Robust shortest path problems with uncertain costs

R. Montemanni L.M. Gambardella

Technical Report No. IDSIA-03-08
April 7, 2008

IDSIA / USI-SUPSI
Istituto Dalle Molle di studi sull'intelligenza artificiale
Galleria 2, 6928 Manno, Switzerland

Robust shortest path problems with uncertain costs

R. Montemanni* L.M. Gambardella

April 7, 2008

Abstract

Data coming from real-world applications are very often affected by uncertainty. On the other hand, it is difficult to translate uncertainty in terms of combinatorial optimization. In this paper we study a combinatorial optimization model to deal with uncertainty in arc costs in shortest path problems. We consider a model where feasible arc cost scenarios are described via a convex polytope. We present a computational complexity result and we discuss exact algorithms for two different robust optimization criteria. We finally present experimental results showing that the proposed approaches are suitable to be used on realistic size instances.

Keywords: Shortest path problem, robust optimization, linear programming.

1 Introduction

Different approaches have been developed to cope with uncertainty about real data in combinatorial optimization. Among them the *scenario model* (see [17]), where a finite set of possible alternative graphs is considered at the same time and the *interval data model* (see [15]), where an interval of possible values is associated with each arc.

In this paper we study the application of a model that extends the interval data model to the shortest path problem. It makes use of a set of linear inequalities to define the values of costs that can be simultaneously taken on different arcs. Using these constraints, a polytope of feasible scenarios is defined. This approach generalizes the interval data model, where only a range constraint for each arc cost is allowed, and provides a tool for describing relations among arc costs. The approach we propose can be seen as an alternative to the scenario model, since it has the same modeling power. In our case we describe the polytope of feasible scenarios via its facets, while in the scenario model the polytope is described by its extreme points. Ideas similar to the one we propose have been applied to generic linear programming in [5]. General theoretical results about convex robust optimization and its generalizations can be found in [6], [16] and [1]. Exact algorithms for the shortest path problem with interval data have been proposed in [9], [15], [13] and [14], while approximation algorithms are discussed in [10]. In [17] the scenario case is analysed.

The *absolute robustness criterion* and the *relative robustness criterion* will be considered in this paper. These criteria have been first introduced in [11]. Formal definitions of these criteria will be given in Sections 3.1 and 4.1.

*Corresponding author. Tel. +41 58 666 6667. Fax +41 58 666 6661. Email: roberto@idsia.ch

The absolute robust shortest path problem with interval data is known to be solvable in polynomial time. Here we prove that the version where uncertainty is modeled via a general convex polytope is NP-hard. The robust deviation shortest path problem has been proven to be NP-hard when interval data are considered ([3, 18]), and consequently remains NP-hard in our case (for which box constraints represent a special case).

Other contributions of this paper are the adaptation of known exact solution approaches originally developed for the interval data shortest path problem to the new extended problem (Sections 3.3 and 4.2), and a computational study proving that the proposed approaches are able to handle realistic-size problems (Section 5).

2 Problem description

A directed graph $G = (V, A)$, where V is a set of vertices and A is a set of arcs is given together with a starting vertex $c \in V$, and a destination vertex $t \in V$. At least a path from s to t exists in G exists by hypothesis. A variable $c(i, j) \geq 0$ is associated with each arc $(i, j) \in A$, and a set of linear constraints on variables c defines F, the polytope of feasible scenarios. Formally we have:

$$F = \left\{ c \in R_+^{|A|} \mid Mc \leq b \right\} \tag{1}$$

Starting from this description of the network, two optimization problems are considered: the absolute robust shortest path problem and the robust deviation shortest path problem.

3 Absolute robust shortest path

3.1 Definition

Definition 1. *A* scenario r *is a feasible realization of arc costs, i.e. a vector* $c^r \in F$ *is selected.*

Definition 2. *A path* p *from* s *to* t *is said to be an* absolute robust shortest path *if it has the smallest (among all paths from* s *to* t*) maximum (among all possible feasible scenarios) cost.*

The absolute robust shortest path problem can be described through the following formulation ABS, that also uses the following variables: path variables y, that are those of a classic flow formulation for the shortest path problem (i.e. $y_{ij} = 1$ if arc (i, j) is on the path selected, 0 otherwise).

$$(ABS) \quad \min_{y \in \{0,1\}^{|A|}} \max_{c \in F} c^T y \tag{2}$$

$$\text{s.t. } Ny = e_s - e_t \tag{3}$$

The matrix N in constraint (3) is the incidence matrix of graph G, while vector e_i is the vector of length $|V|$ with 0 in all the entries but entry i, that contains 1. Constraint (3) models a classic flow formulation for the shortest path problem over y variables, that are then forced to form a path from node s to node t. The objective function (2) combines variables c and y to describe the minimax nature of the problem.

