Scalable Data Center Multicast

Reporter: 藍于翔
Advisor: 曾學文

Outline

- Introduction
- Scalable multicast
- Conclusion
- Comparison

Reference

- **Exploring Efficient and Scalable Multicast Routing in Future Data Center Networks**
  - Dan Li, Jiangwei Yu, Junbiao Yu, Jianping Wu
  - INFOCOM, 2011 Proceedings IEEE
- **ESM: Efficient and Scalable Data Center Multicast Routing**
  - Dan Li, Yuanjie Li, Jianping Wu, Sen Su, Jiangwei Yu.
  - Networking, IEEE/ACM Transactions on
- **Multicast Fat-Tree Data Center Networks with Bounded Link Oversubscription**
  - Zhiyang Guo , Yuanyuan Yang
  - INFOCOM, 2013 Proceedings IEEE

Introduction

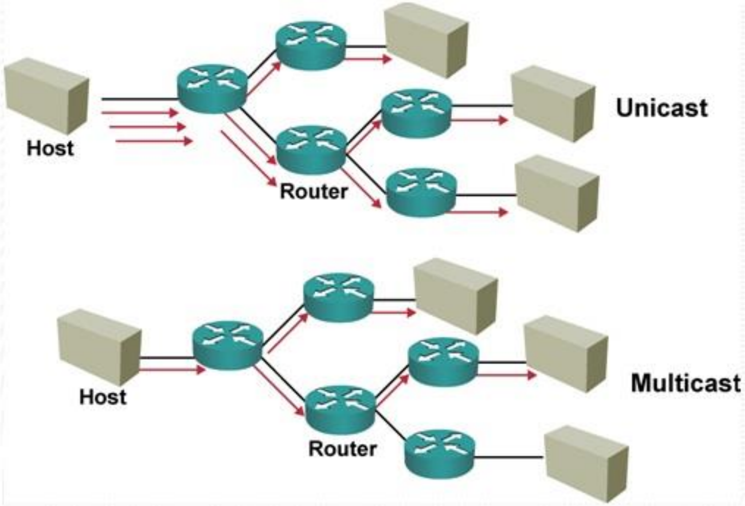
- As the core of cloud services, data centers run online cloud applications.
  - Web search
  - Web mail
  - Interactive games
- Back-end infrastructural computations .
  - Distributed file system

Introduction

- Many of services and applications require one-to-many group communication.
  - Redirecting search queries to multiple indexing servers.
  - Distributing executable binaries to a group of servers participating in Map-Reduce alike cooperative computations.
  - Replicating file chunks in distributed file systems.

Introduction

- Multicast benefits data center group communication.
  - Saving network traffic.
    - Increase the throughput.
  - Reduce the task finish time of delay-sensitive applications.
    - Releasing the sender from sending multiple copies of packets to different receivers.



Scalable multicast

- We explore network-level Multicast routing, which is responsible for building the Multicast delivery tree, in future data center networks.
- Bandwidth-hungry, large-scale data center applications call for efficient and scalable Multicast routing schemes.

Scalable multicast

- There are many equal-cost paths between a pair of servers or switches in data center.
- Multicast trees formed by traditional independent receiver-driven Multicast routing can result in severe link waste compared with efficient ones.

Scalable multicast

- Problem

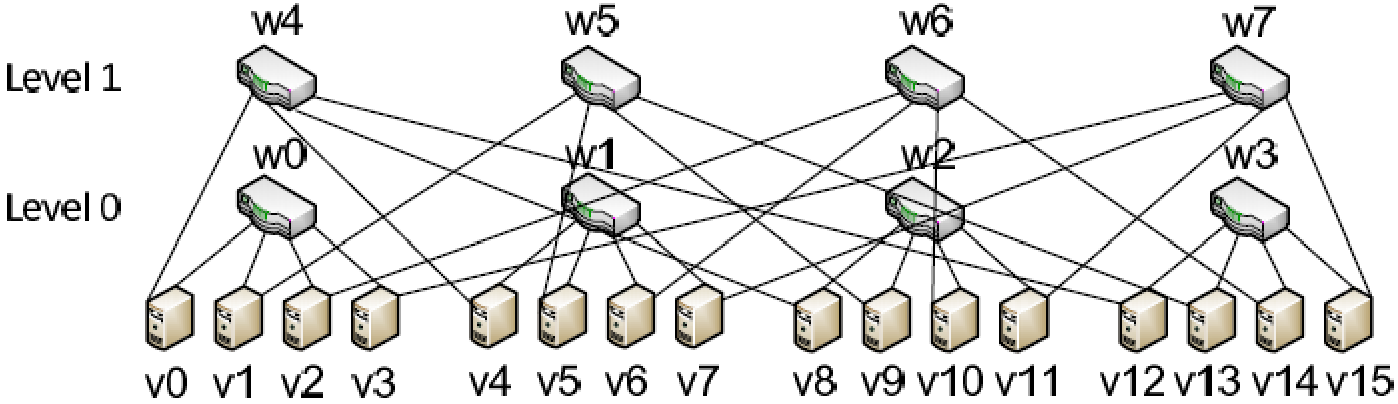


Fig. 1. A BCube(4,1) architecture.

