Improving the Performance of the Gnutella Network

André Dufour
adufour@cs.sfu.ca

Communication Networks Laboratory
http://www.ensc.sfu.ca/research/cnl
School of Engineering Science
Simon Fraser University

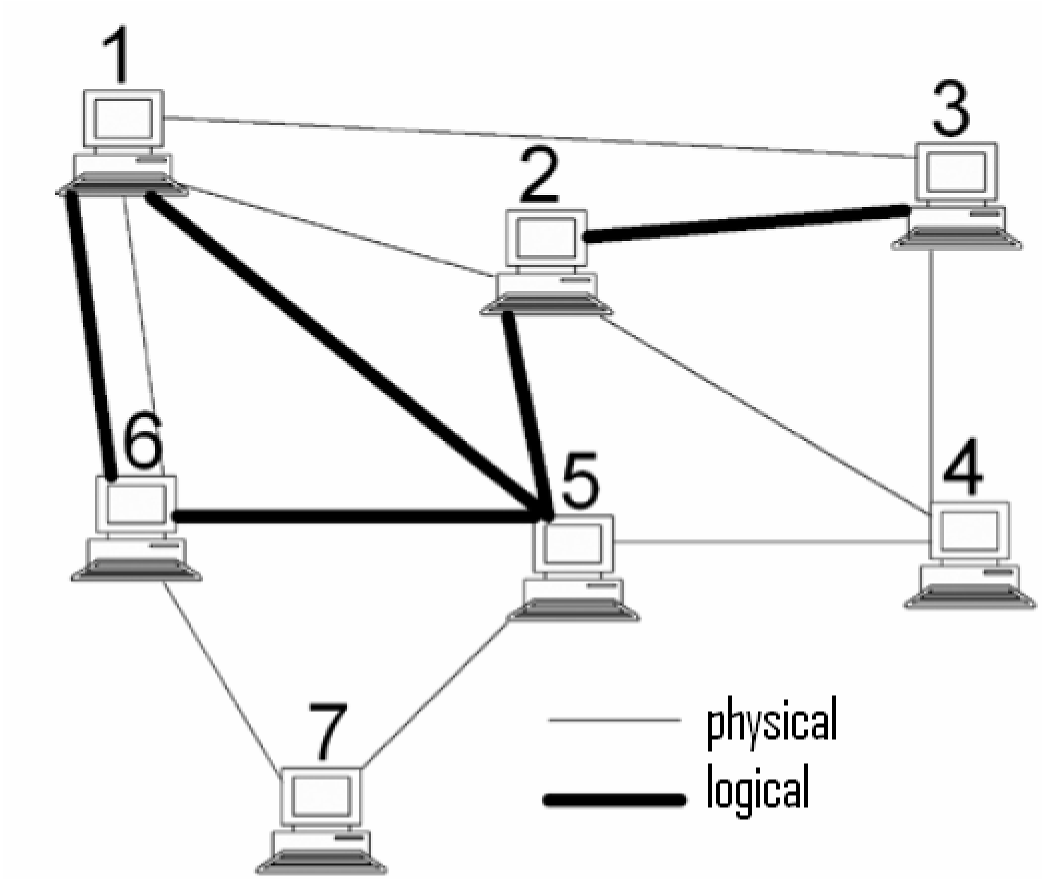
Outline

- Introduction
  - P2P networks
  - Gnutella network
- Gnutella problem: topology mismatch
- Proposal:
  - Vivaldi coordinate system
  - neighbour selection algorithm
  - GnutellaSim and Gnutaldi simulators
- Simulation scenarios
- Simulation results
- Conclusions
- References

Outline

- Introduction
  - P2P networks
  - Gnutella network
- Gnutella problem: topology mismatch
- Proposal:
  - Vivaldi coordinate system
  - neighbour selection algorithm
  - GnutellaSim and Gnutaldi simulators
- Simulation scenarios
- Simulation results
- Conclusions
- References

P2P as an overlay network



P2P network properties

- No centralized point of failure
- More responsibility to nodes at the edge of the network
  - performing routing functions
  - providing content: files, human interaction, processor cycles
- Transient node presence is assumed, unlike client-server model
- Account for up to 90% of Internet traffic

P2P applications

- File sharing: Gnutella, Kazaa, BitTorrent
- Online gaming and chatting: Jabbers, Skype
- Generic development frameworks: JXTA, Pastry, Tapestry
- Ad-Hoc networks: ad-hoc networks are P2P

The Gnutella network

- Decentralized P2P protocol
- Current version is 0.6
  - 0.7 has been proposed
- Open "standard"
- Used for file sharing
- Query messages flooded through network and limited by TTL field
- Nodes with the desired content respond with a Query Hit
- Other messages: ping, pong, push

TTL: time to live

Outline

- Introduction
  - P2P networks
  - Gnutella network
- Gnutella problem: topology mismatch
- Proposal:
  - Vivaldi coordinate system
  - neighbour selection algorithm
  - GnutellaSim and Gnutaldi simulators
- Simulation scenarios
- Simulation results
- Conclusions
- References