3.2 Computational complexity

Theorem 1. *The ABS problem is NP-hard even in layered networks of width 2 and a feasible region for costs with only 2 extreme points.*

Proof. We reduce a well-known NP-complete problem, the 2-partition problem (see [8]), to the ABS problem (the ideas is similar to that used in [17]). First we formally define the 2-partition problem.

The 2-partition problem
Instance: Finite set I of m elements and a size $a_i \in Z_+$ for $i \in I$.
Question: Is there a subset $I' \subseteq I$ such that $\sum_{i \in I'} a_i = \sum_{i \in I \setminus I'} a_i$?

We consider a layered network that consists of $2m+2$ layers, i.e. $V = V_0 \cup V_1 \cup \cdots \cup V_{2m+1}$. The 0th layer contains only the origin node $V_0 = \{s\}$, and the $(2m+1)$th layer contains only the destination node $\{t\}$. The rest of the layers contain 2 nodes each: $V_k = \{v_{ki} \mid i = 1,2\} \forall k = 1,2,\ldots,2m$. Thus, the width of the network is 2. The arc set A contains three sets of arcs that are defined as:

$$A_1 = \{(s, v_{11}), (s, v_{12}), (v_{2m,1}, t), (v_{2m,2}, t)\},$$

$$A_2 = \cup_{k=1}^{m-1} \{(v_{2k,i}, v_{2k+1,j}) \mid \forall i, j = 1, 2\},$$

$$A_3 = \{(v_{2k-1,i}, v_{2k,i}) \mid \forall i = 1, 2, \forall k = 1, 2, \ldots, m\}$$

$A = A_1 \cup A_2 \cup A_3$. We now define an ABS problem with feasible costs described via a convex polytope F defined as follows. All the arcs in A_1 and A_2 have a fixed cost of 0 under any feasible scenario. These arcs are used for making a complete path, together with arcs from A_3, without incurring additional distances. Feasible costs for the arcs in A_3 are defined as follows:

$$c(v_{2k-1,i}, v_{2k,i}) \in [0, a_k] \ \forall i = 1, 2, \forall k = 1, 2, \ldots, m$$

The layered network so obtained is displayed in Figure 1. Moreover, we also have the following constraints integrating the definition of the convex polytope F of feasible scenarios:

$$\frac{1}{a_k} c(v_{2k-1,1}, v_{2k,1}) + \frac{1}{a_k} c(v_{2k-1,2}, v_{2k,2}) = 1 \ \forall k = 1, 2, \ldots, m \tag{4}$$

$$\frac{1}{a_k} c(v_{2k-1,1}, v_{2k,1}) - \frac{1}{a_{k+1}} c(v_{2k+1,1}, v_{2k+2,1}) = 0 \ \forall k = 1, 2, \ldots, m-1 \tag{5}$$

Equations (4) and (5) impose that feasible scenarios are those where the cost vector is a linear combination of the following 2 extreme points:

1. All the costs of arcs in the upper part of the graph of Figure 1 are at their highest possible value, and those of arcs in the lower part of the graph are 0.

2. All the costs of arcs in the lower part of the graph of Figure 1 are at their highest possible value, and those of arcs in the upper part of the graph are 0.

