An Exact Algorithm for Calculating Blocking Probabilities in Multicast Networks

Eeva Nyberg, Jorma Virtamo & Samuli Aalto
Laboratory of Telecommunications Technology
Helsinki University of Technology

samuli.aalto@hut.fi

Background

"Blocking probabilities in dynamic multicast networks"

- Earlier work carried out at HUT on the topic:
  - J. Karvo, J. Virtamo, S. Aalto & O. Martikainen (1997)
    - "Blocking of dynamic multicast connections in a single link"
    - COST 257TD(97)46 (also in BC'98, Stuttgart)
  - J. Karvo, J. Virtamo, S. Aalto & O. Martikainen (1998)
    - "Blocking of dynamic multicast connections"
    - INFORMS TELECOM-4, Boca Raton (to appear in *TS*)
- Current work:
  - Eeva Nyberg's M.Sc. Thesis (November 1999)
    - "Calculation of blocking probabilities and dimensioning of multicast networks"

Contents

Multicast communication

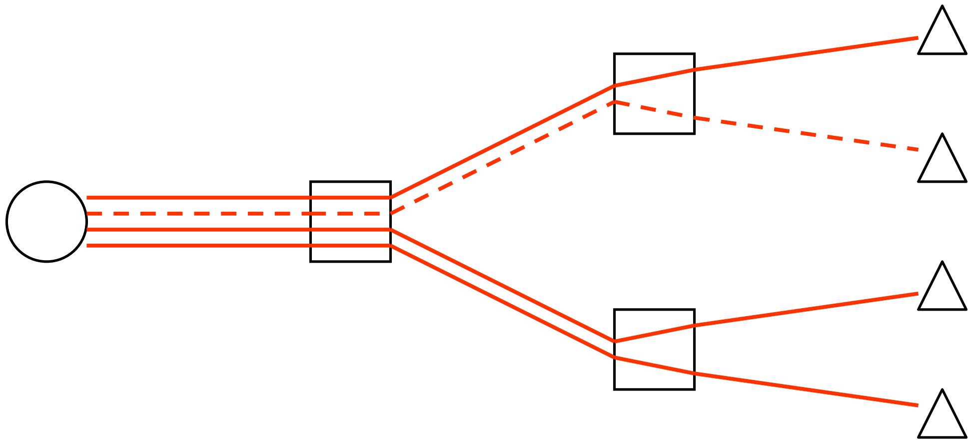
multicast communication = point-to-multipoint communication

- Multicast communication can be implemented by
  - point-to-point connections
  - static point-to-multipoint connections
  - dynamic point-to-multipoint connections

