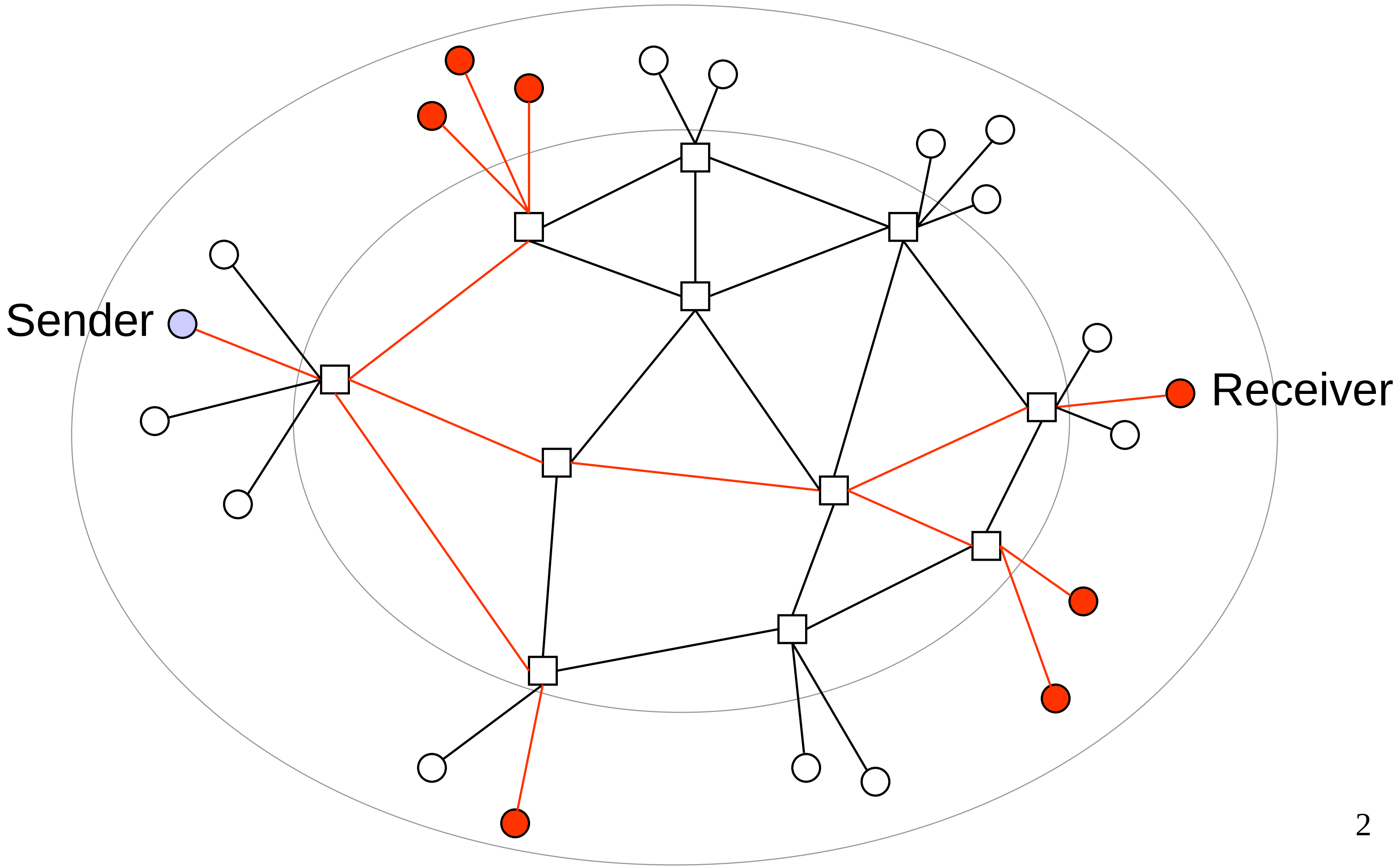
Calculating Blocking Probabilities of Multicast Connections with Dynamic Membership