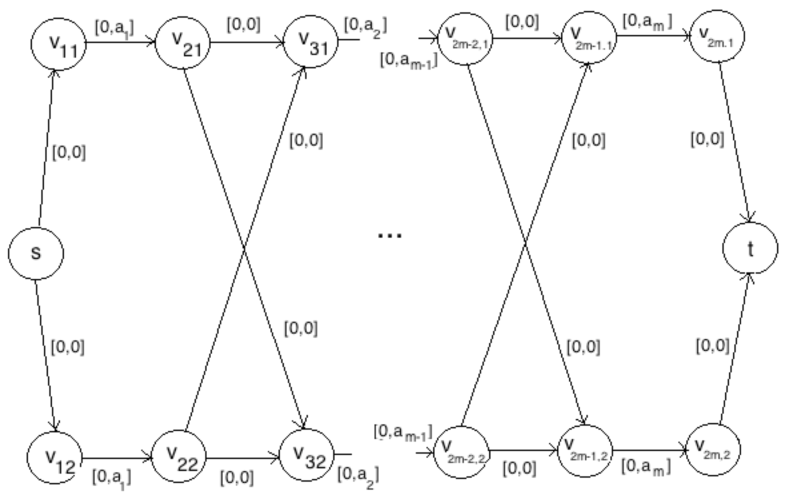

Figure 1: The ABS problem corresponding to a 2-partitioning problem.

With the above construction, let p be a path from s to t, and let

$$I' = \{k \mid \text{ arc } (v_{2k-1,1}, v_{2k,1}) \text{ is traversed by path } p\}$$

Then it must be:

$$I \backslash I' = \{k \mid \text{ arc } (v_{2k-1,2}, v_{2k,2}) \text{ is traversed by path } p\}$$

Let us now consider the scenario $\bar{c}$ where $c(v_{2k-1,1}, v_{2k,1}) = c(v_{2k-1,2}, v_{2k,2}) = \frac{a_k}{2} \forall k = 1, 2, \ldots, m$, and let us indicate with $g(p, c)$ the cost of path p under scenario c and with $g^*(p)$ the absolute robustness cost of path p, which is $g^*(p) = \max_{c \in F}\{g(p, c)\}$. It is easy to verify that scenario $\bar{c}$ is feasible ($\bar{c} \in F$), and that for every path p from s to t we have $g(p, \bar{c}) = \frac{1}{2}\sum_{k \in I} a_k \leq g^*(p)$. If we have a path p^* for which $g^*(p^*) = g(p^*, \bar{c})$, it means that for each feasible scenario $s \in F$, $g(p^*, c) = g^*(p^*)$. This is also true for the feasible scenario where $c(v_{2k-1,1}, v_{2k,1}) = a_k \forall k = 1, 2, \ldots, m$ and $c(v_{2k-1,2}, v_{2k,2}) = 0 \forall k = 1, 2, \ldots, m$ (extreme point 1 of F above). It is easy to verify that I' induces in this case a 2-partition. On the other hand, if a 2-partition exists, it is easy to verify that a path p^* such that $g^*(p^*) = g(p^*, \bar{c})$ must exist.

Thus, a 2-partition exists if and only if the ABS problem has an optimal solution p^* with total length $g^*(p^*) = \frac{1}{2}\sum_{k \in I} a_k$. □

3.3 Algorithms

The algorithms for the absolute robustness criterion described in this section are obtained by specializing the general methods described in [4].

The following formulation $ABS2$ can be obtained from ABS by inserting a free variable z, and by accommodating the original objective function into constraint (7).

$$
\begin{aligned}
(ABS2) \quad & \min z && (6)\\
\text{s.t. } & \text{constraints (3)} \\
& z \geq \max_{c \in F} c^T y && (7)\\
& y \in \{0, 1\}^{|A|} && (8)
\end{aligned}
$$

3.3.1 Algorithm ABSa

The max operator in constraint (7) of formulation $ABS2$ can be eliminated by substituting constraint (7) with the following set of constraints:

$$z \geq c^T y \quad \forall c \in F \qquad (9)$$

We can restrict our attention to the extreme points of F. Let us define P_F as the set of the extreme points of F. We obtain formulation $ABS3$.

$$
\begin{aligned}
(ABS3) \quad & \min z && (10)\\
\text{s.t. } & \text{constraints (3), (8)} \\
& z \geq c_i^T y \qquad \forall c_i \in P_F && (11)
\end{aligned}
$$

A Benders decomposition strategy (see [7]) can be applied to formulation ABS3. The key idea of the method is that only a few constraints of type (11) will be active at optimality: the method generates them step by step only when violated. The algorithm is based on an iterative mechanism. Let ψ represent the iteration number and let P_F^{ψ} represent the restricted set of extreme points of P_F available at iteration ψ. The basic decomposition algorithm can be summarised as follows:

