On Minimizing Packet Loss Rate and Delay for Mesh-based P2P Streaming Services

Zhiyong Liu, CATR

Prof. Zhili Sun, UniS

Dr. Dan He, UniS

Denian Shi, CATR

Agenda

- Introduction
- Background
- Problem Statement and Algorithm Design
- System Design
- Performance Evaluation
- Conclusion and Future Work

Agenda

- *Introduction*
- Background
- Problem Statement and Algorithm Design
- System Design
- Performance Evaluation
- Conclusion and Future Work

Introduction of P2P Streaming

- **Streaming: providing unique experience which enables user to enjoy playback without downloading the entire media file.**
  - **Rapid development due to the increasing availability of bandwidth resource and the technology advances in video coding;**
  - **generate nearly 91% of the global Internet traffic by 2015;**

- **Two type of typical delivery architecture:**
  - **CDN:**
    - client-server based paradigm;
    - Stream content from the closest cache node to end users;
    - Delivers reasonable Quality of Experience (QoE), but deploying this service is costly.

Introduction of P2P Streaming_contd.

- **Two type of typical delivery architecture:**
  - **P2P overlay:**
    - Enables one peer cache and relay content to other peers in the meanwhile downloading the content;
    - Form an overlay without investing on costly and complex CDN network;
    - A lot of solutions identified: PPLive, PPStream, SopCast……
    - Advantages: effectiveness and scalability;
    - Shortcomings:
      - Unacceptable QoE: bad video quality, long playback latency and startup delay when compared with CDN based ones;
      - Generated great traffic on the underlying network, causing congestions
    - Why ???
      - *The network-oblivious policy in constructing P2P overlay*

Issues should be considered

- **Selection of the best peers**
  - **A judicious neighbor peer selection policy which takes into account:**
    - Peers heterogeneity in term of computation and uploading bandwidth
    - Adapting to high churns (because peer may leave/join at any time);
  - **Constructing an efficient overlay which matches the topology of underlying backbone;**
  - **Continuous monitoring of video quality and network status**
    - Timely switching of neighbor peers when video quality seriously degrades due to sender or network fails;

Current Limitations

- **Few considerations on underlying networking issues:**
  - **Peer location;**
  - **Dynamic network status;**
  - **Uploading bandwidth of candidate neighbors;**
- **No video quality monitoring and performance metering;**
- **No dynamic switching policy to adapt to real-time network conditions;**

Our Solutions and Contributions

- ***Building a novel QoS-aware mesh system which minimizes maximum packet loss rate for all participating peers, while the cumulated delay of the most-far-end peer from source is upper bounded.***
- **Contributions:**
  - **Model the problem of constructing a Minimum Packet Loss Mesh (MPLM) network with bounds on delay**
    - NP-complete;
    - To the best of our knowledge, no technique has been unveiled to address the combined requirement on both packet loss and delay up to now.
  - **Optimization of video quality by:**
    - Rational selection of best peers with least possible packet loss and delay;
    - Maintaining high performance overlay topology;

Our Solutions and Contributions_contd.

- **Contributions:**
  - **Adopt a self-adaptive continuous mesh evolvement method to adapt to the high dynamic P2P environment.**
    - Continuous passive video quality measurement and active probing;
    - Self-adaptive evolvement when video quality degrades;
  - **Simulations show**
    - Overall performance advantages over existing scheme (proposed in the paper of IEEE Trans. on Multimedia 2009)

Agenda

- Introduction
- *Background*
- Problem Statement and Algorithm Design
- System Design
- Performance Evaluation
- Conclusion and Future Work

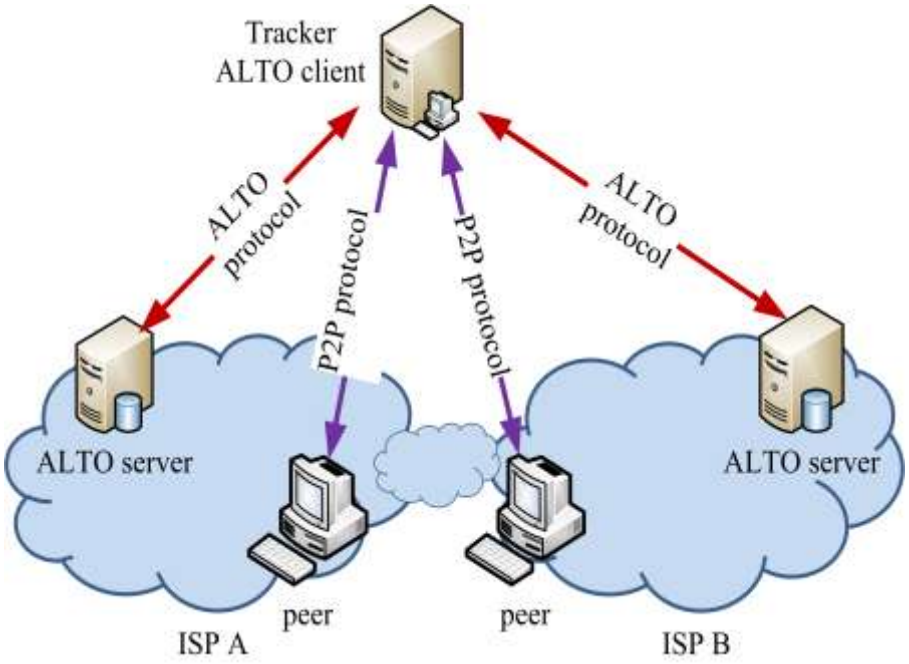
Topology Construction for P2P Streaming

- **Aim to: delivering better streaming quality, and alleviating the serious mismatching problem between P2P overlay and the underlying network;**
- **Typical methods:**
  - **P2P infers and gathers information of the underlying network to derive biased peer selection and better P2P topology, methods include:**
    - PROMISE
    - Taming the Torrent
    - ……
  - **Tree based topology: single-tree or multi-tree;**
  - **P2P and ISP cooperate to build efficient overlay:**
    - ORACLE
    - P4P
    - Using ALTO information
  - **Minimizing mesh delay**

ALTO related Topics

- **Application-Layer Traffic Optimization (ALTO) work item under IETF**

- **Investigating the network information sharing service provided by ISP or third party to applications**
  - **For P2P. perform better than random peer selection and construct rational topology that is correctly mapped on top of the transport network structure.**
  - **For ISP, minimize the effect of P2P traffic on underlying network after P2P topology be optimized**

