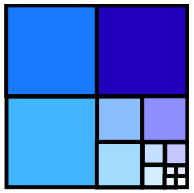


# Degree Hunter: on the Impact of Balancing Node Degrees in de Bruijn-Based Overlay Networks

23 April 2010

Pierre FRAIGNIAUD * - Hoang Anh PHAN**
(*) (**) CNRS/LIAFA and University Paris Diderot

# Loading Balancing: Problematic

- Load Balancing: a technique to distribute load evenly across two or more nodes.

- Objectives:
  - Optimal resource utilization
  - Maximizing throughput
  - Minimizing response time
  - Avoiding overload

- Problems:
  - Bandwidth heterogeneity
  - Load dynamic change
  - Popularity of resources

- Load:
  - stress concerning :
    - Memory,
    - CPU,
    - Bandwith,
    - Storage,
    - Power,…
  - 2 kinds of load:
    - Signaling: lookup, routing,etc.
    - Data transfert load

# Load Balancing: State of the art

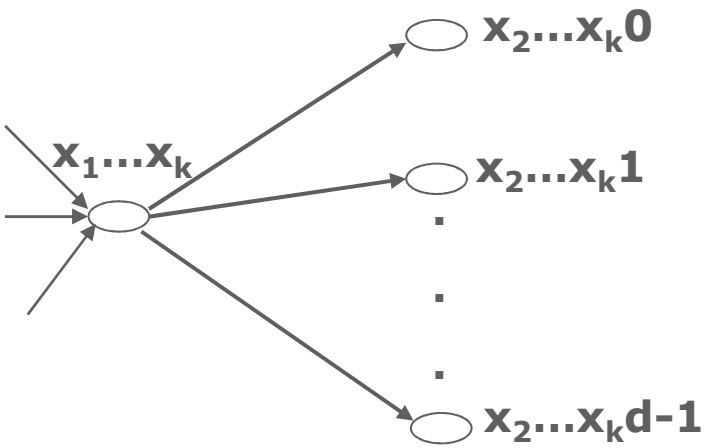
- Virtual server [Rao et al. - IPTPS] [Dabek et al. - SOSP] :
  - Virtual server = subset of the keys
  - Moving virtual servers from a « heavy » node and a « light » node during runtime

- Controlling the resource location [Bayer, Considine, Mitzenmacher – IPTPS'03]
  - Multiple hash functions per resource. Computed key is stored in the peer with the smallest load

- Controlling the node location [Karger & Ruhl – IPTPS '04] [Rieche et al. – IEEE LCN'04]
  - Underloaded node migrates itself to portions of the address space pointing to lots of data items, in order to share the load with the node responsible for this overloaded address space.

# D2B Protocol

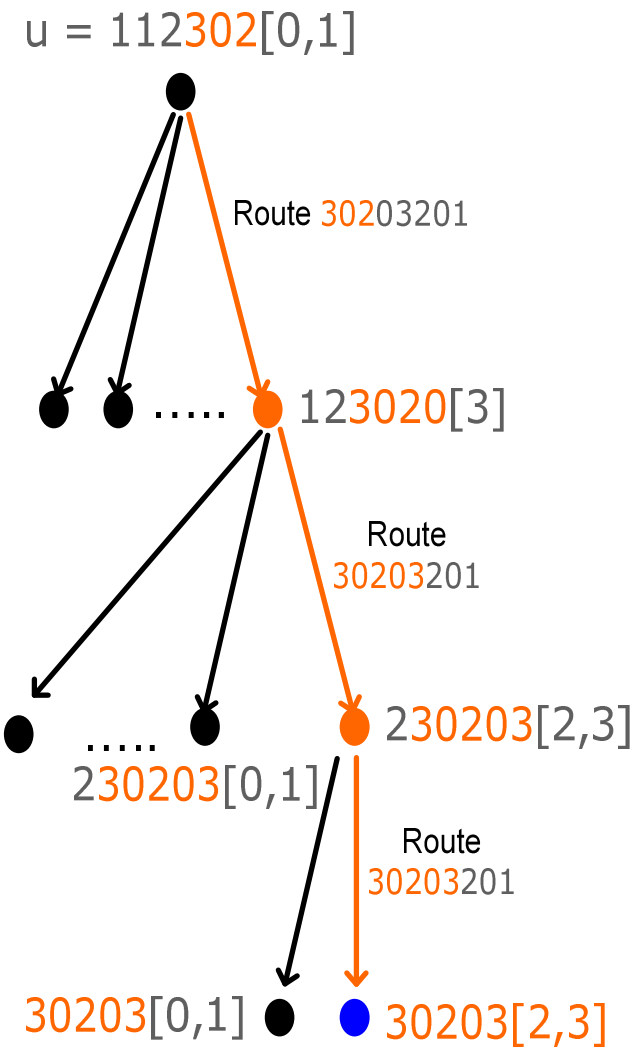
- De Bruijn B(d,k) graph:
  - The nodes are the words of length k on an alphabet of d symbols.




- Features:
  - Number of nodes = $d^k$
  - In-degree = Out-degree = d
  - Diameter (longest shortest path)= k
  - Simple routing

# D2B Protocol

- P2P System D2B-d dimension [P. Fraigniaud & P. Gauron – 2003]:
  - ID of a node: NodeId, words on an alphabet of d symbols.
  - ID of a resource: Key, words of length m on an alphabet of d symbols.
  - The responsible node stores the information, i.e. the key, concerning a resource (IP address of its owner) iff its NodeId is a prefix of the key.
- Example of routing:
  - D2B 4-dimension
  - Key k =30203201
  - Node u = 112302[0,1]



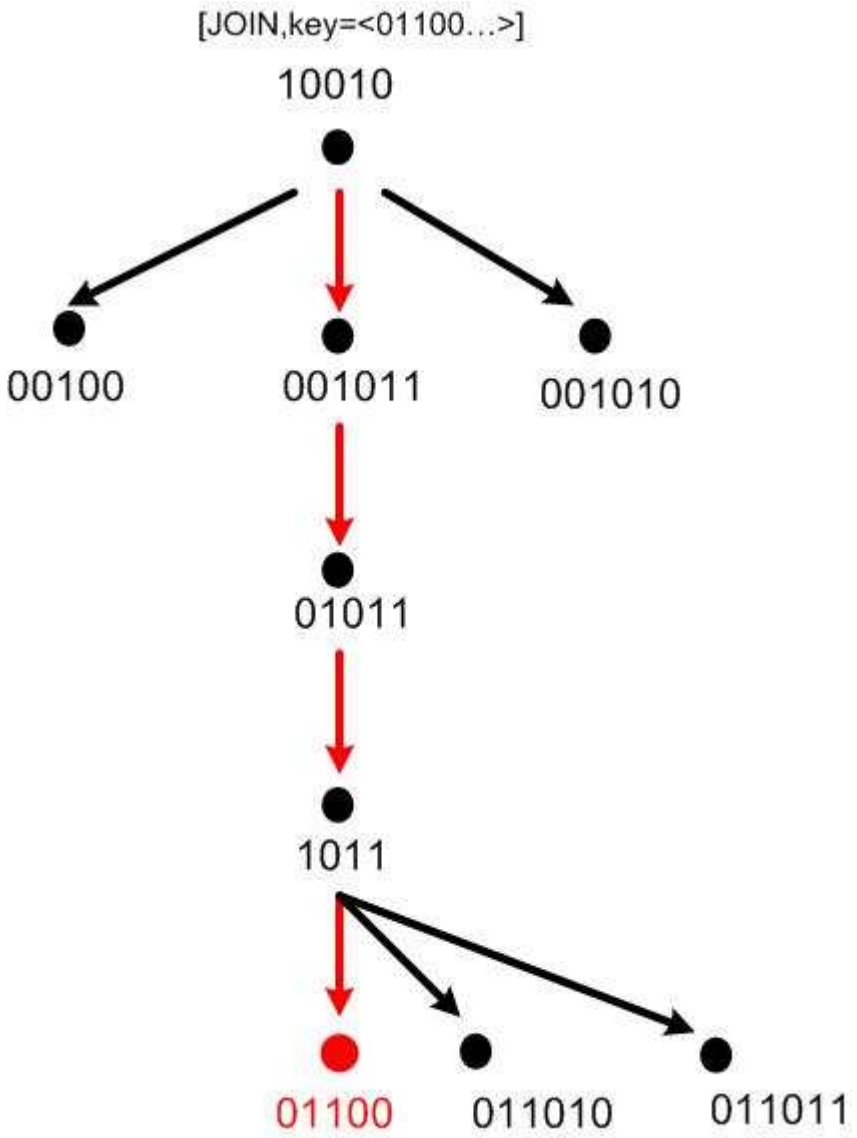
# Load Balancing: Degree Hunter (1/6)