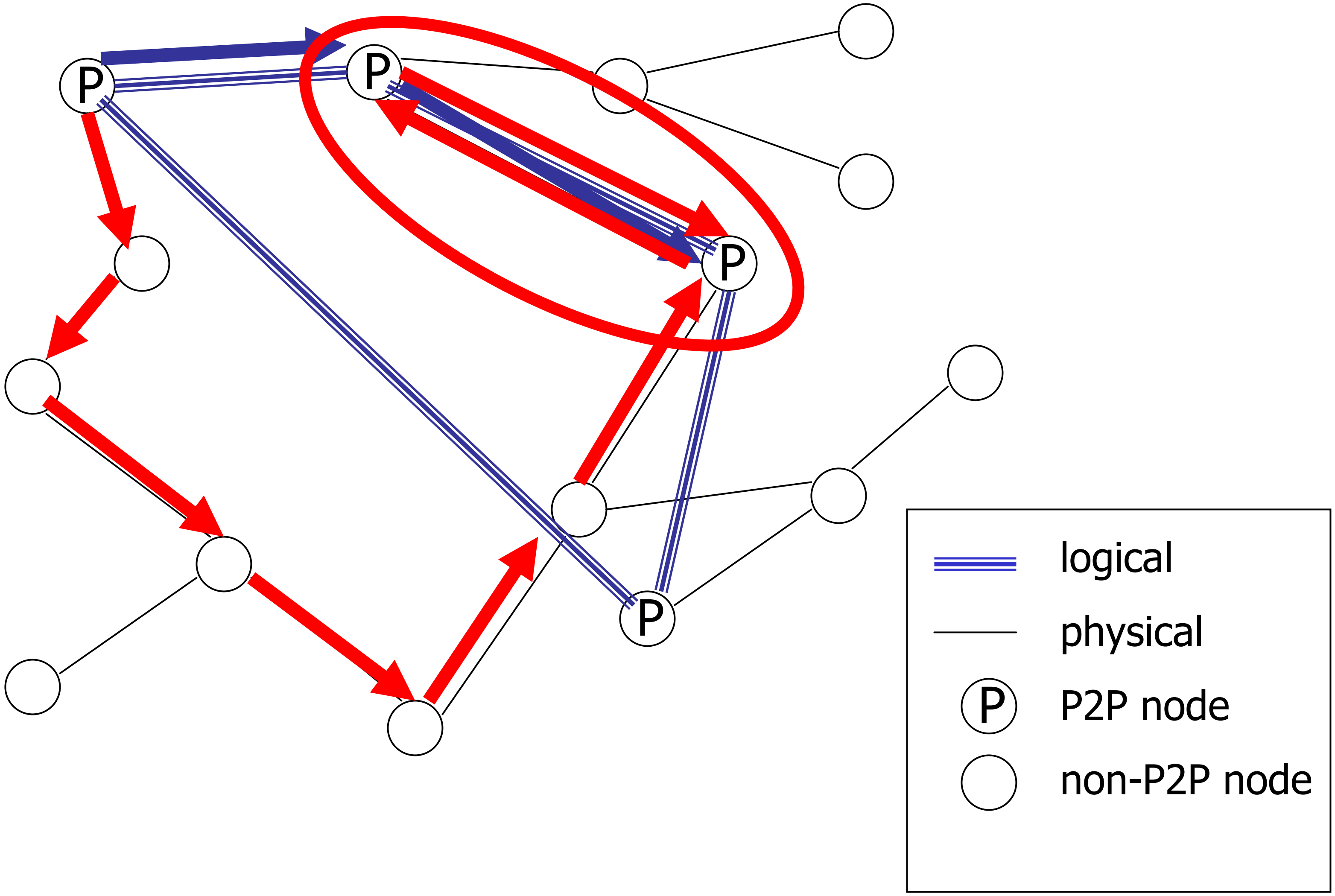
The problem: topology mismatch

- Two topologies:
  - physical: the way routers and end systems are connected in the Internet
  - logical: the neighbour relationships in the Gnutella overlay
- Virtually no correlation between the two topologies[1]
  - inefficient
- Less than 5% of Gnutella connections link nodes in the same AS
- No correlation between domain name hierarchy and Gnutella node clustering

M. Ripeanu and I. Foster, "Mapping the Gnutella network," in *Proc. 1st International Workshop on Peer-to-Peer Systems,* Cambridge, MA, Mar. 2002, pp. 85–93.

AS: Autonomous system

Inefficiency caused by mismatch



Outline

- Introduction
  - P2P networks
  - Gnutella network
- Gnutella problem: topology mismatch
- Proposal:
  - Vivaldi coordinate system
  - neighbour selection algorithm
  - GnutellaSim and Gnutaldi simulators
- Simulation scenarios
- Simulation results
- Conclusions
- References