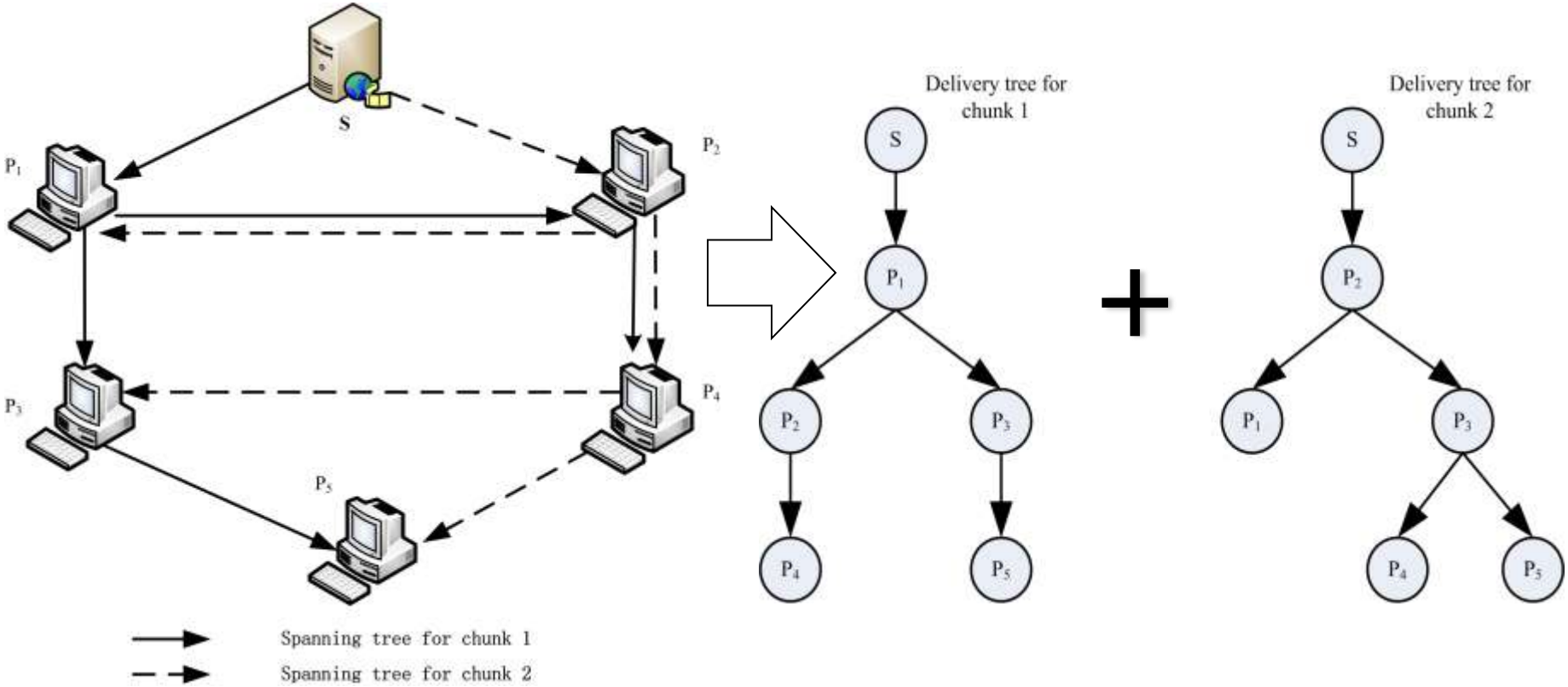


Agenda

- Introduction
- Background
- *Problem Statement and Algorithm Design*
- System Design
- Performance Evaluation
- Conclusion and Future Work

Partitioning Mesh into Spanning Trees

- **Chunks belonging to same content are delivered along independent paths from source to the whole swarm, thus forming *multiple spanning trees*.**



Problem Formulation

Definition :

- directed graph G = (V, E), V denotes the set of peers, and E is the set of overlay edges.
- $p = |V|$;
- D_{ij} : path delay from peer P_i to P_j
- Stochastic process W_t be the set of chunks that should be streamed to peer P_i at time t, $w = |W_t|$ be the number of chunks in W_t
- Let T_l represents the spanning tree of chunk l, $l \in W_t$.
- Denote the packet loss rate between P_i and P_j as L_{ij}

Problem Formulation_contd.

- **Approximation method for propagation packet loss rate**
  - **We consider that there exist m (m ≥0) relay peers between Pi and Pj, we denote the packet loss rate of this m+1 edges as L(x), 0≤L(x),≤1, 0≤ x≤ m, so we have:**

$$L_{ij} = 1 - \prod_{0 \le x \le m} (1 - L(x))$$

$$= \sum_{0 \le x \le m} L(x) - \sum_{x \ne y, 0 \le x, y \le m} L(x) * L(y) + \sum_{x \ne y \ne z, 0 \le x, y, z \le m} L(x) * L(y) * L(z) + \underset{0 \le x \le m}{o} (L(x)) \qquad (1)$$

  - **Because L(x) is relatively small, the high power in (1) can be ignored without causing unacceptable overall deviation. So we omit the second and higher powers, and we can simplify expression (1) into:**

$$L_{ij} \approx \sum_{0 \le x \le m} L(x) \qquad (2)$$

Problem Formulation_contd.

- **We denote the propagation delay from source *S* to peer P_i along spanning trees T_l as $D_i(T_l)$, we have**

$$D_i(T_l) = D_j(T_l) + D_{ij}(T_l) = \sum_{\forall k, k \in [0,m], e_{k,(k+1)} \in T_l} D_{k(k+1)}(T_l)$$

where $D_i(T_l)$ stands for the propagation delay of P_j, and $D_{ij}(T_l)$ stands for the minimum delay along $e_{i,j}$

- **The maximum delay experience by P_i across all possible spanning trees can be derived:**

$$D_i = \arg\max_{T_l} D_i(T_l)$$

Problem Formulation_contd.

- **Similarly, we get maximum propagation packet loss rate of a peer:**

$$L_i(T_l) = L_j(T_l) + L_{ij}(T_l)$$

$$= \sum_{\forall k, k \in [0,m], e_{k,(k+1)} \in T_l} L_{k(k+1)}(T_l)$$

$$L_i = \arg\max_{T_l} L_i(T_l)$$

$$= \arg\max_{T_l} \sum_{\forall k, k \in [0,m], e_{k,(k+1)} \in T_l} L_{k(k+1)}(T_l)$$

in which $L_i(T_l)$ stands for the packet loss rate from source to P_i along spanning trees T_l, $L_{ij}(T_l)$ stands for packet loss rate along $e_{i,j}$, which is the edge of tree T_l, and L_i stands for the maximum propagation packet loss rate of P_i seen at different spanning trees.

Problem Formulation_contd.

- **Given an upper delay bound *D*, the Minimum Packet Loss Mesh (MPLM) problem can be formally defined as: to find a mesh which can minimize the maximum possible packet loss rate experience on all peers in the mesh, while the corresponding mesh delay is no more than upper bound *D*, as follows:**

$$\min \qquad \arg \max_{\forall P_i, P_i \in V} L_i$$

$$s.t.: \qquad \arg \max_{\forall P_i, P_i \in V} \max_{T_l} D_i(T_l) \le D$$

prove the NP-completeness of MPLM problem

- **Method: Reducing Diameter-Constrained Minimum Spanning Tree (DCMST) problem, which is well-known to be an NP-complete problem, to MPLM problem.**
  - **Diameter-Constrained Minimum Spanning Tree (DCMST) problem can be stated as follows: given an undirected edge-weighted graph $G' = (V', E')$ and a positive integer *k*, find a spanning tree with the minimum weight among all spanning trees of which contains no path with more than k edges. It's proven that DCMST is NP-complete.**
  - **Let us consider a MPLM formulation, including a complete graph *G = (V, E),* which is identical to $G' = (V', E')$, and an upper delay bound *D* which can be viewed as *k* in G'**
  - **the maximum packet loss rate of the overlay *G* will have the same meaning just as weight in G'. In other words, the DCMST will have a spanning tree with minimum weight if and only if the same in *G* have the minimum packet loss rate**
  - **Further we can check easily the propagation delay of a mesh against the upper bound *D*, just as verifying maximum path length against bound *k* for DCMST. Thus we prove that MPLM problem is NP-complete.**

Heuristic Algorithm to Address NP-completeness

- **A heuristic solution: OptMesh;**
  - **Build an optimized mesh with:**
    - Least max packet loss;
    - Least max delay;
  - **OptMesh:**
    - Aims to help peers select the best neighbor peers and maintain optimal mesh topology
    - By connecting new joint peer to those neighbors with least possible propagation packet loss and delay.
    - Parameters taken include packet loss rate and delay among peers, and the available uploading bandwidth of the candidate parent peers.

In order to rate candidates and select best neighbors, we propose a rating criterion:

$$R(P_j) = \alpha * f(L_{ij} + L_j) + \beta * f(D_{ij} + D_j) + \gamma * f(1/R_j)$$

$$\alpha + \beta + \gamma = 1, \text{ where } 0 < \alpha, \beta, \gamma < 1$$

$$Function: \quad f(X_k(i)) = \frac{X_k(i)}{\sum_{j=0}^{k-1} X_k(j)}$$

So:

$$f(L_{ij} + L_j) = \frac{L_{ij} + L_j}{\sum_{j=0}^{c-1} (L_{ij} + L_j)}$$

$$f(D_{ij} + D_j) = \frac{D_{ij} + D_j}{\sum_{j=0}^{c-1} (D_{ij} + D_j)}$$

$$f(1/R_j) = \frac{1/R_j}{\sum_{j=0}^{c-1} 1/R_j}$$

Heuristic Algorithm to Address NP-completeness_contd.