Initialization step: Set $\psi = 1$ and $P_F^1 := \emptyset$.
Main step: Solve the mixed integer problem $ABS3^{\psi}$, which is the relaxed version of $ABS3$ obtained by replacing P_F with P_F^{ψ} in (11). Let now (z^{ψ}, y^{ψ}) be an optimal solution of $ABS3^{\psi}$. Solve the following linear program $AUX_{ABS}(y^{\psi})$, where y^{ψ} acts as a constant vector and c is now a vector of variables.

$$(AUX_{ABS}(y^{\psi})) \quad ub^{\psi} = \max_{c \in F} c^T y^{\psi} \tag{12}$$

If $\min_{\xi \in \{1,2,\ldots,\psi\}} ub^{\xi} \leq z^{\psi}$ then the optimal solution of ABS has been found, **stop**. Otherwise, let c^{ψ} be an optimal solution of $AUX_{ABS}(y^{\psi})$. Set $P_F^{\psi+1} := P_F^{\psi} \cup \{c^{\psi}\}$, $\psi := \psi + 1$ and **repeat the Main step**.

3.3.2 Algorithm ABSb

The use of a technique already experimented in [2] (see also [12]) leads to the following result.

Proposition 1. *Mathematic formulation $ABS2$ is equivalent to the following mixed integer program:*

$$
\begin{aligned}
(ABS4) \quad & \min z && (13)\\
s.t. \quad & \textit{constraints (3), (8)} && \\
& z \geq b^T \mu && (14)\\
& M^T \mu \geq y && (15)\\
& \mu \geq 0 && (16)
\end{aligned}
$$

Proof. Start from formulation $ABS2$, treat y variables as parameters and consider constraint (7). The value for variable z is greater than or equal to the optimal solution of the following problem:

$$(P) \max_{c \in F} c^T y \tag{17}$$

We define variables μ as the dual variables associated with the constraints of F. Strong duality yields the equivalent formulation:

$$
\begin{aligned}
& (D) \min b^T \mu && (18)\\
& \text{s.t. constraints (15), (16)} &&
\end{aligned}
$$

By replacing problem P with problem D, we eliminate the non-linearity of the cost for a path, and obtain the mixed integer program $ABS4$. □

Formulation $ABS4$ can be handled directly by any mixed integer linear programming solver.

4 Robust deviation shortest path

4.1 Definition

Definition 3. *The* robust deviation *for a path p from s to t in a scenario r is the difference between the cost of p in r and the cost of the shortest path from s to t in scenario r.*

Definition 4. *A path p from s to t is said to be a* robust deviation shortest path *if it has the smallest (among all paths from s to t) maximum (among all possible scenarios) robust deviation.*

The following formulation RD of the robust deviation shortest path problem is based on variables y and c, that have the same meaning as in Section 3. The difference between formulations RD and ABS is represented by the objective function, that in RD incorporates also a regret term - namely function $sp(c)$ - that returns, for each possible feasible scenario c, the cost of the respective (classic) shortest path.

$$(RD) \quad \min_{y \in \{0,1\}^{|A|}} \max_{c \in F} (c^T y - sp(c)) \tag{19}$$

$$\text{s.t. constraints (3)-(8)}$$

4.2 Algorithm RD

The algorithm presented is an extension of the method described in [14] for the version of the problem where only interval data are considered.

The following formulation $RD2$ can be obtained from RD by inserting a free variable z, by accommodating the original objective function (19) into constraint (21), and by considering the extreme points P_F of F only (analogously to what done in Section 3.3.1).

$$(RD2) \quad \min z \tag{20}$$

$$\text{s.t. constraints (3), (8)}$$

$$z \geq c^T y - sp(c) \qquad \forall c \in P_F \tag{21}$$

Formulation $RD2$ is in a suitable form to use a Benders decomposition approach. Let ψ represent the iteration number, P_F^{ψ} the restricted set of extreme points of P_F available at iteration ψ, and $RD2^{\psi}$ the relaxed version of $RD2$ obtained by replacing P_F with P_F^{ψ}. Let finally (z^{ψ}, y^{ψ}) be an optimal solution of $RD2^{\psi}$. The algorithm works as described in Section 3.3.1.