- Degree Hunter:
  - Main ideas:
    - Hunting the nodes with a large key space: short ID
    - Hunting the nodes with too large or too small number of out-going neighbors (children)
  - Why are these nodes hunted?
    - Node with a small out-degree: fault tolerance, construction of multicast trees
    - Node with a large out-degree: updating connections, large #packet duplications on the multicast trees.
  - Objectives:
    - Balancing the key space among all nodes
    - Balancing the degrees of the nodes
    - Avoiding additional load
    - Avoiding the increase of the response time
    - Avoiding the increase of the time for JOIN/LEAVE
  - 3 decentralized mechanisms and 1 centralized mechanism

# Load Balancing: Degree Hunter (2/6)

- Degree Hunter in D2B-2d
  - STANDARD:

Message [Join] with key = <01100…>



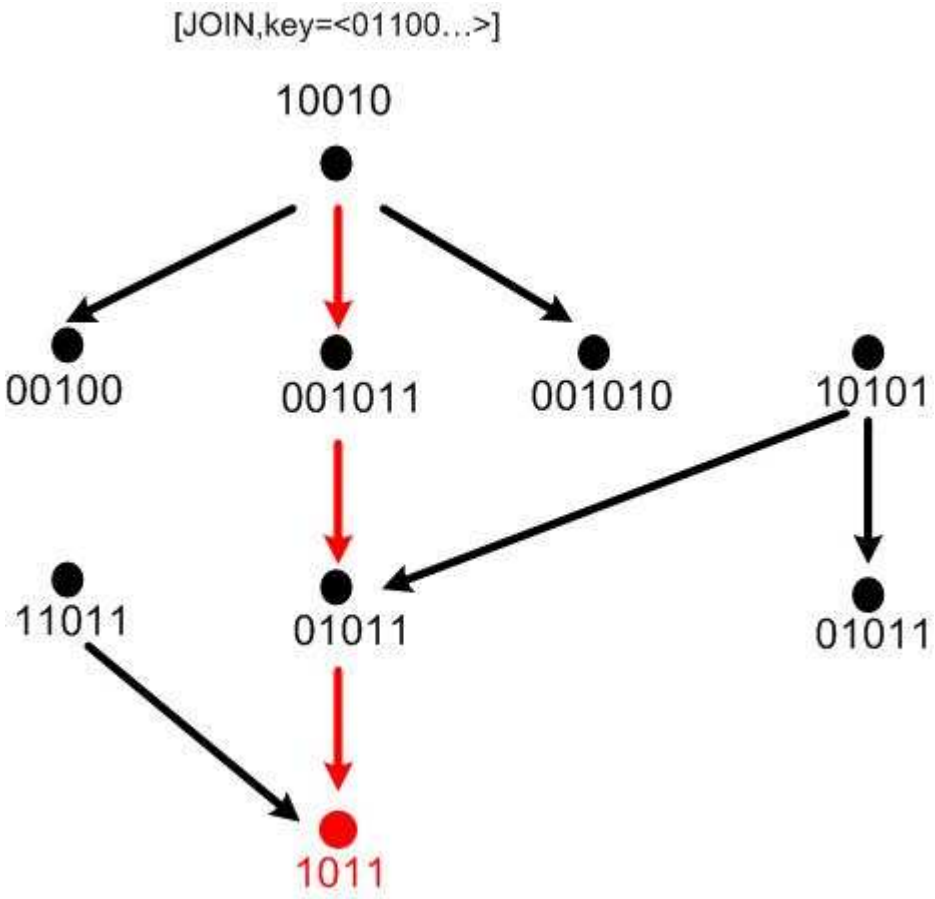
# Load Balancing: Degree Hunter (3/6)

- Degree Hunter in D2B-2d
  - ALL-SINGLE:

    Stop at a node that is the unique out-neighbor of all its in-neighbors

**Properties**:

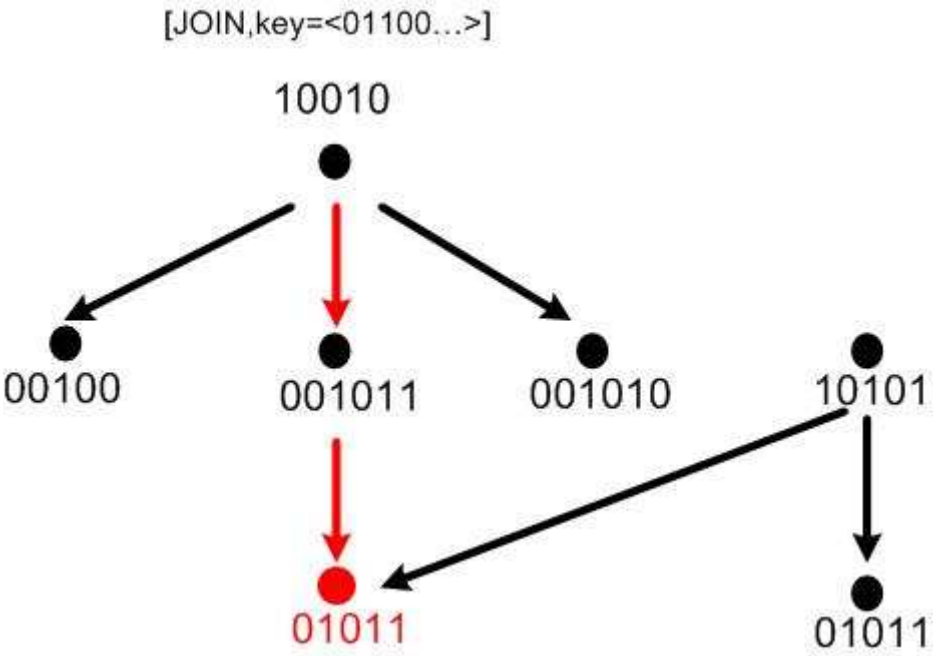
- Reduce the #nodes having a unique child while not creating nodes with high out-degrees.
- Adapted to data streaming



# Load Balancing: Degree Hunter (4/6)

- Degree Hunter in D2B-2d
  - 1-SINGLE:

    Stops when a node which is the unique out-neighbor of at least one of its in-neighbors is found




**Properties:**

- Reduces the #nodes having a unique child but may create nodes with high out-degrees.
- Adapted to file sharing systems.

# Load Balancing: Degree Hunter (5/6)

- Degree Hunter in D2B-2d
  - ID-LENGTH: routes [JOIN] to (among its neighbors):
    - Node having a largest key space (short ID).
    - In case of a tie, node whose in-neighbors have a small out-degree
    - In case of a double tie, node who is the unique child of many nodes
    - Stops when the curent node is locally the best.

**Properties:**

- Balances the key space among nodes while reducing the number of nodes having a unique child, without adding nodes with large out-degree.

- Adapted to streaming + file sharing systems.

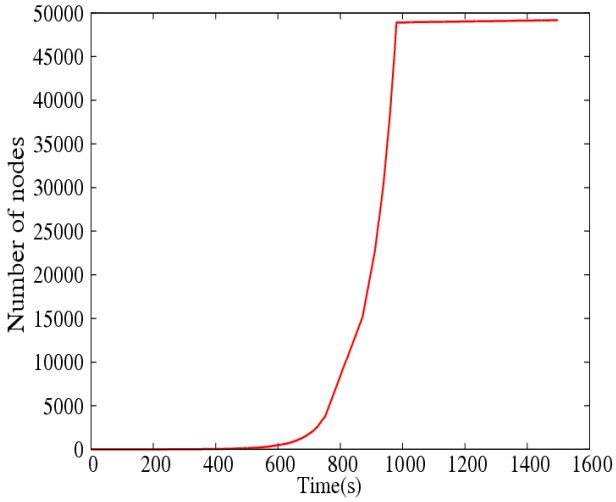
# Load Balancing: Degree Hunter (6/6)

- Degree Hunter dans D2B-2d
  - TRACKER:
    - Maintains a short list of insertion points: the nodes to be contacted when joining the system.
    - List is sorted according to the same rules as ID-LENGTH
    - Dynamic mechanism: limited size (500 nodes), updated after each modification of the network
    - If the list is empty: STANDARD method

