Dealing with residual energy when transmitting data in energy-constrained capacitated networks

Herminia I. Calvete, Lourdes del-Pozo, José A. Iranzo

PII: S0377-2217(18)30174-7
DOI: 10.1016/j.ejor.2018.02.041
Reference: EOR 15001

To appear in: *European Journal of Operational Research*

Received date: 3 April 2017
Revised date: 31 January 2018
Accepted date: 15 February 2018

Please cite this article as: Herminia I. Calvete, Lourdes del-Pozo, José A. Iranzo, Dealing with residual energy when transmitting data in energy-constrained capacitated networks, *European Journal of Operational Research* (2018), doi: 10.1016/j.ejor.2018.02.041

Highlights

- Single objective: To maximize the minimum of the residual energy in a capacitated network
- Bi-objective: To minimize the transmission time and maximize the minimum of the residual energy
- Polynomial time algorithms for both problems
- Only shortest path problems are solved
- Experiments display the valuable contribution of the algorithm

Dealing with residual energy when transmitting data in energy-constrained capacitated networks☆

Herminia I. Calvete[a,*], Lourdes del-Pozo[b], José A. Iranzo[c]

[a] *Dpto. de Métodos Estadísticos, IUMA, Universidad de Zaragoza, Pedro Cerbuna 12, 50009 Zaragoza, Spain. Corresponding author.*

[b] *Dpto. de Métodos Estadísticos, Universidad de Zaragoza, Violante de Hungría 23, 50009 Zaragoza, Spain.*

[c] *Centro Universitario de la Defensa de Zaragoza, Carretera de Huesca s/n, 50090 Zaragoza, Spain.*

Abstract

This paper addresses several problems relating to the energy available after the transmission of a given amount of data in a capacitated network. The arcs have an associated parameter representing the energy consumed during the transmission along the arc and the nodes have limited power to transmit data. In the first part of the paper, we consider the problem of designing a path which maximizes the minimum of the residual energy remaining at the nodes. After formulating the problem and proving the main theoretical results, a polynomial time algorithm is proposed based on computing maxmin paths in a sequence of non-capacitated networks. In the second part of the paper, the problem of obtaining a quickest path in this context is analyzed. First, the bi-objective variant of this problem is considered in which we aim to minimize the transmission time and to maximize the minimum residual energy. An exact polynomial time algorithm is proposed to find a minimal complete set of efficient solutions which amounts to solving shortest path problems. Second, the problem of computing an energy-constrained quickest path which guarantees at least a given residual energy at the nodes is reformulated as a variant of the energy-constrained quickest path problem. The algorithms are tested on a set of benchmark problems providing the optimal solution or the Pareto front within reasonable computing times.

Keywords: networks; quickest path; energy constraint; minsum-maxmin; bi-objective optimization

1. Introduction

When transmitting data in a capacitated network, the transmission time depends on two parameters, an additive function which represents the traversal time or the delay along the path and a bottleneck function which represents the path capacity. The quickest path problem (QPP) has been proposed by Chen and Chin [6] to model these kinds of transmission problems when the goal is to design a path in a directed network which minimizes the time taken to transmit a given amount of data. Previously, the QPP had been introduced by Moore [21] to model flows of convoy-type traffic. Martins and Santos [20] and Pelegrín and Fernández [24] approached the QPP as a special minsum-maxmin bi-objective path problem.

Let $\mathcal{G} = [\mathcal{N}, \mathcal{A}]$ be a directed network without multiple arcs and self loops, where $\mathcal{N}$ denotes the set of nodes and $\mathcal{A}$ the set of directed arcs. Let n be the number of nodes and m the number of arcs. Let s and t be two distinguished nodes in the network called, respectively, origin and destination. Let $\sigma \in \mathbb{R}^+$ be the size of the message, i.e. the data units to be sent from node s to node t. Each arc $(u, v) \in \mathcal{A}$ has associated to it a capacity $c(u, v) > 0$ and a delay time $l(u, v) \geqslant 0$. The capacity represents the amount of data that can be sent through arc (u, v) per time unit. The delay time is the time required to traverse the arc (u, v).