A difference with respect to the method described in Section 3.3, is in the construction of the auxiliary problem. Here it is a more complex task, since function $sp(c)$ has to be expressed in a linear form in order to make the problem tractable by a mixed integer linear programming solver. We can formulate the auxiliary problem $AUX_{RD}(y^{\psi})$ as follows. Let variables x model the shortest path problem on the scenario defined by variables c. Namely, $x_{ij} = 1$ if arc (i, j) is on the selected shortest path of scenario c, 0 otherwise. Constraint (23) models flow constraints and defines the shortest path problem, while the respective costs are derived with the help of continuous variables v and constraints (24), where K is a large constant such that $K \geq c_{ij}^r \; \forall (i, j) \in A, \forall c^r \in P_F$. In particular, v_{ij} is forced to be always positive, and to assume value c_{ij} in case $x_{ij} = 1$. The objective function (22) drives optimization to the scenario c that maximizes the value of $c^T y^{\psi}$ minus the cost of the shortest path in scenario c.

$$(AUX_{RD}(y^{\psi})) \quad ub^{\psi} = \max_{c \in F, v \geq 0, x \in \{0,1\}^{|A|}} (c^T y^{\psi} - \mathbf{1}^T v) \tag{22}$$

$$\text{s.t. } Nx = e_s - e_t \tag{23}$$

$$v \geq c + K(x - 1) \tag{24}$$

Where $\mathbf{1}$ is the all-1 $|A|$-long column vector.

Up to our knowledge, this is the first time a practical algorithm is proposed for a problem where uncertainty is described via a general convex polytope and the robust deviation criterion is used to drive optimization.

5 Computational experiments

The execution times of algorithms described in Sections 3.3 and 4.2 are studied. All the algorithms have been coded in ANSI C and ILOG CPLEX[1] 9.0 has been used to solve mixed integer linear programs. All the tests have been carried out on an Intel P4 1.5 GHz / 256 MB machine.

5.1 Random instances

Each random instance is a problem with interval data and added extra inequalities, describing the polytope of feasible costs F. Each instance is defined by the following parameters:

- **n** : number of nodes in the network;
- **d** : approximate arc density of the network;
- **M** : maximum value for arc costs;
- **q** : number of extra inequalities describing F;
- **B** : maximum number of variables involved in each of the q inequalities describing F;
- **Hl** : minimum coefficient allowed in the left-hand side of the inequalities describing F;
- **Hu** : maximum coefficient allowed in the left-hand side of the inequalities describing F;
- **R** : maximum value allowed in the right-hand side of the inequalities describing F (the minimum value is always 0).

Instances are generated according to the parameters above. Namely, an instance of type *(n-d)-(M-q-B-Hl-Hu-R)* will have n nodes and approximately $|A| = d\frac{n(n-1)}{2}$ random arcs. Feasible scenarios are defined by a polytope, where each arc cost c_{ij} can take values within the interval $[l_{ij}, u_{ij}]$, with $0 \leq l_{ij} \leq u_{ij} \leq M$, and subject to a set of q inequalities in the form:

$$\sum_{(i,j) \in Q_k} h_{ij}^k c_{ij} \leq rhs^k \quad \forall Q_k \in \mathcal{Q} \tag{25}$$

[1] www.cplex.com.

where: $\mathcal{Q} = \{Q_1, Q_2, \ldots, Q_q\}$ is a set containing q subsets of A; $|Q_k| \leq B \ \ \forall Q_k \in \mathcal{Q}$; $Hl \leq h_{ij}^k \leq Hu \ \ \forall Q_k \in \mathcal{Q}, \forall (i,j) \in Q_k$; $0 \leq rhs^k \leq R \ \ \forall Q_k \in \mathcal{Q}$.

Inequalities (25) are very general. It might happen that a set of inequalities (25) implies $F = \emptyset$. In this case, the corresponding instance is discharged and a new one is generated.

5.2 Execution times

The execution times of the algorithms discussed in Sections 3.3 and 4.2 over a wide range of problems are reported in Table 1. Average and standard deviation over 50 runs are presented for each problem. Empty entries in the column of algorithm *RD* are due to memory problems: a more sophisticated technique than the one we implemented should have been used to handle these problems.

A first comparison can be done between algorithms ABSa and ABSb, that solve the same optimization problem. ABSb is faster, although computation times are always comparable with those of ABSa. Computation times of ABSb present a smaller standard deviation. Therefore ABSb should be preferred to ABSa in practice.