# Degree Hunter: Evaluations (1/8)

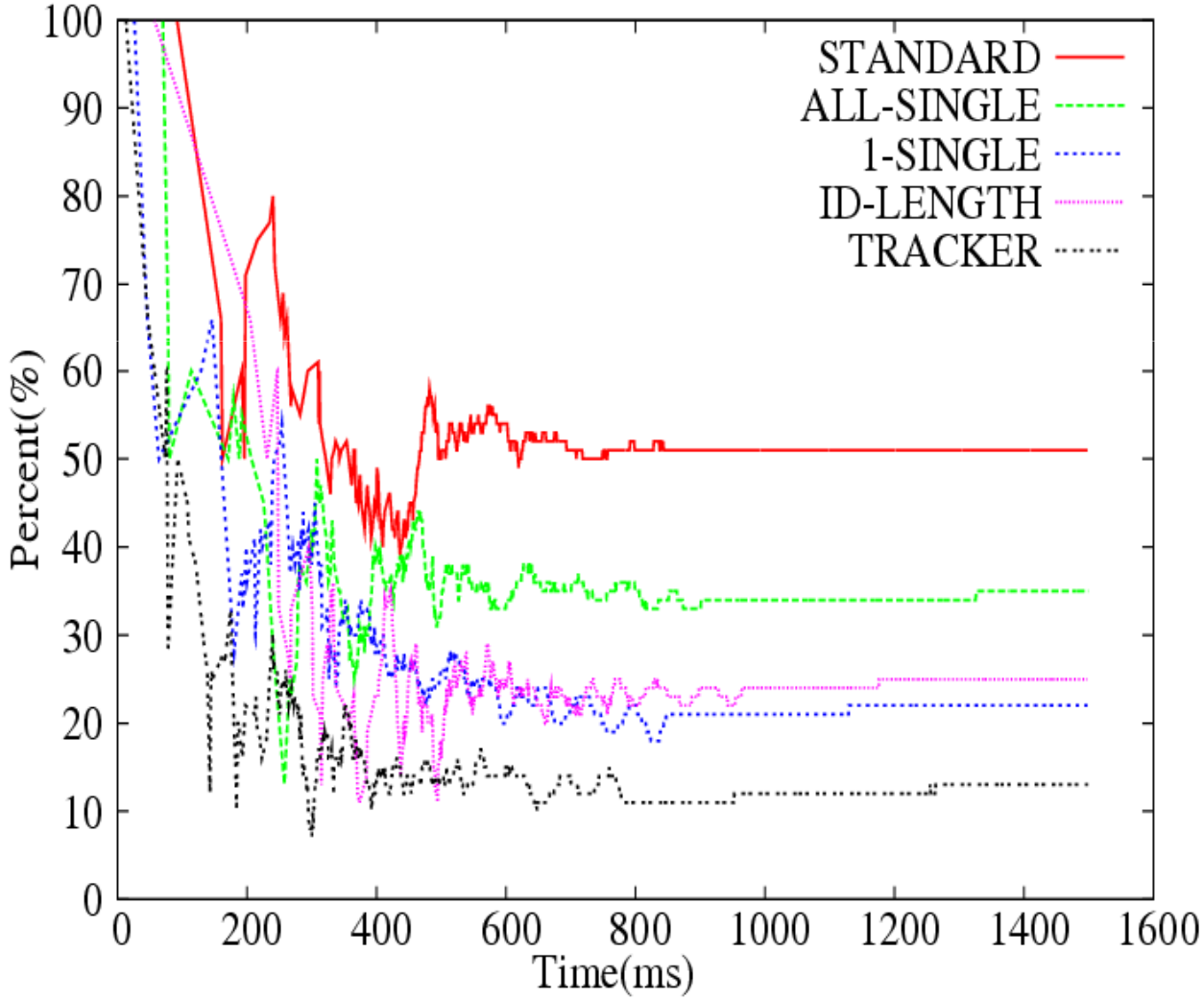
- Simulator: Peersim Jelasity &A. Montresor, G-P Jesi & S. Voulgari]: discret events & packet layer
- Parameters of D2B system:
  - 50.000 nodes
  - Fowarding delay for a message between 2 nodes: 10 to 100ms
  - Buffer = 1Mb at each node
  - Download bandwidth: 6Mb to 20Mb/s, upload: 1Mb/s
- Evolution of system's size during simulation
- 2 phases:
  - Transition phase: 15 mns
  - Steady phase (50000 nœuds): 10 mns,

    entrance freq = leaving freq = 0.5s (average)



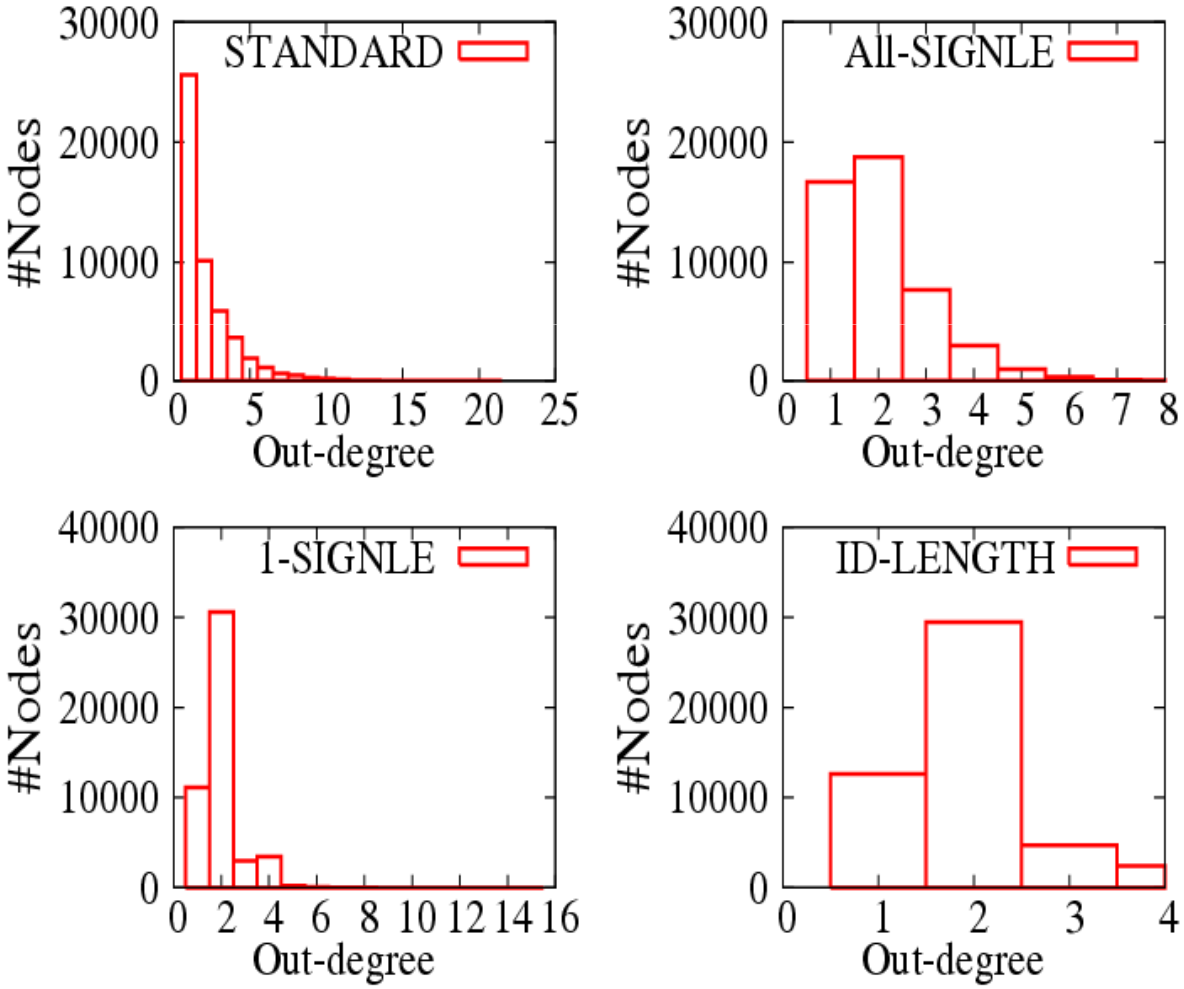
# Degree Hunter: Evaluations (2/8)

Fraction of the number of nodes having only one out-neighbor



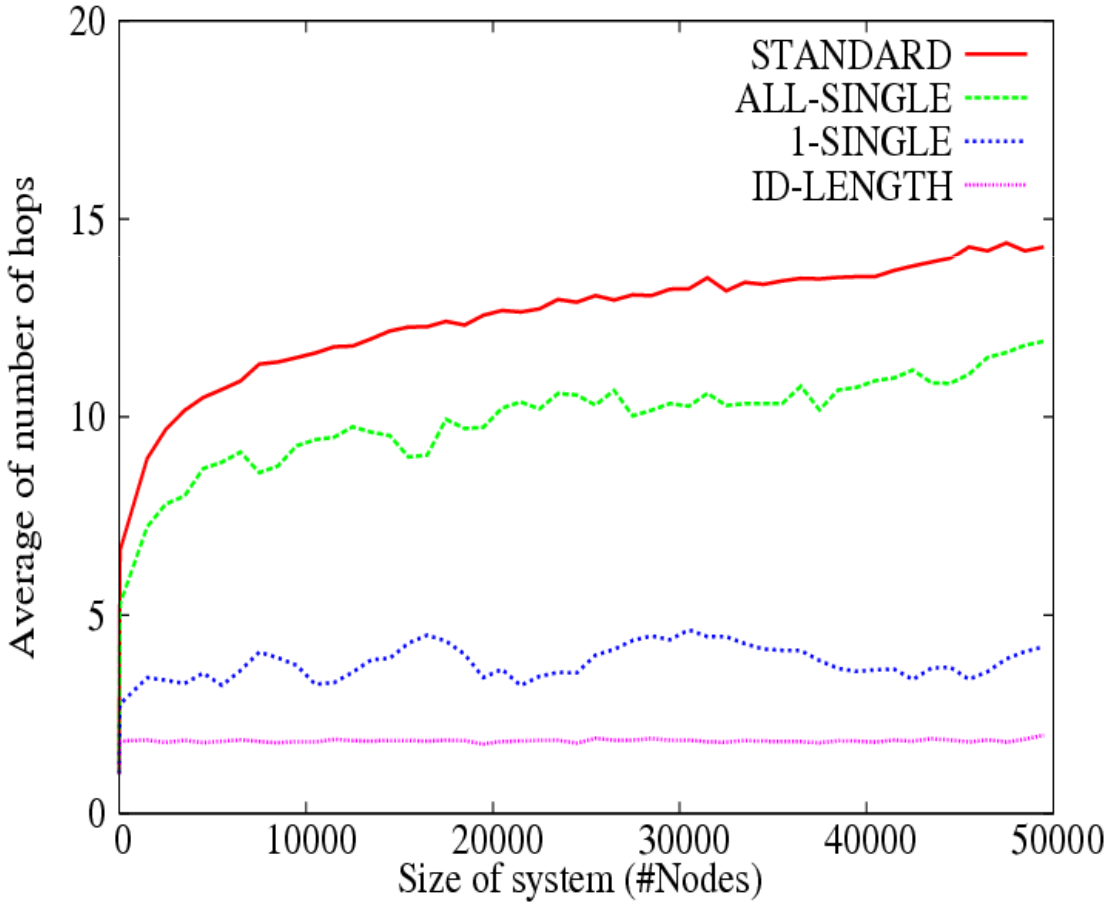
# Degree Hunter: Evaluations (3/8)