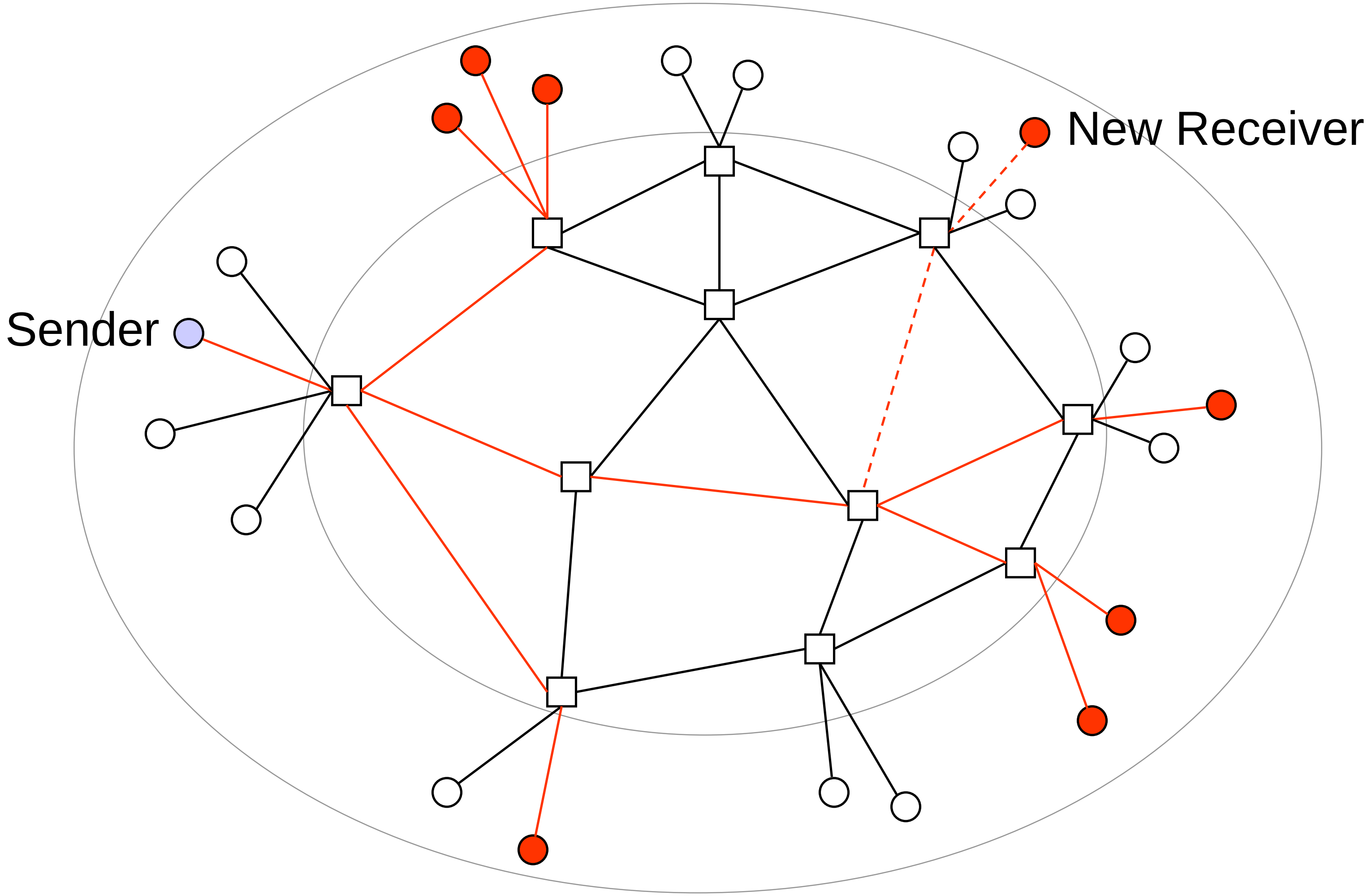
Samuli Aalto, Jouni Karvo, Eeva Nyberg & Jorma Virtamo
Laboratory of Telecommunications Technology
Helsinki University of Technology

