

Resource Usage Prediction for Application-Layer Multicast Networks

Genge Béla and Haller Piroska

“Petru Maior” University of Tîrgu Mureş, Romania
{bgenge, phaller}@engineering.upm.ro

August 27, 2010

Presentation overview

- Introduction to Application-Layer Multicast
- Problem statement
- Predictive model
- Model evaluation
- Conclusions and future work

Application-Layer Multicast (ALM)

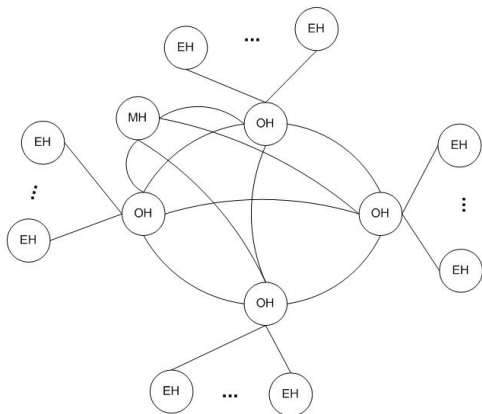
- The main goal of group communication is to enable the exchange of information between group members
- One of the main application of group communications is in the field of multicast: IP and ALM
- After nearly a decade of research in the field of IP multicast, it was never fully adopted because of several technical and administrative issues
- One of the directions that has been clearly adopted over the last few years is ALM

ALM - Existing research and proposal

- Existing research:
 - Construction of the overlay network (K. Ragab et al., 2009)
 - Routing algorithms (S. Jaggi et al., 2005)
 - Optimizing transfer by measuring connection latency (S. Ratnasamy et al., 2001)
- Our proposal:
 - Optimize distribution of end-hosts to an already constructed overlay
 - In our previous work we proposed a heuristic algorithm for end-host distribution (P. Haller et al., 2005)
 - In this paper we extend it with a predictive resource usage model

Proposed multicast network topology

- We use a complete graph overlay topology consisting of:
 - End-Hosts (EH)
 - Overlay-Hosts (OH)
 - Monitor-Hosts (MH)
- Advantages: simplified routing and overlay construction
- Disadvantages: increased number of connections that grows exponentially with the number of OHs; single point of failure of MH
- Applicability: leaf overlay for hierarchical topologies (W. Yong et al., 2009)



Overview of EH distribution algorithm

Algorithm 1

Let Req be the set of OH - measured latency pairs (oh, l)

Let $Chosen_{OH}$ be the set of chosen OHs

Let $eh_{OH} = OH_1$ be the OH chosen for this EH

Let $l_{min} = MAX_VAL$ be the minimal computed latency

for all $(oh, l) \in Req$ **do**

 Let $l = Latency(Chosen_{OH} \cup \{oh\})$

if $l < l_{min}$ **then**

$eh_{OH} = oh$

$l_{min} = l$

end if

end for

$Chosen_{OH} = Chosen_{OH} \cup \{eh_{OH}\}$

Overview of EH distribution algorithm - drawbacks

- Uses only EH count in the distribution process
 - ↳ OH memory and CPU usage can increase dramatically (while maintaining a low EH count)
 - ↳ It can affect the overall performance of the overlay
 - ↳ It can lead to failure of the overlay

Extended EH distribution algorithm

Algorithm 2 (part 1)

Let Req be the set of OH - measured latency pairs (oh, l)

Let $Chosen_{OH}$ be the set of chosen OHs

Let $eh_{OH} = OH_1$ be the OH chosen for this EH

Let $l_{min} = MAX_{VAL}$ be the minimal computed latency

Let (Min, \leq) be a totally ordered set of minimum latency OHs

Let $found = 0$ be a flag denoting that we have found a proper OH

for all $(oh, l) \in Req$ **do**

 Let $l = Latency(Chosen_{OH} \cup \{oh\})$

if $l < l_{min}$ **then**

$Min = Min \cup \{< OH, l >\}$

$l_{min} = l$

end if

end for

Extended EH distribution algorithm

Algorithm 2 (part 2)

```

for  $i = \overline{1, |Min|}$  do
  if  $Pred_{MEM}(Min_i^1) < MAX_{MEM}$  then
    if  $Pred_{CPU}(Min_i^1) < MAX_{CPU}$  then
       $eh_{OH} = oh$ 
       $found = 1$ 
    @EndFor
  end if
end if
end for
if  $found = 0$  then
   $eh_{OH} = Min_1^1$ 
end if
 $Chosen_{OH} = Chosen_{OH} \cup \{eh_{OH}\}$ 