- Histogram of the in-degrees at the end of the steady state period



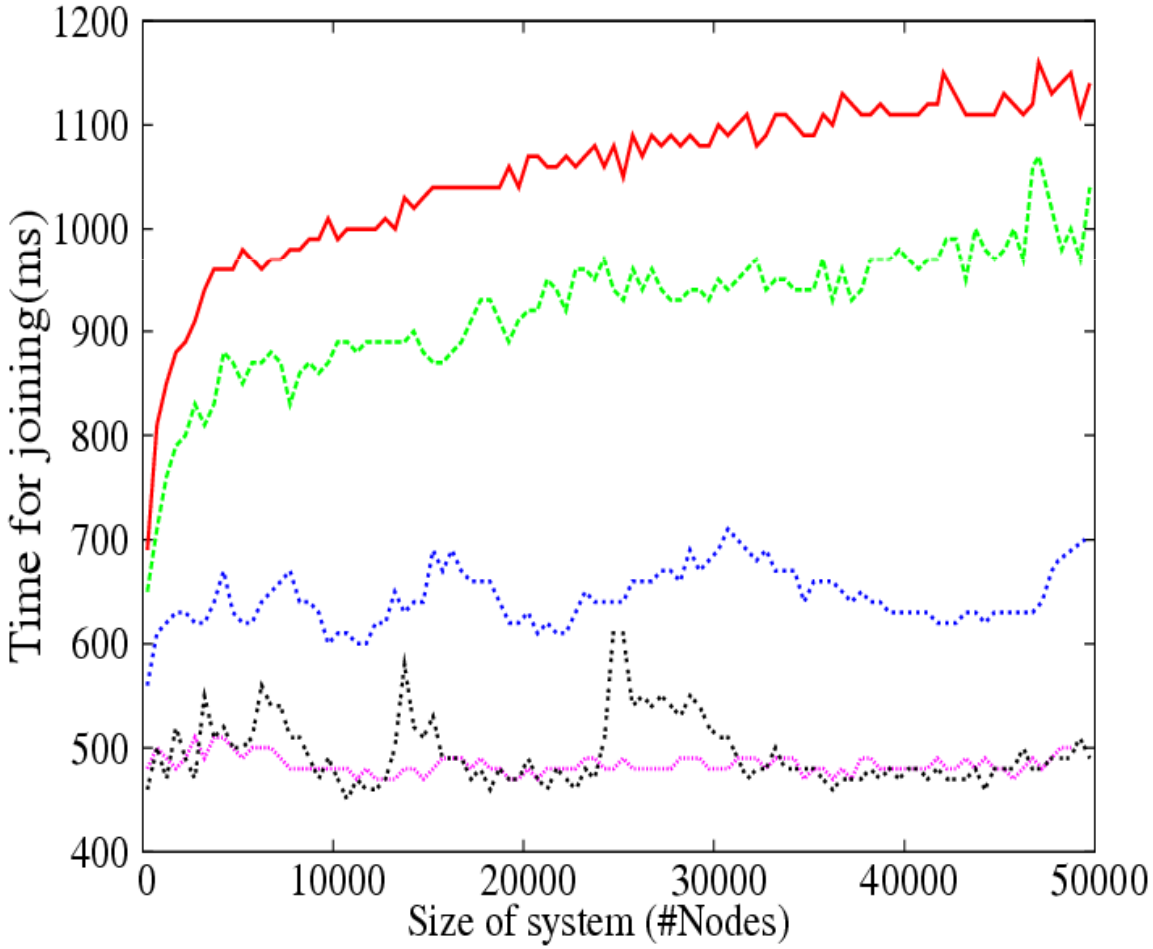
# Degree Hunter: Evaluations (4/8)

- Impact on congestion and latencies (1/3)
  - Average number of hops for a JOIN, as a function of the size of the system



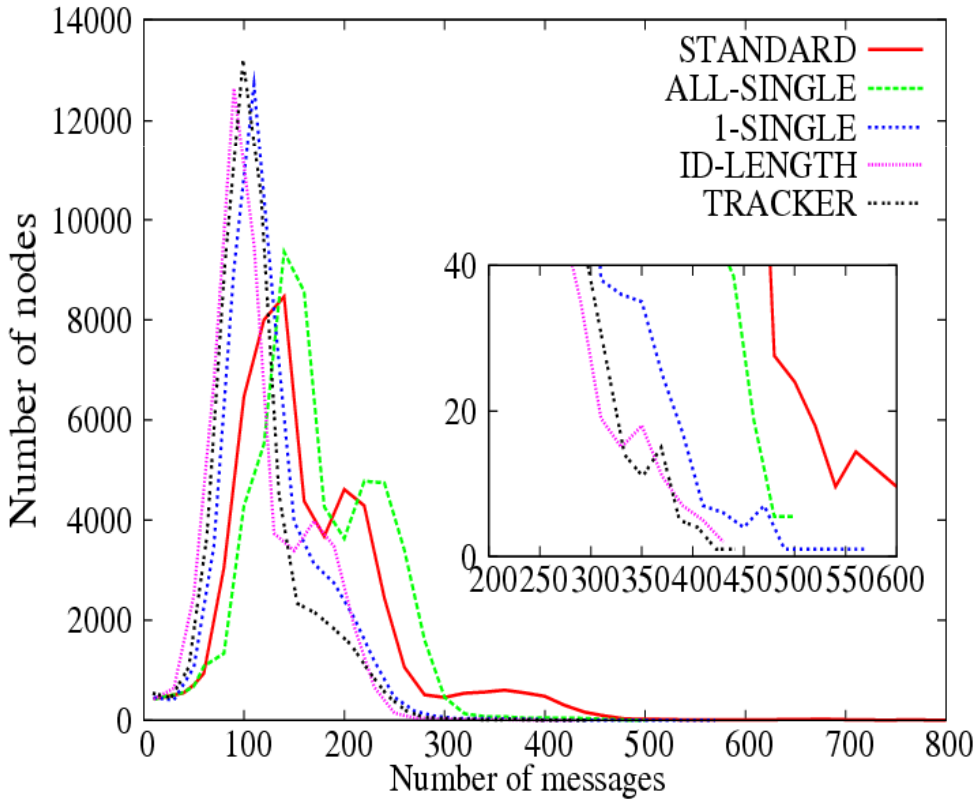
# Degree Hunter: Evaluations (5/8)

- Impact on congestion and latencies (2/3)
  - Average time for a JOIN as a function of the size of the system during the transition phase



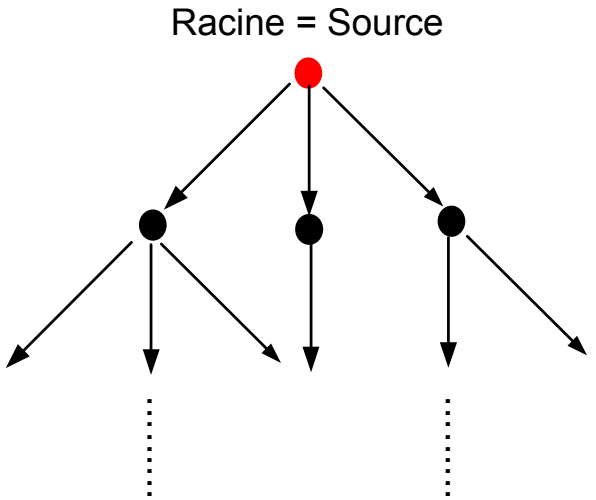
# Degree Hunter: Evaluations (6/8)

- Impact on congestion and latencies (3/3)
  - Distribution of the nb of LOOKUP/PUBLISH requests traversing a node at the steady state period (about 1000 requests/s for all nodes of the system).