A simple path or loopless path P from node s to node t is a sequence of distinct nodes and arcs $P = (s = u_1, u_2, \ldots, u_k = t)$ such that $u_i \in \mathcal{N}$, $i = 1, \ldots, k$, and $(u_i, u_{i+1}) \in \mathcal{A}$, $i = 1, \ldots, k-1$. In the paper, we use the term path instead of simple or loopless path for short as well as the term $s-t$ path instead of a path from s to t. We assume that the set of $s-t$ paths in the network $\mathcal{G}$ is nonempty.

☆This research work has been funded by the Gobierno de Aragón under grant E58 (FSE).
*Corresponding author

Email addresses: herminia@unizar.es (Herminia I. Calvete), lpozo@unizar.es (Lourdes del-Pozo), joseani@unizar.es (José A. Iranzo)

 Received: date / Accepted: date

Assuming that a message is transmitted as a continuous stream along the arc (u, v) at a constant flow rate $\rho \leqslant c(u, v)$, a message of σ data units is sent from node u to node v through arc (u, v) in $l(u, v) + \frac{\sigma}{\rho}$ time. This expression takes its minimum value when $\rho = c(u, v)$. Hence, the minimum required transmission time is $l(u, v) + \frac{\sigma}{c(u,v)}$.

If σ data units are sent at a constant rate from s to t along the $s - t$ path P with no buffering at intermediate nodes (circuit switching mode), the minimum transmission time or end-to-end delay of path P is [26]:

$$T_\sigma(P) = l(P) + \frac{\sigma}{c(P)} \tag{1}$$

where $l(P)$ and $c(P)$ denote the delay time and the capacity of path P, respectively:

$$\begin{aligned} l(P) &= \sum_{i=1}^{k-1} l(u_i, u_{i+1}) \\ c(P) &= \min_{i=1,\ldots,k-1} c(u_i, u_{i+1}) \end{aligned} \tag{2}$$

Hence, the QPP is formulated as:

$$\begin{aligned} \min_{P} \quad & T_\sigma(P) \\ \text{s.t.} \quad & \\ & P \text{ is an } s - t \text{ path in the network } \mathcal{G} \end{aligned} \tag{3}$$

In a sense, the QPP can be viewed as a generalization of the shortest path problem. However, it is worth mentioning that the QPP does not satisfy the property known as 'the optimality principle', that is to say, an $s' - t'$ subpath of an optimal $s - t$ path is not necessarily an $s' - t'$ optimal path. Several polynomial time algorithms have been proposed in the literature for solving the QPP, all of them with time complexity $O(r(m + n\log(n)))$, where r is the number of distinct capacities in the network [3, 6, 20, 21, 22, 24, 27, 28]. They are based on solving a sequence of shortest path problems, using label-setting techniques or applying the fact that a quickest path is a supported efficient solution of the bi-objective problem whose objectives are to minimize the delay time and maximize the capacity of the path. Although all the algorithms have the same time complexity, it is worth mentioning at this point that they do not have the same space complexity. The algorithms developed in [3, 6, 22] use $O(rm + rn)$ space, whereas the algorithms proposed in [20, 21, 24, 27, 28] use $O(m + n)$ space.

Clímaco et al. [7] applied the QPP to the routing of data packets in Internet networks. Hamacher and Tjandra [11] proposed the model in a special evacuation problem where evacuees may use only a single path or tunnel to move from their initial position. Problems in which the QPP is constrained in some way have also been considered in the literature. Chen and Hung [5] and Rosen et al. [27] studied the QPP constrained to contain a given subpath. Calvete and del-Pozo [2] analyzed the problem of determining the transmission process when data are transmitted in batches of variable size but with required limits. The QPP when traversal times can fluctuate depending on traffic conditions was considered in [1]. The problem of computing a quickest path constrained to have a given reliability was studied in [3]. Lin [17, 18] considered the case when the capacity of arcs is assumed to be stochastic. Lin then evaluated the probability that a stochastic flow network allows the transmission of a given amount of flow through one path [17] or multiple disjoint paths [18] within a fixed amount of time. Exact algorithms were developed for solving these NP-hard problems. El Khadiri and Yeh [14] proposed to perform estimations by a Monte-Carlo simulation method to deal with the problem addressed in [17]. Pascoal et al. [23] provide a survey on the subject.