Towards a better topology…

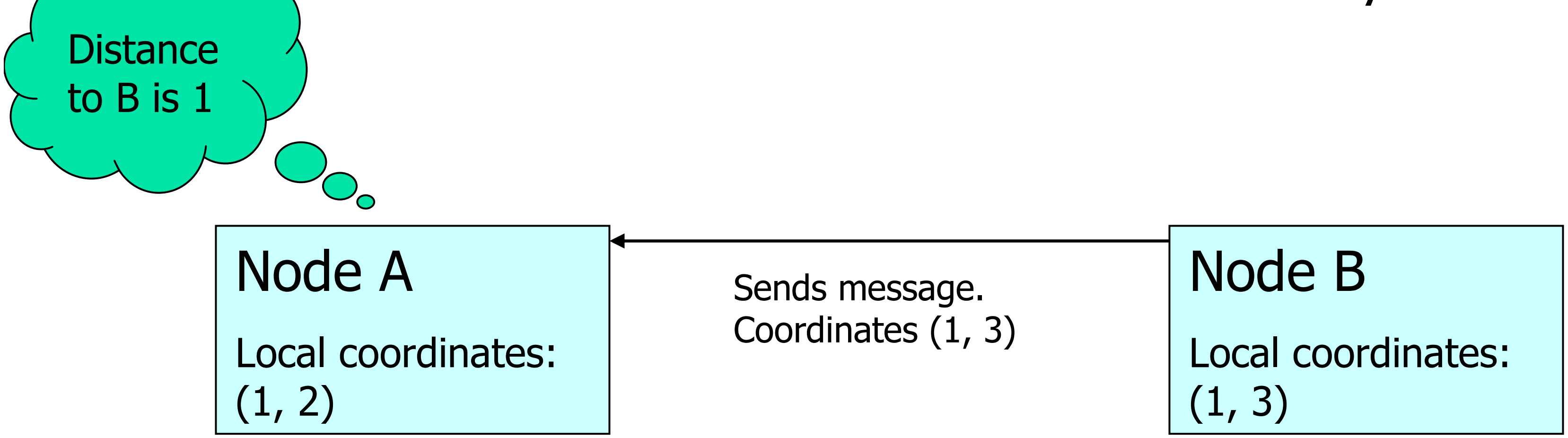
- Bias selection of neighbours to favour nodes that are physically close:
  - utilize network resources more efficiently
  - improve response time to queries
    - better QoE for users
- Unfeasible to directly measure the RTT to each potential neighbour directly, say with ping requests
- When a node receives a message saying that connections are available, it must be able to decide whether or not to connect

QoE: Quality of experience

RTT: Round trip time

Internet coordinate systems

- Assign coordinates to nodes in the network
- Coordinates are exchanged along with messages
- A node knows its coordinates
  - it can calculate the distance to any node that sends it a message
- We decided to use the Vivaldi coordinate system



The Vivaldi coordinate system

- Assigns coordinates without using a fixed infrastructure or landmark nodes
- Euclidian distance predicts latency between two hosts – median relative error of 14%
- Proposed without reference to specific applications
- Used by the Chord P2P network for performing lookups, but not in Gnutella and not in the formation of the overlay
- Uses an algorithm based on a network of physical springs

Landmark: node assumed to be always available, such as a DNS server or other stable entity. Sometimes called beacon.

F. Dabek, R. Cox, F. Kaashoek, and R. Morris, "Vivaldi: a decentralized network coordinate system," in *Proc. SIGCOMM'04*, Portland, OR, Aug. 2004, pp. 15–26.

Springs and error minimization

- Model a spring between each pair of nodes
- Length of the spring: distance between the nodes given their current coordinates
- Potential energy of a spring: error of the system – square of its displacement from its rest position
- Minimizing potential energy of spring system: minimizing the error in the coordinate system
  - most accurate estimates of latency

Centralized Vivaldi algorithm

- Let *M* be a matrix of latencies and M_{xy} be the latency between nodes *x* and *y*
- Use a simple squared error function (Hooke's law):

$$E = \sum_x \sum_y \left(M_{xy} - dist(x, y)\right)^2$$

- … and minimize *E*
- However, P2P networks do not lend themselves to centralized algorithms

Vivaldi: distributed calculation

- Whenever two nodes communicate, they
  - measure the latency between them
  - include their coordinates in the message
- With each measurement, nodes adjust their coordinates to reduce the error
  - a node moves its coordinates towards a point p, at which the difference between the measured and predicted latency is zero
  - it only moves a fraction δ of the way towards p, to avoid oscillation

Pseudocode

vivaldi(rtt, x_j, e_j)

(1) Calculate the sample weight, which balances the local and remote error

$$w = \frac{e_i}{e_i + e_j}$$

(2) Compute the relative error of this sample

$$e_s = \frac{\left| \left\| x_i - x_j \right\| - rtt \right|}{rtt}$$

F. Dabek, R. Cox, F. Kaashoek, and R. Morris, “Vivaldi: a decentralized network coordinate system,” in *Proc. SIGCOMM’04*, Portland, OR, Aug. 2004, pp. 15–26.

x_i: local coordinates
x_j: remote coordinates
rtt: sampled round trip time
w: weight of current sample
e_i: local error
e_j: remote error
e_s: error on this sample

Pseudocode (2)

```
(3) Update the weighted moving average of the
   local error
```

$$e_i = e_s \times c_e \times w + e_i \times (1 - c_e \times w)$$

```
(4) Update local coordinates
```

$$\delta = c_c \times w$$

$$x_i = x_i + \delta \times \left(rtt - \left\| x_i - x_j \right\|\right) \times u(x_i - x_j)$$

δ: timestep
c_e: tuning parameter (0.25)
c_c: tuning parameter (0.25)
$U(x_i - x_j)$: unit vector from x_i to x_j

Initial conditions and special cases

- If two nodes have the same coordinates, they move in a random direction
  - especially when the network is young
- Coordinates are initialized to the origin
- Initial error value is an arbitrary large integer because nodes have no idea where they are initially