14 links

$$v0 \rightarrow w0 \rightarrow v1 \rightarrow w5 \rightarrow v5,$$
$$v0 \rightarrow w4 \rightarrow v4 \rightarrow w1 \rightarrow v6,$$
$$v0 \rightarrow w4 \rightarrow v8 \rightarrow w2 \rightarrow v9,$$
$$v0 \rightarrow w0 \rightarrow v2 \rightarrow w6 \rightarrow v10.$$

9 links

$$v0 \rightarrow w0 \rightarrow v1 \rightarrow w5 \rightarrow v5,$$
$$v0 \rightarrow w0 \rightarrow v2 \rightarrow w6 \rightarrow v6,$$
$$v0 \rightarrow w0 \rightarrow v1 \rightarrow w5 \rightarrow v9,$$
$$v0 \rightarrow w0 \rightarrow v2 \rightarrow w6 \rightarrow v10.$$

Scalable multicast

- Source Driven Tree Building
  - For example, if the sender is v0 and the receiver set is {v1,v5,v9,v10,v11,v12,v14}

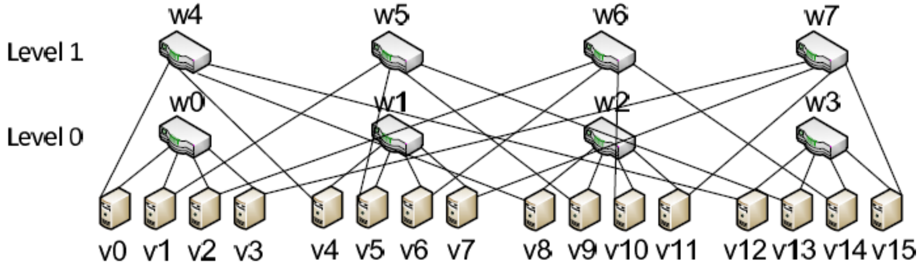
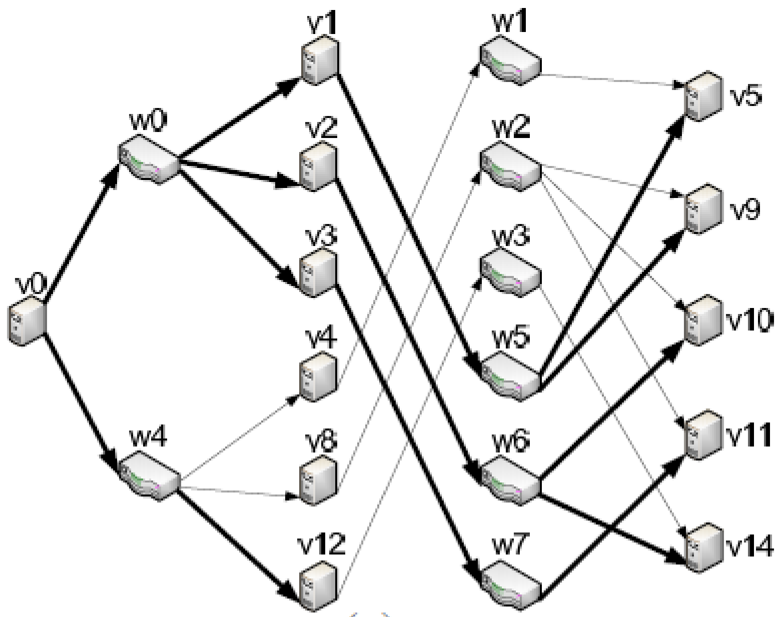


Fig. 1. A BCube(4,1) architecture.



(a)

Scalable multicast

- Source Driven Tree Building
  - For example, if the sender is v0 and the receiver set is {v1,v5,v9,v10,v11,v12,v14}

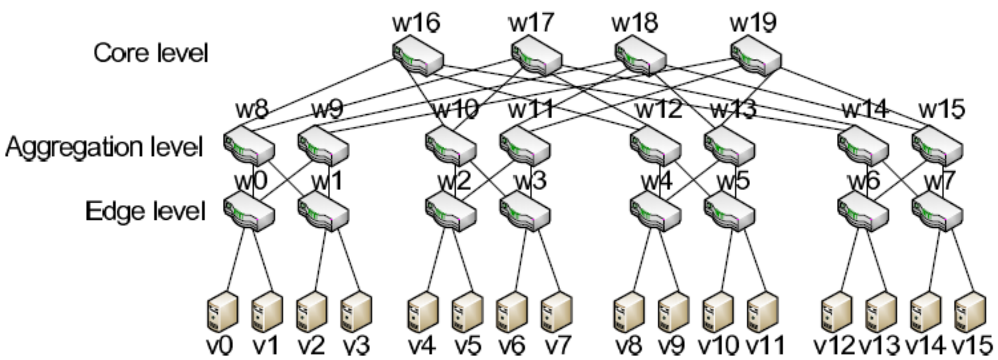
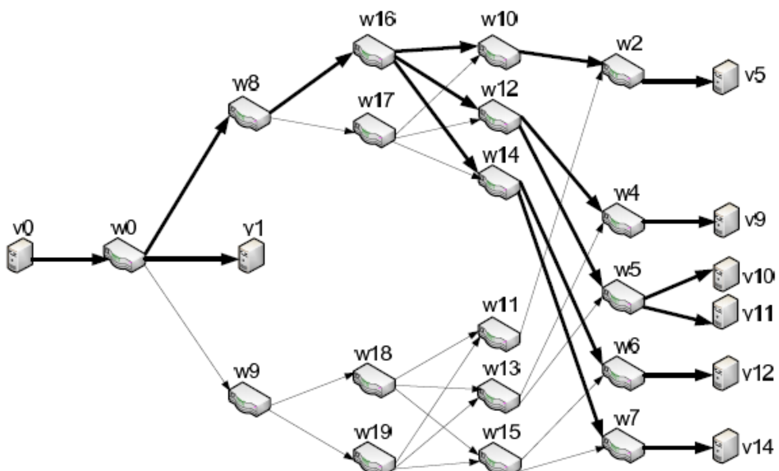


Fig. 2. A PortLand architecture with 4-port switches.