multicast connection = point-to-multipoint connection

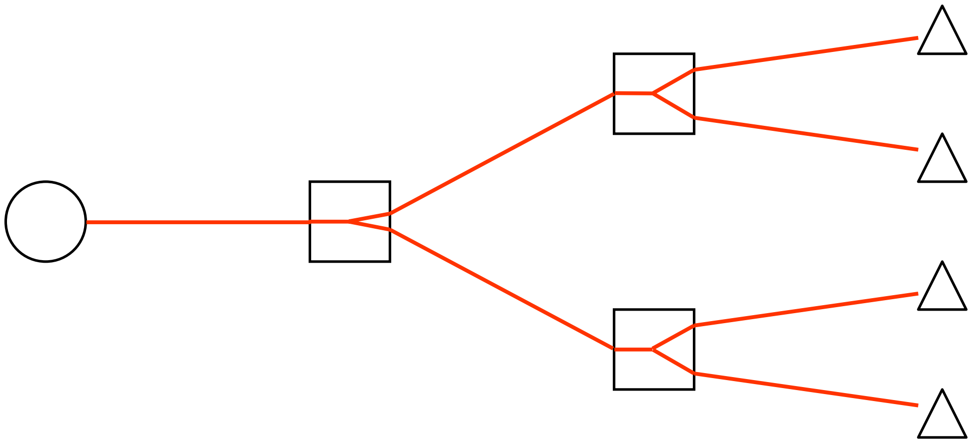
Point-to-point connections

- Flexible but …
- … wasting resources



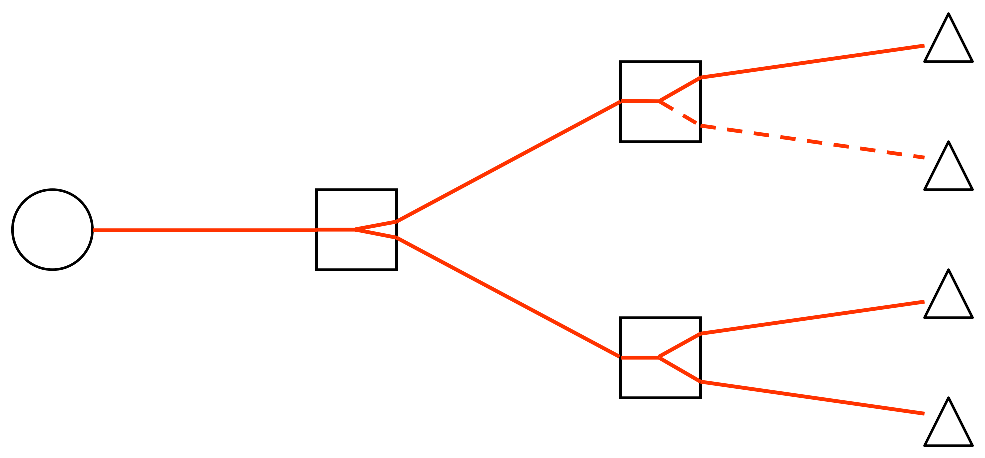
Static point-to-multipoint connections

- Saving resources but …
- … inflexible



Dynamic point-to-multipoint connections

- Flexible
- Saving resources



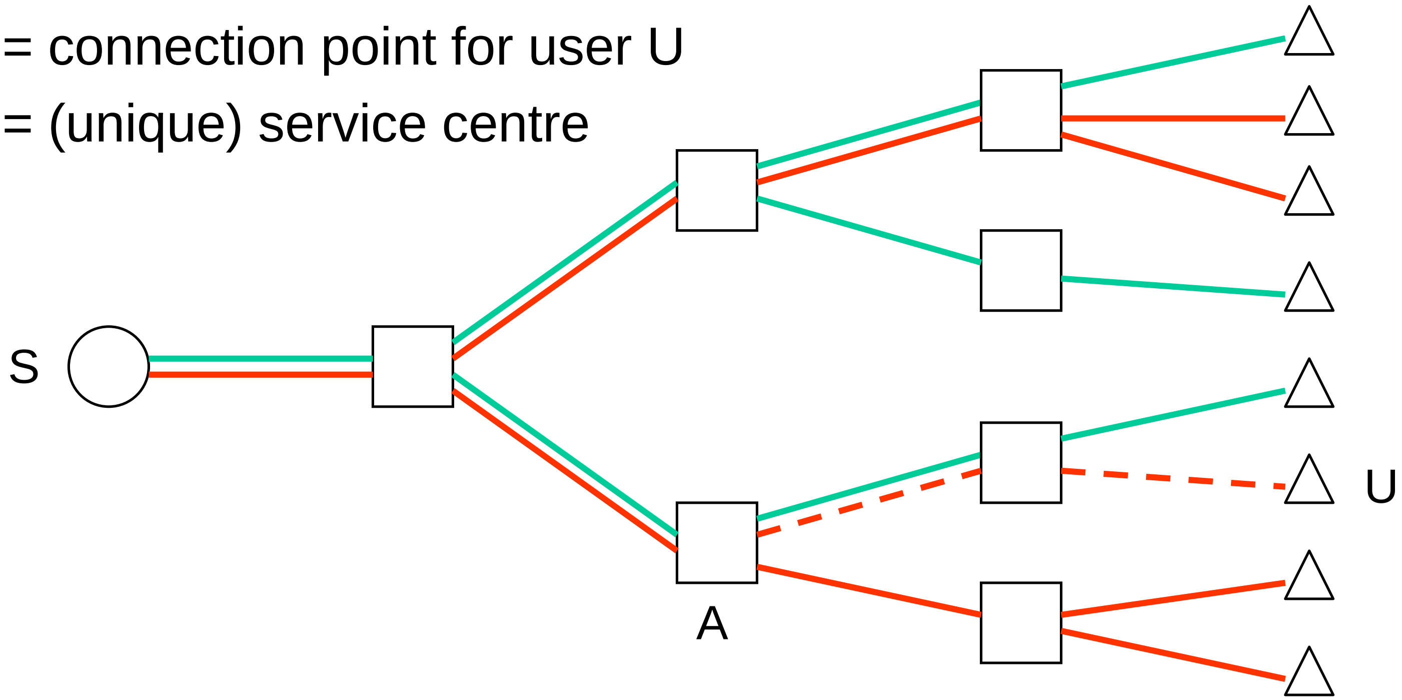
Dynamic multicast network model (1)

