A bicriterion approach for routing problems in multimedia networks

João Clímaco [a]

José Craveirinha

Marta Pascoal

jclimaco@inescc.pt, jcrav@dee.uc.pt, marta@mat.uc.pt

University of Coimbra

PORTUGAL

[a] The research of João Clímaco was partially supported by project POCTI/GES/37707/2001, *Multicriteria programming methods in aided support decision networks*.

Abstract

Routing problems in communication networks supporting multiple services, involve the selection of paths satisfying multiple constraints (of a technical nature) and seeking to “optimise” the associated metrics.

Traditional models were single-objective but, in many situations it is important to consider different objectives.

We consider a bicriterion model dedicated to calculate non-dominated paths for specific traffic flows (associated with video services), in multiservice high-speed networks.

The mathematical formulation and the bicriterion algorithmic approach developed for its resolution will be presented together with computational results regarding an application to video traffic in a high-speed network.

The approach is an adaptation of recent works by Ernesto Martins and his collaborators, namely the MPS algorithm.

Introduction

Routing problems in communication networks supporting multiple services, namely multimedia applications, involve the **selection of paths satisfying multiple constraints** of a technical nature, designated as QoS (Quality of Service) requirements and **seeking simultaneously to "optimise" the chosen objective functions.**

The **objective functions** are concerned with the **necessity of minimizing the consumption of (transmission) resources** along a path and **to obtain a minimum negative impact** in all other traffic flows that may use the network.

The specific models of these cost functions and of the QoS constraints depend on the type of multimedia service associated with the "calls" which are being routed from origin to destination.

Multimedia Applications

Objective functions:

- number of arcs (usually designated in telecommunications as hops or links);
- cost of accepting a call in each arc, as measured by an appropriate traffic model related with the bandwidth available in each link.

Constraints on the paths:

- minimum bandwidth required by the call;
- maximum allowed delay;
- maximum allowed jitter.

Multimedia Applications

Although traditional models in this area were single-objective, in many situations it is important to consider different, eventually conflicting objectives.

Routing algorithms that have been employed in current networks or proposed for this type of problem, are heuristics based on Dijkstra or Bellman-Ford shortest path algorithm.

Mathematical Formulation

In a teletraffic routing problem we consider a representation of a communication network the nodes of which may represent routers, servers or switches and the arcs of which represent links in the network with a certain transmission capacity expressed in terms of bandwidth.

Let $(\mathcal{N}, \mathcal{A})$ be an **undirected network** where:

- $\mathcal{N} = \{v_1, \ldots, v_n\}$ denotes the **set of nodes**;
- $\mathcal{A} = \{a_1, \ldots, a_m\}$ denotes the **set of arcs** (or links);
- a_k corresponds to pair (i, j), for some $i, j \in \mathcal{N}$;
- s is the **initial node** and t is the **terminal node.**

Path from s to t is an alternating sequence of nodes and arcs,

$$p = \langle v'_1, a'_1, v'_2, a'_2, \ldots, v'_\ell \rangle,$$

where $s \equiv v'_1$ and $t \equiv v'_\ell$.

Mathematical Formulation

Multimedia traffic routing problem represented as a bicriterion network problem.

Values associated with each arc (or link) (i, j) in $(\mathcal{N}, \mathcal{A})$:

- $c_{ij} > 0$ representing the **cost** of (i, j),
- $b_{ij} > 0$ representing the available **bandwidth** of (i, j),
- $d_{ij} > 0$ representing the associated **delay**.

Moreover:

- $c(p) = \sum_{(i,j) \in p} c_{ij}$,
- $b(p) = \min_{(i,j) \in p} \{b_{ij}\}$,
- $d(p) = \sum_{(i,j) \in p} d_{ij}$,
- h assigns to each path p its number of arcs.

Mathematical Formulation

Given $\Delta_{\text{rjiter}} \in \mathbb{N}$, $\Delta_{\text{bandwidth}} \in \mathbb{R}^+$ and $\Delta_{\text{delay}} \in \mathbb{R}^+$, where Δ_{rjiter} is a constraint on the number of arcs related with a bound on jitter.

The problem is to determine loopless paths p with minimum cost and minimum number of arcs, and satisfying the constraints:

- $b(p) \geq \Delta_{\text{bandwidth}}$;
- $d(p) \leq \Delta_{\text{delay}}$;
- p has at the most Δ_{rjiter} arcs.

That is, considering $f : \mathcal{P} \longrightarrow \mathbb{R}^2$ s.t. $f(p) = (c(p), h(p))$:

$$\begin{array}{lll} \text{"min"} & \{f(p) : p \in \bar{\mathcal{P}}\} & (1) \\ \text{s. t.} & b(p) \geq \Delta_{\text{bandwidth}} & (2) \\ & d(p) \leq \Delta_{\text{delay}} & (3) \\ & h(p) \leq \Delta_{\text{rjiter}} & (4) \end{array}$$

Mathematical Formulation

In general such a problem does not have a solution due to possible conflict between the considered functions.

Thus, it will be determined the set of “efficient” solutions, there is no other feasible path which improves one objective function without worsening at least one of the other objective functions.

Given paths p and q, p **dominates** q ($p_D q$) if and only if $c(p) \leq c(q)$, $h(p) \leq h(q)$ and at least one of the inequalities is strict.

Path q is **dominated** if and only if there is another path p such that $p_D q$.

$\mathcal{P}_N$ denotes the **set of non-dominated paths**.

Bicriterion algorithms

Labelling algorithms:

- Generalization of labelling algorithms for the shortest path problem.
- Supported by the Optimality Principle:
  Every non-dominated path is formed by non-dominated subpaths.
- However, the problem to be solved does not verify the OP (given the additional constraints), therefore these algorithms cannot be used.

Bicriterion algorithms

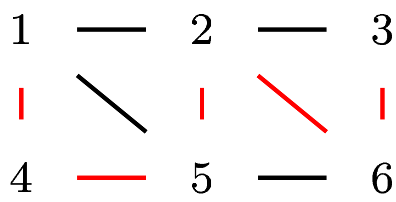
Ranking algorithms:

- Use a ranking algorithm to list paths by non-decreasing costs. Use that ordered list to choose the non-dominated paths. Paths are ranked until its cost exceeds a previously determined c^*.
- For ranking paths (or loopless paths) a deviation algorithm can be used.

Deviation algorithms:

- Use a set X which contains paths candidates to p_k
  1. Compute the shortest path and store it in X
  2. Successively pick up the shortest element in X, p_k. Analyse nodes of p_k, in order to compute shortest paths, not determined yet, and deviating from p_k in the analysed node. Store them in X.

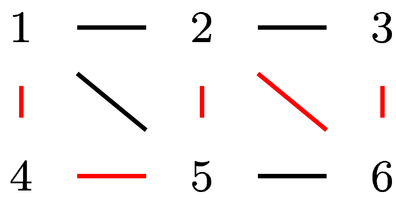
MPS algorithm



i	1			2			3	4	5
j	2	4	5	3	5	6	6	5	6
c_{ij}	20	0	10	0	0	0	0	0	10

Compute $\mathcal{T}_t$. (Red arcs in the network)
Compute $\bar{c}_{ij} = c_{ij} + \pi_i - \pi_j$, for any $(i, j) \in \mathcal{A}$. (Here $\bar{c}_{ij} = c_{ij}$.)
Compute shortest path.
While (there are paths to analyse) and
        (less than K loopless paths were identified) Do
    Choose the shortest determined path, p
    Analyse nodes of p and generate paths deviating from p
    If (p is loopless) Then
        Store p as p_k and increase k