- Fully served: When a peer can download stream at an aggregate rate of *s* (the fan-out speed of single stream from video source) from its neighbors
- All peers in candidate set *C* will be sorted according to rating function with the smallest rating in the top
- Peer selection prefers those peers with lower packet loss rate, lower delay and higher available bandwidth.
- P_i will connect to the neighbors at the top until it is fully served
- After selected neighbors, P_i exchanges chunks within them, thus joining a mesh

Peer Selection and Mesh Evolvement

- **Decompose the peer selection and mesh evolvement process into two phases:**
  - **Bootstrapping phase when peer joins**
  - **Self-adaptive evolvement phase**
- **Bootstrapping phase:**
  - **Tracker generates candidate set *C* with the main objective to minimize AS-hop:**

    Find candidate set C from swarm S, which can:

$$\min \sum_{P_j \in C} H_{ij}$$

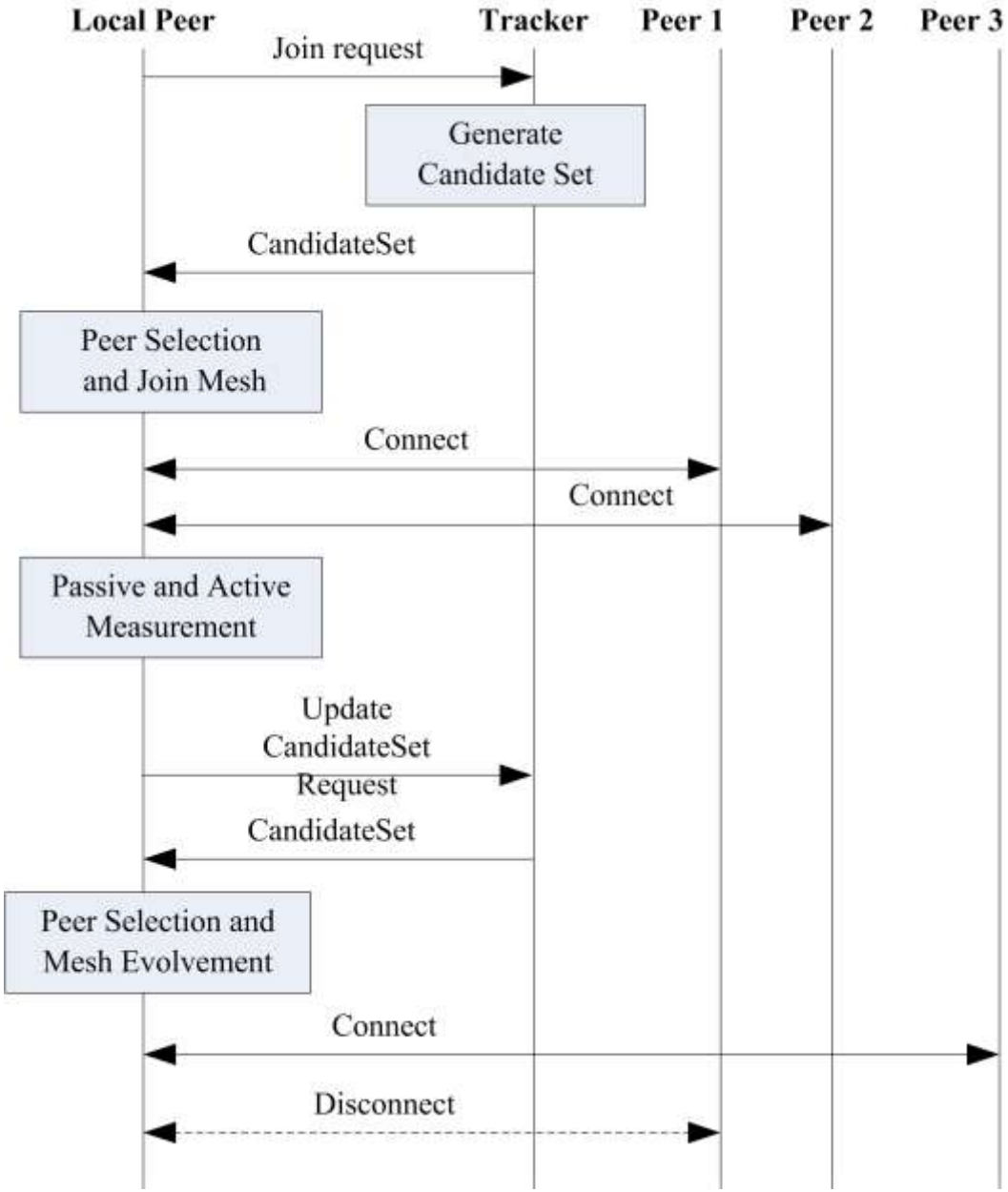
  - **Peer sort neighbor set *N* out of candidate set *C* by using rating criterion**

Peer Selection and Mesh Evolvement_contd.

- **Self-adaptive evolvement phase:**
  - **estimate video quality by passive measurement and use it to trigger self-adaptive evolvement process**
- **video quality threshold Q_T**
- **packet loss rate threshold L_T**

- **When video quality fall bellow Q_T, neighbor peers with packet loss rate higher than L_T will deleted, and re-select a net neighbor set.**

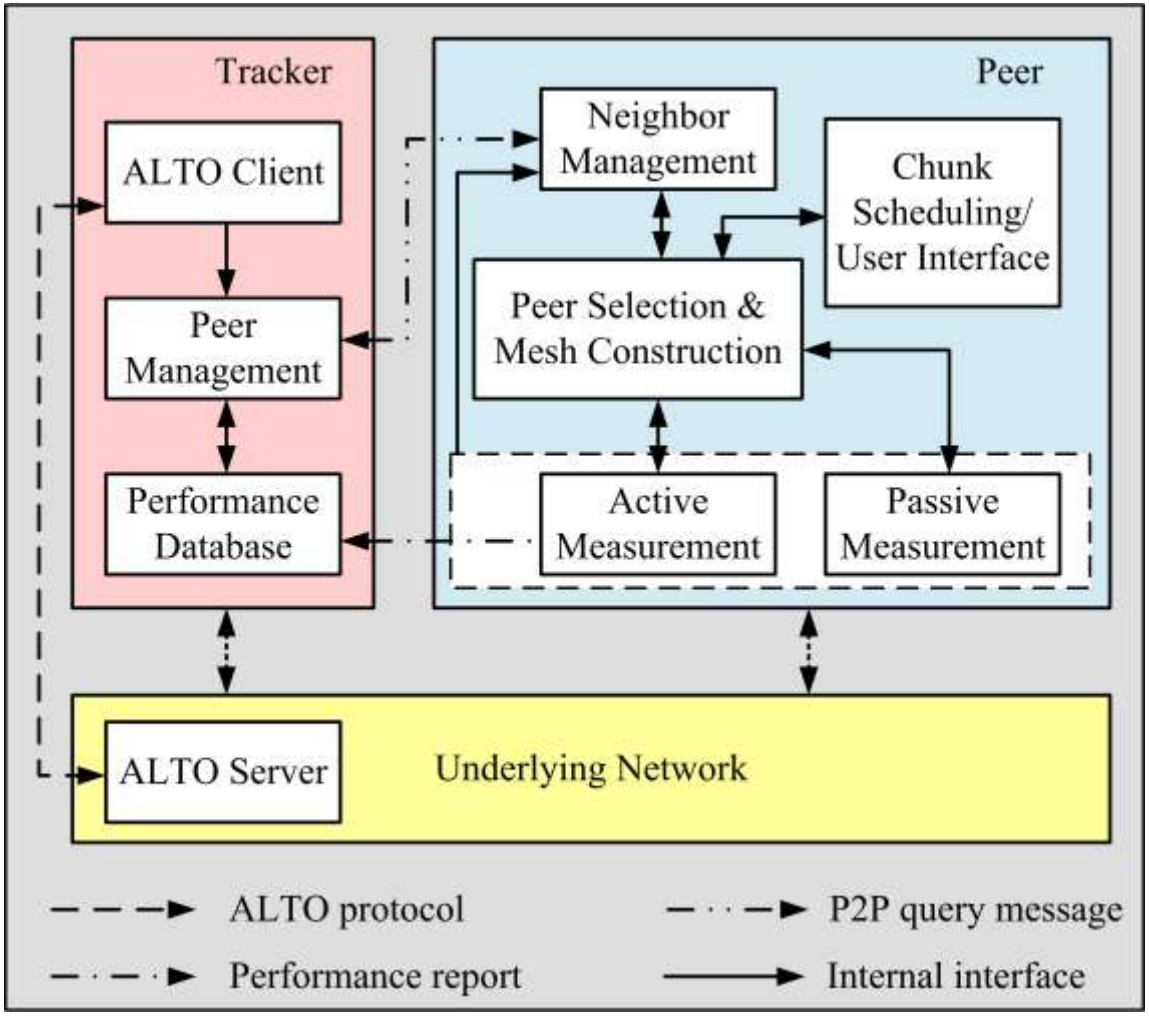
Flow chart of peer selection and mesh evolvement



Agenda

- Introduction
- Background
- Problem Statement and Algorithm Design
- *System Design*
- Performance Evaluation
- Conclusion and Future Work

Functional architecture



Key Elements

- **Passive measurement module:**
  - **Monitoring video quality**
  - **Assessing the average packet loss rate by the following:**

$$\bar{L}_i = \frac{\sum_{j=0}^{n-1} R_{ij} * L_{ij}}{R_0} \approx \frac{\sum_{j=0}^{n-1} R_{ij} * L_{ij}}{\sum_{j=0}^{n} R_{ij}}, \quad P_j \in N$$

where R_{ij} denotes the offered transfer rate from neighbor peer P_j, and R_0 refers to the original playback rate of the video content.