- Setup:
  - Unique **service center** offers a variety of **channels**
  - Each channel is delivered by a **dynamic multicast connection**
  - **Fixed routing** of these multicast channels
  - Each multicast connection uses the same **multicast tree**
  - Service center located at the **root node** of the multicast tree
  - Users located at the **leaf nodes** of the multicast tree

- Possible application:
  - TV or radio delivery via a telecommunication network

Dynamic multicast network model (2)

- U = new user of channel ‘red’
- A = connection point for user U
- S = (unique) service centre



Blocking of dynamic multicast connections

- Ordinary **loss network model** is suitable for
  - networks with
    - point-to-point or
    - **static** multicast connections
- But the ordinary loss network model is **not** suitable for
  - networks with
    - **dynamic** multicast connections
- Thus, new methods are needed to
  - calculate blocking probabilities in dynamic multicast networks

Blocking in a single link

- Karvo, Virtamo, Aalto & Martikainen (1997) :
  - Assumptions:
    - single finite capacity link (the others being infinite)
    - infinite user populations in the leaves of the multicast tree subscribe to different channels according to independent Poisson processes
    - channel subscription times (of individual users) generally distributed with channel-wise means
  - Results:
    - exact results for the call blocking probability

End-to-end blocking in a network

- Karvo, Virtamo, Aalto & Martikainen (1998):
  - Assumptions:
    - as before but now all the links of finite capacity
  - Results:
    - approximative results for the end-to end call blocking probability by applying the Reduced Load Approximation (RLA) method
    - verification by simulations
  - Conclusions:
    - RLA seems to give an upper bound
    - results of approximations and simulations on the same scale
    - difference about 10 - 50 % (not totally satisfactory)

Contents

Notation and assumptions

- Notation:
  - J = set of links (indexed by j)
  - C_j = capacity of link j
  - $U \subset J$ = set of leaf links = set of user populations (indexed by u)
  - I = set of channels (indexed by i)
  - d_i = capacity requirement of channel i
  - $a_{ui} = \lambda_{ui}/\mu_i$ = traffic intensity of traffic class (u,i) $\in U \times I$
- Assumptions:
  - infinite user populations in the leaves of the multicast tree subscribe to different channels according to independent Poisson(λ_{ui})-processes
  - channel subscription times (of individual users) independent and generally distributed with channel-wise means $1/\mu_i$

Contents

Network with infinite link capacities

- Assume that
  - $C_j = \infty$ for all j
- Denote
  - X_{ui} = 1{channel i of leaf link u is 'on'} $\in$ {0,1}
  - $p_{ui} = 1 - \exp(-a_{ui})$
- Earlier result (based on M/G/∞ model):

$$\pi_{ui}(x) := P\{X_{ui} = x\} = (p_{ui})^{x}(1 - p_{ui})^{1-x}$$

Network state

- Network state:
  - $\mathbf{X} = (X_{ui} \mid u \in U, i \in I) \in \Omega$
- Network state space:
  - $\Omega = \{0,1\}^{U \times I}$
- Stationary distribution (by independence of user populations):

$$\pi(\mathbf{x}) := P\{\mathbf{X} = \mathbf{x}\} = \prod_{u \in U} \prod_{i \in I} \pi_{ui}(x_{ui})$$

Leaf link state

- Leaf link state (for leaf link $u \in U$):
  - $\mathbf{X}_u = (X_{ui} \mid i \in I) \in S$
- Link state space:
  - $S = \{0,1\}^I$
- Stationary distribution (by independence of user populations):

$$\pi(\mathbf{x}_u) := P\{\mathbf{X}_u = \mathbf{x}_u\} = \prod_{i \in I} \pi_{ui}(x_{ui})$$