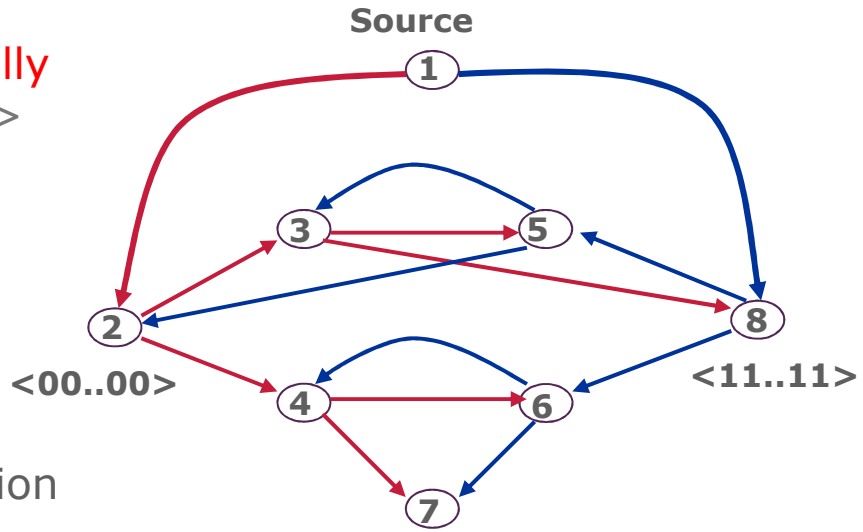


# Degree Hunter: Evaluations (7/8)

- Impact on multicast performances (1/2)
  - BFS Protocol
    - Each multicast group use one BFS tree to distribute datas to all the members
    - The root (= source) of the multicast tree is responsible for distributing the datas in its tree



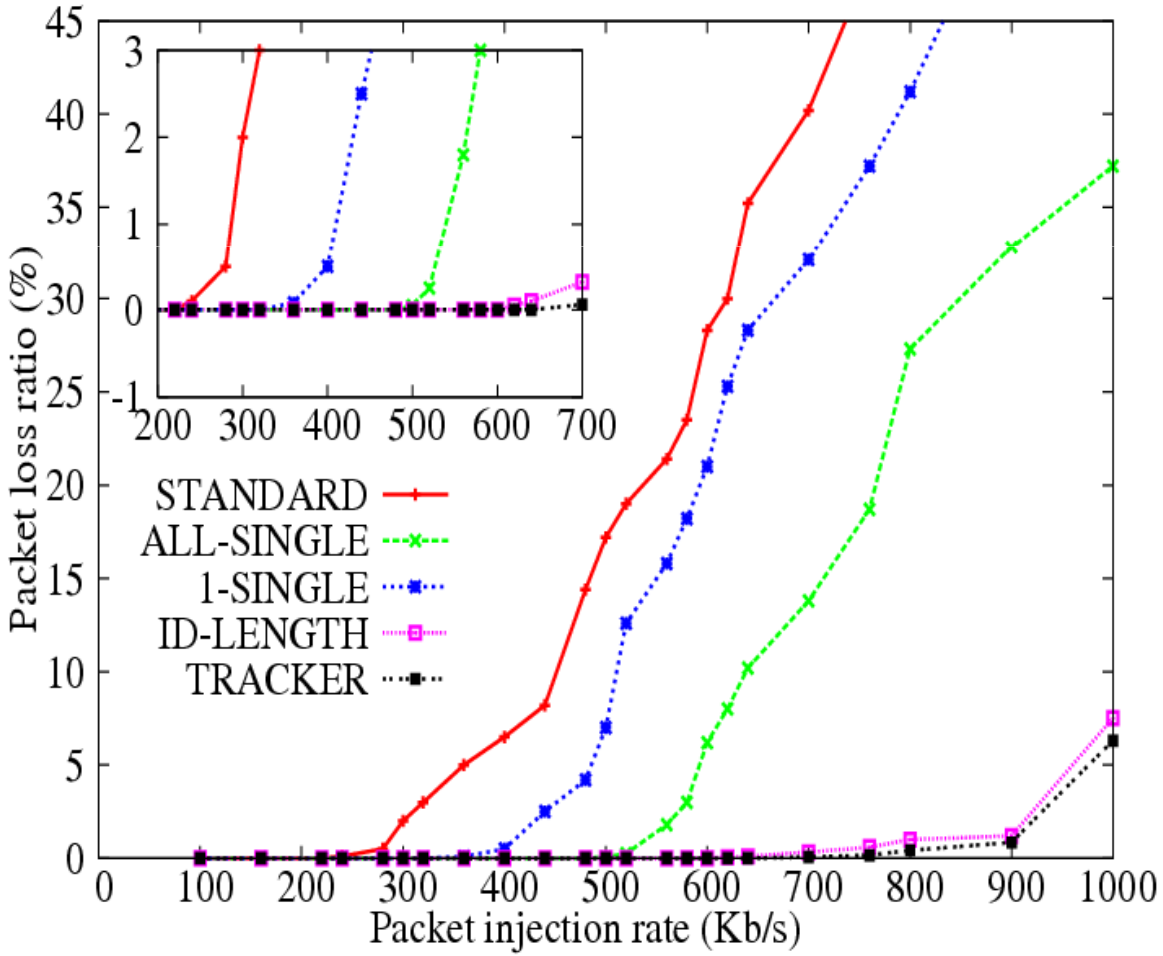

  - 2TREES Protocol :
    - Each group composed of 2 internally disjoint BFS trees rooted at <0..0> and <1..1> [J-C. Bermond & P. Fraigniaud -1994]




  - Critical injection rate: maximum injection rate enabling no packet to be lost

# Degree Hunter: Evaluations (8/8)

- Impact on multicast performances (2/2)
  - Packet loss ratio as a function of the application packet injection rate for the multicast protocol 2TREES (1 group, 100 members).



# Conclusions

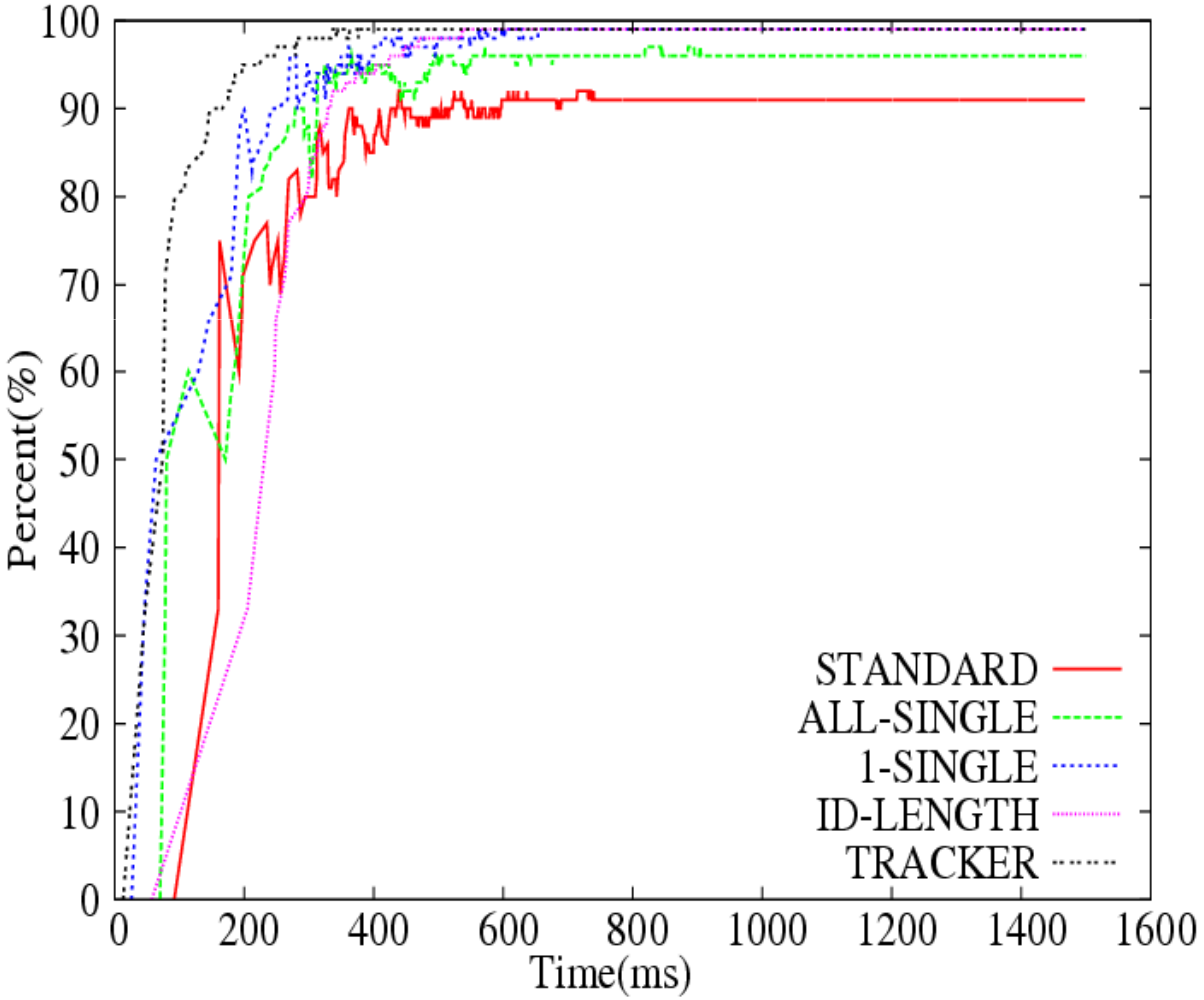
- 3 decentralized mechanisms for load balancing:
  - 1-SINGLE: adapted to file sharing systems
  - ALL-SINGLE: adapted to media streaming
  - ID-LENGTH: adapted to both (but less local)

- Controling dynamicaly the degree and key space of nodes

Thank you for your attention!