(b)

Scalable multicast

A **covers** B (or B **is covered by** A): For any two node sets A and B in a group spanning graph, we call A *covers* B (or B is *covered* by A) if and only if for each node $j \in B$, there exists a directed path from a node $i \in A$ to j in the group spanning graph.

A **strictly covers** B (or B **is strictly covered by** A): If A covers B and any subset of A does not cover B, we call A *strictly covers* B (or B is *strictly covered* by A).

Scalable multicast

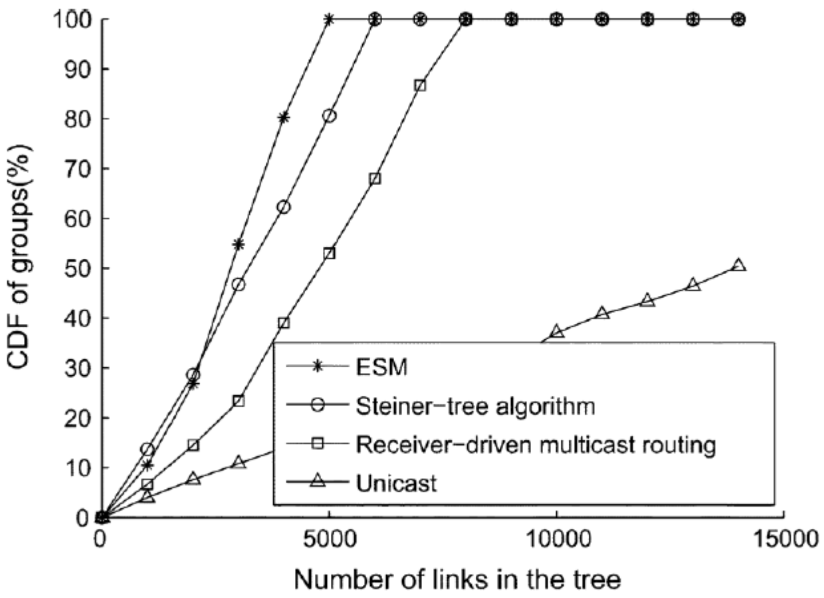
- Source-to-receiver expansion
  - Source Driven + Strictly covering
- Benefit
  - Many unnecessary intermediate switches used in receiver-driven Multicast routing are eliminated.
  - Source-to-receiver latency is bounded by the number of stages of the group spanning graph.

Scalable multicast

- Dynamical Receiver Join/Leave.
  - Given multicast receivers dynamically join or leave a group, the multicast tree should be rebuilt to encompass group dynamics.
  - ESM can gracefully embrace this case.
    - Receiver join/leave does not change the source-to-end paths of other receivers in the group

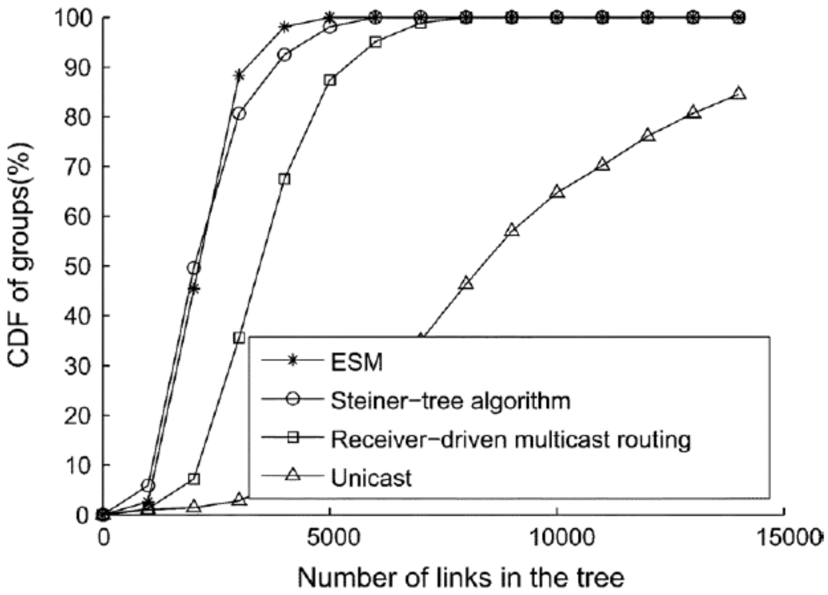
Scalable multicast

- Number of links - BCube



(a)

Uniform group size distribution

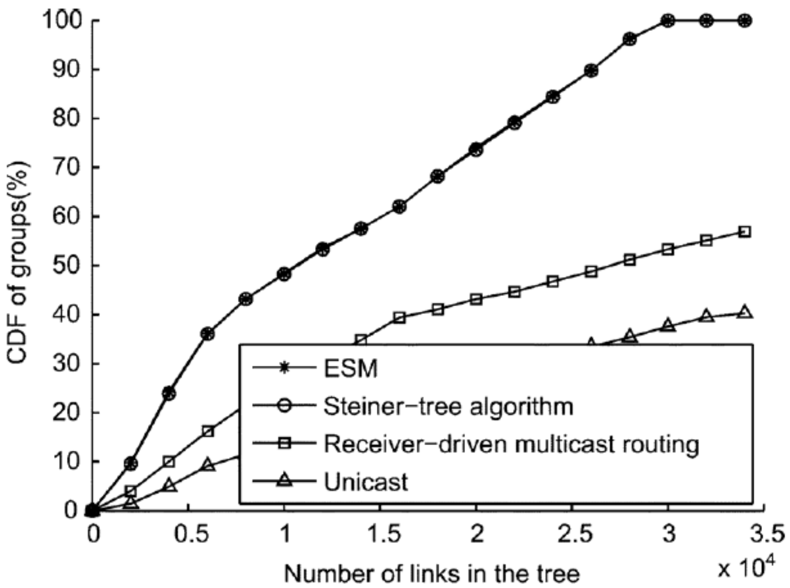


(b)