```

MIMO model

General MIMO model

$$\begin{aligned}
 mem(t+1) &= a_1 mem(t) + b_1 meh(t) + c_1 seh(t) \\
 cpu(t+1) &= a_2 cpu(t) + b_2 meh(t) + c_2 seh(t)
 \end{aligned}
 \tag{1}$$

Criteria functions

$$\begin{aligned}
 J(a_1, b_1, c_1) &= \sum_{t=1}^N (mem'(t+1) - mem(t+1)), \\
 J(a_2, b_2, c_2) &= \sum_{t=1}^N (cpu'(t+1) - cpu(t+1))
 \end{aligned}
 \tag{2}$$

- $meh(t)$ is the measuring EH
- $seh(t)$ is the streaming EH count
- a_i, b_j are model parameters

MIMO model (cont'd)

Parametric MIMO model

$$\begin{aligned}
 P_{mem}^{\alpha}(t) &= a_1 mem(t) + b_1 meh(t) + c_1(seh(t) + \alpha), \\
 P_{cpu}^{\alpha}(t) &= a_2 cpu(t) + b_2 meh(t) + c_2(seh(t) + \alpha)
 \end{aligned}
 \tag{3}$$

Quantified resource prediction

$$\begin{aligned}
 Q_{mem}(t) &= \sum_{i=1}^{\beta_t} (P_{mem}^{\alpha_i}(t) - mem(t)) + mem(t), \\
 Q_{cpu}(t) &= \sum_{i=1}^{\beta_t} (P_{cpu}^{\alpha_i}(t) - cpu(t)) + cpu(t)
 \end{aligned}
 \tag{4}$$

- α is the number of new requests
- β_t is the total number of requests between t and $t + 1$

MIMO model (cont'd)

Current predicted resource usage

$$C_{mem}(t) = mem(t) + \sum_{k=1}^{t-1} \lambda_{mem}^{t-k} \Delta Q_{mem}(k),$$

$$C_{cpu}(t) = cpu(t) + \sum_{k=1}^{t-1} \lambda_{cpu}^{t-k} \Delta Q_{cpu}(k) \quad (5)$$

Delta values

$$\Delta Q_{mem}(k) = Q_{mem}(k) - mem(k),$$

$$\Delta Q_{cpu}(k) = Q_{cpu}(k) - cpu(k) \quad (6)$$

- λ_{mem} and λ_{cpu} are exponential forgetting factors, larger values increase the importance of older predictions
- In control theory $\lambda_{mem}, \lambda_{cpu} \in [0.95, 0.99]$ (M. Haeri et al., 2006)

MIMO model (cont'd)

Real predicted resource usage

$$R_{mem}(t) = \begin{cases} C_{mem}(t), & \text{if } C_{mem}(t) < \delta_{mem}, \\ \delta_{mem}, & \text{otherwise.} \end{cases}$$
$$R_{cpu}(t) = \begin{cases} C_{cpu}(t), & \text{if } C_{cpu}(t) < \delta_{cpu}, \\ \delta_{cpu}, & \text{otherwise.} \end{cases} \quad (7)$$

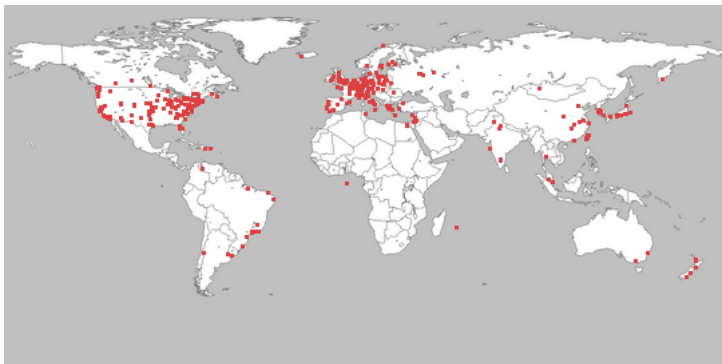
- δ_{mem} and δ_{cpu} are the maximum possible values for memory and CPU usage, respectively

Main issues in the deployment of Algorithm 2

- Requires a real networking environment with several hundred nodes available
- Requires implementing/maintaining the overlay structure
- Requires measuring latency between each OH-OH and each EH-OH
- Requires measuring resource usage (i.e. memory and CPU)
- Requires implementing the proposed algorithm in the MH