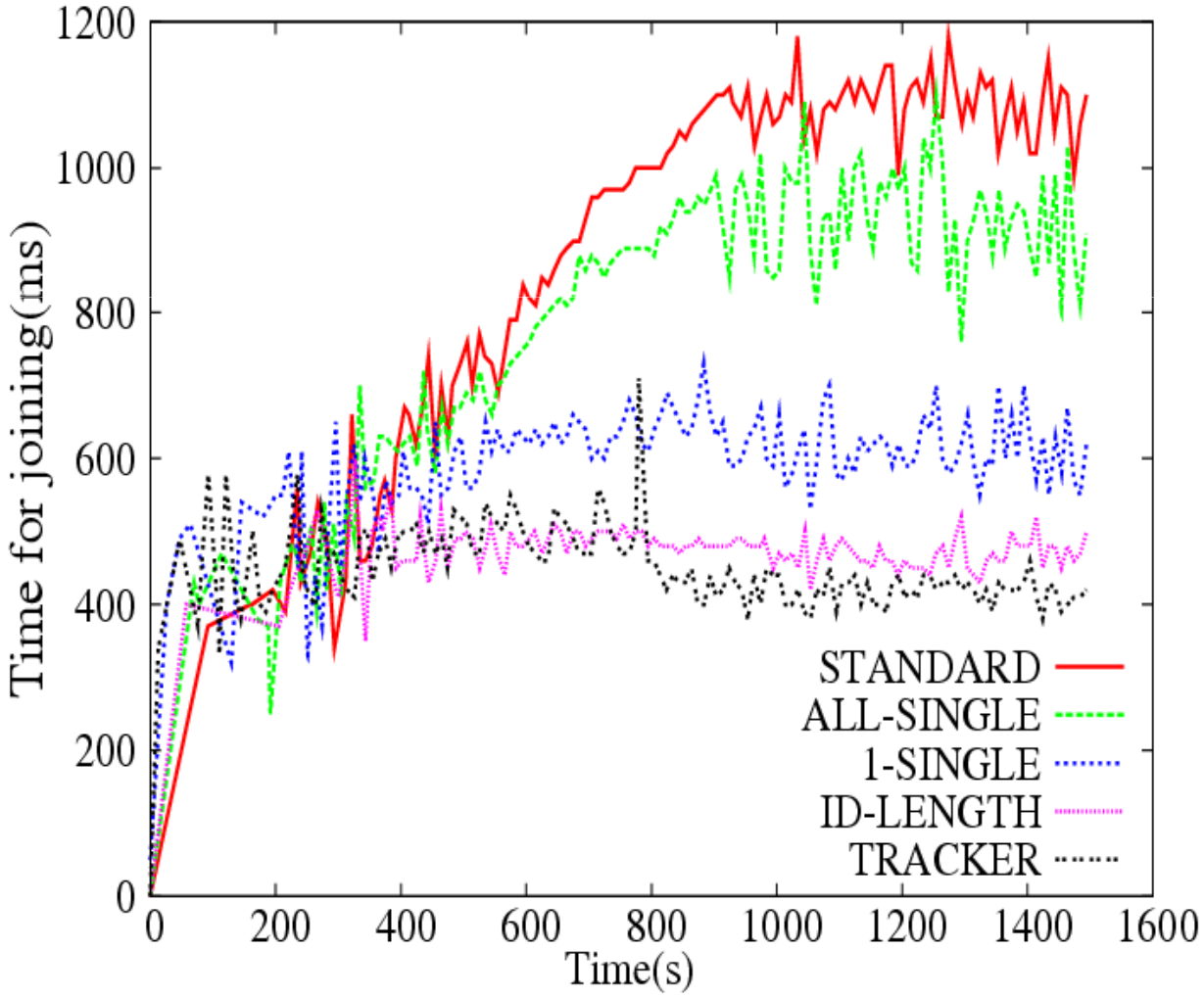
# Backup Slide: Evaluations

- Fraction of the number of nodes having 2 in-neighbors, as a function of time



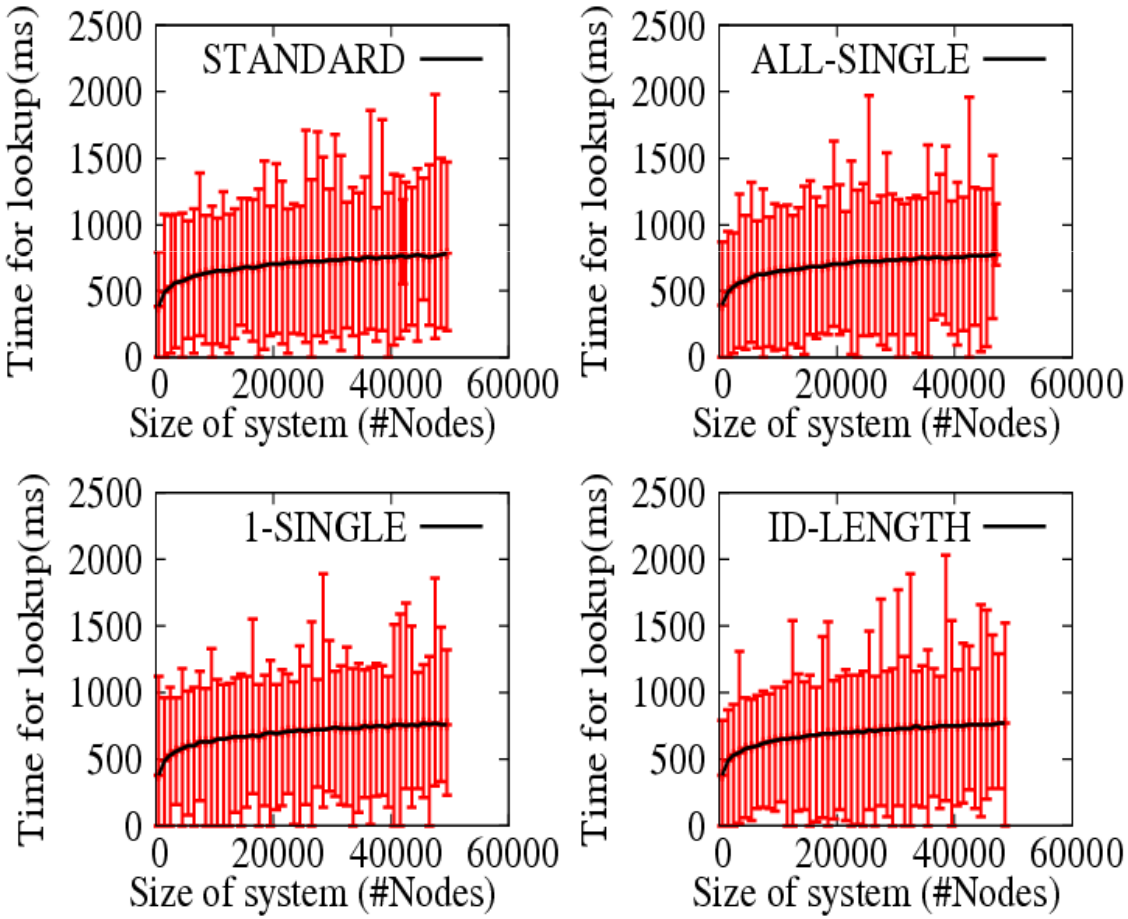
# Backup Slide: Evaluations

- Impact on congestion and latencies
  - Average time for a JOIN as a function of time