  - **Then assessing video quality using MPQM model**

$$Q_i = Q_E (1 - \bar{L}_i)^{\frac{\bar{L}_i * 100}{r_R}}$$

  - **Trigger mesh evolvement when video quality fall bellow threshold Q_T**

Key Elements

- **Active measurement module:**
  - **Parameters collected include packet loss rate, delay and available uploading bandwidth of candidate peers**
  - **Parameters will be fed into peer selection and mesh evolvement process**
  - **Active measurement will be activated under two circumstances:**
    - At the bootstrapping phase
    - At mesh evolvement phase when user perceived video quality falls bellow threshold Q_T

Agenda

- Introduction
- Background
- Problem Statement and Algorithm Design
- System Design
- *Performance Evaluation*
- Conclusion and Future Work

Simulation Setup

- **Generating network topology:**
  - **We develop an topology generator with MATLAB**
  - **Generates connections, link delay, link packet loss rate**

- **Generating initial uploading bandwidth for each peer**
  - **By exploring the distribution model of peers' average uploading bandwidth of PPlive streaming service**
  - **Without loss of generality, we normalize 400k bps as source streaming rate**

DISTRIBUTION OF INITIAL UPLOADING BANDWIDTH

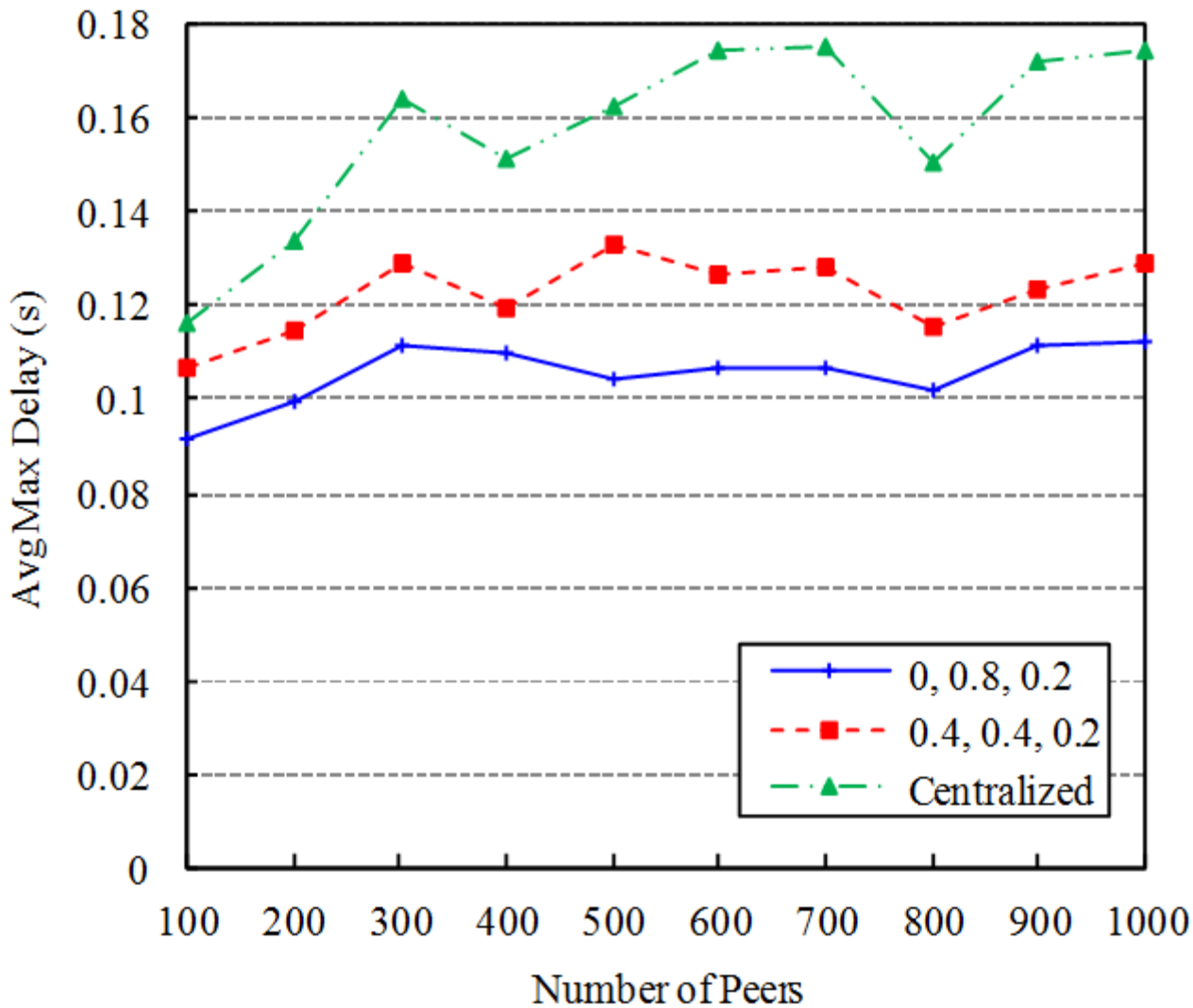
Initial Uploading Bandwidth (unit)	# of Peers (Percentage)
0 - 0.5	36%
0.5 - 1	28%
1 - 2	25%
2 - 3	5%
> 3 (3 - 20)	6%

Evaluation Metrics

- **Max packet loss rate from source to peer**
  - **closely related to video quality**
- **Max delay from source to peer**
  - **Influence playback latency and channel switch lag**
- **Max hop count**
  - **Indicate how deep the overlay topology is**
- **The amount of neighbors**