If the computation times of algorithm ABSa and ABSb are compared with those of RD, it emerges that the use of criterion ABS implies faster algorithms. The explosion of the computation times of RD as the number of nodes of the network increases was expected, and is connected with the complex nature of the robust deviation criterion.

The number of nodes of the network appears to be the most critical factor for the computation times of the algorithms. The other problem-defining parameters seem to have a marginal impact on the performance of the methods.

The aim of the experiments was however to validate algorithms ABSa, ABSb and RD from an experimental point of view. We can therefore conclude that these methods are suitable to be used on realistic size problems.

6 Conclusion

Robust shortest path problems, where arc cost uncertainty is modeled by the meaning of a convex polytope of feasible scenarios, have been discussed. Computational complexity results on an open problem have been presented. Algorithms for the studied problems have been presented and tested from an experimental point of view, showing that these methods are able to handle realistic size instances in reasonable time.

References

[1] H. Aissi, C. Bazgan, and D. Vanderpooten. Approximation of min-max and min-max regret versions of some combinatorial optimization problems. *European Journal of Operational Research*, 179(2):281–290, 2007.

[2] A. Altin, E. Amaldi, P. Belotti, and M.C. Pinar. Provisioning virtual private networks under traffic uncertainty. *Networks*, 49((1)):100–115, 2005.

[3] I. Averbakh and V. Lebedev. Interval data minmax regret network optimization problems. *Discrete Applied Mathematics*, 138:289–301, 2004.

[4] W. Ben-Ameur and H. Kerivin. Routing of uncertain demands. *Optimization and Engineering*, 3:283–313, 2005.

[5] A. Ben-Tal and A. Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming*, 88:411–424, 2000.

[6] A. Ben-Tal and A. Nemirovski. Robust optimization - methodology and applications. *Mathematical Programming Series B*, 92:453–480, 2002.

[7] J.F. Benders. Partitioning procedures for solving mixed integer variables programming problems. *Numerische Mathematik*, 4:238–252, 1962.

[8] M.R. Garey and D.S. Johnson. *Computers and Intractability: A guide to the theory of NP-completeness*. W.H. Freemam and Co., San Francisco, 1979.

[9] O.E. Karaşan, M.Ç. Pinar, and H. Yaman. The robust shortest path problem with interval data. Bilkent University, 2001.

[10] A. Kasperski and P. Zieliński. An approximation algorithm for interval data minmax regret combinatorial optimization problems. *Information Processing Letters*, 97:177–180, 2006.

[11] P. Kouvelis and G. Yu. *Robust Discrete Optimization and its applications*. Kluwer Academic Publishers, 1997.

[12] R. Montemanni. A mixed integer programming formulation for a single machine robust scheduling with interval data. *Journal of Mathematical Modelling and Algorithms*, 6(2):287–296, 2007.

[13] R. Montemanni and L.M. Gambardella. An exact algorithm for the robust shortest path problem with interval data. *Computers and Operations Research*, 31(10):1667–1680, 2004.

[14] R. Montemanni and L.M. Gambardella. The robust shortest path problem with interval data via Benders decomposition. *4OR*, 3(4):315–328, 2005.

[15] R. Montemanni, L.M. Gambardella, and A.V. Donati. A branch and bound algorithm for the robust shortest path problem with interval data. *Operations Research Letters*, 32(3):225–232, 2004.

[16] M. Salazar-Neumann. The robust minimum spanning tree problem: compact and convex uncertainty. *Operations Research Letters*, 35:17–22, 2007.

[17] G. Yu and J. Yang. On the robust shortest path problem. *Computers and Operations Research*, 25(6):457–468, 1998.

[18] P. Zieliński. The computational complexity of the relative robust shortest path problem with interval data. *European Journal of Operational Research*, 158(3):570–576, 2004.

Table 1: Execution times (in seconds) of the algorithms described in Sections 3.3 and 4.2. Averages and standard deviations over 50 instances.