OR-convolution

- Denote:
  - $\oplus$ = component-wise OR-operation for *S*-vectors
  - $\otimes$ = OR-convolution for real-valued *S*-functions
- OR-convolution:
  - Let f and g be real-valued *S*-functions
  - Then define

$$[f \otimes g](\mathbf{y}) = \sum_{\mathbf{y}' \oplus \mathbf{y}'' = \mathbf{y}} f(\mathbf{y}')g(\mathbf{y}'')$$

Link state

- Denote
  - $\mathbf{Y}_{ji}$ = 1(channel i of link j is 'on') in {0,1}
- Link state (for link j ∈ $\mathcal{J}$):
  - $\mathbf{Y}_j = (Y_{ji} \mid i \in \mathcal{I}) \in \mathcal{S}$

$$\mathbf{Y}_j = g_j(\mathbf{X}) := \bigoplus_{u \in U_j} \mathbf{X}_u$$

- Stationary distribution (by independence of user populations):

$$c_j(\mathbf{y}) := P\{\mathbf{Y}_j = \mathbf{y}\} = [\bigotimes_{u \in U_j} \pi_u](\mathbf{y}) = \begin{cases} \pi_j(\mathbf{y}), & j \in U \\ [\bigotimes_{k \in N_j} \sigma_k](\mathbf{y}), & j \notin U \end{cases}$$

Contents

- Introduction
- Notation and assumptions
- Network with infinite link capacities
- Network with finite link capacities
- Algorithm for calculating end-to-end blocking probabilities
- Numerical results
- Generalization: inclusion of background traffic
- Summary and open problems

Network with finite link capacities

- Assume that
  - $C_j \leq \infty$ for all j
- Denote

$$\tilde{X}_{ui} = 1\{\text{channel } i \text{ of leaf link } u \text{ is 'on'}\} \in \{0,1\}$$

Network state

- Network state (truncated):

$$\tilde{\mathbf{X}} = (\tilde{X}_{ui} \mid u \in U, i \in I) \in \tilde{\Omega}$$

- Truncated network state space:

$$\tilde{\Omega} = \{\mathbf{x} \in \Omega \mid \mathbf{d} \cdot \mathbf{g}_j(\mathbf{x}) \le C_j, \forall j \in J\}$$

- Denote (for any A ∈ Ω):
  - $G(A) = \Sigma_{\mathbf{x} \in A}\, \pi(\mathbf{x})$
- Stationary distribution (by truncation and insensitivity principles):

$$\tilde{\pi}(\mathbf{x}) := P\{\tilde{\mathbf{X}} = \mathbf{x}\} = \frac{\pi(\mathbf{x})}{G(\tilde{\Omega})}$$

End-to-end blocking probabilities

- Non-blocking states for traffic class (u,i)

$$\tilde{\Omega}_{ui} = \{\mathbf{x} \in \Omega \mid \mathbf{d} \cdot (\mathbf{g}_j(\mathbf{x}) \oplus (\mathbf{e}_i 1_{j \in R_u})) \leq C_j, \forall j \in J\}$$

- Time blocking probability b_{ui}^{t} for class (u,i):

$$b_{ui}^{\mathrm{t}} := 1 - P\{\tilde{\mathbf{X}} \in \tilde{\Omega}_{ui}\} = 1 - \frac{G(\tilde{\Omega}_{ui})}{G(\tilde{\Omega})}$$

- Call blocking probability b_{ui}^{c} for class (u,i) (due to PASTA):

$$b_{ui}^{\mathrm{c}} = b_{ui}^{\mathrm{t}} = 1 - \frac{G(\tilde{\Omega}_{ui})}{G(\tilde{\Omega})}$$

Contents

- Introduction
- Notation and assumptions
- Network with infinite link capacities
- Network with finite link capacities
- Algorithm for calculating end-to-end blocking probabilities
- Numerical results
- Generalization: inclusion of background traffic
- Summary and open problems

Algorithm (1)

- Define (for all j ∈ *J*):

$$\tilde{S}_j = \{\mathbf{y} \in S \mid \mathbf{d} \cdot \mathbf{y} \le C_j\}$$

$$Q_j(\mathbf{y}) = P\{\mathbf{Y}_j = \mathbf{y}; \mathbf{Y}_k \in \tilde{S}_j, \forall k \in M_j\}$$

$$\tilde{S}_j^{ui} = \{\mathbf{y} \in S \mid \mathbf{d} \cdot (\mathbf{y} \oplus (\mathbf{e}_i 1_{j \in R_u})) \le C_j\}$$

$$Q_j^{ui}(\mathbf{y}) = P\{\mathbf{Y}_j = \mathbf{y}; \mathbf{Y}_k \in \tilde{S}_j^{ui}, \forall k \in M_j\}$$