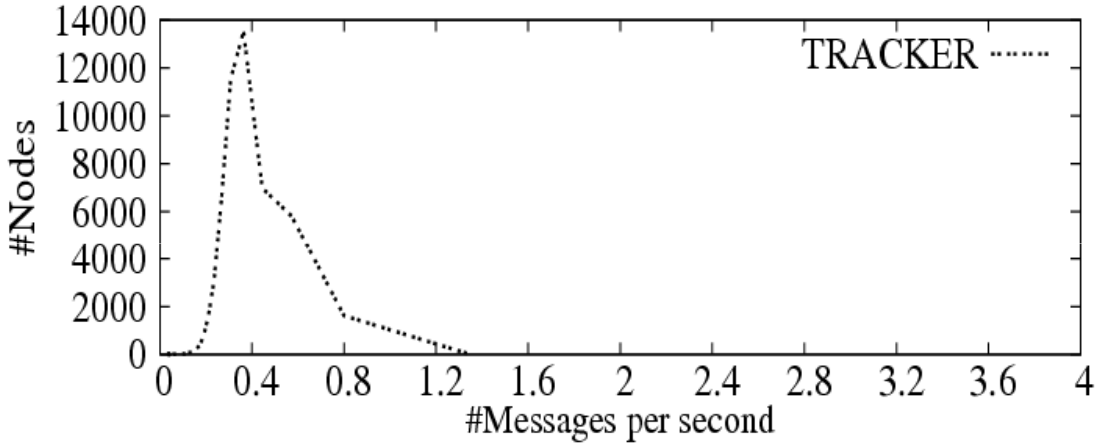
# Backup Slide: Evaluations

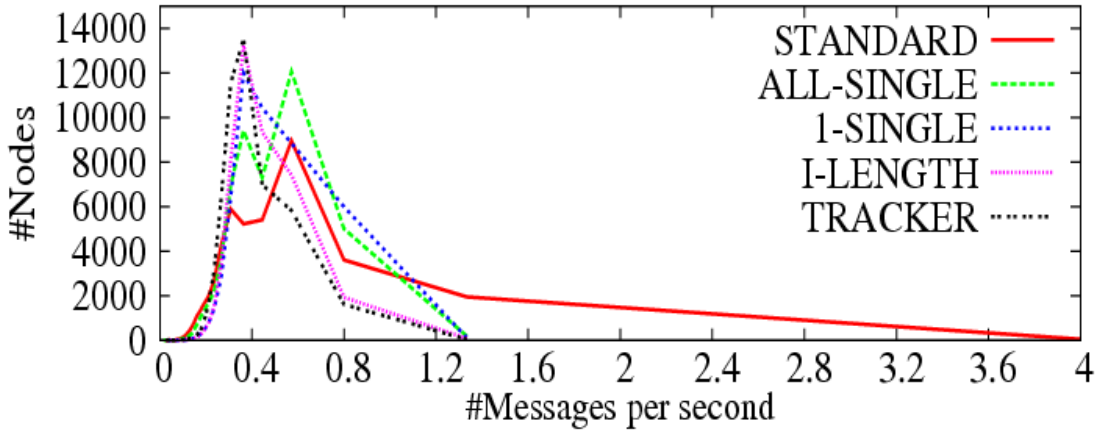
- Impact on congestion and latencies
  - Average and maximum times for LOOKUP/PUBLISH as a function of the size of the system



# Backup Slide: Evaluations

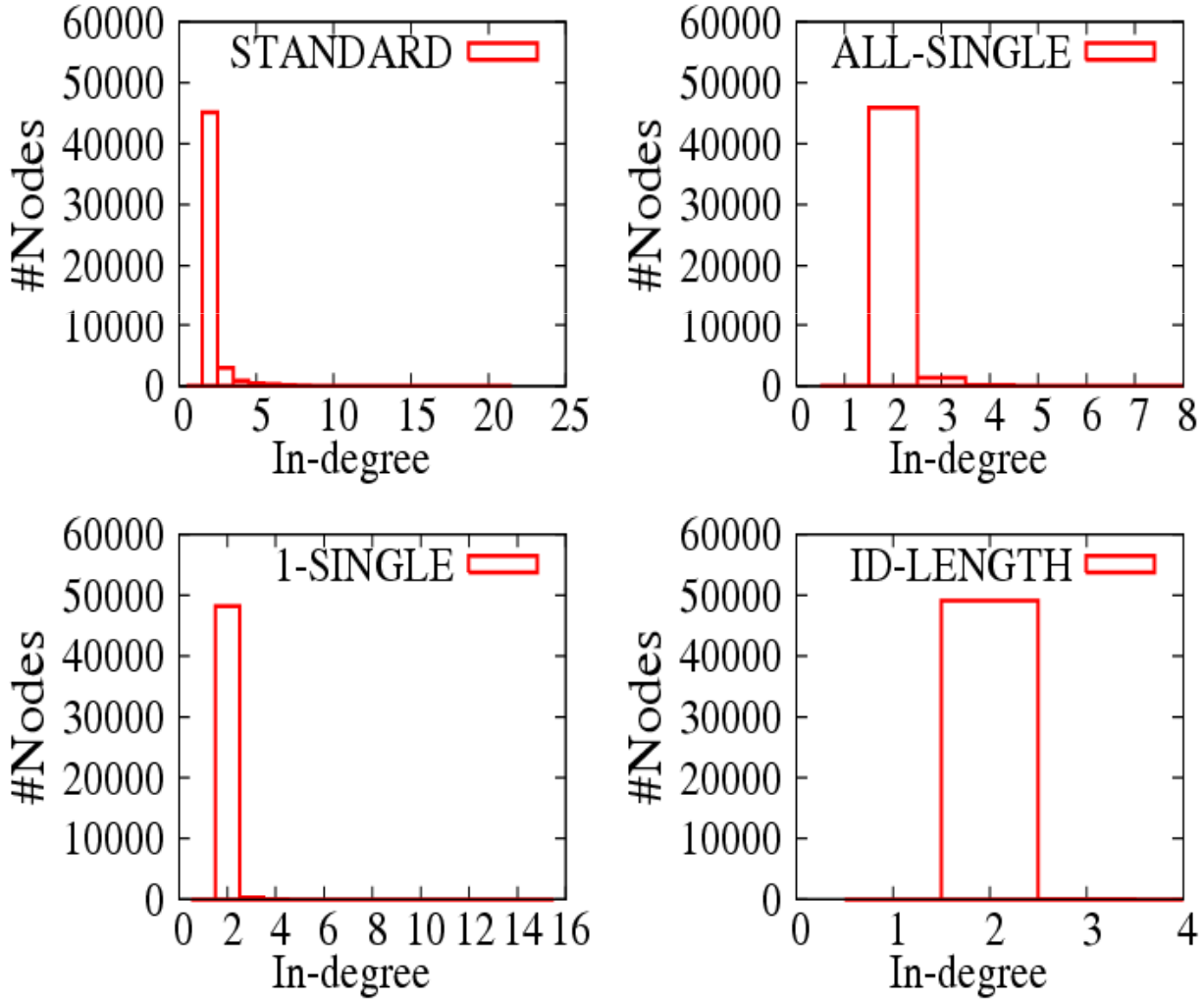
- Impact on congestion and latencies
  - Distribution of nb of LOOKUP/PUBLISH requests par seconde traversing a node per second, at the steady state.