Proposed Neighbour Selection Algorithm

- (Key contribution)
- With every Gnutella message, include a tuple:
  coord, error, time
  - used by remote node to update its coordinates
- When a connection request is received:
  - if there is room for another connection, accept it (as in regular Gnutella)
  - if there is no room for new connections, consider dropping connections to accept the new one using connection selection algorithm

Connection selection

- Estimate the distance between the two nodes
  - the difference between the two nodes' coordinates
- If this distance is less than the distance to any existing connection, drop the worst existing connection and accept the new one
- Estimated distances are updated with each message exchange, since all messages bear the Vivaldi coordinates

GnutellaSim simulator

- Developed Georgia Tech in 2003
- Simulates the Gnutella 0.6 network
  - two-tiered hierarchy Gnutella nodes
  - support for version 0.4 peers
- Uses the so-called UMASS model for peer behaviour
- Based on the ns-2 simulator

Q. He, M. Ammar, G. Riley, H. Raj, and R. Fujimoto, "Mapping peer behavior to packet-level-details: a framework for packet-level simulation of peer-to-peer systems," in *Proc. MASCOTS'03*, Orlando, FL, Oct. 2003, pp. 71–78.

UMASS model

- Developed at the University of Massachusetts
- Model for peer behaviour in P2P networks
- Classes of peers described by:
  - average time they are offline
  - average time they are idle (not sending queries)
  - probability of going offline after a successful query
  - whether they share content or not (freeloaders)
  - number of files they share

Z. Ge, D. Figueiredo, S. Jaiswal, J. Kurose, and D. Towsley "Modeling peer-peer file sharing systems," in *Proc. INFOCOM'03*, San Francisco, CA, Apr. 2003, pp. 2188–2198.

Modifications to GnutellaSim

- Modified simulator is called Gnutaldi (Gnutella + Vivaldi)
- Extensive rewriting of the code:
  - performance: much unused state information is stored and circulated in messages
  - maintainability: instead of a single source file with thousands of lines, dozens of separate, decoupled files
  - modularity: separate functionally distinct modules
  - correctness: state transitions of nodes did not conform to the spec in some cases
- Modified UMASS model:
  - presence of content is no longer probabilistic; deterministic behaviour for the purpose of comparison
- Addition of Vivaldi coordinates and neighbour selection algorithm

Gnutaldi high level architecture

- Protocol layer
  - largely unchanged from GnutellaSim implementation
  - encapsulates and forwards Gnutella messages
- Application layer (key contribution)
  - implements Gnutella application logic, including generation of all messages and neighbour selection algorithm
- Vivaldi classes (key contribution)
  - represent the Vivaldi coordinate system logic
- Messaging classes (key contribution)
  - represent the different messages Gnutella/Vivaldi can exchange

Topology generation

- Used BRITE tool to generate a power-law physical topology
- Exported topology to ns-2 simulator
- Barabasi-Albert model:
  - incremental growth and preferential connectivity
  - at each step, add m nodes
  - probability of a new node connecting to a particular existing node proportional to that node's connectivity
- "Rich get richer" connectivity: popular nodes are most likely to receive new connections
- 92-node topologies includes 42 Gnutella servents

Outline

- Introduction
  - P2P networks
  - Gnutella network
- Gnutella problem: topology mismatch
- Proposal:
  - Vivaldi coordinate system
  - neighbour selection algorithm
  - GnutellaSim and Gnutaldi simulators
- Simulation scenarios
- Simulation results
- Conclusions
- References

Simulation scenario

- Distribute content randomly throughout the network
  - uniform distribution
- Simulate Gnutella operation
  - bootstrapping
  - connections
  - message exchange
  - nodes joining and leaving the network dynamically
- Every few seconds, the same set of randomly chosen nodes sends a query. Observe:
  - unique nodes the query reaches versus time
  - average time to receive query hits
- Also track the median coordinate error