In a recent paper, Calvete et al. [4] have introduced the problem of computing a quickest path problem in an energy-constrained capacitated network. In this context, each arc $(u, v) \in \mathcal{A}$ has an additional parameter associated to it representing the energy consumed at node u during the transmission of the message along the arc, while each node $u \in \mathcal{N}$ is endowed with a limited power to transmit messages. Then, the energy-constrained quickest path problem is formulated as the problem of finding a quickest path whose nodes are able to support the transmission of the given data units.

The importance of dealing with the residual energy at nodes has been considered in some papers [9]. As indicated in [13], a precise management of energy affecting factors is critical in order to obtain network lifetimes which are long enough. For instance, a load-balancing of the energy consumption of individual sensor nodes is critical for data preservation in wireless networks under bandwidth constraints [29]. Also, preserving and balancing the residual energy capacities in a wireless sensor network is addressed in [16]. Following this line of thinking, in this paper we focus on the residual energy at the nodes after transmitting a given amount of data in an energy-constrained capacitated network. In the first part of the paper, we introduce the maxmin energy-constrained path problem (mm-EPP) whose goal is to find a path which allows the data transmission and maximizes the minimum residual energy at the nodes after the transmission of the message. The main theoretical result proves that an optimal solution can be obtained by solving a maxmin problem without additional constraints on the available energy in a subnetwork of the original network. Based on this result, and taking into account that this subnetwork is unknown a priori, a polynomial time algorithm is developed which computes maxmin paths in a sequence of subnetworks of the original network which allows the transmission at a certain flow rate. In the second part of the paper, we introduce the minsum-maxmin bi-objective energy-constrained quickest path problem (msmm-EQPP) which aims to find a path which is able to transmit the given data units while minimizing the transmission time and maximizing the minimum residual energy. We prove that the set of all efficient paths can be obtained by solving minsum-maxmin bi-objective path problems in the above mentioned subnetworks and we develop a polynomial time algorithm to determine the set of non-dominated points. Finally, the problem of obtaining an energy-constrained quickest path which preserves at least a given residual energy at the nodes is analyzed. It is proved that this can be reformulated as an energy-constrained quickest path problem by redefining the residual energy. The paper is structured as follows. In Section 2 additional notations and definitions are provided. Section 3 analyzes the mm-EPP and proves the main theoretical results which support the polynomial algorithm developed for solving it. Section 4 formally sets out the msmm-EQPP and goes on to study its properties and develop a polynomial algorithm for finding a minimal complete set of efficient paths. Section 5 reformulates the problem of computing a QPP in the set of paths which guarantee a certain residual energy at the nodes as an energy-constrained quickest path problem. Section 6 presents the results of the computational experiment carried out to assess the performance of the algorithms. Finally, our conclusions are presented in Section 7.

2. Preliminaries

Before setting out the problems which are the subject of this paper, in this section we introduce some additional definitions and notations. Let $\mathcal{G} = [\mathcal{N}, \mathcal{A}]$ be the network introduced in Section 1. We assume that each arc $(u, v) \in \mathcal{A}$ has associated to it an energy rate $\omega(u, v) \geqslant 0$, which measures the energy required at node u to transmit data units along the arc (u, v) per time unit. Each node $u \in \mathcal{N}$ has an associated power $b_u > 0$ which represents the limited energy available for transmission at node u. This available power must be considered when selecting a path to transmit the message since the energy consumed at node u due to the transmission of the message along the arc (u, v) depends on the units of time during which node u is active, i.e. while it is sending data. Hence, it depends on the rate at which data are transmitted.

If σ data units are transmitted from node u to node v through arc (u, v) at a constant flow rate $\rho \leqslant c(u, v)$, the node u is active during $\frac{\sigma}{\rho}$ time units. Hence the required energy at node u is $\omega(u, v)\ \frac{\sigma}{\rho}$. Without loss of generality, we assume that

$$\omega(u, v)\frac{\sigma}{c(u, v)} \leqslant b_u, \ \ \forall (u, v) \in \mathcal{A} \tag{4}$$

Otherwise, the arc (u, v) cannot support the transmission of the σ data units and should be removed.