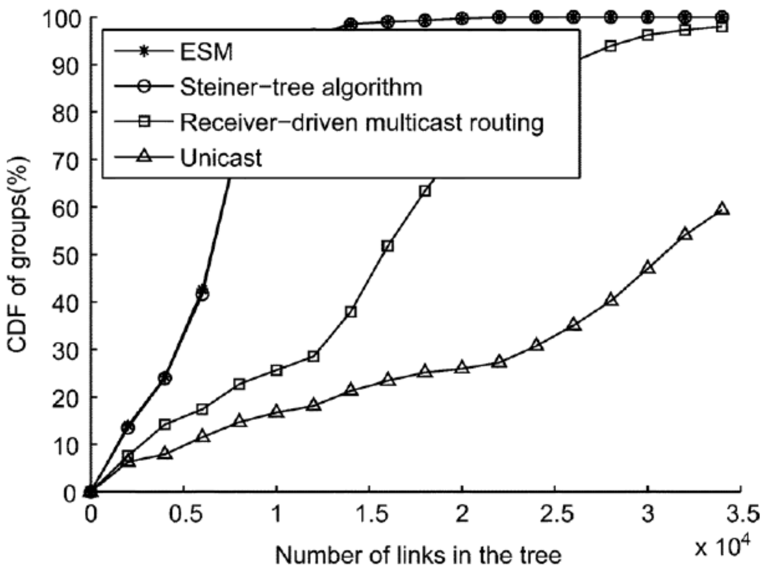
Power-law group size distribution

Scalable multicast

- Number of links - Fat-Tree



(a)



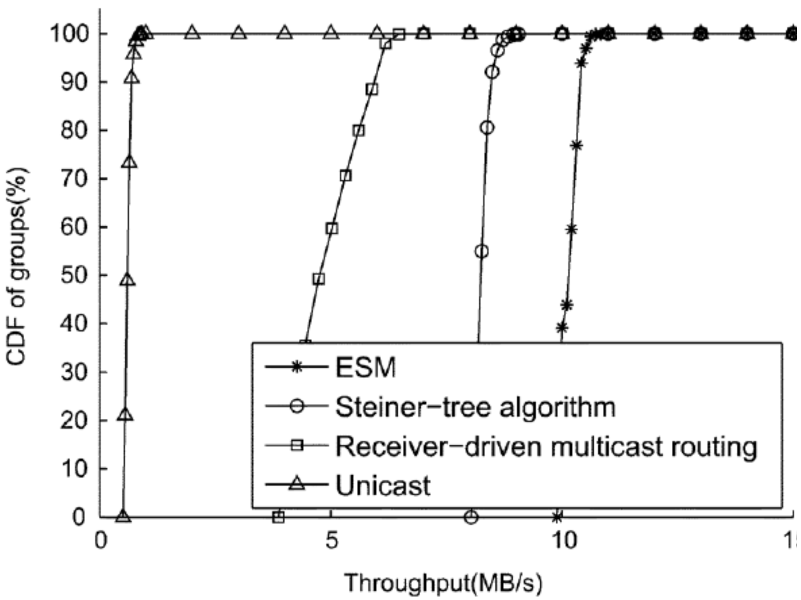
(b)

Uniform group size distribution

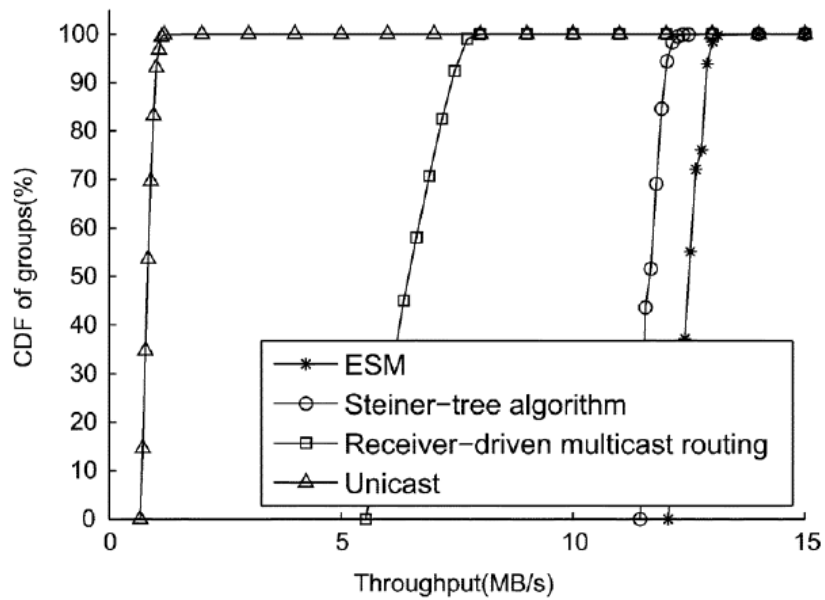
Power-law group size distribution

Scalable multicast

- Throughputs - BCube



(a)