MPS algorithm

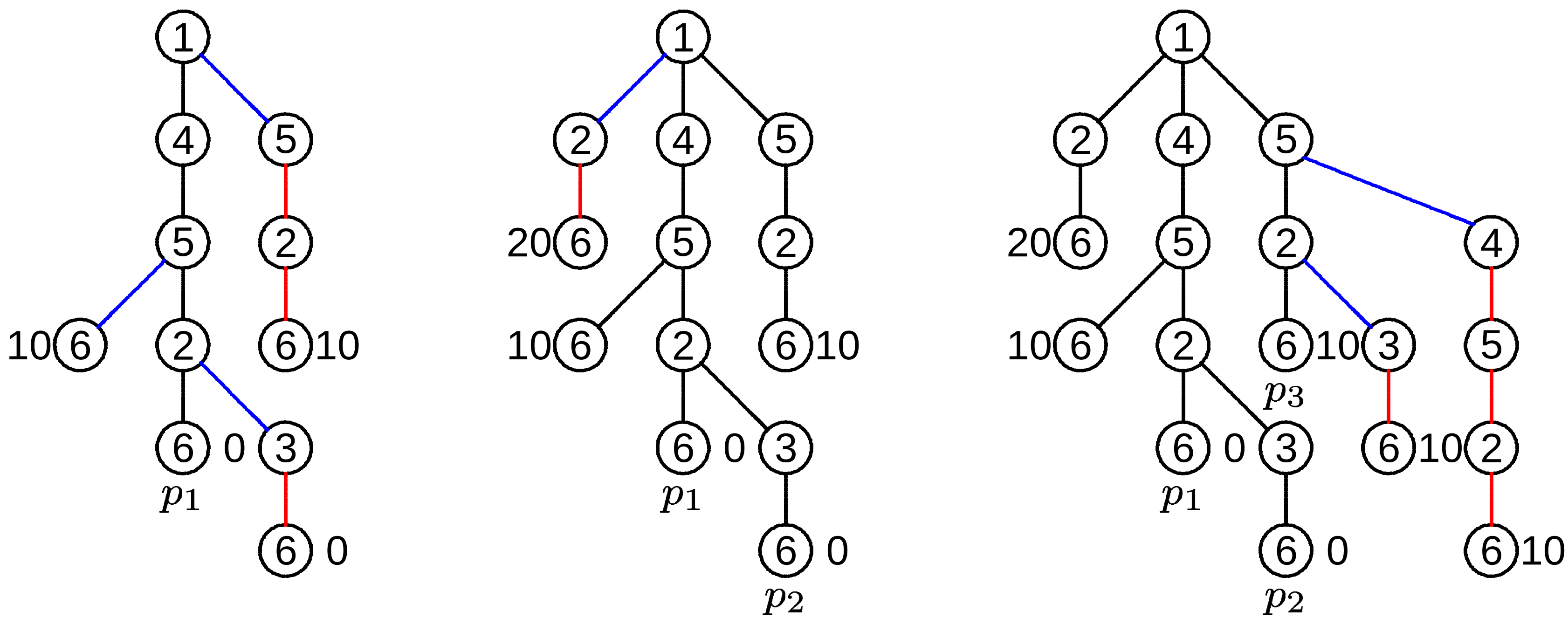


i	1			2			3	4	5
j	2	4	5	3	5	6	6	5	6
c_{ij}	20	0	10	0	0	0	0	0	10

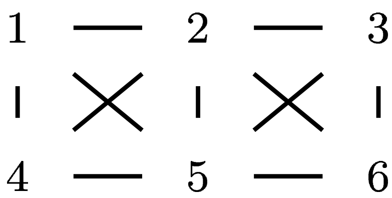
Analyse p_1.
For 4 ignore $(4, 1)$ (forms a loop).
Same for 5 and 2.

Analyse p_2

Analyse p_3.
Although $\langle 1, 5, 4, 5, 2, 6 \rangle$ contains a loop,
it will allow obtaining $\langle 1, 2, 6 \rangle$.



Rank paths with additional constraints

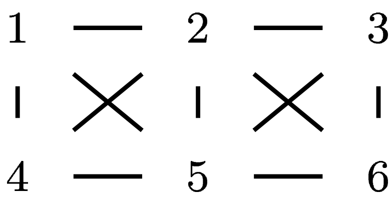


i	1			2				3		4	5
j	2	4	5	3	4	5	6	5	6	5	6
c_{ij}	20	0	10	0	0	0	0	20	0	0	10
b_{ij}	10	12	30	10	2	22	22	4	16	10	26

Constraint $b(p) \geq \Delta_{\text{bandwidth}}$, with $b(p) = \min_{(i,j)\in p}\{b_{ij}\}$

- Delete arcs (i, j) s.t. $b_{ij} < \Delta_{\text{bandwidth}}$

Rank paths with additional constraints



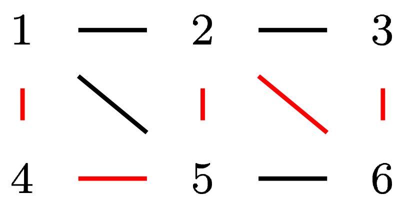
i	1			2				3		4	5
j	2	4	5	3	4	5	6	5	6	5	6
c_{ij}	20	0	10	0	0	0	0	20	0	0	10
b_{ij}	10	12	30	10	2	22	22	4	16	10	26
d_{ij}	25	15	25	5	15	0	25	25	5	5	5

Constraint $d(p) \leq \Delta_{\text{delay}}$, with $d(p) = \sum_{(i,j)\in p} d_{ij}$

- "Ignore" non-feasible paths
  Stop analysing p when $d(\text{sub}_p(s,v)) > \Delta_{\text{delay}}$
- Minimize the generation of non-feasible paths
  Ignore a new deviation at v if $d(\text{sub}_p(s,v) \diamond (v,u)) > \Delta_{\text{delay}}$

Constraint (4) (involving the number of arcs) is treated using an analogous procedure.

Rank paths with additional constraints



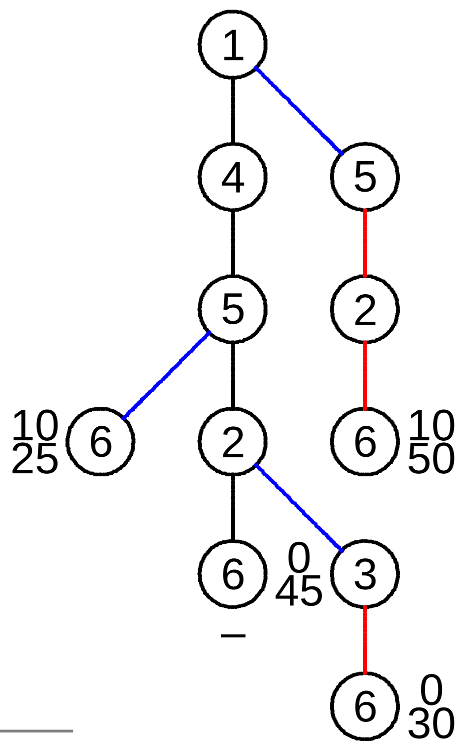
i	1			2			3	4	5
j	2	4	5	3	5	6	6	5	6
c_{ij}	20	0	10	0	0	0	0	0	10
d_{ij}	25	15	25	5	0	25	5	5	5

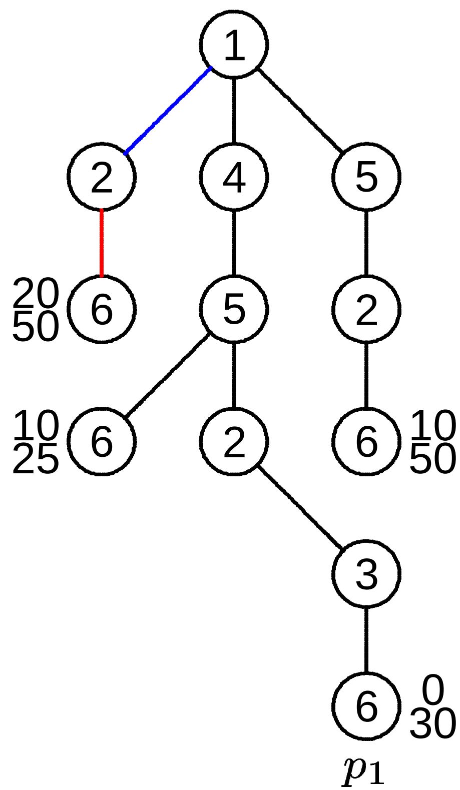
$\Delta_{\text{bandwidth}} = 8$, $\Delta_{\text{delay}} = 40$