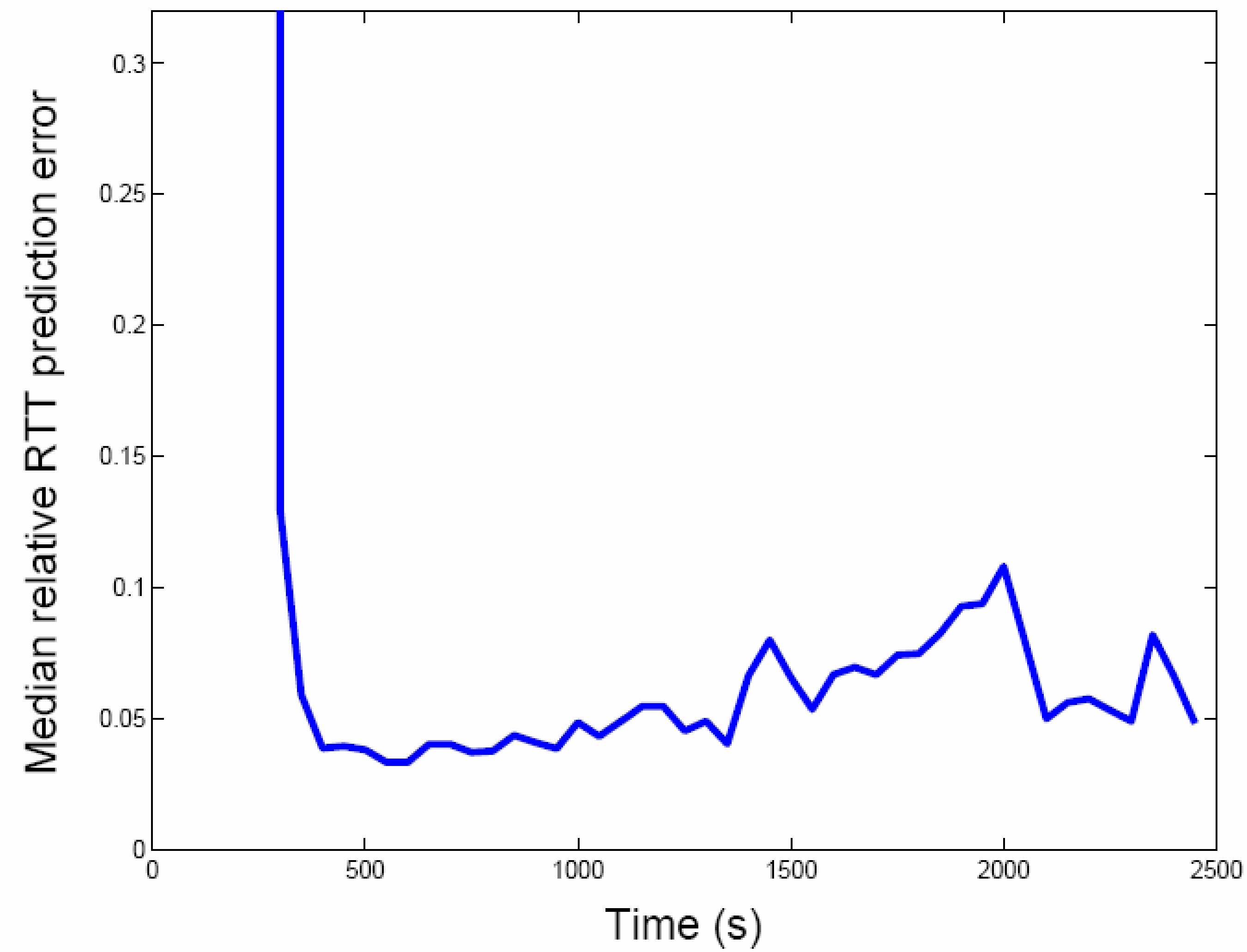
Expectation

- As time progresses
  - median error in the coordinates will diminish: the spring system will stabilize
- As compared to the original Gnutella simulation:
  - more nodes will be reached vs. time
  - mean time to get query responses will be reduced

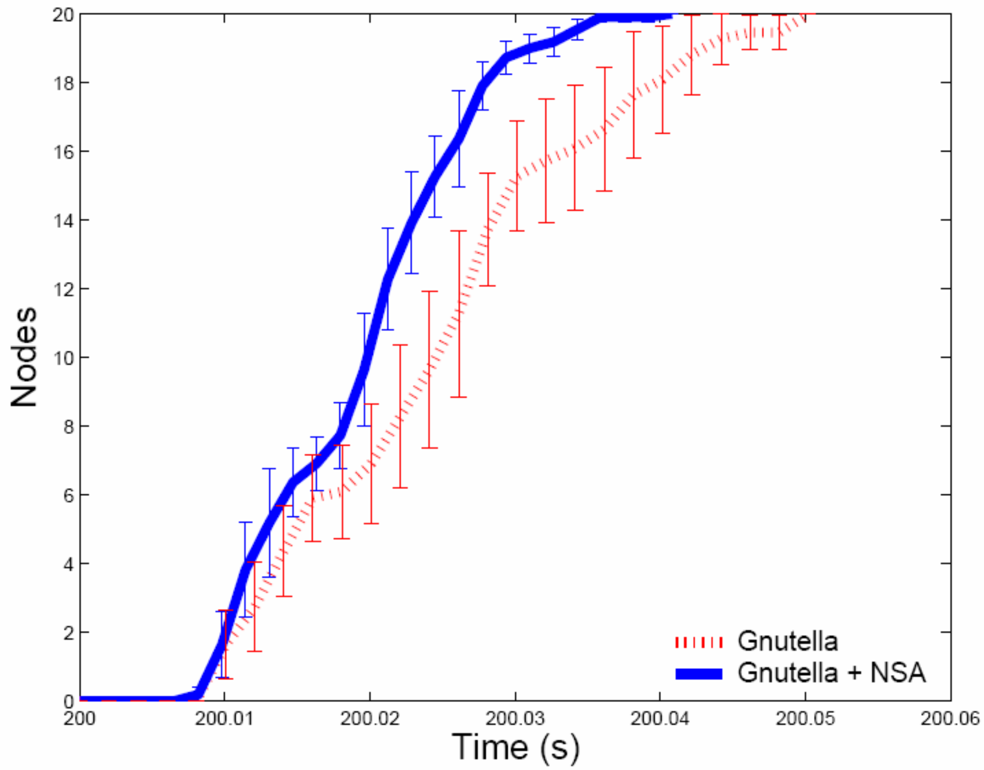
Outline

- Introduction
  - P2P networks
  - Gnutella network
- Gnutella problem: topology mismatch
- Proposal:
  - Vivaldi coordinate system
  - neighbour selection algorithm
  - GnutellaSim and Gnutaldi simulators
- Simulation scenarios
- **Simulation results**
- Conclusions
- References

Vivaldi coordinate convergence



Propagation early in the simulation



NSA: neighbour selection algorithm

- Average number of nodes reached for queries sent with 42 dynamic Gnutella servents when the median relative RTT prediction error is above 100%. Confidence interval = 95%.