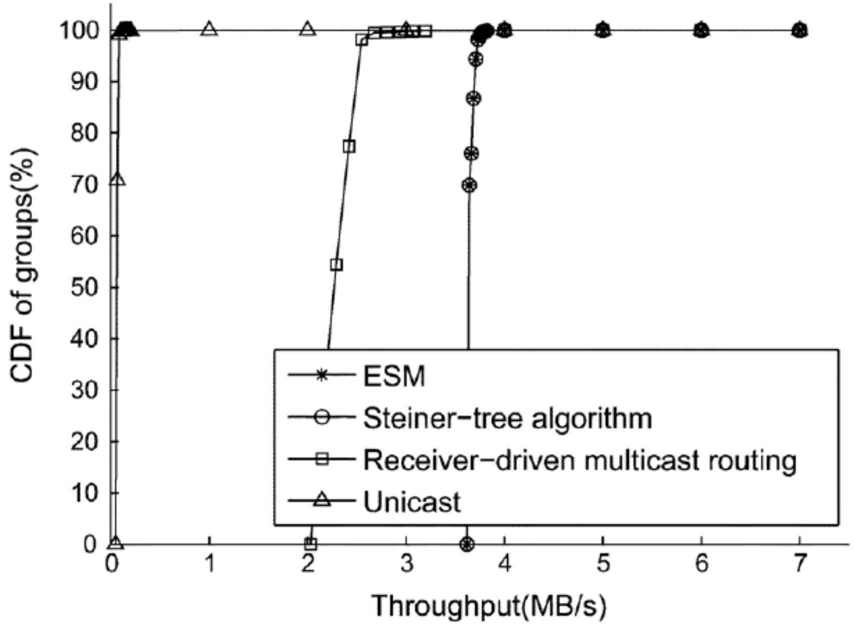
(b)

Uniform group size distribution

Power-law group size distribution

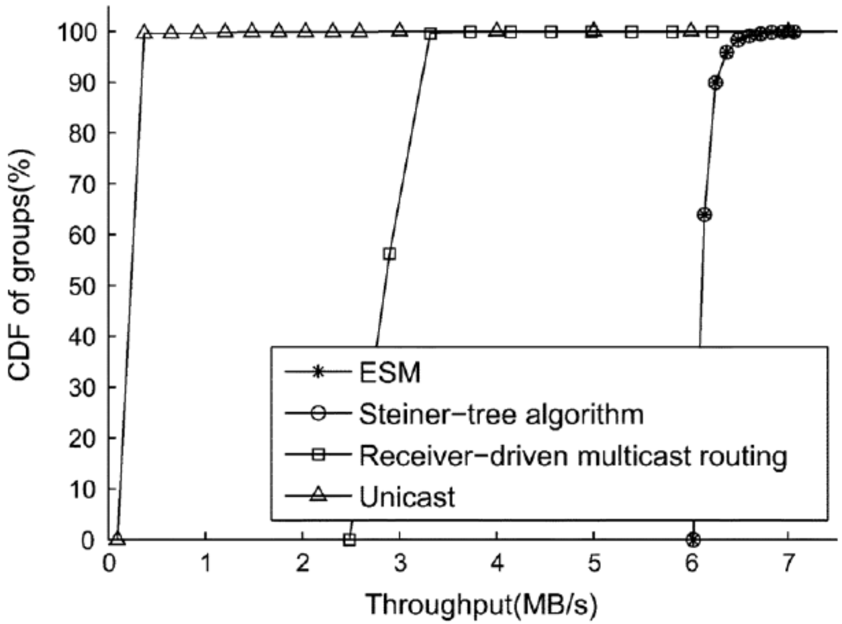
Scalable multicast

- Throughputs - Fat-Tree



(a)

Uniform group size distribution

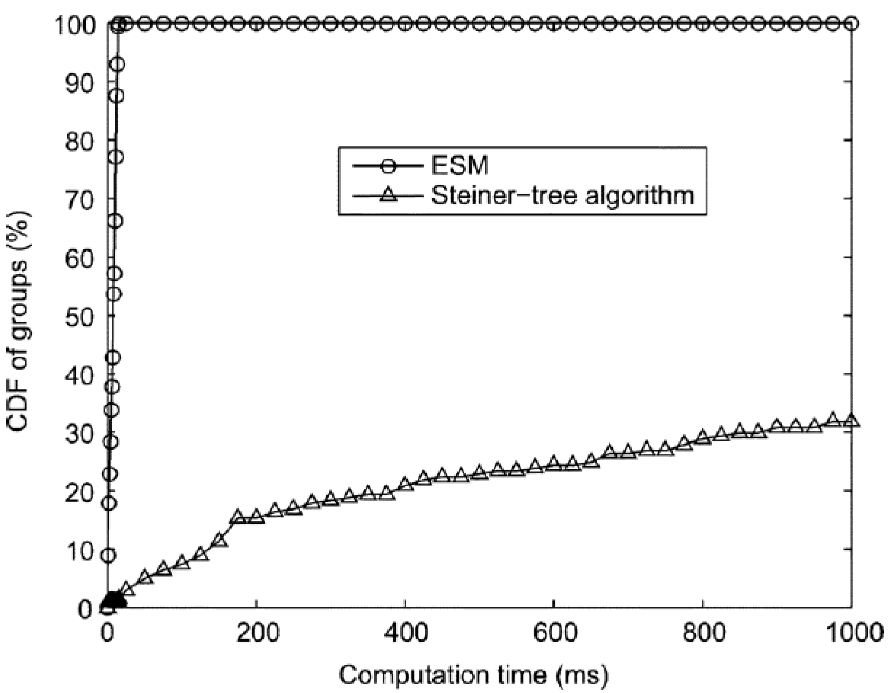


(b)