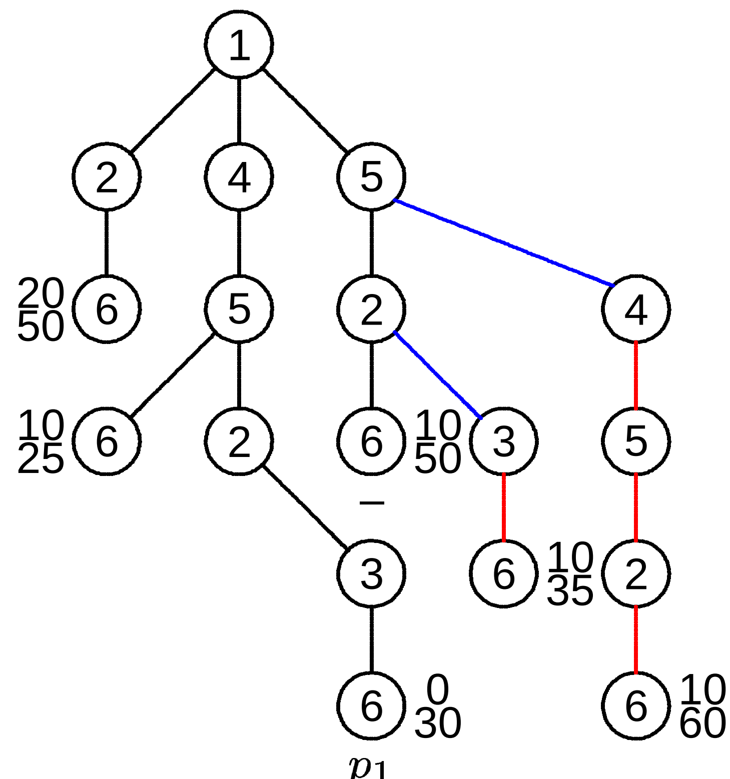
Analyse $\langle 1, 4, 5, 2, 6 \rangle$, not feasible.

Analyse $\langle 1, 4, 5, 2, 3, 6 \rangle$. It is p_1, the 1st feasible path.

Analyse $\langle 1, 5, 2, 6 \rangle$. Not feasible but a feasible path $\langle 1, 5, 2, 3, 6 \rangle$ is obtained from this analysis.







Bicriterion algorithm description

The proposed problem is a **bicriterion problem** where only **feasible loopless paths** (verifying constraints (2), (3) and (4)) are considered.
Therefore, an adaptation of MPS algorithm which ranks only feasible paths is used.

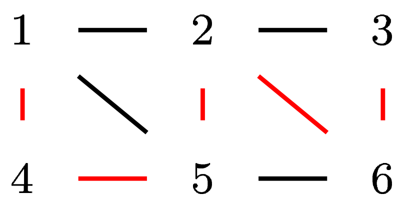
- Compute p the feasible path with less arcs, $\hat{c} = c(p)$.
- Compute q the shortest feasible path, $M_h = h(q)$, $m_c = c(q)$.
- Rank feasible paths p_k by non-decreasing c, until $c(p_k) > \hat{c}$.
  Dominance test to p_k (compare p_k with previous paths).

Bicriterion algorithm description

Dominance test to p_k (compare p_k with previous paths):

- $c(p_k) = M_c$
  If $h(p_k) < m_h$, then p_k dominates the stored candidates and it is candidate to non-dominated.
  If $h(p_k) = m_h$, then p_k is candidate to non-dominated.
  If $h(p_k) > m_h$, then $p_k \notin \mathcal{P}_N$.
- $c(p_k) > M_c$
  If $h(p_k) < m_h$, then the stored candidates belong to $\mathcal{P}_N$, p_k is candidate to non-dominated.
  If $h(p_k) \geq m_h$, then $p_k \notin \mathcal{P}_N$.

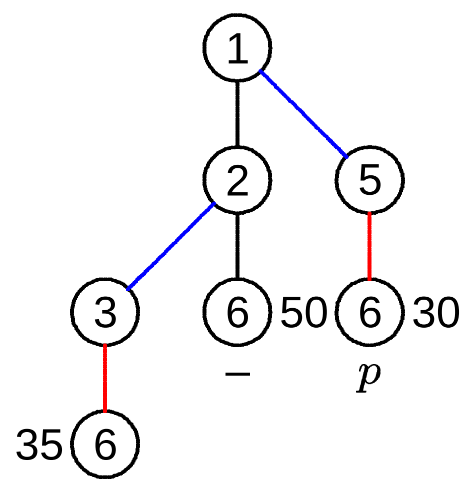
Bicriterion algorithm description



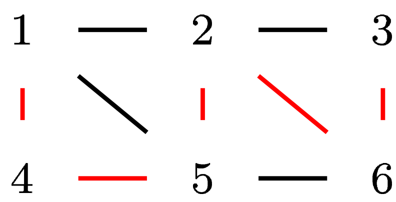
i	1			2			3	4	5
j	2	4	5	3	5	6	6	5	6
c_{ij}	20	0	10	0	0	0	0	0	10
d_{ij}	25	15	25	5	0	25	5	5	5

$\Delta_{\text{bandwidth}} = 8$, $\Delta_{\text{delay}} = 40$, $\Delta_{\text{rjiter}} = ma(s,t) + \Delta_{\text{arcs}} = 6$, where $ma(s,t)$ is the minimum number of arcs of a feasible path from s to t and $\Delta_{\text{arcs}} = 4$.

Compute the feasible path with less arcs. $p = \langle 1, 5, 6 \rangle$, $\hat{c} = 20$.



Bicriterion algorithm description



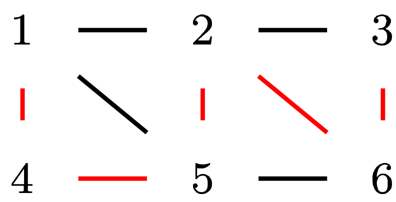
i	1			2			3	4	5
j	2	4	5	3	5	6	6	5	6
c_{ij}	20	0	10	0	0	0	0	0	10
d_{ij}	25	15	25	5	0	25	5	5	5

$\Delta_{\text{bandwidth}} = 8$, $\Delta_{\text{delay}} = 40$, $\Delta_{\text{rjiter}} = 6$.

Compute the shortest feasible path. $q = \langle 1, 4, 5, 2, 3, 6 \rangle$, $M_h = 5$, $m_c = 0$.

Similarly to ranking paths with additional constraints, but additionally verifying the jitter related constraint.