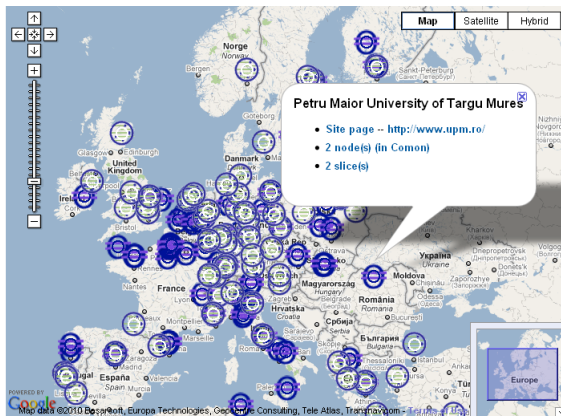
Deployment environment: PlanetLab

- *PlanetLab* is a “geographically distributed overlay network designed to support the deployment and evaluation of planetary-scale network services” (A. Bavier et al., 2004)
- It currently consists of 1098 nodes spread across the entire globe



Deployment environment: PlanetLab (cont'd)

- In 2010 we have also established a site at the “Petru Maior” University of Tîrgu Mureş - it is the only site available in Romania
- Our site contributes with 2 nodes to the PlanetLab network



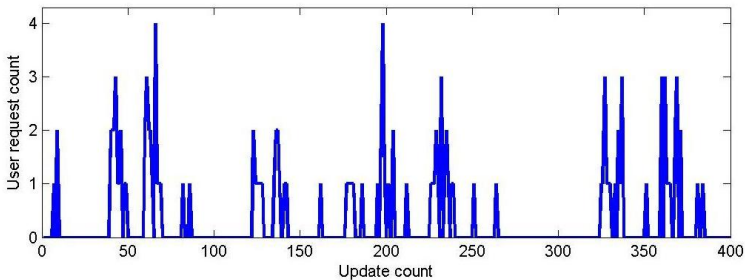
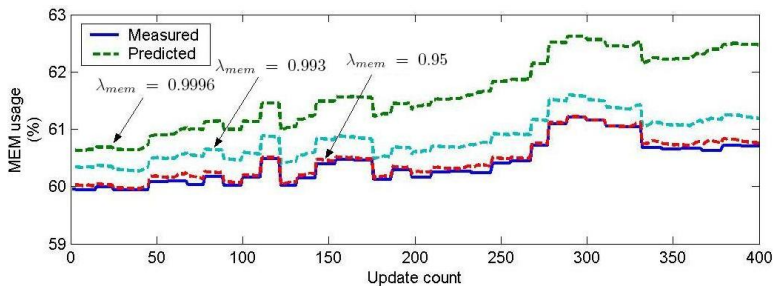
Characteristics of PlanetLab nodes

- Minimum hardware requirements:
 - PCU: Built-in, remote-access power-reset capability, accessible from PLC, such as IntelAMT, HPiLO, DellRAC, IPMIv2, etc
 - CPU: 4x Intel cores @ 2.4Ghz (e.g., quad core or 2x dual core)
 - RAM: 4GB
 - Disk: 500GB
- Virtualization: Linux VServer providing appearance of different Linux kernels (root privileges that do not affect other vservers)
- Networking: public IPs, without any firewalls/proxies
- User access: SSH/SCP-based remote access for configuring/deploying applications

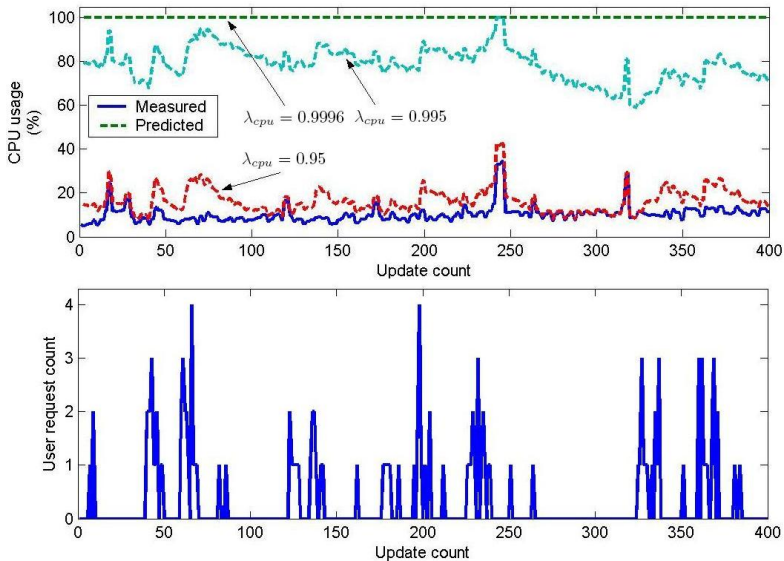
Model evaluation setup

- We used PlanetLab nodes to implement the proposed algorithm and to evaluate the predictive model
- PlanetLab nodes have a heterogeneous hardware setup and a homogeneous software setup
- Hardware heterogeneity \rightarrow model parameters must be determined for each OH
- OH transmit frames of variable size (between 10K and 20K Bytes) to each EH with a frame-rate of 5 frames/second