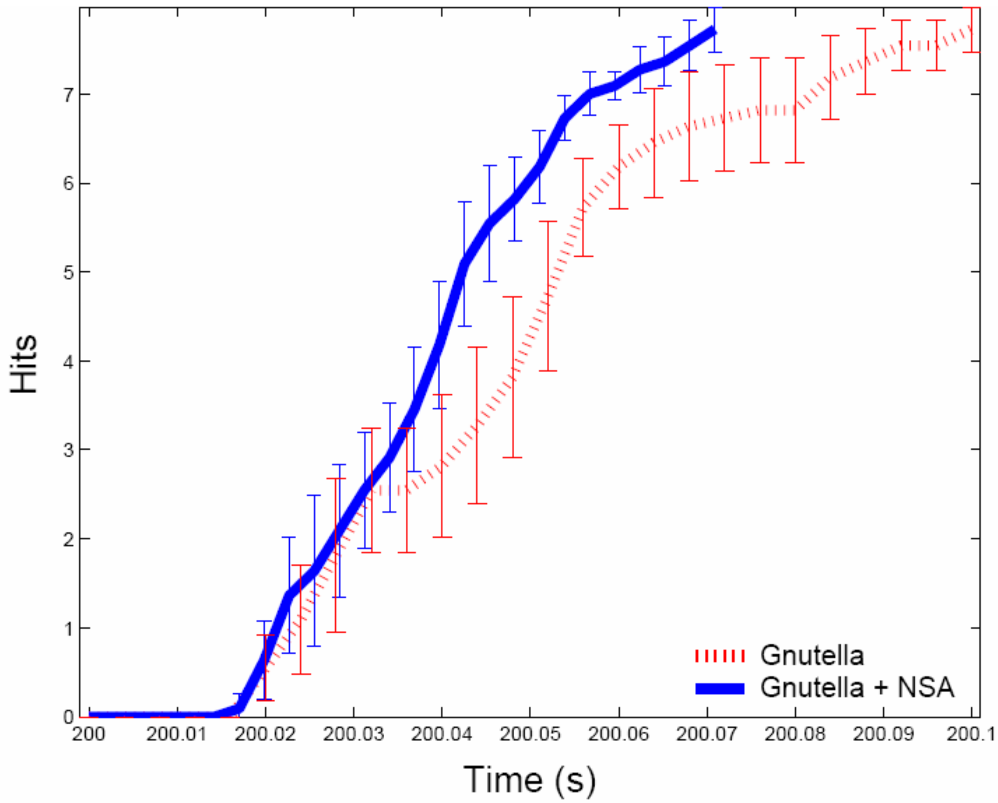
Comments

- The median RTT prediction error was above 100%, yet the performance of the neighbour selection algorithm was better than Gnutella
- The reason is that the neighbour selection algorithm increases connectivity by making it easier for new nodes to find neighbours

Connectivity at 200 Seconds

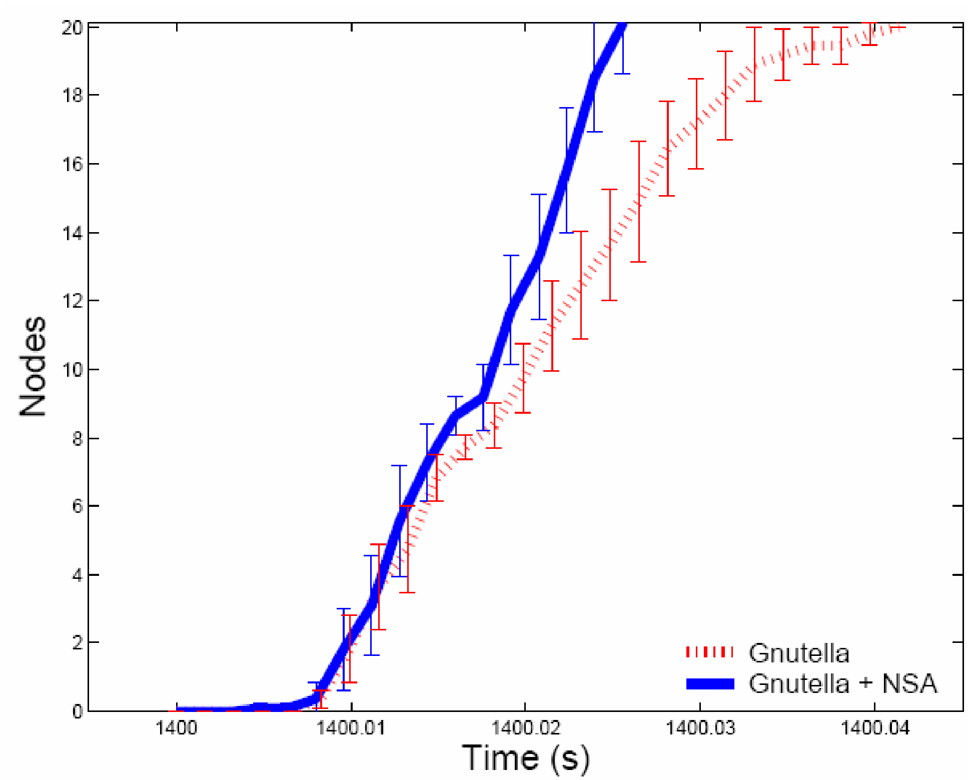
Random Seed	Average Connections per Servent		Standard Deviation	
	Gnutella	Gnutella + NSA	Gnutella	Gnutella + NSA
7	6.5714	6.8571	2.2265	1.8784
12	7.0476	7.5238	1.3593	0.8136
17	6.7619	7.3333	1.8949	0.7303