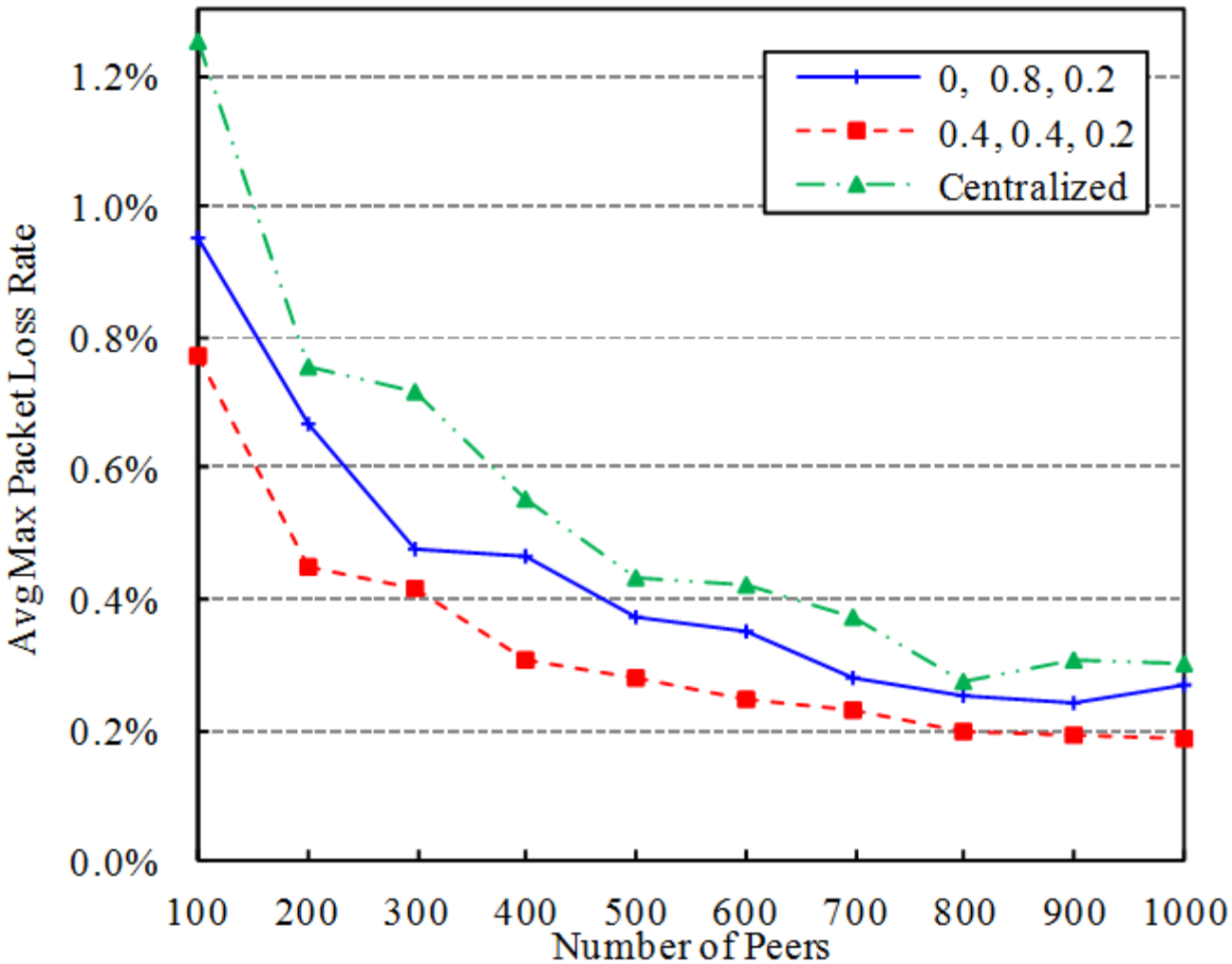
Performance Comparison: Avg. max. delay

- **With: centralized scheme proposed in IEEE trans. on Multimedia 2009**
- **when the number of peers evolves from 100 to 1000, avg. max. delay for:**
  - **Centralized: increase 51% ;**
  - $\alpha = 0, \beta = 0.8 \text{ and } \gamma = 0.2$ **Increase 23%**
  - $\alpha = 0.4, \beta = 0.4, \gamma = 0.2$ **Increase only 20%**



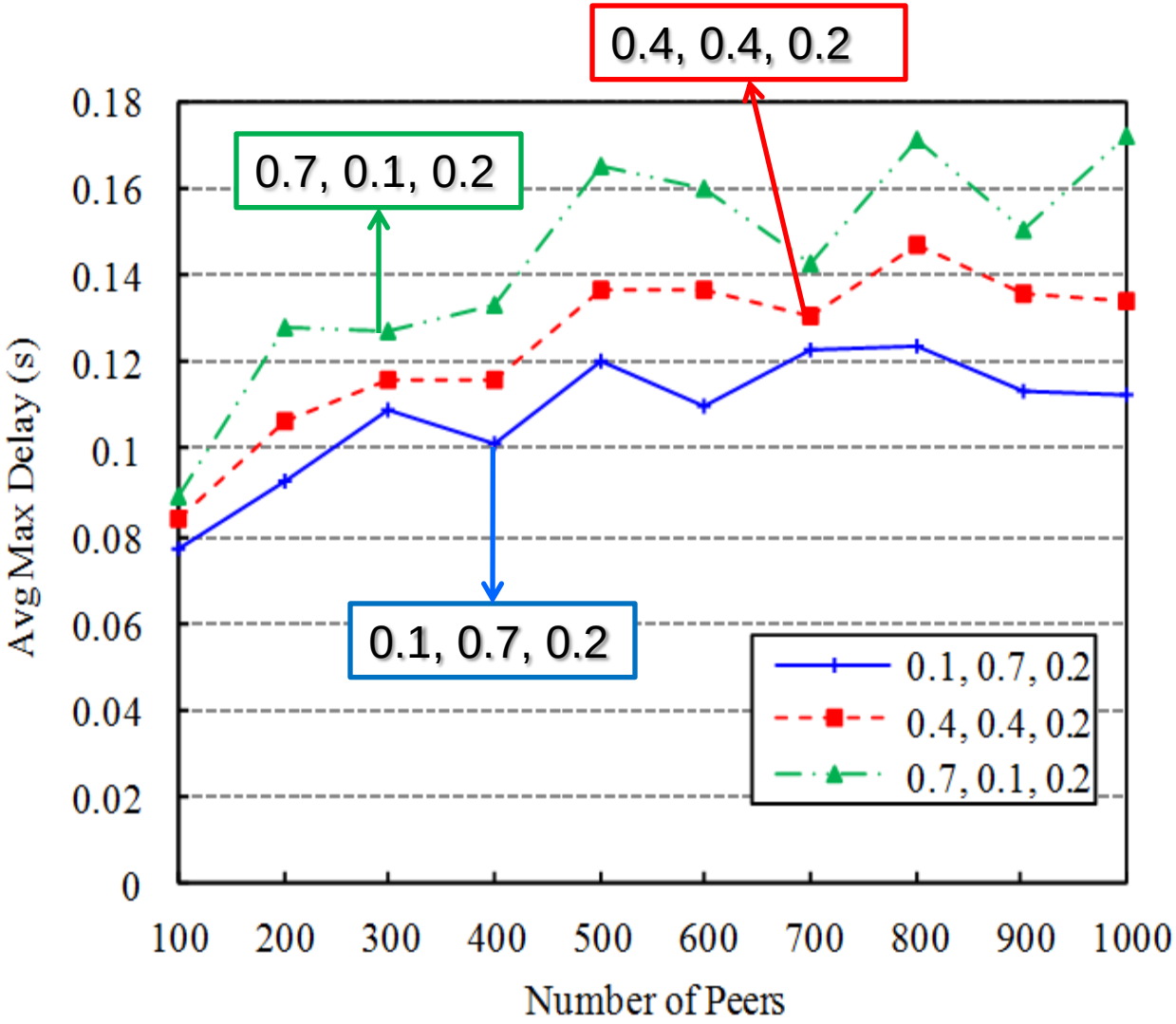
Performance Comparison: Avg. max. packet loss rate

- $\alpha = 0.4, \beta = 0.4 \text{ and } \gamma = 0.2$
  Performs better than centralized scheme
- **A tendency of decline with the swarm enlarges from 100 peers to 1000 peers**



Impact of Weighting Coefficients: Avg. max. delay

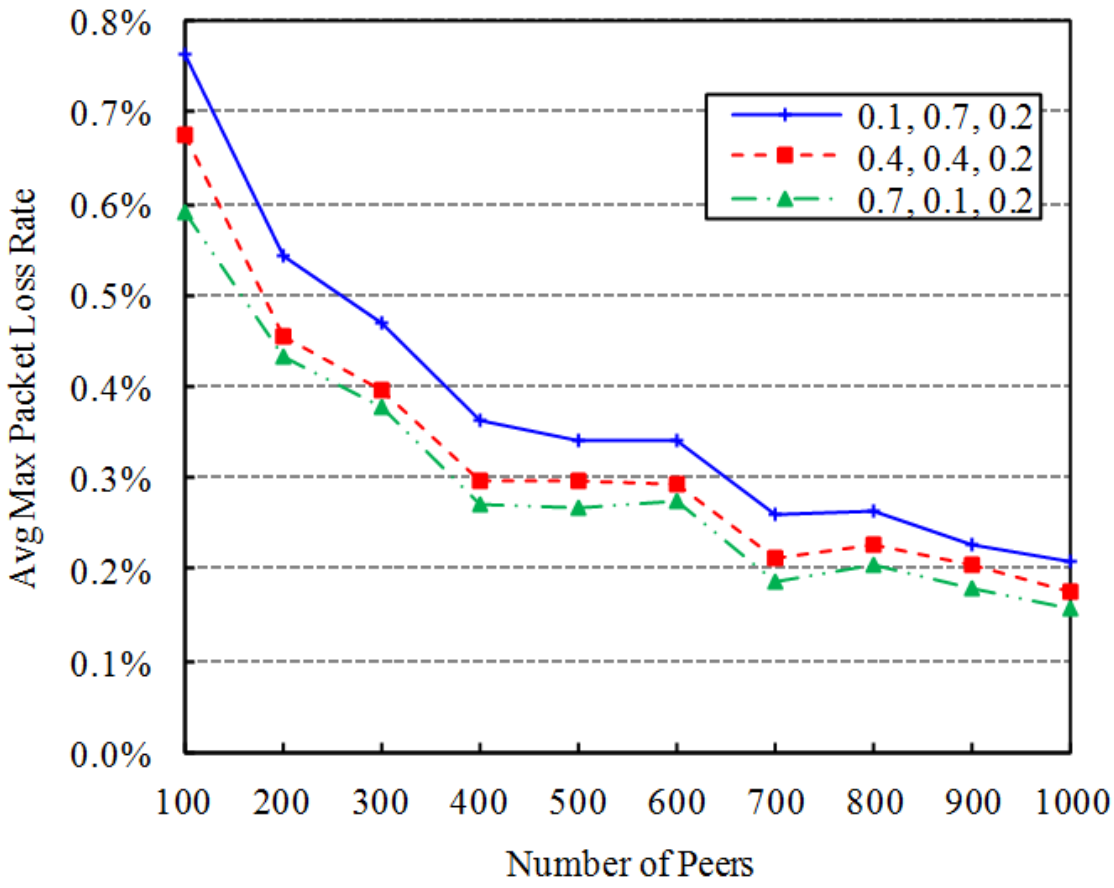
1. *if more weighting is given to weighting coefficient of delay, candidates with lowest maximum delay have more priority to be selected as neighbors, thus reducing max delay for local peer*
2. *Using OptMesh, Max delay does not increase fast, thus enable mesh overlay to scale to larger topology with reasonable max delay*