Power-law group size distribution

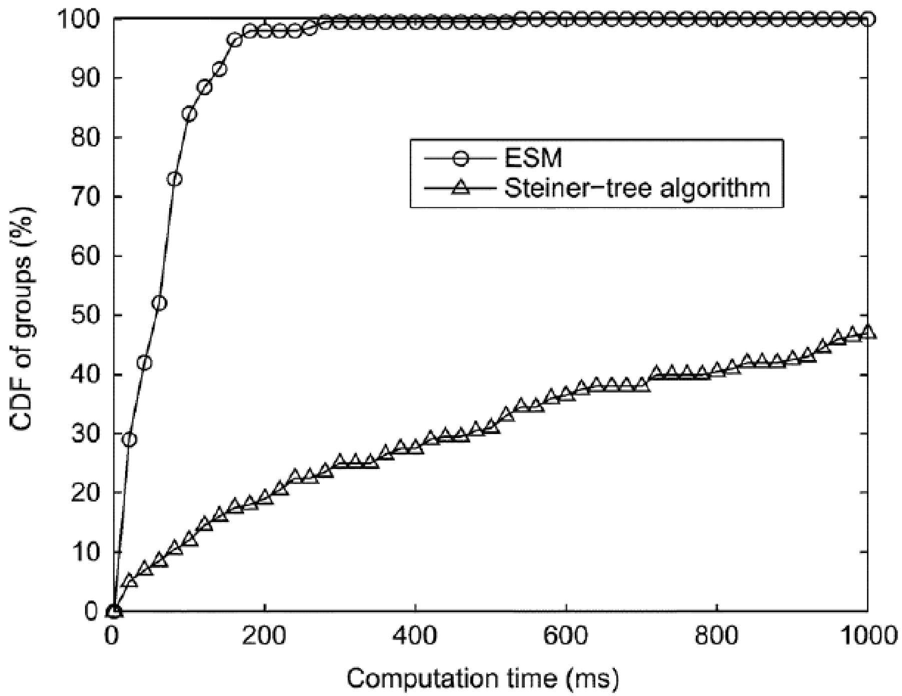
Scalable multicast

- Computation time



(a)

BCube



(b)

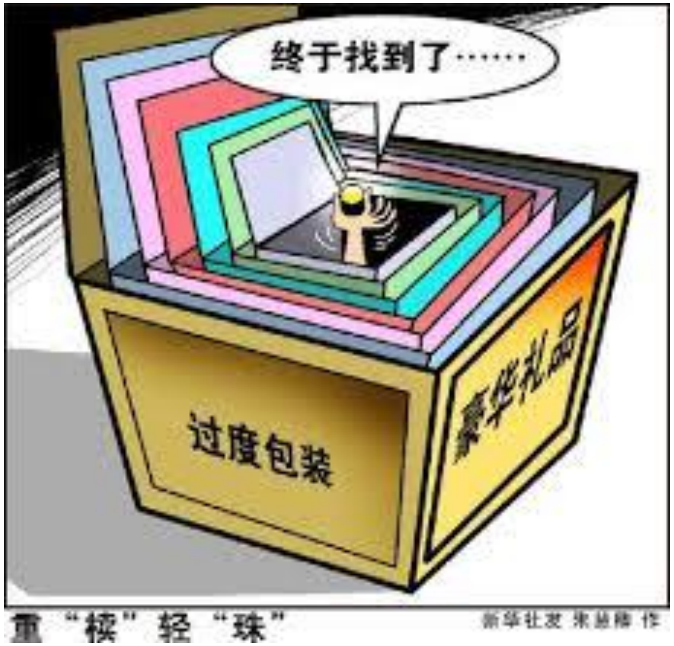
Fat-Tree

Scalable multicast

- The memory space of the routing table in low-end commodity switches is relatively narrow.
  - Switches can hold no more than 1500 multicast group states.

Scalable multicast

- In-packet Bloom Filter eliminates the necessity of in-switch routing entries.
  - Result in bandwidth waste.



Scalable multicast

- ESM combines both in-packet Bloom Filter and in-switch entries.
  - In-packet Bloom Filters are used for small-sized groups to save routing space in switches.
  - Routing entries are installed into switches for large groups to alleviate the bandwidth overhead.

Scalable multicast

- The bandwidth waste of in-packet Bloom Filter comes from two aspects.
  - The Bloom Filter field in the packet brings network bandwidth cost.
  - False-positive forwarding by Bloom Filter causes traffic leakage

Scalable multicast

- Bandwidth Overhead Ratio

length of the in-packet Bloom Filter

the number of actual links covered by Bloom Filter based forwarding

$$r = \frac{t * f + (c - t) * p}{t * (p - f)}.$$

number of links in the multicast tree

packet length

Scalable multicast

- Bandwidth Overhead Ratio

additional traffic in the multicast tree resulting from the addition of the Bloom Filter field in the packet