Query hit response early in the simulation



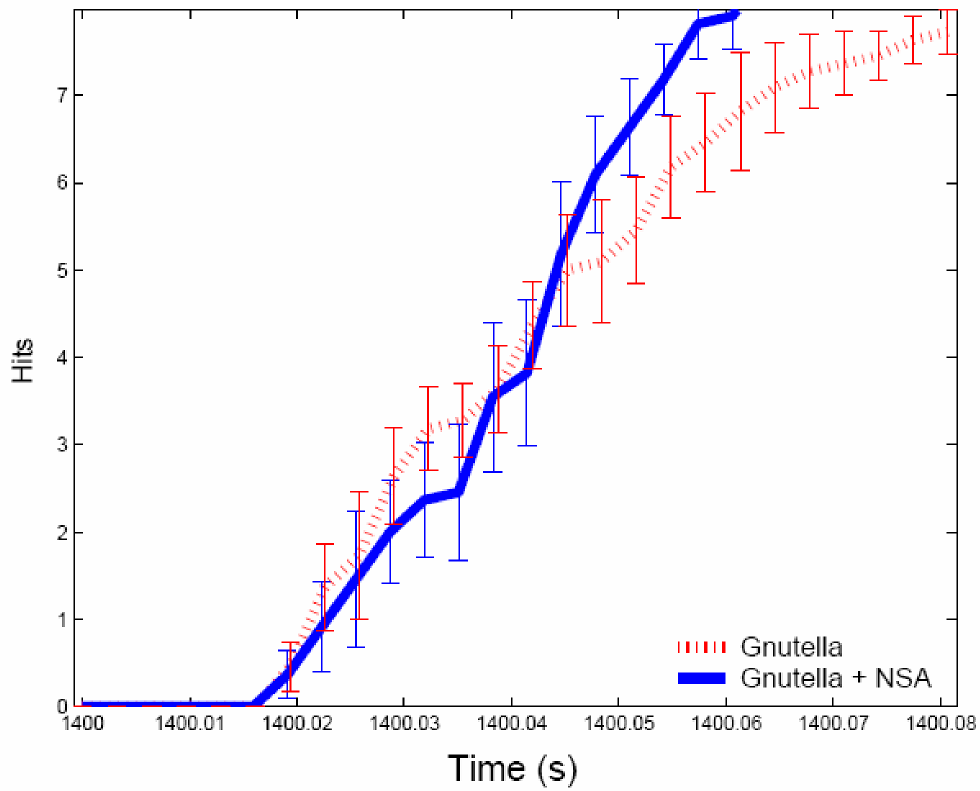
- Average number of query hits received for queries sent with 42 Gnutella servents joining and leaving the network when the median relative RTT prediction error is above 100%. Confidence interval = 95%.

Propagation later in the simulation



- Average number of nodes reached for queries sent with 42 Gnutella servents joining and leaving the network at time 1,400 s. At this instant, the network most closely resembles real-world network conditions. The median relative RTT prediction error is 0.051884. Confidence interval = 95%.

Query hit response later in the simulation



- Average number of query hits received for queries sent with 42 Gnutella servents joining and leaving the network at time 1,400 s. At this instant, the network most closely resembles real-world network conditions. The median relative RTT prediction error is 0.051884. Confidence interval = 95%.

Outline

- Introduction
  - P2P networks
  - Gnutella network
- Gnutella problem: topology mismatch
- Proposal:
  - Vivaldi coordinate system
  - neighbour selection algorithm
  - GnutellaSim and Gnutaldi simulators
- Simulation scenarios
- Simulation results
- Conclusions
- References

Conclusions

- We proposed a neighbor selection algorithm for the Gnutella peer-to-peer network to improve the performance of queries and query hits
- The algorithm employs the Vivaldi coordinate system
- The performance of the proposed algorithm, was characterized using the new network simulator Gnutaldi
- The Vivaldi coordinates converged with a low error
- Using the neighbor selection algorithm, queries reached nodes faster and query hits were returned to the originator more quickly
- The simulation results indicate that the propsed neighbour selection algorithm improves users quality of experience

References

[1] (2005) Napster: All the music you want. Any way you want it. [Online]. Available: http://www.napster.com.