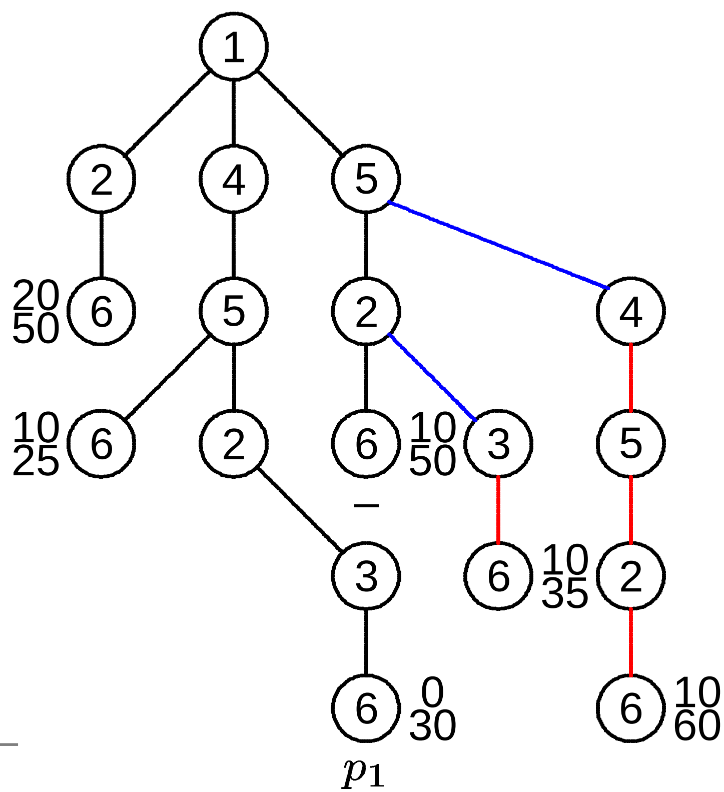
Bicriterion algorithm description



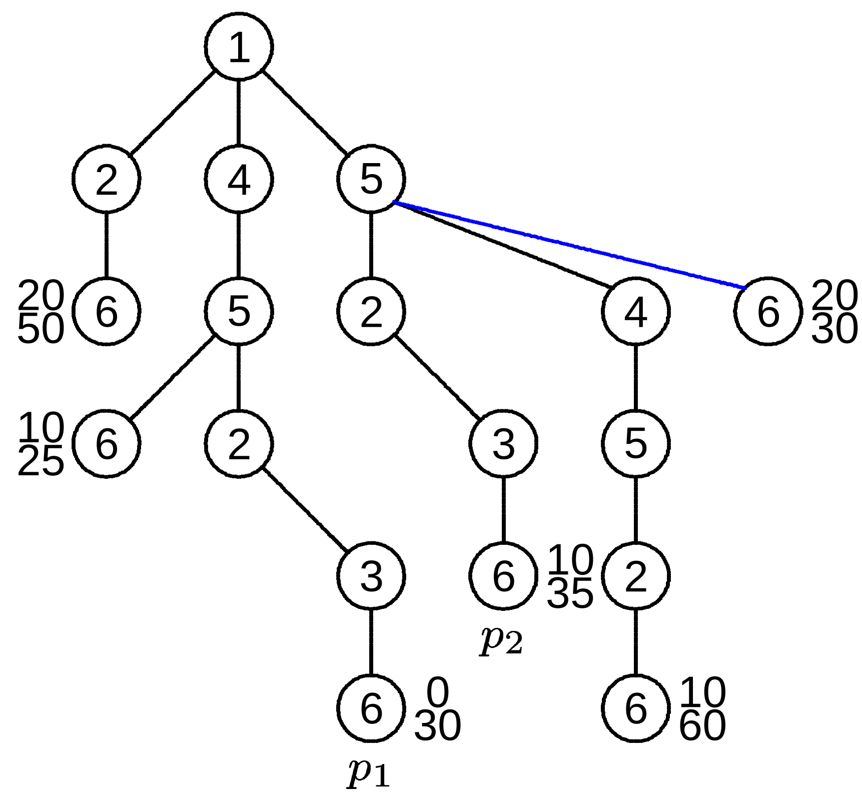
i	1			2			3	4	5
j	2	4	5	3	5	6	6	5	6
c_{ij}	20	0	10	0	0	0	0	0	10
d_{ij}	25	15	25	5	0	25	5	5	5

$\Delta_{\text{delay}} = 40$, $\Delta_{\text{rjitter}} = 6$.

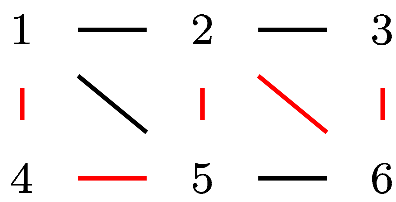
Analyse p_1, it might belong to $\mathcal{P}_N$.



Analyse p_2, it might belong to $\mathcal{P}_N$.
$p_1 \in \mathcal{P}_N$, $M_h = 4$, $m_c = 10$.



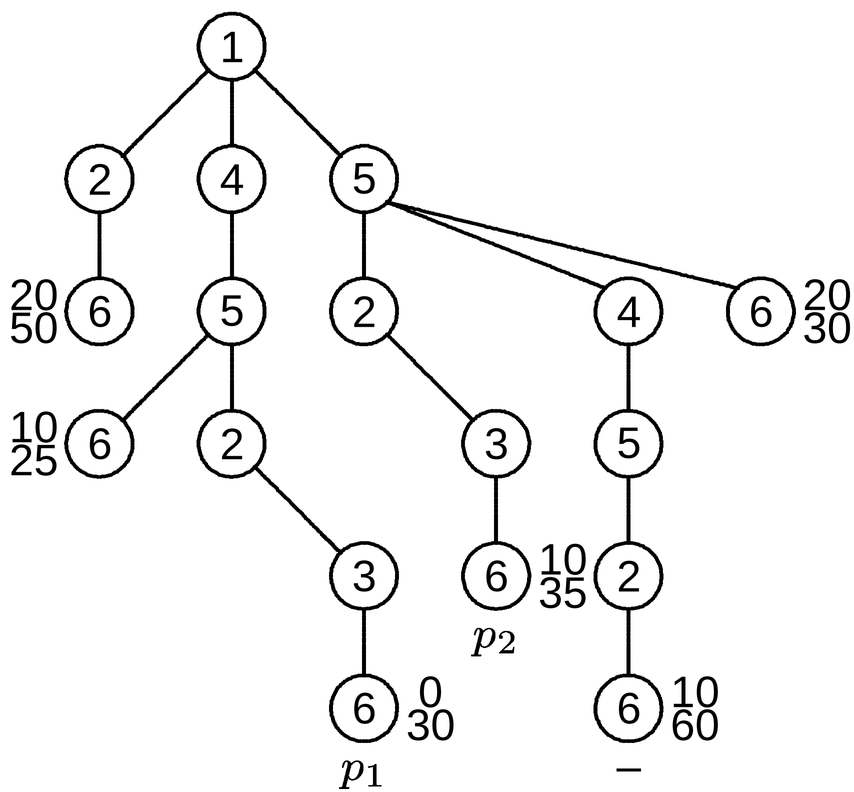
Bicriterion algorithm description



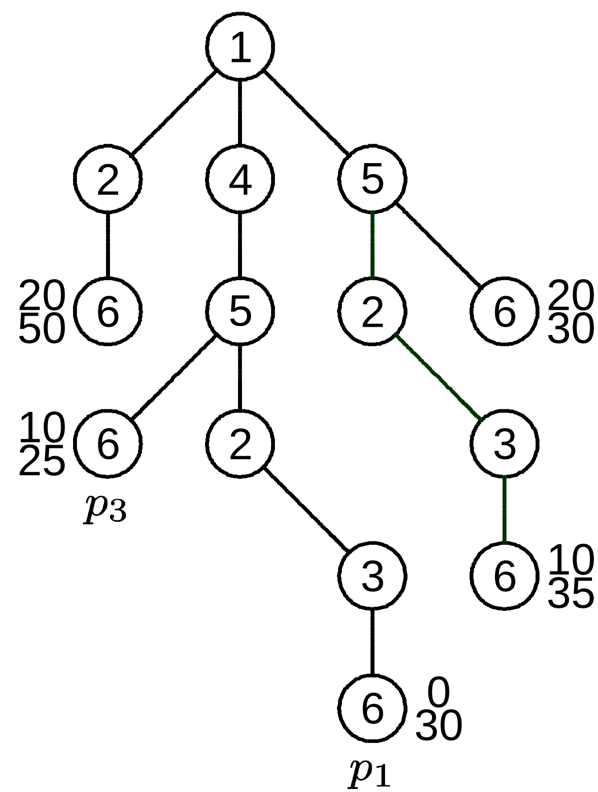
i	1			2			3	4	5
j	2	4	5	3	5	6	6	5	6
c_{ij}	20	0	10	0	0	0	0	0	10
d_{ij}	25	15	25	5	0	25	5	5	5

$\Delta_{\text{delay}} = 40$, $\Delta_{\text{rjitter}} = 6$.

Analyse non-feasible $\langle 1, 5, 4, 5, 2, 6 \rangle$.



Analyse p_2, it might belong to $\mathcal{P}_N$.
$p_3 \,_D p_2 \Longrightarrow p_2 \notin \mathcal{P}_N$. $M_h = 3$, $m_c = 10$.