samuli.aalto@hut.fi

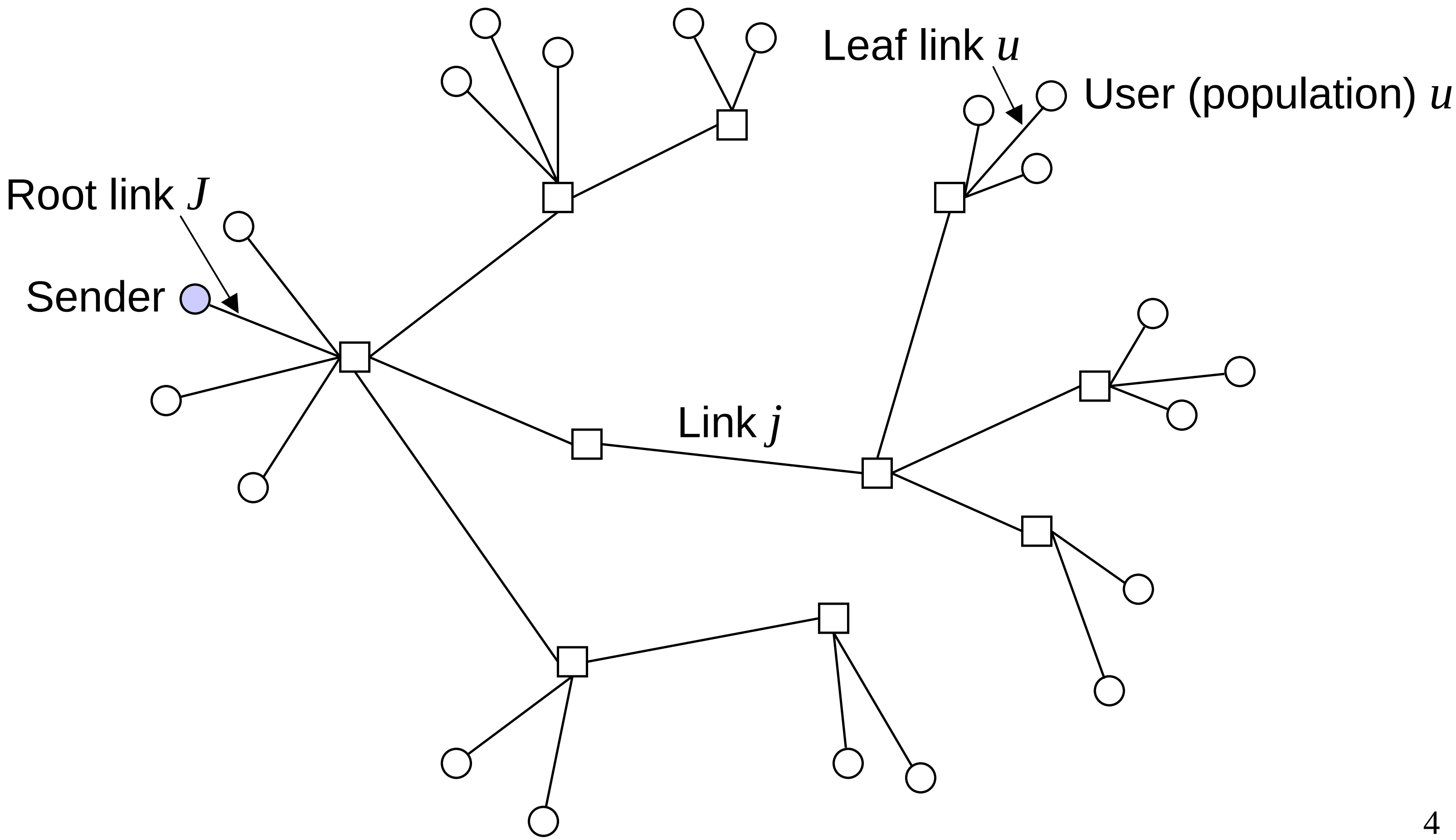
Multicast connection with dynamic membership (1)



Multicast connection with dynamic membership (2)



Routing tree



Teletraffic study

- Calculation of
  the call blocking for the specified user
  = probability that the user fails to join the multicast connection

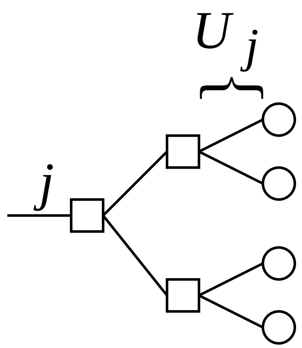
- Assumptions:
  - links $j \in J$ have finite capacities C_j
  - fixed routing tree for the multicast connection
  - when active on a link, the multicast connection takes capacity d
  - independent on-off type users $u \in U$ (cf. Engset model)
  - random background traffic independently on each link j

Infinite capacity system

- Y_u = connection state on leaf link u
  - $Y_u \sim$ Bernoulli(p_u), $\pi_u(y) := P\{Y_u = y\}$
- Y_j = connection state on link j

OR-operation

$$Y_j = \bigoplus_{u \in U_j} Y_u$$



- Z_j = capacity required by background traffic on link j
  - $Z_j \sim$ Poisson(a_j), $\pi_j^Z(z) := P\{Z_j = z\}$
- S_j = total capacity requirement on link j

$$S_j = Y_j d + Z_j$$

Steady state probabilities in an infinite capacity system

- $\mathbf{X}$ = network state (**without** capacity constraints)

state space

$$\mathbf{X} = (\mathbf{Y}, \mathbf{Z}) = ((Y_u, u \in U), (Z_j, j \in J)) \in \Omega$$