Impact of Weighting Coefficients: Avg. max. packet loss rate

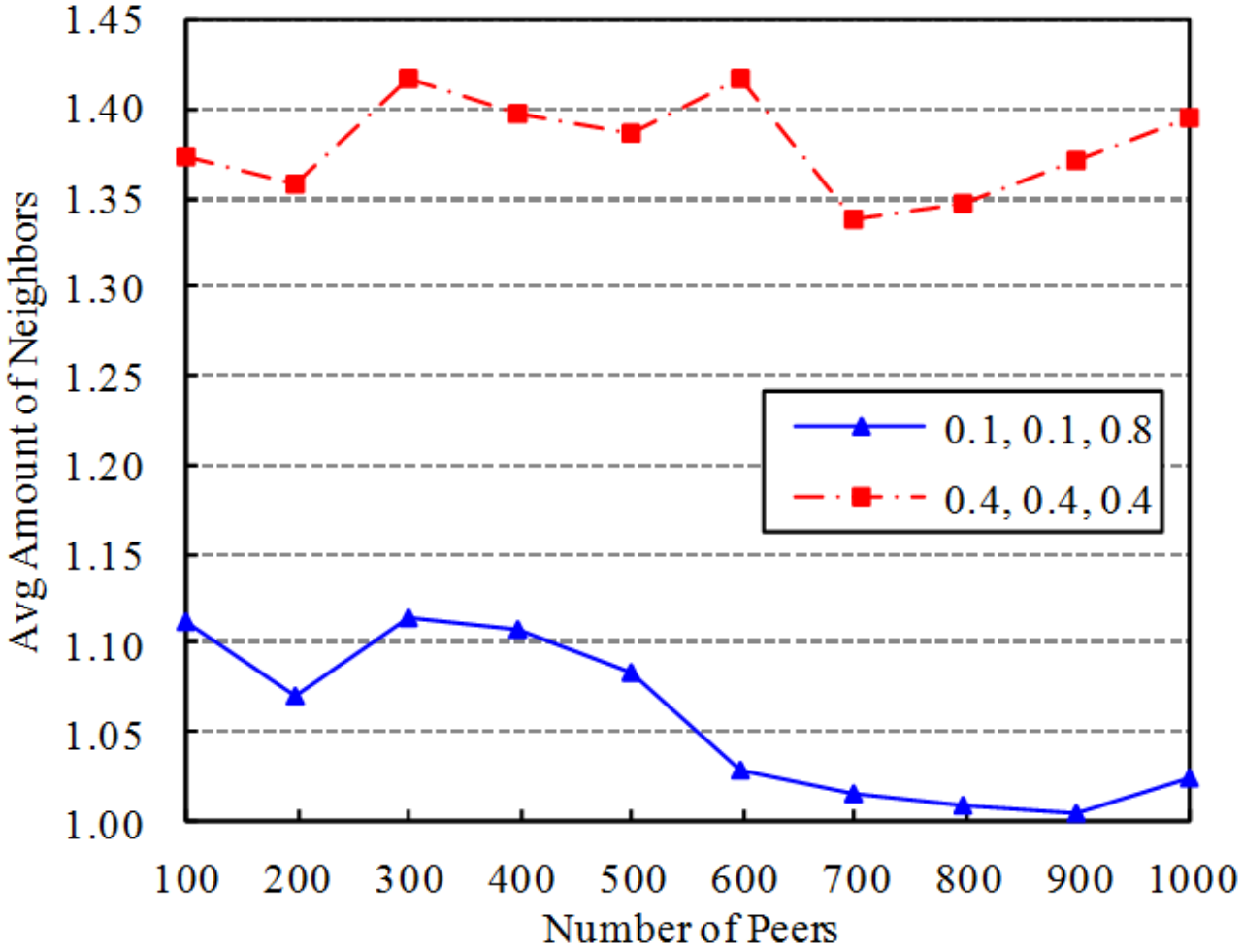
1. *Similarly, if more weighting is given to weighting coefficient of delay, candidates with lowest maximum delay have more priority to be selected as neighbors, thus reducing max delay for local peer*

2. *An interesting observation is that the average maximum packet loss rate will decrease with larger swarm sizes. We expect such a behavior because when the topology become larger, the new joint peer are more likely to find such neighbors with low packet loss rate*



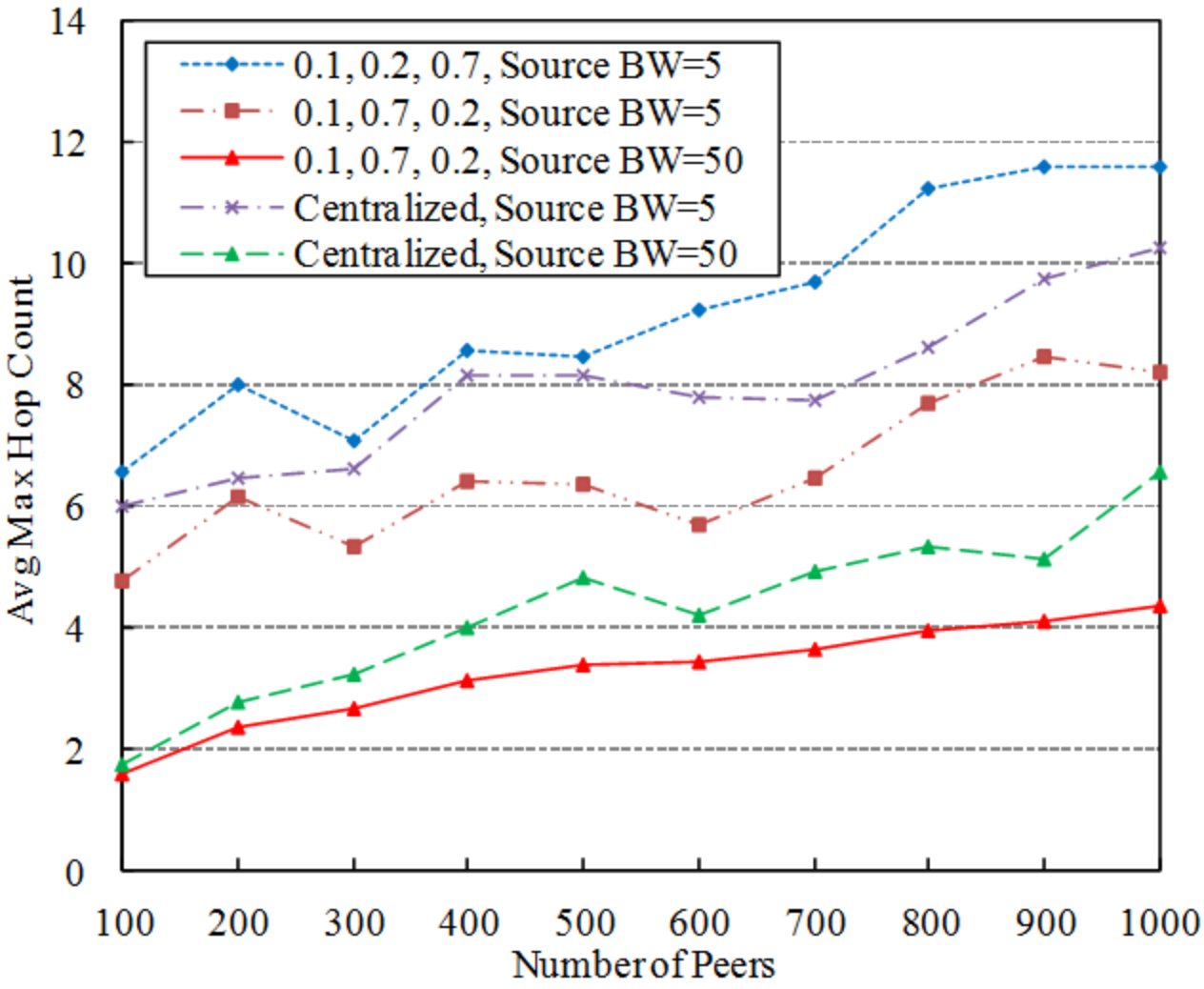
Impact of Weighting Coefficients: Avg. amount of neighbors

1. *Average amount of neighbors show a tendency of decline if more weighting is given to weighting coefficient for uploading bandwidth.*
2. *Because candidate peers with high available uploading bandwidth gain more opportunity to be selected as neighbors, thus leading to small neighbor count whilst meeting the streaming rate requirements*