- Then call blocking probability for class (u,i) is

$$b_{ui}^{\mathrm{c}} = 1 - \frac{G(\tilde{\Omega}_{ui})}{G(\tilde{\Omega})} = 1 - \frac{\sum_{\mathbf{y} \in S} Q_J^{ui}(\mathbf{y})}{\sum_{\mathbf{y} \in S} Q_J(\mathbf{y})}$$

Algorithm (2)

- Truncation operator 1:
  - Let f be any real-valued *S*-function
  - Then define

$$T_j[f](\mathbf{y}) = \begin{cases} f(\mathbf{y}), & j \in \tilde{S}_j \\ 0, & j \notin \tilde{S}_j \end{cases}$$

- Recursion 1 to calculate $Q_J(\mathbf{y})$:

$$Q_j(\mathbf{y}) = \begin{cases} T_j[\pi_j](\mathbf{y}), & j \in U \\ T_j[\bigotimes_{k \in N_j} Q_k](\mathbf{y}), & j \notin U \end{cases}$$

Algorithm (3)

- Truncation operator 2:
  - Let f be any real-valued *S*-function
  - Then define

$$T_j^{ui}[f](\mathbf{y}) = \begin{cases} f(\mathbf{y}), & j \in \tilde{S}_j^{ui} \\ 0, & j \notin \tilde{S}_j^{ui} \end{cases}$$

- Recursion 2 to calculate $Q_J^{ui}(\mathbf{y})$:

$$Q_j^{ui}(\mathbf{y}) = \begin{cases} T_j^{ui}[\pi_j](\mathbf{y}), & j \in U \\ T_j^{ui}[\bigotimes_{k \in N_j} Q_k^{ui}](\mathbf{y}), & j \notin U \end{cases}$$

Contents

Contents

- Introduction
- Notation and assumptions
- Network with infinite link capacities
- Network with finite link capacities
- Algorithm for calculating end-to-end blocking probabilities
- Numerical results
- Generalization: inclusion of background traffic
- Summary and open problems

Inclusion of background traffic

- Assumption:
  - In addition to multicast traffic, each link j has independent (unicast) background traffic of intensity A_j
  - These background connection requests arrive according to independent Poisson-processes
  - Background connection holding times are independent and generally distributed
  - Each background connection reserves one channel
- Result:
  - algorithm for calculating end-to-end call blocking probabilities for dynamic multicast connections in this generalized setting, which is a slight modification of the original algorithm

Modifications needed

- Define (for all j ∈ *J*):

$$q_j(z) = \frac{(A_j)^z}{z!} e^{-A_j}$$

- Modified truncation operator 1:

$$\hat{T}_j[f](\mathbf{y}) = \begin{cases} \sum_{z=0}^{C_j - \mathbf{d}\cdot\mathbf{y}} q_j(z) f(\mathbf{y}), & j \in \tilde{S}_j \\ 0, & j \notin \tilde{S}_j \end{cases}$$

- Modified truncation operator 2:

$$\hat{T}_j^{ui}[f](\mathbf{y}) = \begin{cases} \sum_{z=0}^{C_j - \mathbf{d}\cdot\mathbf{y}} q_j(z) f(\mathbf{y}), & j \in \tilde{S}_j^{ui} \\ 0, & j \notin \tilde{S}_j^{ui} \end{cases}$$

Contents

Summary

- Assumptions:
  - dynamic multicast network (with finite capacity links)
  - infinite user populations in the leaves of the multicast tree subscribe to different channels according to independent Poisson processes
  - channel subscription times (of individual users) generally distributed with channel-wise means
- Results:
  - exact results for the end-to-end call blocking probability for each class
  - algorithm for calculating these probabilities based on truncation and OR-convolution operators
  - generalization to the case with independent (unicast) background traffic

Open problems

- Finite user population case
  - studied in Eeva Nyberg's M.Sc. Thesis
- Due to state space explosion, need for
  - improved simulations methods
  - improved approximation methods for the end-to-end blocking

THE END