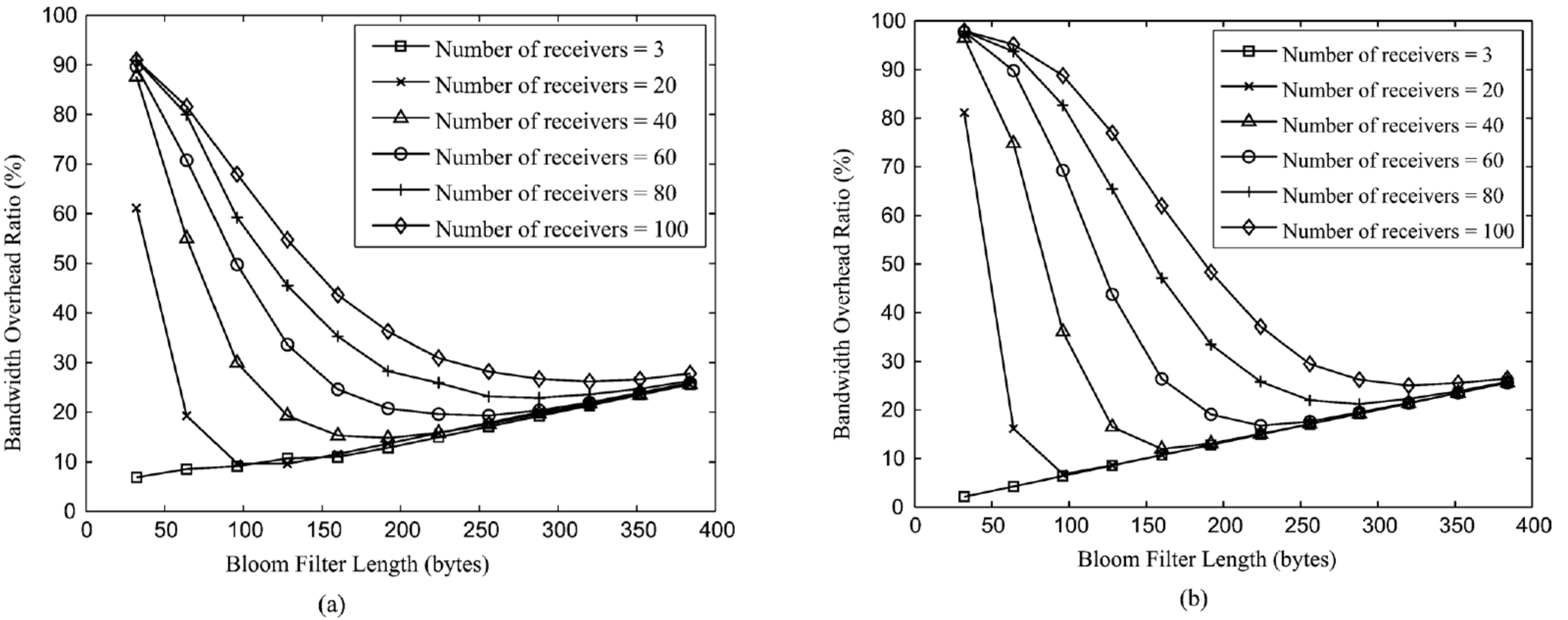
total traffic carried by the links beyond the tree

$$r = \frac{t * f + (c - t) * p}{t * (p - f)}.$$

actual payload traffic on the tree

Scalable multicast

- Bandwidth Overhead Ratio

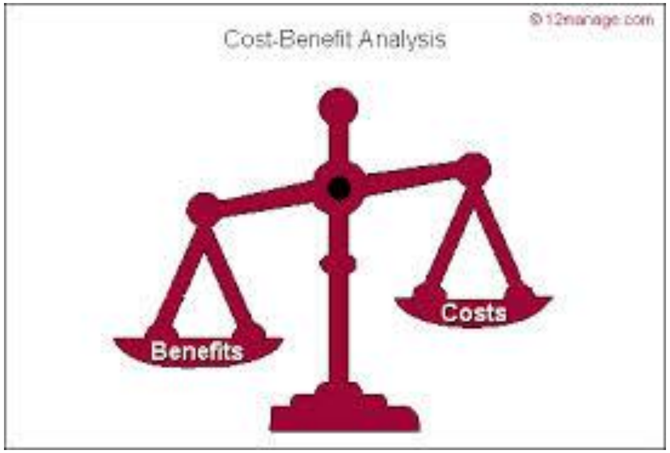


BCube

Fat-Tree

Scalable multicast

- In here, we approach the multicast traffic load balance problem in fat-tree DCNs from a novel angle, aiming to find the most cost-effective way to build a fat-tree DCN with bounded link subscription ratio.



Scalable multicast

- Core switches are expensive high-end commodity switch modules with large port count and high port speed, we analyze the minimum number of core switches required to achieve bounded oversubscription in a multicast fat-tree DCN in order to ensure the cost-effectiveness of data centers.

