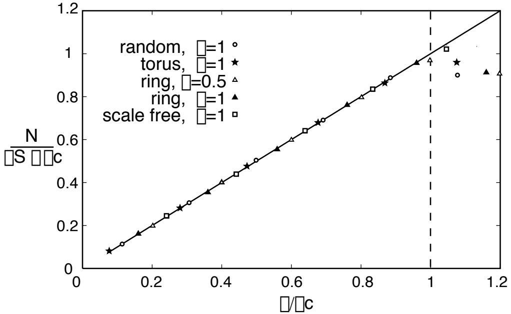
"The average number of customers in a queuing system is equal to the average arrival rate of customers to that system, times the average time spent in the system", Kleinrock 1975

Formulation

$$\frac{d\,N(t)}{d\,t} = \rho S \lambda - \frac{N(t)}{\tau(t)}.$$

- $\rho S \lambda$ is the average arrival rate to the queues per unit of time,
- $\tau(t)$ is the average time spent in the system, and
- $N(t)/\tau(t)$ is the number of packets delivered per unit of time.

Mean Field Approximation. Little's Law



⇐ From the steady state solution

$$\rho S\lambda - \frac{N}{\tau} = 0$$

Little's law does not depends on

- the arrival distribution of packets to the queue
- or the service time distribution of the queues.
- Also it does not depends upon the number of queues in the system or upon the queuing discipline within the system.

Mean Field Approximation. Congestion

Estimating the time delay

- If the load is low, the delay is time is given by the length of the shortest path, then the average delay is the average of the shortest paths $\bar{\tau} \approx \bar{\ell}$.
- If the load is high, the delay time is the length of the shortest path plus the time a packet spends on the queues

Mean Field Approximation. Congestion

Estimating the time delay

- If the load is low, the delay is time is given by the length of the shortest path, then the average delay is the average of the shortest paths $\bar{\tau} \approx \bar{\ell}$.
- If the load is high, the delay time is the length of the shortest path plus the time a packet spends on the queues
- if we assume that, on average each queue contains $\bar{N}/S$ packets

$$\tau(t) \approx \bar{\tau} \approx \bar{\ell}(1+\bar{Q}) = \bar{\ell}\left(1+\frac{\bar{N}}{S}\right)$$

Mean Field Approximation. Congestion

Estimating the time delay

- If the load is low, the delay is time is given by the length of the shortest path, then the average delay is the average of the shortest paths $\bar{\tau} \approx \bar{\ell}$.
- If the load is high, the delay time is the length of the shortest path plus the time a packet spends on the queues
- if we assume that, on average each queue contains $\bar{N}/S$ packets

$$\tau(t) \approx \bar{\tau} \approx \bar{\ell}(1 + \bar{Q}) = \bar{\ell}\left(1 + \frac{\bar{N}}{S}\right)$$

- ! we are approximating the delay time with the queue's average size

QUEEN MARY
UNIVERSITY OF LONDON

Mean Field Approximation. Congestion

Estimating the critical load λ_c

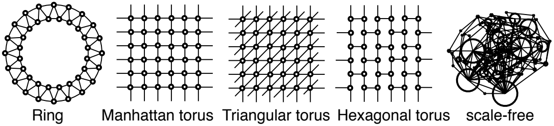
- From the steady state solution $dN(t)/dt = 0$ the traffic load generated is

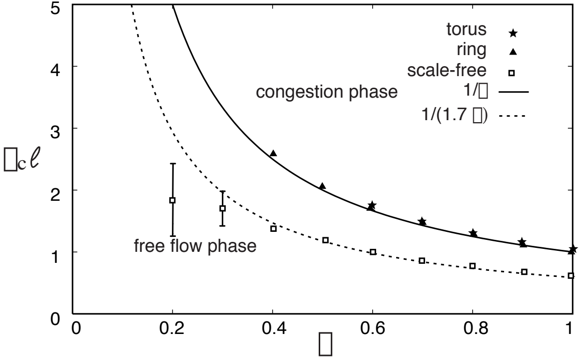
$$\lambda = \frac{1}{\rho\bar{\ell}(1 + S/\bar{N})}$$

- At the congestion point the average number of packets diverges, i.e. $\bar{N} \to \infty$ so the critical load is

$$\lambda_c = \frac{1}{\rho\bar{\ell}}$$

Mean Field Approximation. Congestion





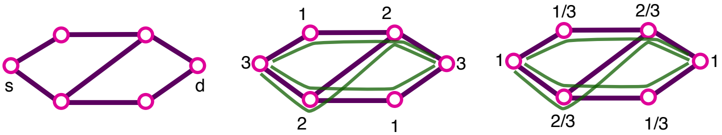
Betweenness Centrality

- Consider the journey time between two nodes in the network where there is at least two shortest paths between the nodes
- The journey time of two shortest paths with the same length can be very different due to the different patterns of usage of the routes
- The reason is that some nodes are more “prominent” because they are highly used when transferring packet-data.
- A way to measure this “importance” is by using the concept of *node betweenness centrality* (also called *load* or just *betweenness*).