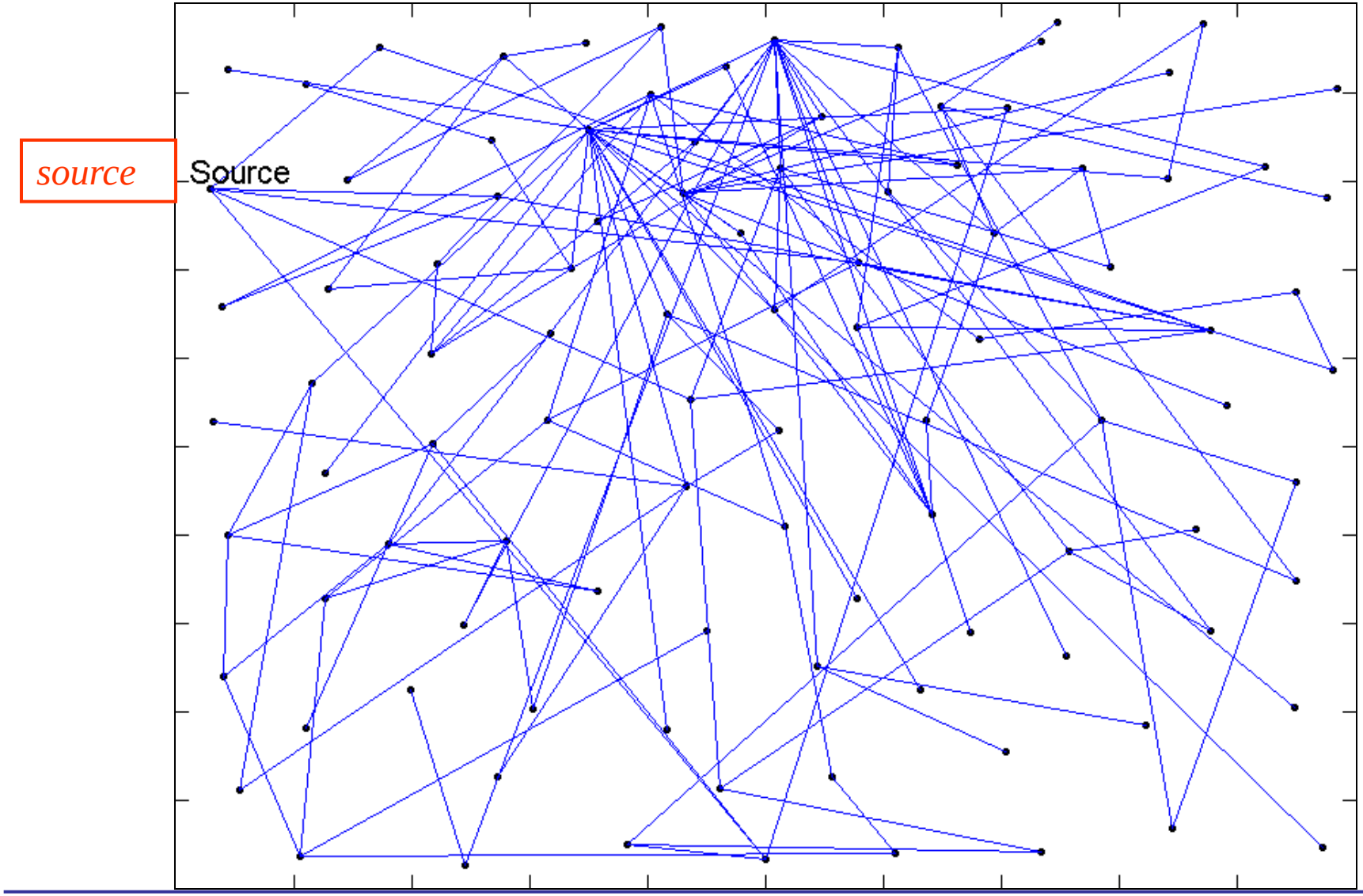
Hop Count Analysis

1. *Average hop count decreases when delay coefficient and uploading bandwidth of video source increase*
2. *Low hop count when compared with centralized scheme*