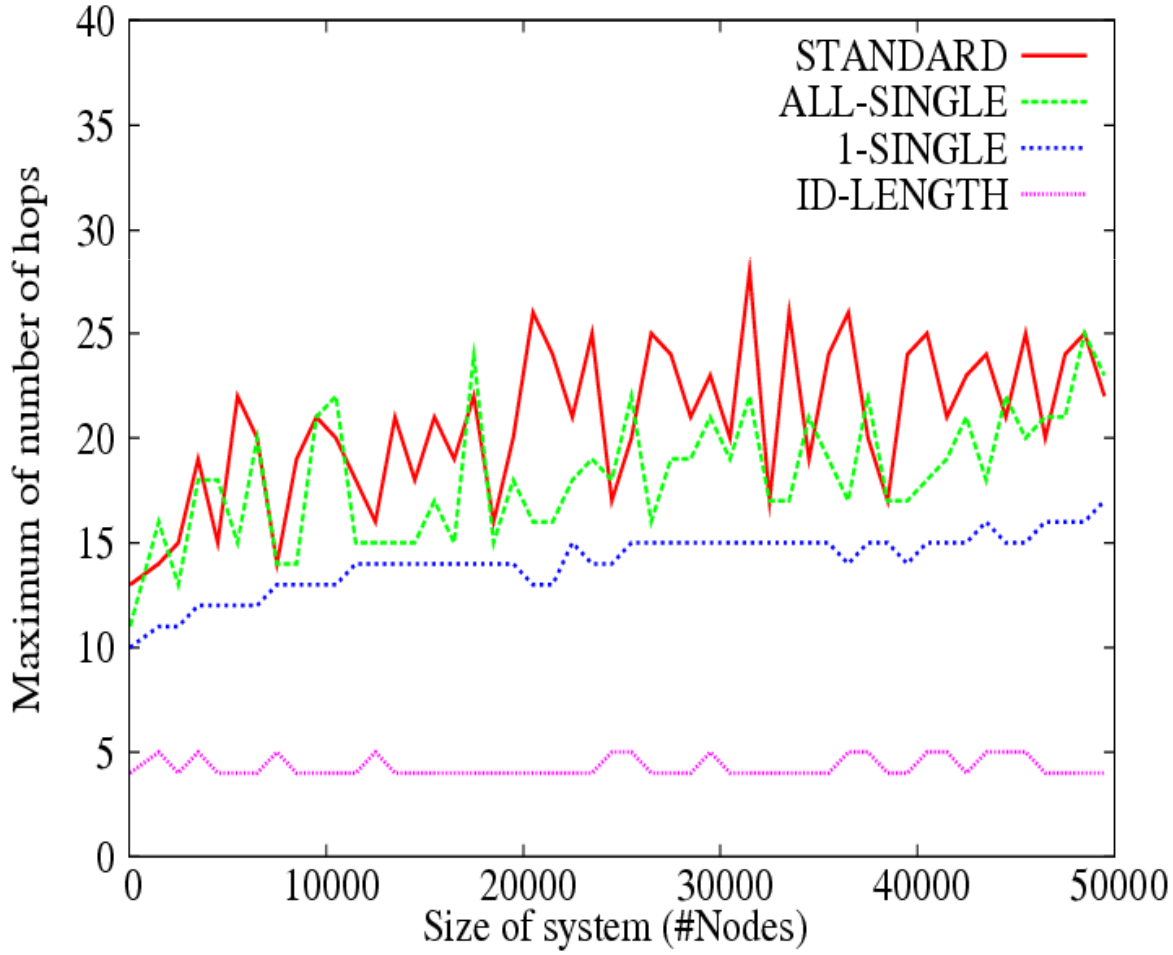
# Backup Slide: Evaluations

- Histogram of the out-degrees at the end of the steady state period



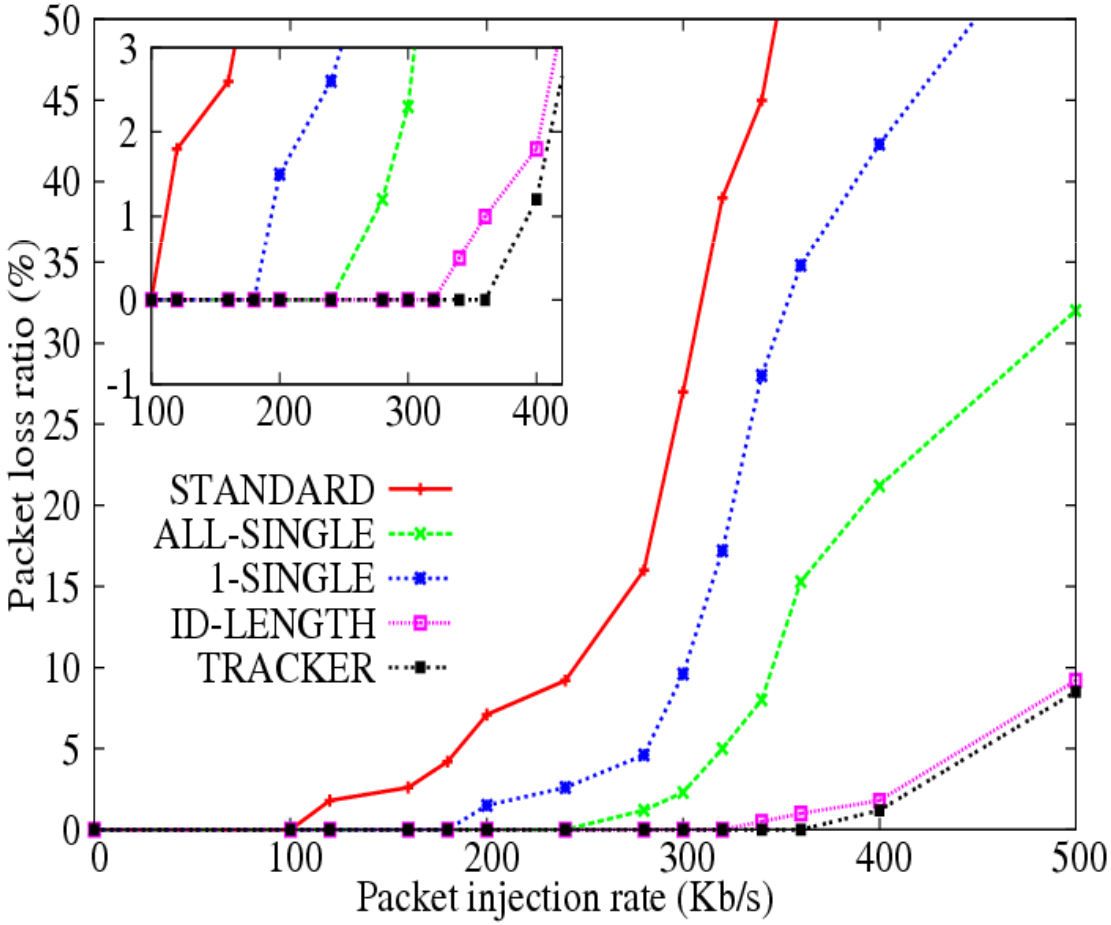
# Backup Slide: Evaluations

- Impact on congestion and latencies
  - Maximum number of hops for a JOIN, as a function of the size of the system



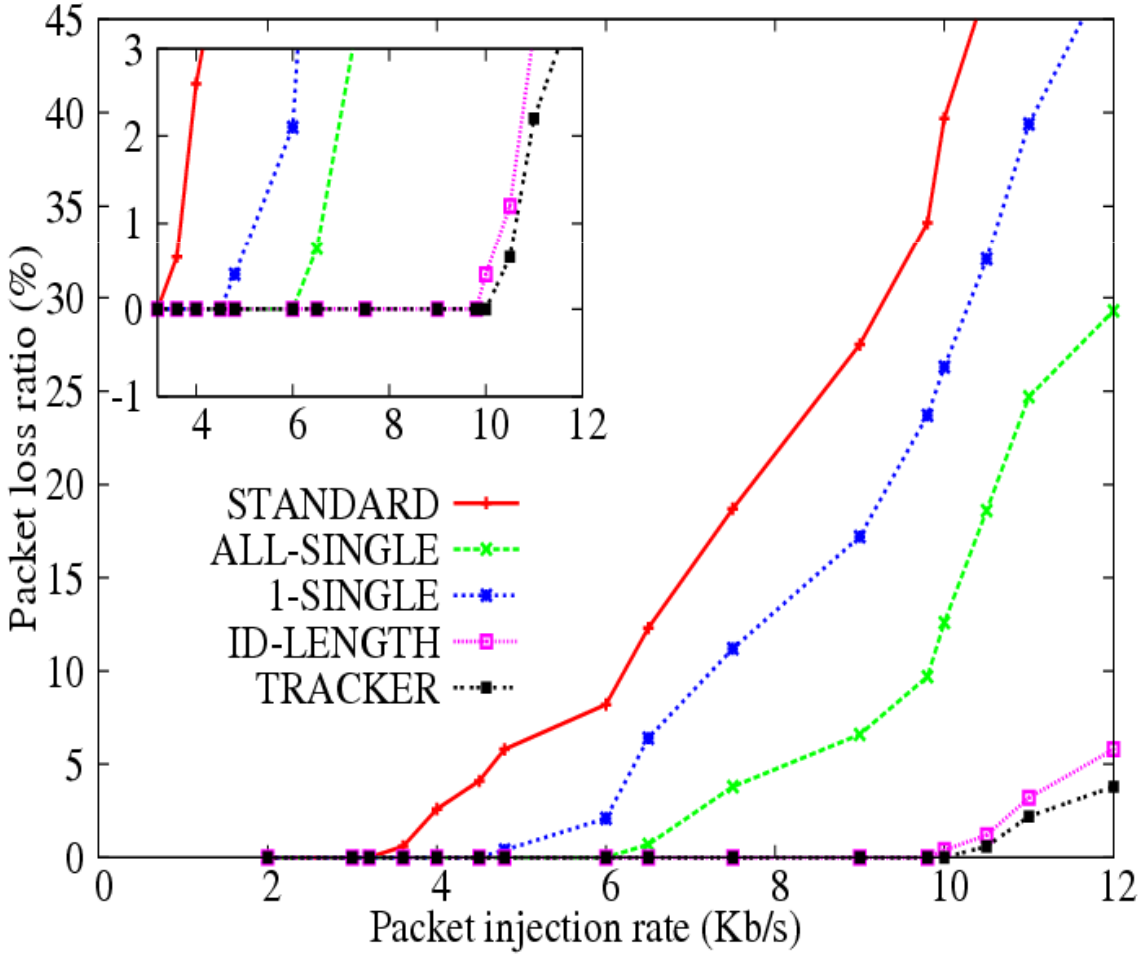
# Backup Slide: Evaluations

- Impact on multicast performances
  - Packet loss ratio as a function of the application packet injection rate for the multicast protocol BFS (1 group, 100 members).



# Backup Slide: Evaluations

- Impact on multicast performances
  - Packet loss ratio as a function of the application packet injection rate for the multicast protocol BFS (100 groups, 100 members/groups).



# Backup Slide: Evaluations

- Impact on multicast performances
  - Packet loss ratio as a function of the application packet injection rate for the multicast protocol 2TREES (100 groups).