Let $P = (s = u_1, u_2, \ldots, u_k = t)$ be an $s - t$ path. The capacity $c(P)$ provides the maximum rate at which data can be transmitted along the path P, hence it is the flow rate which needs the least time to send the σ data units along P. At this flow rate every node in the path is active for the least time and so the required energy at each node is the least. Hence, assuming that this flow rate is used for the transmission, we define the residual energy at node u after transmitting σ data units along the path P as:

$$b_u(\sigma, P) = \begin{cases} b_u - \omega(u, u_{i+1})\dfrac{\sigma}{c(P)} & \text{if } u = u_i, i = 1, \dots, k-1 \\ b_u & \text{otherwise} \end{cases} \tag{5}$$

An $s-t$ path P is of interest if it is able to transmit the σ data units. Hence, we say that P is an $s-t$ feasible path if $b_u(\sigma, P) \geqslant 0$, $\forall u \in P$. In other words, an $s-t$ path is feasible if all its nodes are able to transmit the message of size σ at a rate $c(P)$.

Without loss of generality, in what follows we will assume that there are r different arc capacities $c_1 < c_2 < \cdots < c_r$ in the network $\mathcal{G}$.

3. The maxmin energy-constrained path problem

In this section, we focus on the problem of obtaining an $s-t$ feasible path in $\mathcal{G}$ which maximizes the minimum of the residual energy remaining at the nodes after transmitting σ data units. No attention is paid to the transmission time, which will be dealt with in Section 4. Therefore, only the capacity and energy rate parameters associated to the arcs as well as the energy available at the nodes are relevant.

Let $P = (s = u_1, u_2, \dots, u_k = t)$ be an $s-t$ feasible path. From (5), after transmitting σ data units by using the path P, only the energy available at nodes $u_1, \dots, u_{k-1}$ is modified. The power of node $u_k = t$ is not altered because this is the end node of the path, so it is not a 'sender' node. We define the minimum residual energy of the path P as:

$$R_\sigma(P) = \min_{i=1,\dots,k-1} b_{u_i}(\sigma, P) \tag{6}$$

Hence, the problem of finding a path which maximizes the minimum residual energy at the nodes, called the maxmin energy-constrained path problem, can be formulated as:

$$\begin{aligned} \text{mm-EPP}: \quad & \max_{P} \quad R_\sigma(P) \\ & \text{s.t.} \\ & \qquad b_u(\sigma, P) \geqslant 0, u \in \mathcal{N} \\ & \qquad P \text{ is an } s-t \text{ path in } \mathcal{G} \end{aligned} \tag{7}$$

The mm-EPP is a maxmin problem with an additional constraint on the residual energy. Next we will prove that, in order to solve this problem, it is enough to solve a sequence of raw maxmin problems in subnetworks of the original network in which it is assured that the energy available at the nodes allows the transmission of the message at a certain flow rate. For this purpose, for each arc $(u, v) \in \mathcal{A}$ we define:

$$c^{\min}(u, v) = \min_{i=1,\ldots,r} \left\{ c_i : b_u - \omega(u, v)\frac{\sigma}{c_i} \geqslant 0 \right\}$$

This value provides the minimum rate at which the node u can support the transmission of the σ data units along the arc (u, v). As a consequence, the arc (u, v) can be in an $s-t$ feasible path P only if $c(P) \geqslant c^{\min}(u, v)$.

Let us define $\mathcal{G}_j = [\mathcal{N}, \mathcal{A}_j]$, $j = 1, \ldots, r$, a subnetwork of $\mathcal{G}$ where

$$(u, v) \in \mathcal{A}_j \quad \text{iff} \quad (u, v) \in \mathcal{A},\ c(u, v) \geqslant c_j \ \text{ and } \ c^{\min}(u, v) \leqslant c_j \tag{8}$$