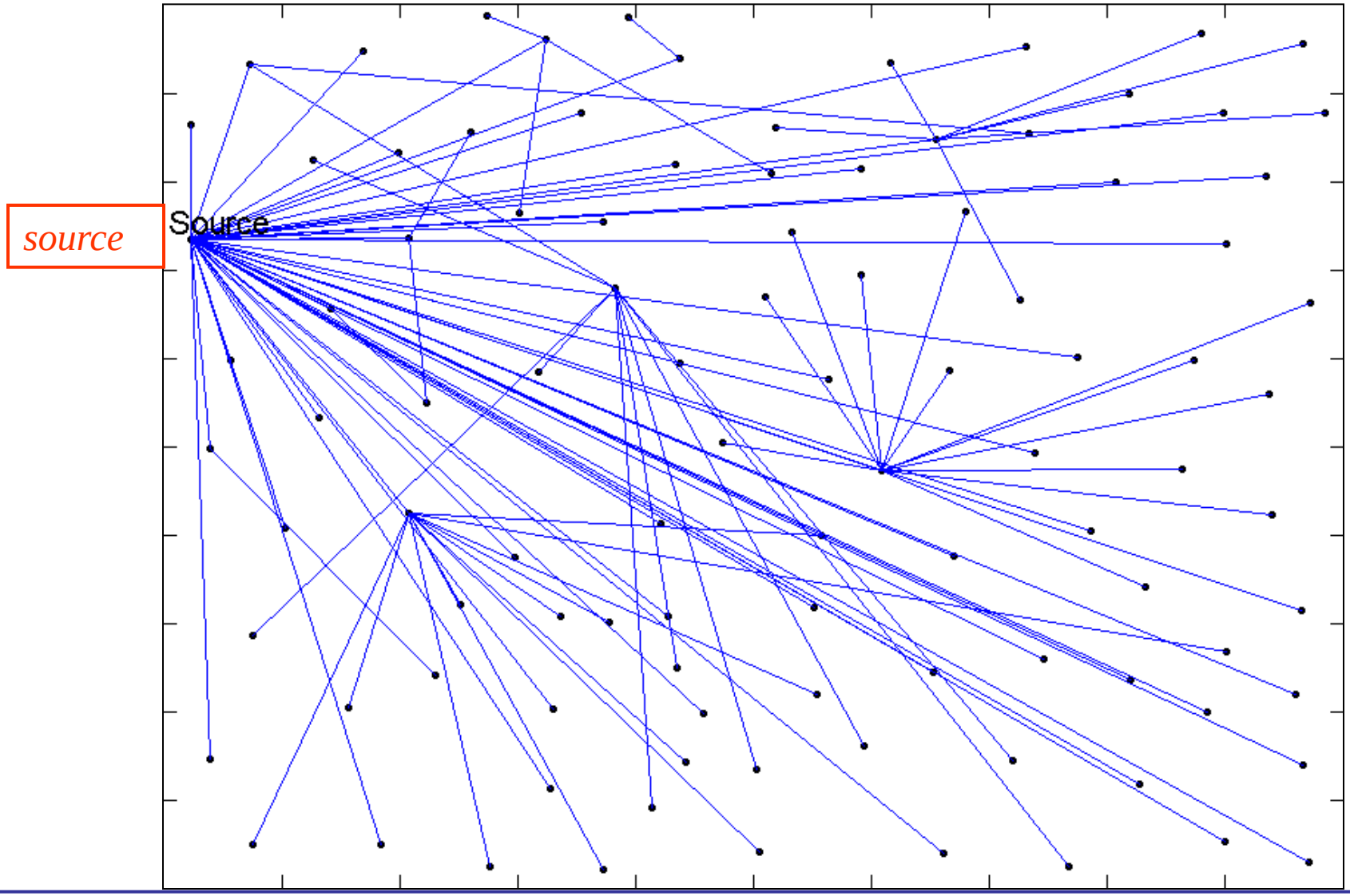
The Formed Mesh Topology

- **Swarm size: 100, source uploading bandwidth:5**



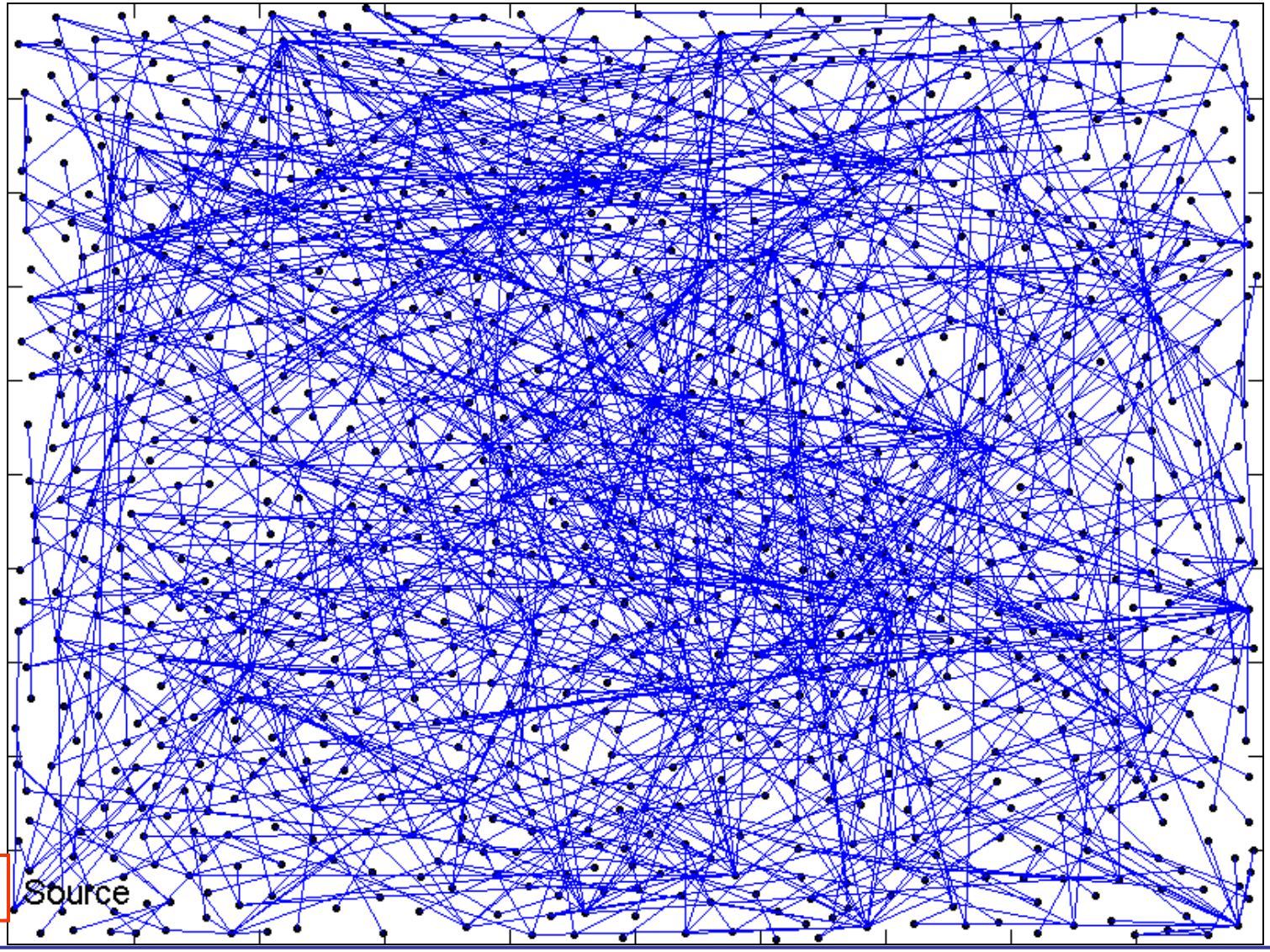
The Formed Mesh Topology

- **Swarm size: 100, source uploading bandwidth:50**



The Formed Mesh Topology

- **Swarm size: 1000, source uploading bandwidth:5**

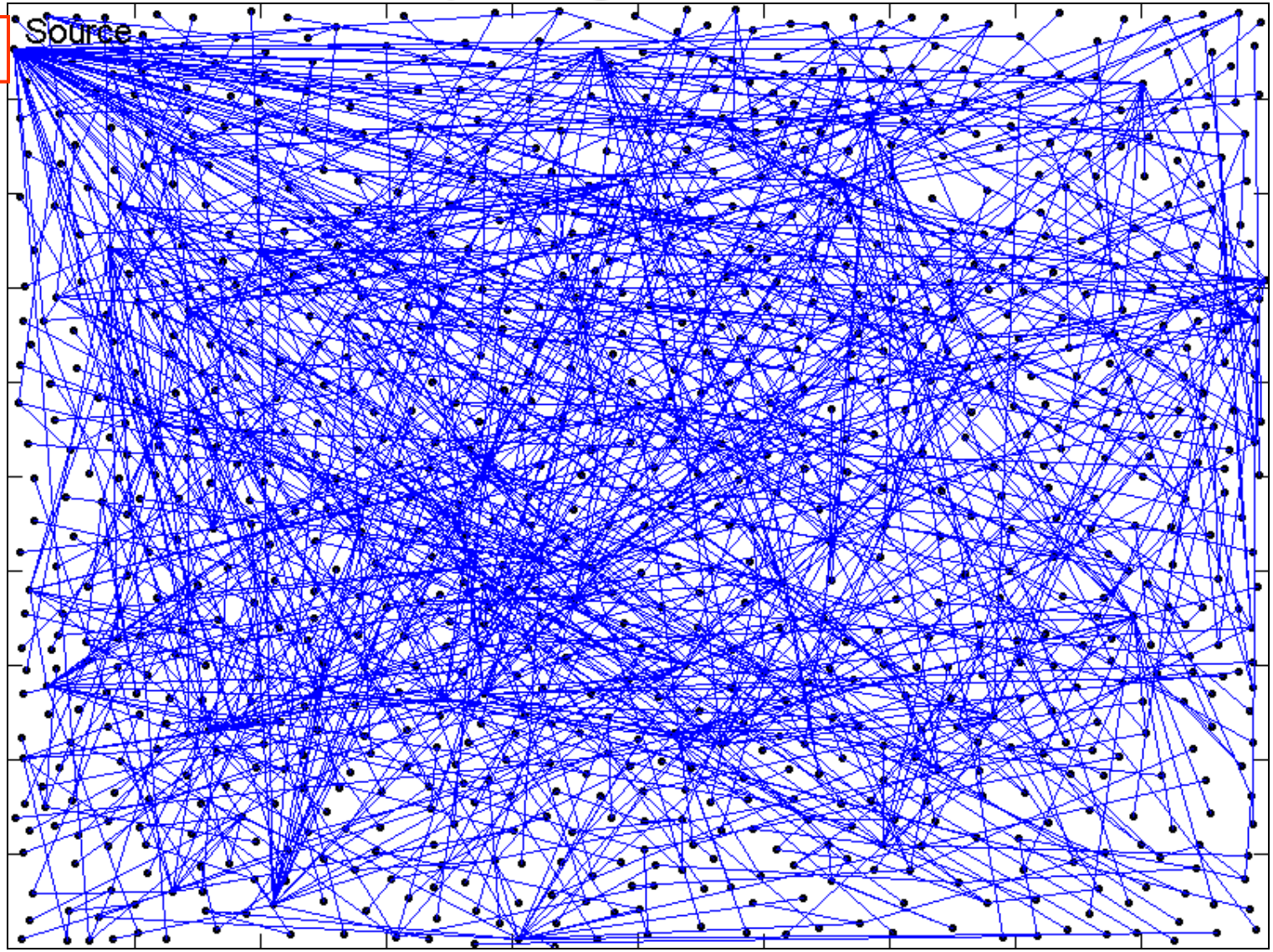


source

The Formed Mesh Topology

- **Swarm size: 1000, source uploading bandwidth:50**

source



The Formed Mesh Topology

- **Conclusion**
  - **When uploading bandwidth is increased to 50 units, it is observed that more peers select source as their neighbors thanks to OptMesh algorithm**
  - **This property further proves that effectiveness of OptMesh algorithm on selecting best neighbors with least propagation delay and packet loss rate.**

Conclusion and Future Works

- **In this paper, we**
  - **Present the model of constructing a Minimum Packet Loss Mesh (MPLM) network with bounds on delay and prove this problem is NP-complete**
  - **Design one algorithm called OptMesh to select best neighbors for peers;**
  - **Present one self-adaptive mesh evolvement method for maintaining best performance mesh topology;**
  - **Prove the combined requirement on both maximum packet loss rate and delay can be meet with careful mesh construction method;**
- **In the future:**
  - **we plan to implement the whole system we proposed and to experiment it on testbed and further on real network**

Questions are welcome!

liuzhiyong@chinattl.com

Z.Sun@surrey.ac.uk

D.He@surrey.ac.uk

shidenian@chinattl.com