Problems	Absolute robustness				Robust deviation	
	ABSa		ABSb		RD	
	Avg	StD	Avg	StD	Avg	StD
(100-0.75)-(100-0-40-0-10-0.2)	0.747	0.291	0.694	0.287	17.479	15.477
(100-0.75)-(100-25-40-0-10-0.2)	0.737	0.292	0.795	0.293	6.644	5.507
(100-0.75)-(100-50-40-0-10-0.2)	0.818	0.291	0.843	0.287	4.332	2.372
(100-0.75)-(1000-0-40-0-10-0.2)	0.774	0.290	0.639	0.282	27.5	30.477
(100-0.75)-(1000-25-40-0-10-0.2)	0.847	0.317	0.832	0.291	11.806	14.631
(100-0.75)-(1000-50-40-0-10-0.2)	0.875	0.364	0.813	0.296	6.81	4.643
(100-0.25)-(100-0-40-0-10-0.2)	0.548	0.542	0.617	0.550	5.78	5.508
(100-0.25)-(100-25-40-0-10-0.2)	0.609	0.290	0.645	0.285	1.559	0.994
(100-0.25)-(100-50-40-0-10-0.2)	0.619	0.295	0.668	0.282	0.834	0.534
(100-0.75)-(100-0-20-0-10-0.2)	0.691	0.292	0.676	0.288	13.691	15.425
(100-0.75)-(100-25-20-0-10-0.2)	0.742	0.290	0.752	0.290	8.031	4.990
(100-0.75)-(100-50-20-0-10-0.2)	0.802	0.325	0.795	0.291	6.487	4.769
(100-0.75)-(100-0-40-0-100-0.2)	0.693	0.702	0.742	0.697	17.443	15.416
(100-0.75)-(100-25-40-0-100-0.2)	0.747	0.765	0.789	0.745	6.518	6.194
(100-0.75)-(100-50-40-0-100-0.2)	0.790	0.793	0.848	0.856	4.114	2.876
(200-0.75)-(100-0-40-0-10-0.2)	1.864	0.295	1.422	0.285	83.35	72.494
(200-0.75)-(100-25-40-0-10-0.2)	2.186	0.474	1.710	0.266	68.716	63.612
(200-0.75)-(100-50-40-0-10-0.2)	2.238	0.700	1.965	0.276	59.136	53.825
(200-0.75)-(1000-0-40-0-10-0.2)	1.861	0.287	1.413	0.297	200.154	261.825
(200-0.75)-(1000-25-40-0-10-0.2)	2.162	0.528	1.719	0.314	110.727	107.65
(200-0.75)-(1000-50-40-0-10-0.2)	2.480	0.870	1.940	0.311	139.742	249.659
(200-0.25)-(100-0-40-0-10-0.2)	0.919	0.294	0.825	0.292	38.329	31.748
(200-0.25)-(100-25-40-0-10-0.2)	1.121	0.286	0.934	0.315	25.05	20.429
(200-0.25)-(100-50-40-0-10-0.2)	1.144	0.372	1.043	0.270	21.174	24.26
(200-0.75)-(100-0-20-0-10-0.2)	1.804	0.286	1.379	0.289	92.664	97.894
(200-0.75)-(100-25-20-0-10-0.2)	1.924	0.321	1.594	0.305	88.381	88.841
(200-0.75)-(100-50-20-0-10-0.2)	1.966	0.362	1.809	0.303	75.194	73.324
(200-0.75)-(100-0-40-0-100-0.2)	1.900	0.295	1.419	0.290	83.188	72.218
(200-0.75)-(100-25-40-0-100-0.2)	2.069	0.429	1.724	0.305	60.868	59.815
(200-0.75)-(100-50-40-0-100-0.2)	2.265	0.628	1.972	0.325	54.402	54.839
(100-0.75)-(100-0-40-0-10-80)	0.666	0.293	0.737	0.286	17.426	15.4
(100-0.75)-(100-25-40-0-10-80)	1.031	0.333	0.758	0.302	16.813	14.905
(100-0.75)-(100-50-40-0-10-80)	1.427	0.822	0.881	0.276	15.822	14.383
(200-0.75)-(100-0-40-0-10-80)	1.881	0.295	1.357	0.298	83.426	72.286
(200-0.75)-(100-25-40-0-10-80)	2.187	0.518	1.675	0.271	80.31	69.67
(200-0.75)-(100-50-40-0-10-80)	2.600	0.914	1.949	0.297	78.748	67.645
(500-0.75)-(100-0-40-0-10-80)	15.098	1.087	9.928	0.295	557.503	22.360
(500-0.75)-(100-25-40-0-10-80)	15.768	2.722	10.958	0.329	-	-
(500-0.75)-(100-50-40-0-10-80)	16.011	2.185	12.239	0.386	-	-