Networks $\mathcal{G}_j$ were introduced in [4] to solve the energy-constrained quickest path problem. As mentioned there, in general the network $\mathcal{G}_{j+1}$ is not a subnetwork of $\mathcal{G}_j$ and so the number of arcs in the successive networks does not necessarily decrease. Hence, paths can 'appear' or 'disappear' in successive subnetworks. By way of illustration, let us consider a network $\mathcal{G}$ with three different arc capacities $c_1 < c_2 < c_3$. Let P be an $s-t$ path with capacity $c(P) = c_2$ such that $c^{\min}(u_1, u_2) = c_1$ and $c^{\min}(u_i, u_{i+1}) = c_2$, $i = 2, \ldots, k-1$. Then, P is not an $s-t$ path in $\mathcal{G}_1$ since $c^{\min}(u_i, u_{i+1}) \nleqslant c_1$, $i = 2, \ldots, k-1$. Indeed, P is an $s-t$ path in $\mathcal{G}_2$ since $c(u_i, u_{i+1}) \geqslant c_2$ and $c^{\min}(u_i, u_{i+1}) \leqslant c_2$, $i = 1, \ldots, k-1$. Finally, P is not an $s-t$ path in the network $\mathcal{G}_3$ since $c(P) = c_2$ and so $c(u_i, u_{i+1}) \ngeqslant c_3$, for some $i = 1, \ldots, k-1$.

Notice that the network $\mathcal{G}_j$ contains the arcs $(u,v) \in \mathcal{A}$ with capacity greater than or equal to c_j, such that its start node u is able to support the transmission of the message through this arc at a flow rate c_j. Therefore, by construction, the network $\mathcal{G}_j$ contains the $s-t$ paths P with capacity $c(P) \geqslant c_j$ which can transmit the given data at a rate c_j. In particular, $\mathcal{G}_j$ contains all the $s-t$ feasible paths with capacity c_j.

In the network $\mathcal{G}_j$, we associate to each arc $(u,v) \in \mathcal{A}_j$ a weight:

$$b^j_\sigma(u,v) = b_u - \omega(u,v)\ \frac{\sigma}{c_j} \tag{9}$$

This parameter represents the residual energy at node u after transmitting σ data units along the arc (u,v) at a constant flow rate c_j.

Let $P = (s = u_1, u_2, \ldots, u_k = t)$ be an $s-t$ path in $\mathcal{G}_j$. We define the minimum residual energy of the path P in the network $\mathcal{G}_j$ as:

$$R^j_\sigma(P) = \min_{i=1,\ldots,k-1} b^j_\sigma(u_i, u_{i+1}) \tag{10}$$

i.e. it is assumed that the transmission is made at flow rate c_j. It is worth pointing out that if P is an $s-t$ path in $\mathcal{G}_j$ and $c(P) = c_j$, then $R_\sigma(P) = R^j_\sigma(P)$. In contrast, if $c(P) > c_j$, then $R_\sigma(P) > R^j_\sigma(P)$.

Let us introduce the following maxmin path problem in $\mathcal{G}_j$:

$$\begin{aligned} \text{mm-PP}_j: \quad & \max_P \quad R^j_\sigma(P) \\ & \text{s.t.} \\ & \qquad\quad P \text{ is an } s-t \text{ path in } \mathcal{G}_j \end{aligned} \tag{11}$$

To simplify the notation, a path P which solves problem (11) will be called an $s-t$ maxmin path in $\mathcal{G}_j$. The following results establish the relationship between problems mm-EPP and mm-PP$_j$, $j = 1, \ldots, r$.

Lemma 1. *Let $P = (s = u_1, u_2, \ldots, u_k = t)$ be an $s-t$ path in $\mathcal{G}_j$. Then, P is an $s-t$ feasible path for the mm-EPP.*

Proof. As P is an $s-t$ path in $\mathcal{G}_j$, it is an $s-t$ path in $\mathcal{G}$. Moreover, $c(P) \geqslant c_j \geqslant c^{\min}(u_i, u_{i+1})$, $i = 1, \dots, k-1$. Thus

$$b_{u_i}(\sigma, P) = b_{u_i} - \omega(u_i, u_{i+1})\frac{\sigma}{c(P)} \geqslant b_{u_i} - \omega(u_i, u_{i+1})\frac{\sigma}{c^{\min}(u_i, u_{i+1})} \geqslant 0$$