- Due to the independence assumptions, we have

$$\pi(\mathbf{x}) := P\{\mathbf{X} = \mathbf{x}\} = \prod_u \pi_u(y_u) \prod_j \pi_j^Z(z_j)$$

Steady state probabilities in a finite capacity system

- $\tilde{\mathbf{X}}$ = network state (**with** capacity constraints)

$$\tilde{\mathbf{X}} \in \tilde{\Omega} := \{\mathbf{x} \in \Omega \mid s_j(\mathbf{x}) \le C_j,\ j \in J\}$$

allowed states

- It can be shown that, under the traffic assumptions made (concerning both the user model and background traffic), the **Truncation Principle** applies. Thus,

$$P\{\tilde{\mathbf{X}} = \mathbf{x}\} = P\{\mathbf{X} = \mathbf{x} \mid \mathbf{X} \in \tilde{\Omega}\} = \frac{P\{\mathbf{X} = \mathbf{x}\}}{P\{\mathbf{X} \in \tilde{\Omega}\}}, \quad \mathbf{x} \in \tilde{\Omega}$$

Time blocking

- Nonblocking states for user u:

$$\tilde{\Omega}_u := \{\mathbf{x} \in \tilde{\Omega} \mid d + z_j \le C_j,\ j \in R_u\}$$

- B^t_u = time blocking probability for user u

$$B_u^t := 1 - P\{\tilde{\mathbf{X}} \in \tilde{\Omega}_u\} = 1 - \frac{P\{\mathbf{X} \in \tilde{\Omega}_u\}}{P\{\mathbf{X} \in \tilde{\Omega}\}}$$

numerator

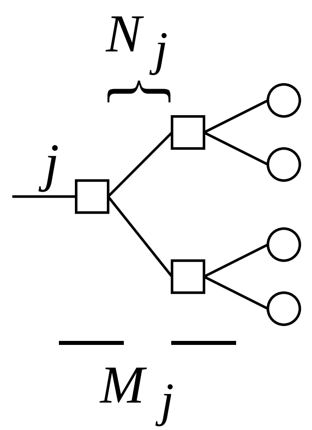
denominator

- **Note**: Due to applicability of the Truncation Principle, we are able to calculate the blocking probability in a finite capacity system by analysing the (much easier) infinite capacity system!
- **Problem**: computationally extremely complex
  - exponential in U and J

Recursive algorithm for the denominator (1)

- Define for all links j

$$Q_j(y) = P\{Y_j = y; S_k \le C_k, \forall k \in M_j\}$$



- Key observation:

$$P\{\mathbf{X} \in \tilde{\Omega}\} = \sum_y Q_J(y)$$

root link

- Recursive **Convolution-Truncation** algorithm for Q_j's:

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

Recursive algorithm for the denominator (2)

- **Truncation** T_j:
  - Let f be a real-valued function defined on $\{0,1\}$. Define

$$T_j[f](y) = f(y) \cdot \boxed{P\{Z_j \le C_j - yd\}}$$

cdf of Z_j

- **OR-convolution** $\otimes$:
  - Let f and g be real-valued functions defined on $\{0,1\}$. Define

$$\begin{cases} [f \otimes g](0) = f(0)g(0) \\ [f \otimes g](1) = f(0)g(1) + f(1)g(0) + \boxed{f(1)g(1)} \end{cases}$$

- **Note**: A similar recursive algorithm is valid for the numerator
  - truncation modified along the route R_u

Call blocking

- In this case,
  call blocking (for user u) equals
  time blocking in a modified network
  where the user u is removed!
- Thus,
  a similar Convolution-Truncation algorithm
  can be developed for the calculation of call blocking

Generalizations

- **Several user population models**: single / finite / infinite
- **Multiple parallel multicast connections** $i = 1,\ldots,I$ with dynamic membership (using the same routing tree)
  - e.g. distribution of TV or radio channels
  - link state: $(Y_{j1},\ldots,Y_{jI}, Z_j)$
  - convolution-truncation algorithm with a **modified OR-convolution**
  - complexity: linear in U but exponential in I
- **Multiple groups** $k = 1,\ldots,K$ of parallel multicast connections with dynamic membership (using the same routing tree)
  - within a group: connections symmetric
  - link state: $(N_{j1},\ldots,N_{jK}, Z_j)$
  - convolution-truncation algorithm with a **combinatorial convolution**
  - complexity: polynomial in I for any fixed K (e.g. $O(I^3)$ for $K = 1$)

Future work

- Development of approximate algorithms for large networks with multiple connections/classes
  - Quick Simulation approach
  - Reduced Load Approximation approach

- Application of the Convolution-Truncation algorithm for layered multicast connections

THE END