[2] G. Bennett. (Dec. 30, 2003) Controlling P2P traffic. [Online]. Available: http://www.lightreading.com/document.asp?doc id=44435

[3] V. Pande, "Folding@home: advances in biophysics and biomedicine from world-wide grid computing," in Proc. 19th IEEE International Parallel and Distributed Processing Symposium, Denver, CO, Apr. 2005, pp. 101–107.

[4] E. Korpela, D. Werthimer, D. Anderson, J. Cobb, and M. Leboisky, "Seti@home-massively distributed computing for seti," Computing in Science & Engineering, vol. 3, no. 1, pp. 78–83, Jan. 2001.

[5] (2005) Icq.com–community, people search and messaging service! [Online]. Available: http://www.icq.com/.

[6] A. El Saddik and A. Dufour, "Peer-to-peer suitability for collaborative multiplayer games," in Proc. Seventh IEEE International Symposium on Distributed Simulation and Real Time Applications, Delft, Netherlands, Oct. 2003, pp. 101–107.

[7] A. El Saddik and A. Dufour, "Peer-to-peer communication through the design and implementation of xiangqi," in Proc. International Conference on Parallel and Distributed Computing, Klagenfurt, Austria, Aug. 2003, pp. 1309–1313.

References

[8] (2003) The annotated Gnutella protocol specification v0.4. [Online]. Available: http://rfc-gnutella.sourceforge.net/developer/stable/index.html

[9] M. Ripeanu and I. Foster, "Mapping the Gnutella network," in *Proc. 1st International Workshop on Peer-to-Peer Systems*, Cambridge, MA, Mar. 2002, pp. 85–93.

[10] F. Dabek, R. Cox, F. Kaashoek, and R. Morris, "Vivaldi: a decentralized network coordinate system," in *Proc. SIGCOMM'04*, Portland, OR, Aug. 2004, pp. 15–26.

[11] Q. He. (2005) Packet-level peer-to-peer (Gnutella) simulation. [Online]. Available: http://www.cc.gatech.edu/computing/compass/gnutella/usage.html.

[12] S. Jiang and X. Zhang, "FloodTrail: an Efficient File Search Technique in Unstructured Peer-to-Peer Systems," in Proc. IEEE GLOBECOM'03, San Francisco, CA, Dec. 2003, pp. 2891–2895.

[13] M.W. Jiang, M.M. Morgado Costa, J.M.Almeida, and V.A.F. Almeida, "Using Locality of Reference to Improve Performance of Peer-to-Peer Applications," in Proc. ACM WOSP'04, Redwood City, CA, Jan. 2004, pp. 216–227.

[14] I. Stoica, R. Morris, D. Liben-Nowell, D. Karger, M. K. F. Dabek, and H. Balakrishnan, "Chord: a scalable peer-to-peer lookup protocol for Internet applications," IEEE/ACM Transactions on Networking, vol. 11, no. 1, pp. 17–32, Feb. 2003.

[15] T. Ng and H. Zhang, "Predicting Internet network distance with coordinates-based approaches," in *Proc. IEEE INFOCOM'02*, New York, NY, June 2002, pp. 170–179.

References

[16] T. Ng and H. Zhang, "A network positioning system for the Internet," in *Proc. of the 1st USENIX Symposium on Networked Systems Design and Implementation*, San Francisco, CA, Mar. 2004, pp. 141–154.

[17] M. Pias, J. Crowcroft, S. Wilbur, T. Harris, and S. Bhatti, "Lighthouses for scalable distributed location," in *Proc. 2nd International Workshop on Peer-to-Peer Systems*, Berkeley, CA, Feb. 2003. pp. 278–291.

[18] M. Costa, M. Castro, A. Rowstron, and P. Key, "Pic: practical Internet coordinates for distance estimation," in *Proc. 24th International Conference on Distributed Computing Systems*, Tokyo, Japan, Mar. 2004, pp.178–187.

[19] (2005) Mechanics of materials: Hooke's law. [Online]. Available: http://www.efunda.com/formulae/solid mechanics/mat mechanics/hooke.cfm

[20] K. Gummadi, S. Saroiu, and S. King, "Estimating latency between arbitrary Internet end hosts," in *Proc. SIGCOMM'02*, Pittsburgh, PA, Nov. 2002, pp. 5–18.

[21] T. Klingberg and R. Manfredi. (2002) Gnutella protocol development. [Online]. Available: http://rfc-gnutella.sourceforge.net/src/rfc-0 6-draft.html.

[22] Q. He, M. Ammar, G. Riley, H. Raj, and R. Fujimoto, "Mapping peer behavior to packet-level details: a framework for packet-level simulation for peer-to-peer systems," in Proc. 11th IEEE/ACM International Symposium on MASCOTS, Orlando, FL, Oct. 2003, pp. 71–78.

References

[23] A. Medina, I. Matta, and J. Byers, “BRITE: a flexible generator of Internet topologies,” in Proc. MASCOTS, Cincinnati, OH, Aug. 2001,pp. 346–353.

[24] A. Barabasi, R. Albert, and H. Jeon, “Scale-free characteristics of random networks: the topology of the World Wide Web,” *Physica A*, vol. 281, no. 1-4, pp. 69–77, June 2000.

[25] A. Dufour and Lj. Trajkovic, “Using synthetic coordinates to improve the performance of the Gnutella network,” in Proc. The Third International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks, Waterloo, Canada, Aug. 2006, to be published.