Nodes which are not in P do not consume energy. Hence, $b_u(\sigma, P) \geqslant 0$, $\forall u \in \mathcal{N}$. □

Lemma 2. *Let $P = (s = u_1, u_2, \dots, u_k = t)$ be an $s-t$ feasible path for the mm-EPP with capacity $c(P) = c_j$. Then, P is an $s-t$ path in $\mathcal{G}_j$.*

Proof. Since the path P is feasible, $b_{u_i}(\sigma, P) \geqslant 0$, $i = 1, \dots, k-1$. Therefore,

$$c^{\min}(u_i, u_{i+1}) \leqslant c(P) = c_j \leqslant c(u_i, u_{i+1}), \quad i = 1, \dots, k-1$$

Hence, the arc $(u_i, u_{i+1}) \in \mathcal{A}_j$, $i = 1, \dots, k-1$, and the conclusion follows. □

Lemma 3. *If P is an $s-t$ maxmin path in $\mathcal{G}_j$ and $c(P) = c_h > c_j$, then there is no optimal solution of the mm-EPP with capacity c_j.*

Proof. Let Q be an $s-t$ feasible path for the mm-EPP with capacity c_j. Then Q is a path in $\mathcal{G}_j$ and

$$R_\sigma(P) > R^j_\sigma(P) \geqslant R^j_\sigma(Q) = R_\sigma(Q)$$

Thus, Q cannot be an optimal solution of the mm-EPP. □

Theorem 4. *Let P^* be an optimal solution of the mm-EPP and $c(P^*) = c_h$. Then P^* is an $s-t$ maxmin path in $\mathcal{G}_h$ and any $s-t$ maxmin path in $\mathcal{G}_h$ is an optimal solution of the mm-EPP.*

Proof. The path P^* is an $s-t$ feasible path for the mm-EPP with capacity c_h, therefore P^* is an $s-t$ path in $\mathcal{G}_h$.

Let Q be an $s-t$ path in $\mathcal{G}_h$. If $R^h_\sigma(Q) > R^h_\sigma(P^*)$, then

$$R_\sigma(Q) \geqslant R^h_\sigma(Q) > R^h_\sigma(P^*) = R_\sigma(P^*)$$

which contradicts the optimality of P^*. Therefore, P^* is an $s-t$ maxmin path in $\mathcal{G}_h$.

On the other hand, let $\widetilde{P}$ be an $s-t$ maxmin path in $\mathcal{G}_h$. If $c(\widetilde{P}) > c_h$, by applying Lemma 3, there would be no optimal solution of the mm-EPP with capacity c_h, which contradicts the optimality of P^*. Hence, any $s-t$ maxmin path $\widetilde{P}$ in $\mathcal{G}_h$ has $c(\widetilde{P}) = c_h$ and so can transmit the given data units at flow rate $c(\widetilde{P})$. As a consequence, $\widetilde{P}$ is an $s-t$ feasible path for the mm-EPP verifying $R^h_\sigma(\widetilde{P}) = R^h_\sigma(P^*)$ and so is an optimal solution of the mm-EPP. □

Theorem 4 allows us to conclude that any optimal solution to the mm-EPP can be obtained as an $s-t$ maxmin path in $\mathcal{G}_j$, for some $j \in \{1,\dots,r\}$. Based on this property, we propose to find an optimal solution of the mm-EPP by solving all the mm-PP$_j$, $j = 1,\dots,r$. Then, from the set of optimal solutions to these problems, we select the paths P which maximize $R_\sigma(P)$. Next we describe the algorithm in a precise way:

The algorithm mm-EPA

Step 0.

Set $j = 1$, $\mathcal{E} = \emptyset$

Step 1.

Solve the mm-PP$_j$ in $\mathcal{G}_j$.

If there is no $s-t$ maxmin path in $\mathcal{G}_j$, go to Step 2.

Let P_j be an $s-t$ maxmin path in $\mathcal{G}_j$. If $c(P_j) = c_j$ set $\mathcal{E} = \mathcal{E} \cup P_j$.

Step 2.