Application

Application to a **routing problem of video traffic** in an ATM (Asynchronous Transfer Mode).

Undirected networks (connected) with n nodes and $m = 4n$ arcs.

Each node is a point in a 400×240 grid with mesh size unit of 10 Km.

Each node is adjacent to at least 2 and at most 10 other nodes.

Problem simulated with flow $(\sigma_k, r_k, S^k_{\max})$ s.t.:

- $\sigma_k = 10\, S_{\max}$: token bucket size (in bits),
- $r_k = r = 1.5 \times 10^6$ bit/s: token generation rate of the leaky bucket (stochastic model associated with the nodes),
- $S^k_{\max} = S_{\max}$: maximum packet size of the flow k (in bits),

where $S_{\max} = 53 \times 8$ bit is the size of an ATM cell.

Application Model

Each node is modeled as a queueing system using Weighted Fair Queueing (WFQ) service discipline, enabling to represent the bound on jitter through a constraint on the number of arcs Δ_{rjiter} .

Given (i, j), where i and j correspond to (x_i, y_i) and (x_j, y_j):

- random $b_{ij} \in \{0.52, 2.52, \ldots, 150.52\}$ (in Mb/s), corresponds to a link capacity of 155.52 Mb/s;
- $c_{ij} = \frac{1}{b_{ij}}$ (in order to obtain lowest call blocking probability);
- $d_{ij} = \left(\frac{S^k_{\max}}{r_k} + \frac{S_{\max}}{R_{ij}}\right) + \frac{\ell_{ij}}{2c/3}$ (in ms), where ℓ_{ij} is the Euclidean distance between (x_i, y_i) and (x_j, y_j), c is the speed of light and $R_{uv} = 155.52 \times 10^6$ bit/s is the bandwidth capacity of (u, v).

$\Delta_{\text{bandwidth}} = 1.5$, $\Delta_{\text{delay}} \in \{10, 15, \ldots, 60\}$, $\Delta_{\text{rjiter}} = ma(s, t) + \Delta_{\text{arcs}}$, with $\Delta_{\text{arcs}} \in \{2, 4\}$.

Application Model

1. Partition $\{0.52, 2.52, \ldots, 150.52\}$ in 5 classes with equal amplitude, each with a predefined percentage of b_{ij} values.

	I_0	I_1	I_2	I_3	I_4
Dist 1	20%	20%	20%	20%	20%
Dist 2	50%	20%	15%	10%	5%
Dist 3	41%	6%	6%	6%	41%

2. Partition the original grid according to x: $\{0, 10, \ldots, 400\} = \{0, \ldots, 120\} \cup \{130, \ldots, 250\} \cup \{260, \ldots, 400\}$. Given (i, j), where i and j correspond to (x_i, y_i) and (x_j, y_j):
   - if $x_i, x_j \in \{0, 10, \ldots, 120\}$, $b_{ij} \in \{0.52, \ldots, 48.52\}$;
   - if $x_i, x_j \in \{130, 140, \ldots, 250\}$, $b_{ij} \in \{50.52, \ldots, 98.52\}$;
   - if $x_i, x_j \in \{260, 270, \ldots, 400\}$, $b_{ij} \in \{100.52, \ldots, 150.52\}$;
   - otherwise $b_{ij} \in \{0.52, \ldots, 150.52\}$.

Application Model

Other type of network was used, obtained by using geographic coordinates of $n = 1088$ US cities.
(`http://www.realestate3d.com/gps/latlong.htm`)

Considering those cities as nodes of the network $m = 5n$ arcs between them were randomly generated (assuring the network is connected).

b_{ij} were randomly generated in $\{0.52, 2.52, \ldots, 150.52\}$ and the additional constraints considered were the previous ones.

Computational tests

Random networks:
$n \in \{500, 1000, 1500, 2000, 2500, 3000\}$ and $\frac{n^2}{25000}$ s-t node pairs.

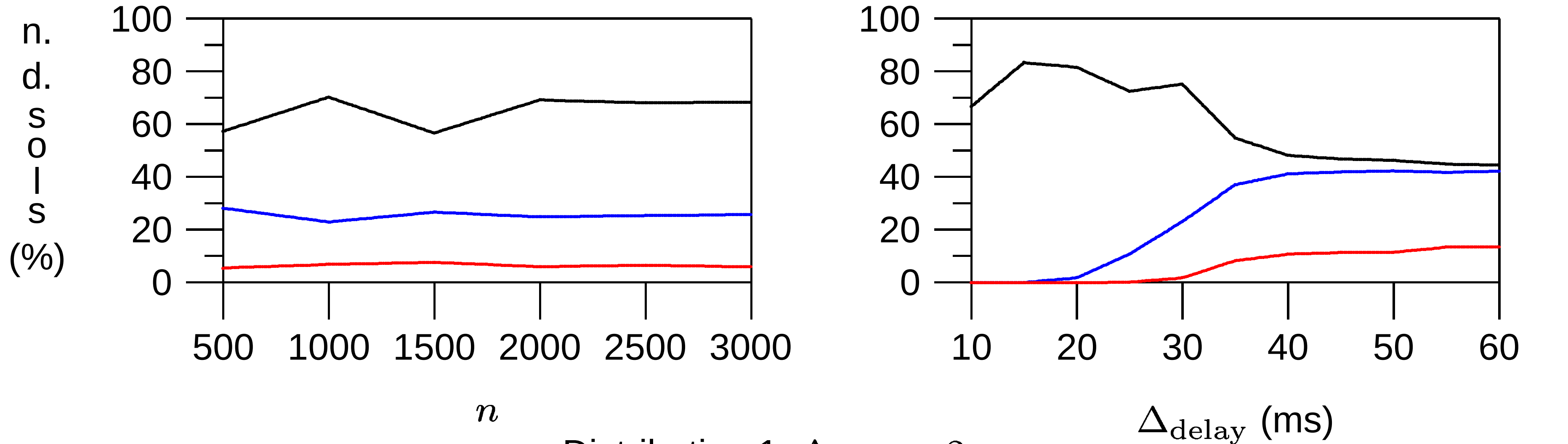
US coordinates network:
20 s-t node pairs.

10 different seeds.

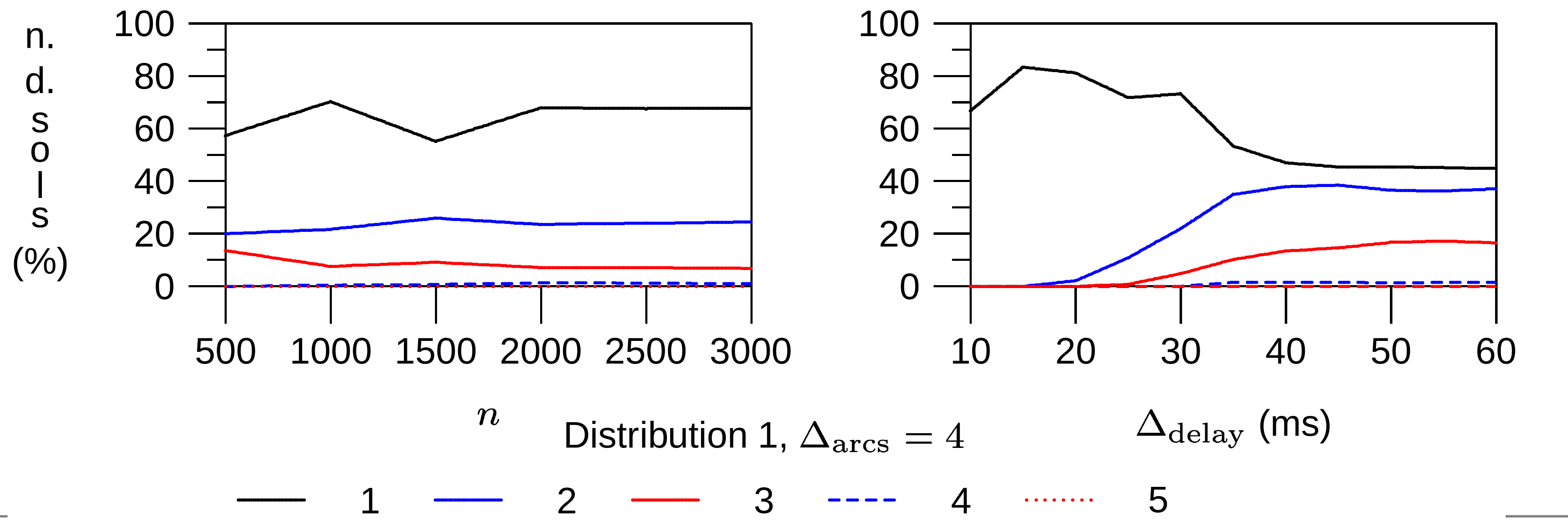
Code and computer:

- C language,
- AMD Athlon server at 1.5 GHz, with 256 Mbytes of RAM,
- Linux.