Scalable multicast

- Architecture

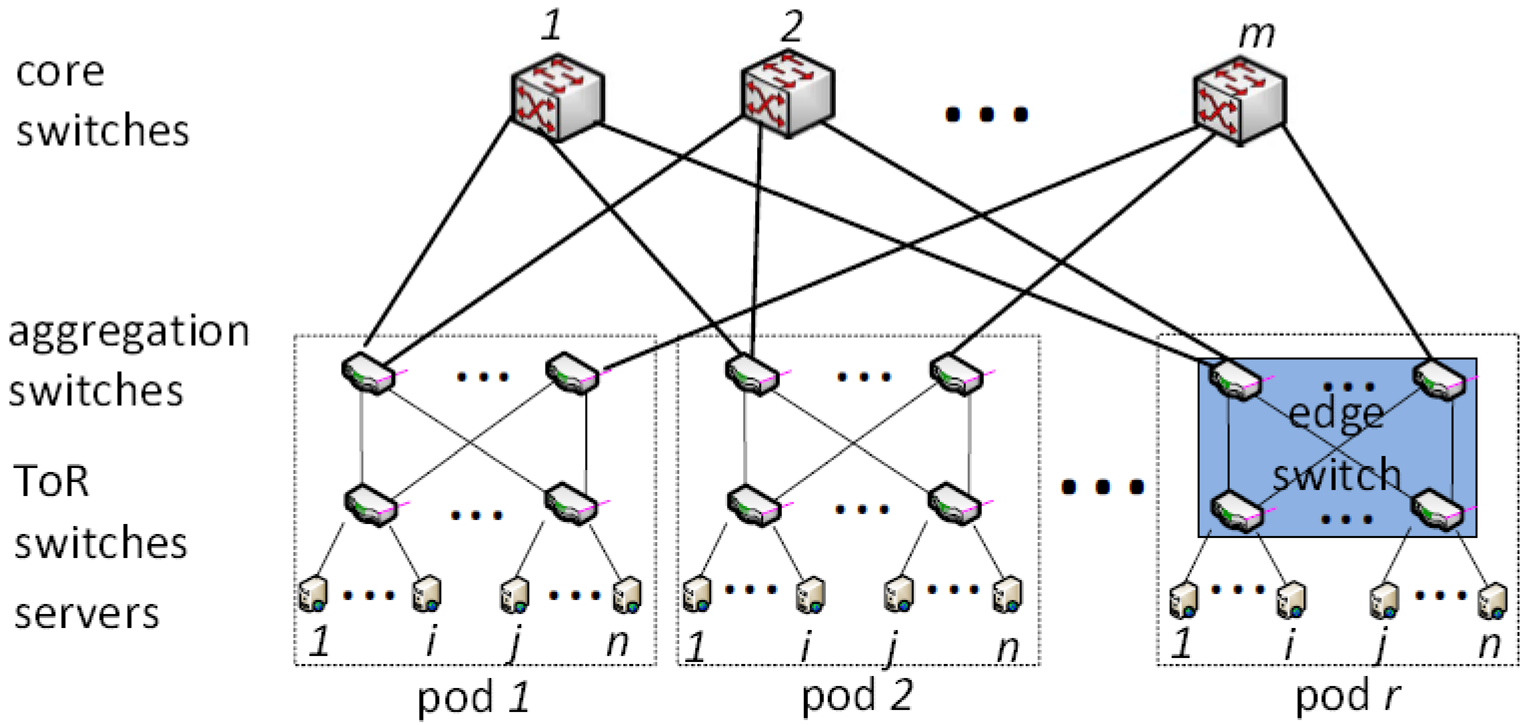


Fig. 1. A level-3 fat-tree DCN, which can be reduced to a level-2 fat-tree $ftree(m, n, r)$ if ToR switches and aggregation switches in each pod are considered as a nonblocking edge switch.

Scalable multicast

- Number of core switches

不可用

$$m > \min_{1 \leq x < r} \max_{\omega \in B} \{ J_x(O,\omega) + J_1(O,\omega)(r-1)^{1/x} \}$$

可用

Scalable multicast

- Number of core switches

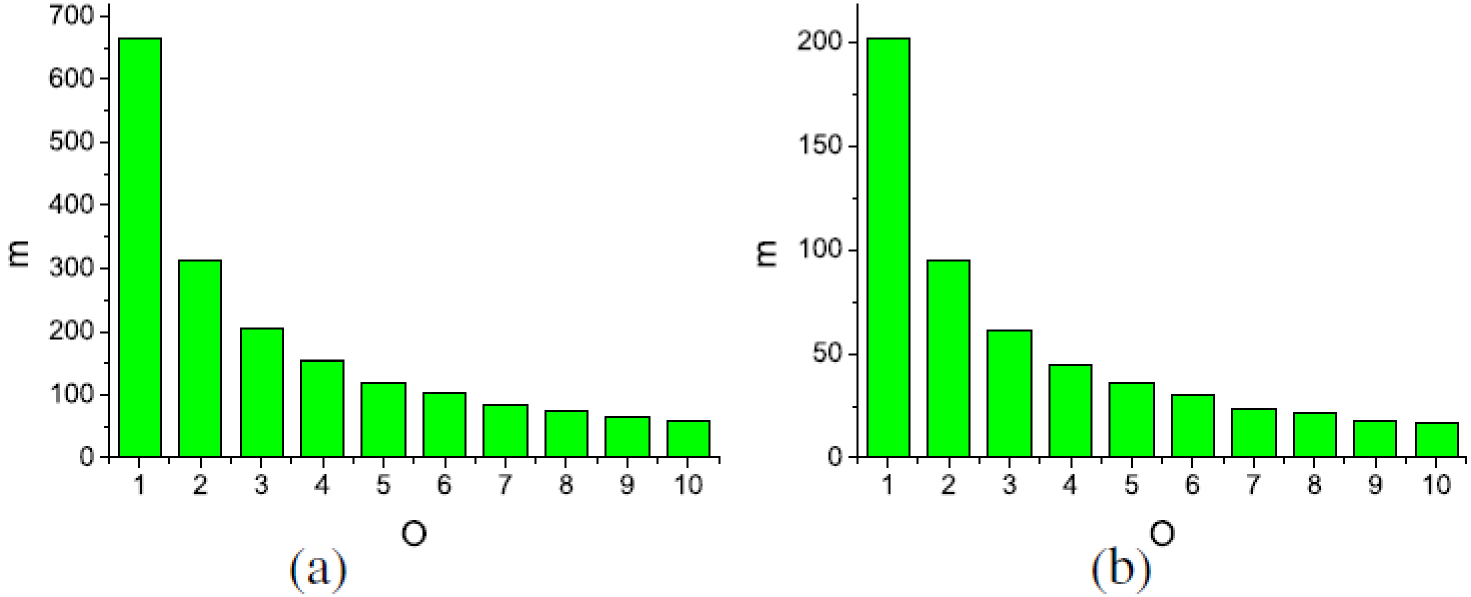


Fig. 2. Minimum number of core switches required, m, in an O-oversubscribed multicast $ftree(m, n, r)$ DCN with respect to different over-subscription bounds O. (a) $n = 1024, r = 25$. (b) $n = 256, r = 100$.

Conclusion

- Source-to-receiver tree building eliminated unnecessary intermediate switches used in receiver-driven .
- ESM combines both in-packet Bloom Filter and in-switch entries to achieve scalable.
- Finding minimum number of core switches in load balance ensure the cost-effectiveness.
- With the above three points can construct an efficient low-cost scalable data center.

Comparison

	Routing Optimization	Scalable	Load Balance	Cost
[1]	✓	✓		
[2]	✓	✓		
[3]			✓	✓