Centrality: Definitions

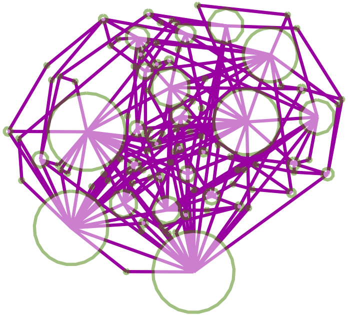
- The number of shortest paths from $s \in \mathcal{V}$ to $d \in \mathcal{V}$ is denoted by σ_{sd}
- The number of shortest paths from s to d that some $v \in \mathcal{V}$ lies on is denoted by $\sigma_{sd}(v)$
- The *pair-dependency* of a pair $s, d \in \mathcal{V}$ on an intermediary $v \in \mathcal{V}$ is

$$\delta_{sd}(v) = \frac{\sigma_{sd}(v)}{\sigma_{sd}}$$



Betweenness/load/Betweenness Centrality

$$\mathcal{C}_B(v) = \sum_{s \in \mathcal{V}} \sum_{d \in \mathcal{V} \setminus s} \delta_{sd}(v), \quad v \in \mathcal{V}$$



A small modification

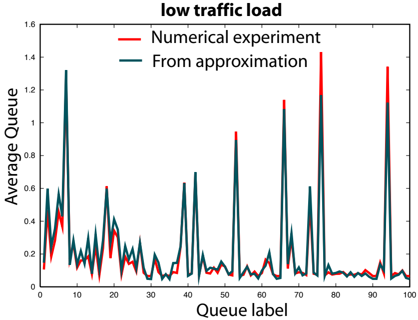
$$C_B(v) = \mathcal{C}_B(v) - 1$$

An improvement to τ

- Characterise the node usage using the normalised betweenness centrality

$$\hat{C}_B(w) = \frac{C_B(w)}{\sum_{v \in \mathcal{V}} C_B(v)}$$

- approximate the average queue size using $\bar{Q}_w \approx \hat{C}_B(w)\bar{N}$

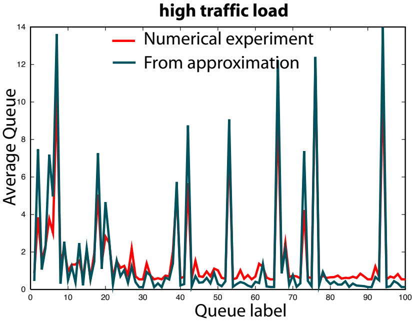


An improvement to τ

- Characterise the node usage using the normalised betweenness centrality

$$\hat{C}_B(w) = \frac{C_B(w)}{\sum_{v \in \mathcal{V}} C_B(v)}$$

- approximate the average queue size using $\bar{Q}_w \approx \hat{C}_B(w)\bar{N}$



An improvement to τ

- the approximation to the average delay time is

$$\bar{\tau} \approx \bar{\ell} + \frac{1}{S(S-1)} \sum_{s \in V} \sum_{d \in \mathcal{V} \setminus v} \left(\sum_{v \in \mathcal{R}_{sd}} \hat{C}_B(v) \bar{N} \right) = \bar{\ell} + D\bar{N}$$

where $v \in \mathcal{R}_{sd}$ is the set of nodes visited by the route

- ! we are taking the average of averages

An improvement to τ

- the approximation to the average delay time is

$$\bar{\tau} \approx \bar{\ell} + \frac{1}{S(S-1)} \sum_{s \in V} \sum_{d \in \mathcal{V} \setminus v} \left(\sum_{v \in \mathcal{R}_{sd}} \hat{C}_B(v) \bar{N} \right) = \bar{\ell} + D\bar{N}$$

where $v \in \mathcal{R}_{sd}$ is the set of nodes visited by the route

- ! we are taking the average of averages
- Using the new approximation to $\bar{\tau}$