Results – Random networks

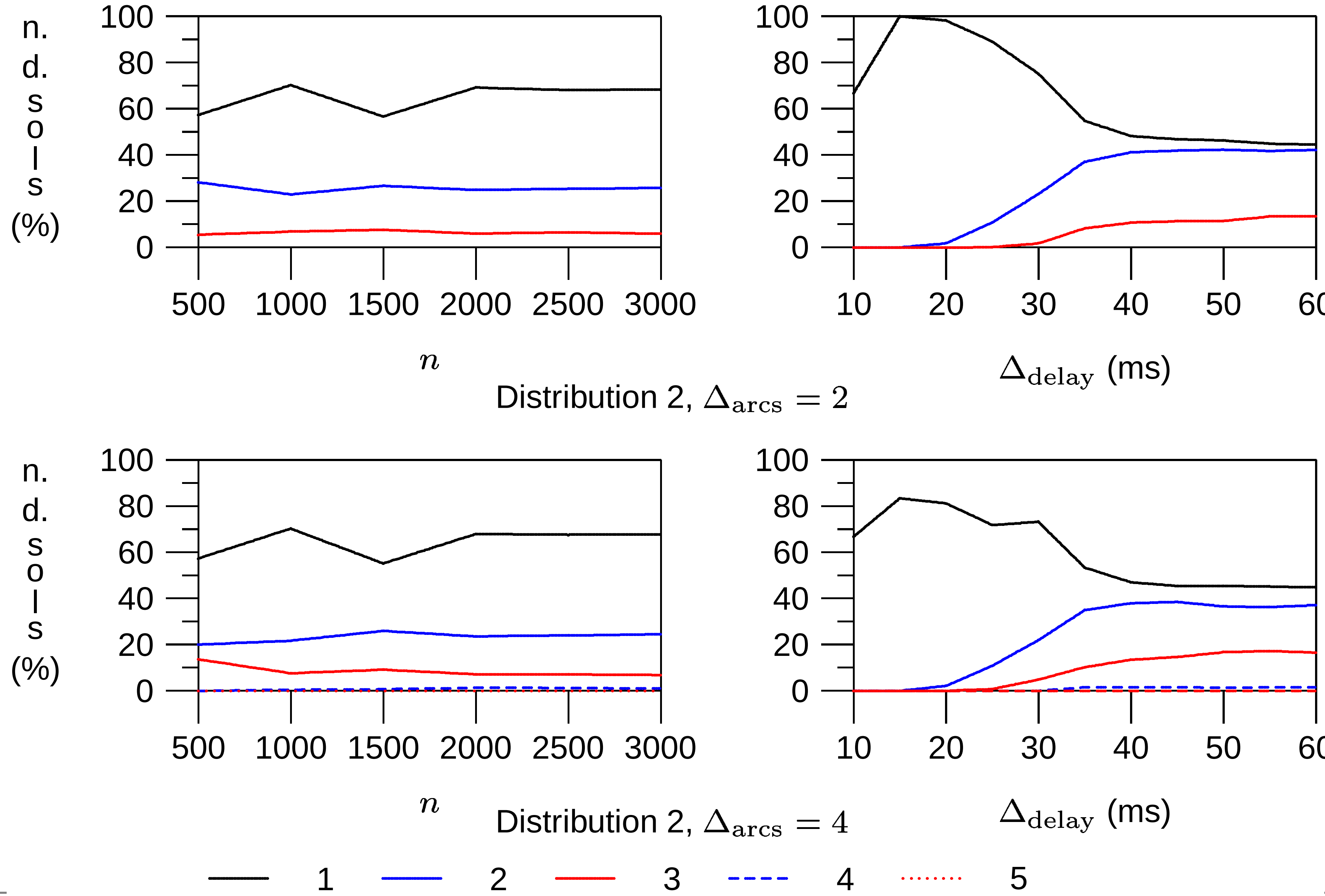


Distribution 1, $\Delta_{\mathrm{arcs}} = 2$

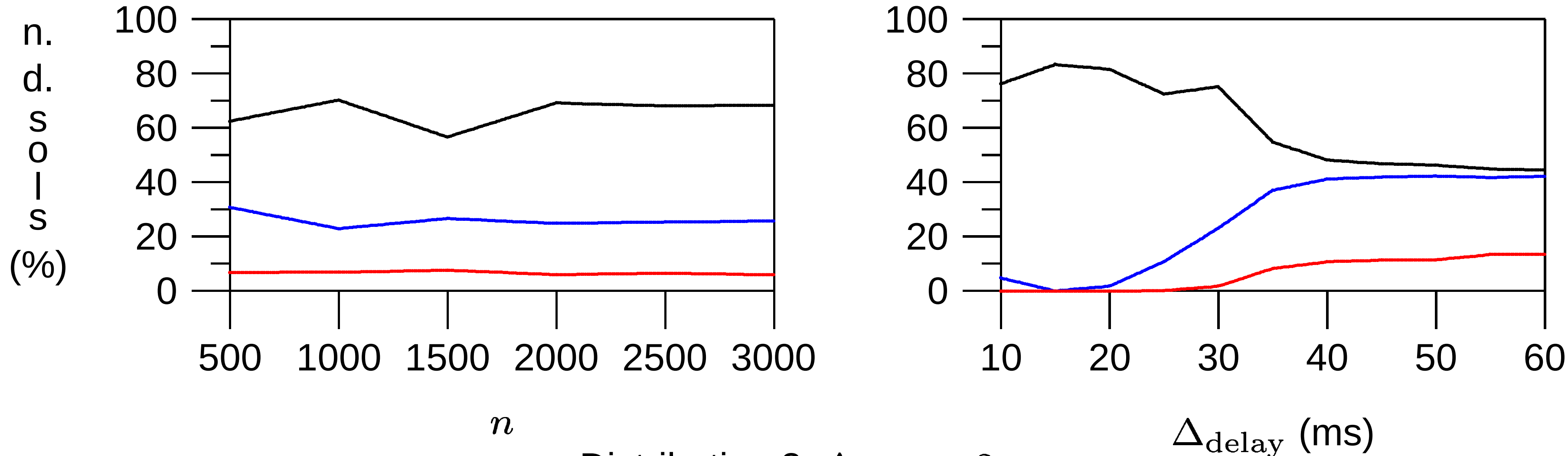


Distribution 1, $\Delta_{\mathrm{arcs}} = 4$

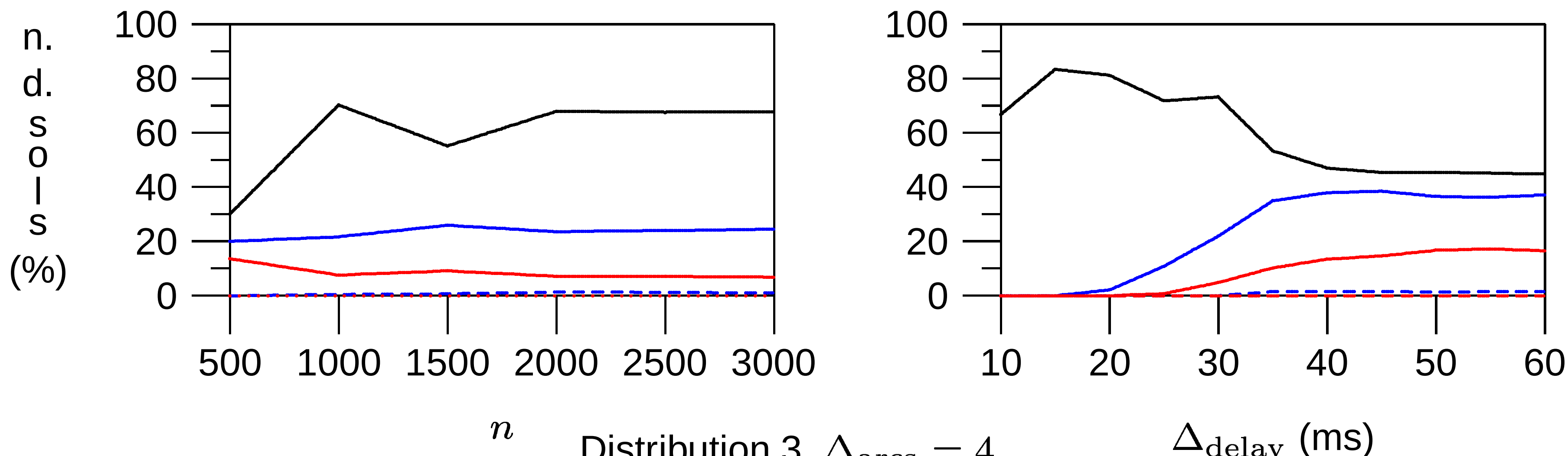
Results – Random networks



Results – Random networks

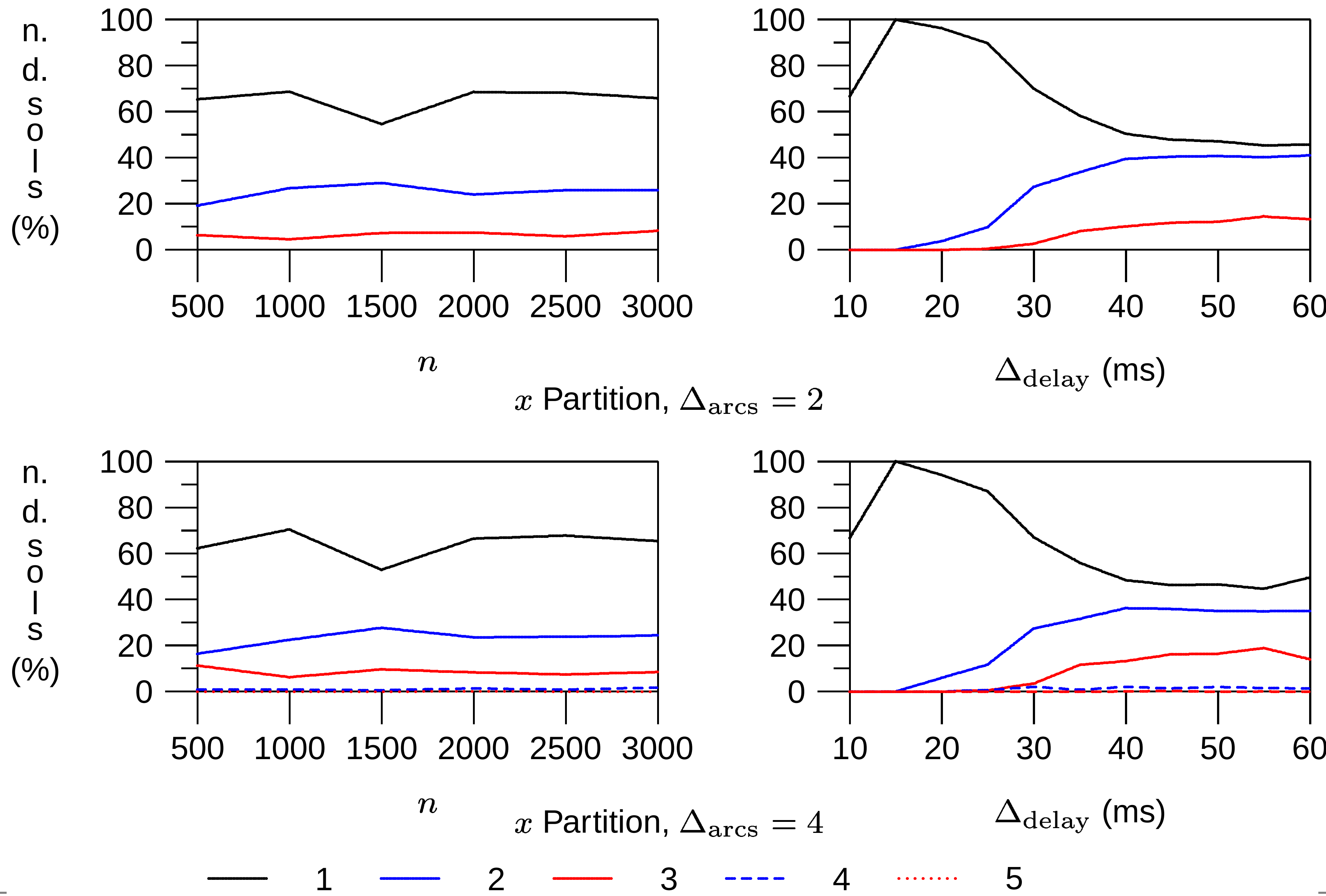


Distribution 3, $\Delta_{\text{arcs}} = 2$

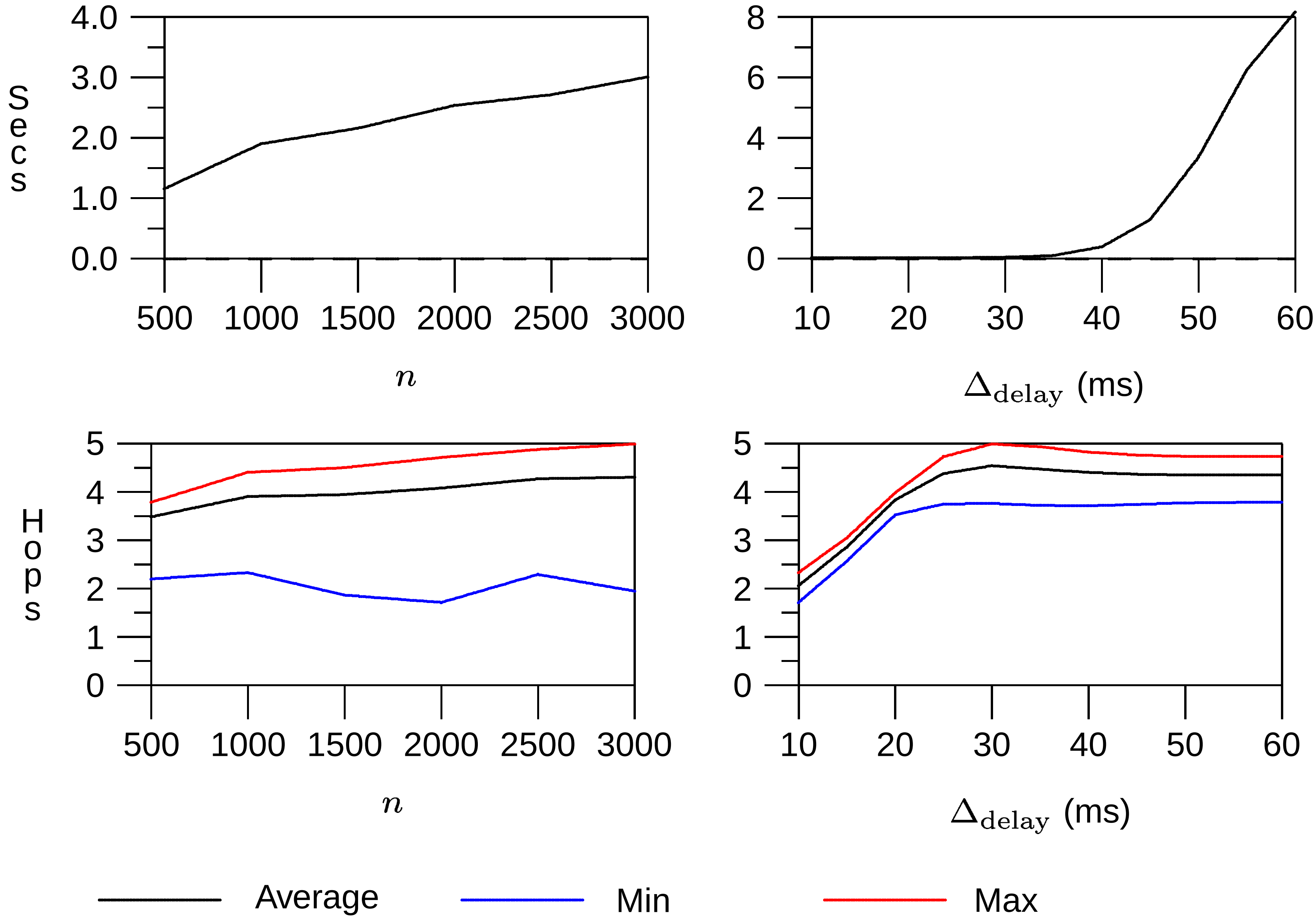


Distribution 3, $\Delta_{\text{arcs}} = 4$

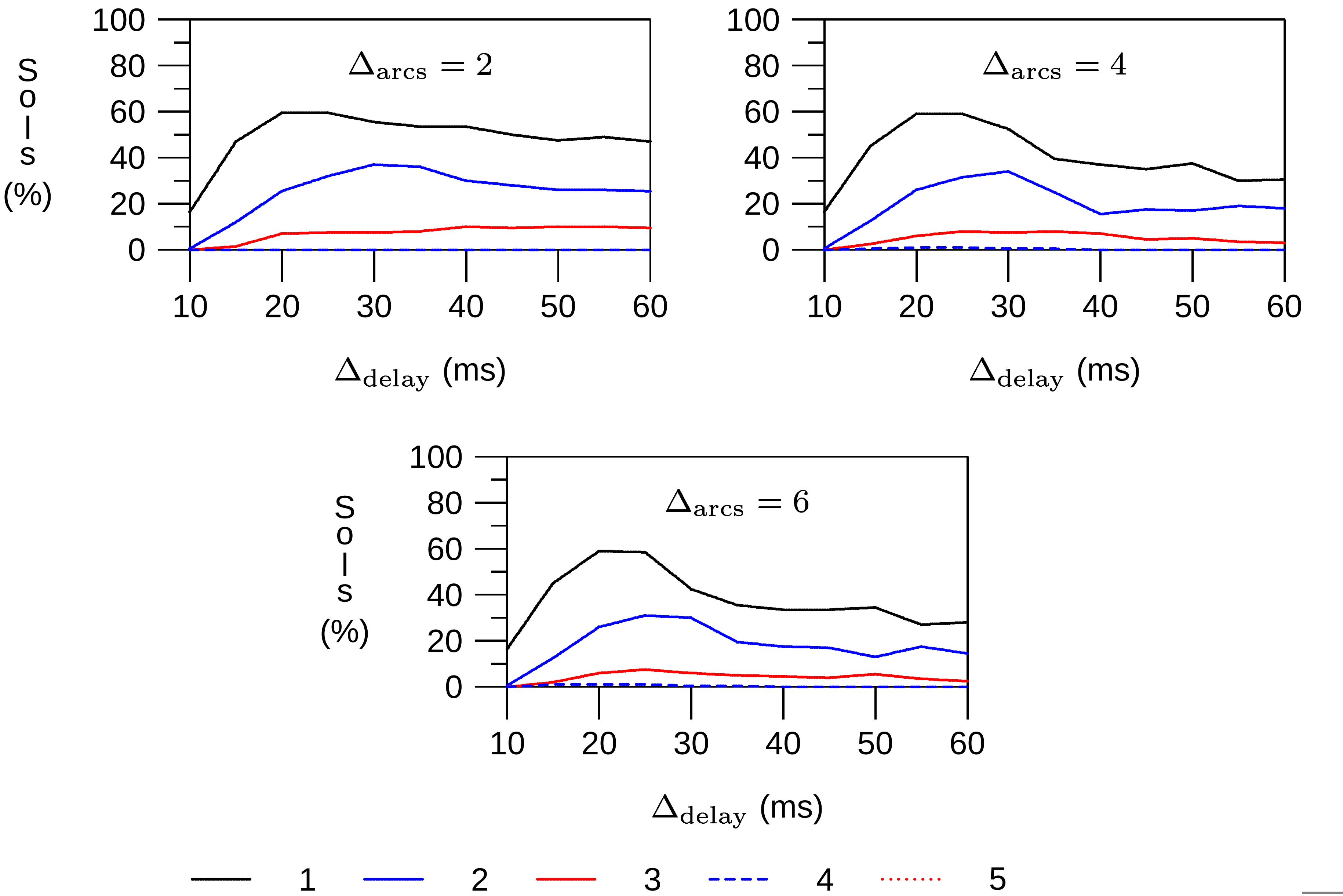
Results – Random networks



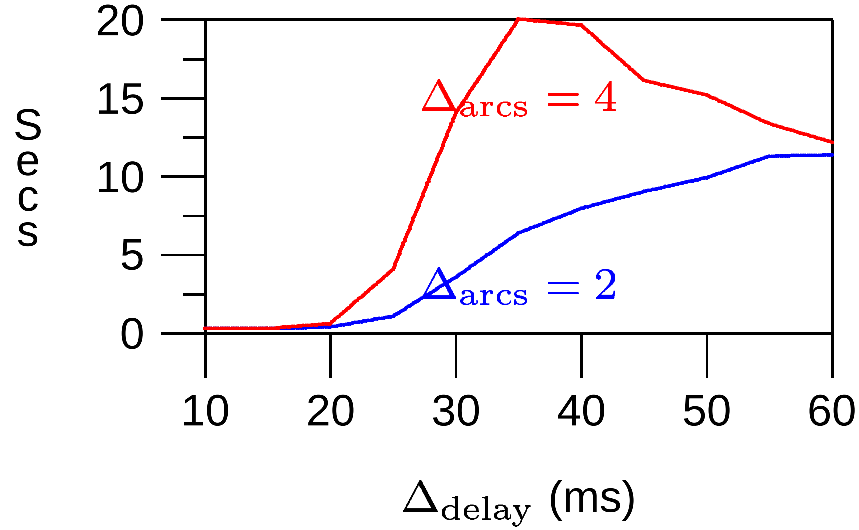
Results – Random networks



Results – US network



Results – US network



Conclusions

Increase Δ_{delay} provokes an increase in the number of problems with more non-dominated solutions (increased conflict between the objective functions for the feasible paths).

Increase Δ_{arcs} from 2 to 4 has even a more relevant impact in the increase in the number of non-dominated solutions.
However it implies a larger number of feasible paths, leading in some cases to lack of memory and thus the computation is not completed.

The profile of the problem non-dominated solutions does not vary significantly with the distribution of b_{ij}.

Some problems have no feasible solution (in practice this would naturally require the relaxation of some constraints).

The results with US network follow the same trends obtained for the randomly generated networks.

Conclusions

Developed a bicriterion model to calculate the whole set of non-dominated paths for traffic flows associated with multimedia type services in multiservice networks.

Adaptation of a ranking type approach for a bicriterion shortest path problem including additional constraints.

The presented algorithm is based on the bicriterion shortest path algorithm by Clímaco & Martins and on the MPS algorithm.

The model was applied to a specific routing problem of video traffic.

Although the objective functions are not strongly conflicting there is a significant percentage of problems with several non-dominated solution, calculated exactly in short processing times and with modest memory requirements; even so in certain real-time applications may be useful the implementation of a distributed calculation version of this procedure.