Predicting future memory usage



Predicting future CPU usage



Conclusions and future work

- We proposed a resource usage prediction model for ALM
- Our proposal can also be used in other environments where prediction is required
- By using a prediction model we can increase the overall performance of the multicast layer
- As future work we intend to investigate on bandwidth usage prediction and of a weighted function for EH distribution

References

- F. Bacelli, A. Chaintreau, Z. Liu, A. Riabov, and S. Sahu, "Scalability of Reliable Group Communication Using Overlays", Proceedings of INFOCOMM, 2004.
- S. M. Venilla, and V. Sankaranarayanan, "Threat Analysis for P-LeaSel, a Multicast Group Communication Model", Asian Journal of Information Technology, Vol. 7, 2008, pp. 6468.
- S. Deering, and D. Cheriton, "Multicast Routing in Datagram Internetworks and Extended LANS", ACM Transactions on Computer Systems, Vol. 8, No. 2, 1990, pp. 85111.
- C. Diot, B.N. Levine, B. Lyles, H. Kassem, and D. Balensiefen, "Deployment issues for the IP multicast service and architecture", IEEE Network Magazine, Vol. 14, No. 1, 2000, pp. 7888.
- A. El-Sayed, and V. Roca, "A Survey of Proposals for an Alternative Group Communication Service, IEEE Network, Vol. 17, No. 1, 2003, pp. 4651.
- V. Roca, and A. El-Sayed, "A Host-Based Multicast (HBM) Solution for Group Communications", Proceedings of the First International Conference on Networking, LNCS, Vol. 2093, 2001, pp. 610619.
- K. Ragab, and A. Yonezawa, "A Self-organized Clustering-based Overlay Network for Application Level Multicast", Journal of Networks, Vol. 4, No. 2, 2009, pp. 8591.
- S. Jaggi, P. Sanders, P. A. Chou, M. Effros, S. Egnér, K. Jain, and L. Tolhuizen, "Polynomial Time Algorithms for Multicast Network Code Construction", IEEE Transactions on Information Theory, Vol. 51, No. 6, 2005, pp. 19731982.
- N. Shanthi, and L. Ganesan, "Security In Multicast Mobile Ad-Hoc Networks", International Journal of Computer Science and Network Security, Vol. 8, No. 7, 2008, pp. 326330.
- H. Piroška, and R. Balint, "Optimal server distribution in multimedia communication", IN the Proc. of the 4th International Conference on RoEduNet, 2005, pp. 142147.
- A. Bavier, M. Bowman, B. Chun, D. Culler, S. Karlin, S. Muir, L. Peterson, T. Roscoe, T. Spalink, and M. Wawrzoniak, "Operating System Support for Planetary-Scale Network Services", Networked Systems Design and Implementation, 2004.

References

- T. L. Huang, and D. T. Lee, "A distributed multicast routing algorithm for real-time applications in wide area networks", *Journal of Parallel and Distributed Computing*, Vol. 67, Issue 5, 2007, pp. 516530.
- W. Jia, W. Tu, and J. Wu, "Hierarchical Multicast Tree Algorithms for Application Layer Mesh Networks", *Networking and Mobile Computing*, LNCS, Vol. 3619, 2005, pp. 549559.
- W. Yong, W. Seng, and H. Xianying, "A new Hierarchical Application Layer Multicast algorithm for large-scale video broadcasting", In the Proc. of the 2nd IEEE International Conference on Computer Science and Information Technology, 2009, pp.610613.
- S. Ratnasamy, M. Handley, R. Karp, and S. Shenker, "Application-Level Multicast Using Content-Addressable Networks", *Networked Group Communication*, LNCS, Vol. 2233, 2001, pp. 1429.
- K. Park, and V. Pai, "Deploying Large File Transfer on an HTTP Content Distribution Network", In *Proceedings of the First Workshop on Real, Large Distributed Systems (WORLDS 04)*, 2004.

Thanks for your attention!

Questions?