$$\lambda_c = \frac{1}{\rho S\, D}$$

An improvement to λ_c

- The equation

$$\lambda_c = \frac{1}{\rho S\, D}$$

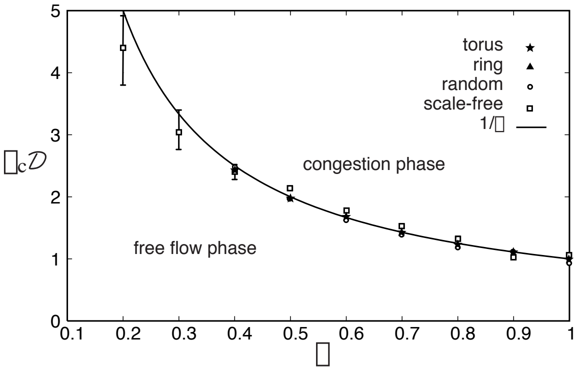
- simplifies to $\lambda_c = 1/(\rho\bar{\ell})$ in the case of regular networks.
- this is obtained by exploiting the property that

$$\ell_{sd} = \sum_{w \in \mathcal{W}} \delta_{sd}(w) - 1$$

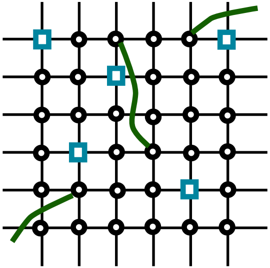
where $\mathcal{W}$ is the set of nodes visited by the shortest paths from s to d.

QUEEN MARY
UNIVERSITY OF LONDON

Betweenness Centrality



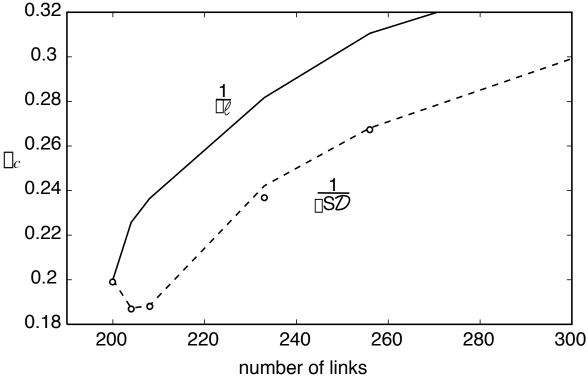
Fukś and Lawcnizack Observation



- Take a Manhattan toroidal network
- add a few new random links
- the onset of congestion occurs more readily when adding these new links that in the original network
- this is because, the new links “attract” traffic, the nodes containing the extra links congest more easily

Fukś and Lawcnizack Observation

- The original network has 200 links
- we compare the prediction using $\lambda_c = 1/(\rho\ell)$ and $\lambda_c = 1/(\rho SD)$



- Similarities with Braess' Paradox?

Another improvement

The queue discipline is M/D/1

- The average of the queue is approximated by

$$\bar{Q}_i = \Lambda_i + \frac{\Lambda_i^2}{2(1-\Lambda_i)} \approx \bar{N} D_i$$

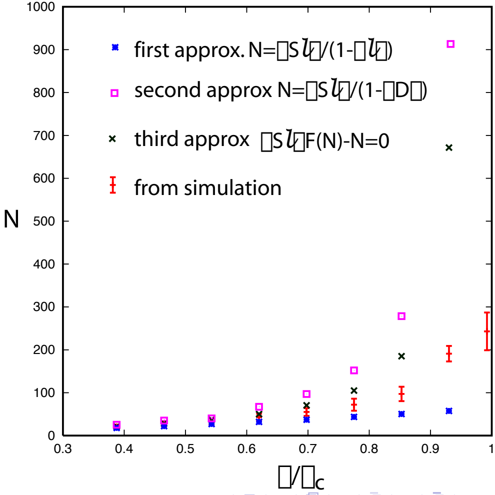
- The average delay is

$$\tau_i = \frac{\Lambda_i}{2(1-\Lambda_i)} \approx \frac{1 + \bar{N} D_i - \sqrt{1 + (\bar{N} D_i)^2}}{2(\sqrt{1 + (\bar{N} D_i)^2} - \bar{N} D_i)}$$

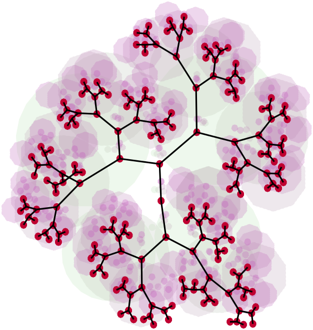
Number of Packets in the Network

Using

$$\rho S \lambda - \frac{N}{\tau} = 0$$



Limitations



- The prediction doesn't work for trees
- The delay time approximation is poor