If $j = r$ go to Step 3. Otherwise, set $j = j+1$ and go to Step 1.

Step 3.

If $\mathcal{E} = \emptyset$, then the mm-EPP is not feasible.

Otherwise, find a path $P \in \mathcal{E}$ such that $R_\sigma(P) = \max_{P_j \in \mathcal{E}} R_\sigma(P_j)$

P solves the mm-EPP.

Remark 5.

It is worth pointing out that in Step 1, at the iteration j, it is only necessary to keep the $s-t$ maxmin path if its capacity equals c_j. Otherwise, it can be skipped. Indeed, let Q be an optimal solution of the mm-PP$_j$. If $c(Q) > c_j$, from Lemma 3 no path with capacity c_j can be an optimal solution of the mm-EPP. On the other hand, Q is an $s-t$ path in the network $\mathcal{G}_h$ where $c_h = c(Q)$. If it is an $s-t$ maxmin path in the network $\mathcal{G}_h$ it will be a candidate to be an optimal solution of the mm-EPP and at this point it will deserve to be included in $\mathcal{E}$.

Moreover, it is not necessary to compute all the maxmin paths in $\mathcal{G}_j$ with capacity c_j. Indeed, if P_1 and P_2 are $s-t$ maxmin paths in $\mathcal{G}_j$ such that $c(P_1) = c(P_2) = c_j$ then $R_\sigma(P_1) = R^j_\sigma(P_1) = R^j_\sigma(P_2) = R_\sigma(P_2)$. Hence, they provide the same objective function value with respect to the mm-EPP.

Remark 6.

The algorithm allows us to compute optimal solutions of the mm-EPP with different capacities, if they exist. Let us assume that P_1 and P_2 solve the mm-EPP and $c(P_1) \neq c(P_2)$. Then, P_1 and P_2 (or alike paths with these capacities) will be obtained when solving the mm-PP$_j$ for $c_j = c(P_1)$ and $c_j = c(P_2)$, respectively.

Theorem 7. *The time complexity of the Algorithm mm-EPA is $O(r(m + n\log(n)))$.*

Proof. The algorithm consists of solving r times the maxmin path problem (11). This is a maximum capacity path problem [25] considering $b^j_\sigma(u, v)$ as the capacity of the arc (u, v). Therefore, it can be solved with a straightforward set of modifications to most shortest-path algorithms. Thus, we apply r times an algorithm running in $O(m + n\log(n))$ time [10] and the conclusion follows. □

4. The minsum-maxmin bi-objective energy-constrained quickest path problem

This section addresses simultaneously the goals of finding a quickest path and maximizing the minimum residual energy. For this purpose we now assume that every arc (u, v) in $\mathcal{G}$ has the three parameters associated to it: delay time $l(u, v)$, capacity $c(u, v)$ and energy rate $\omega(u, v)$. Also, we assume that each node u in $\mathcal{G}$ has a limited power b_u. Using the notations introduced in Sections 1 and 3, the minsum-maxmin bi-objective energy-constrained quickest path problem which aims to find an $s - t$ feasible path to transmit the given data units while minimizing the transmission time and maximizing the minimum residual energy can be formulated as:

$$\begin{aligned} \text{msmm-EQPP}: \quad & \min_{P} \quad T_\sigma(P) \\ & \max_{P} \quad R_\sigma(P) \\ & \text{s.t.} \\ & \qquad b_u(\sigma, P) \geqslant 0, u \in \mathcal{N} \\ & \qquad P \text{ is an } s-t \text{ path in } \mathcal{G} \end{aligned} \tag{12}$$

For a nontrivial multi-objective optimization problem, there is no single solution that simultaneously optimizes each objective. Multi-objective optimization problems have been studied from different points of view [8]. Here we focus on the construction of the Pareto front, which is formed by the images in the objective space of the efficient solutions. A feasible solution is said to be efficient if no other feasible solution is at least as good for all the objectives and strictly better for at least one objective. The images in the objective function space of the efficient paths are the non-dominated points. The set of all non-dominated points is the Pareto front. Formally, with respect to the msmm-EQPP, an $s - t$ path P which is a feasible solution of (12) is efficient if and only if there is no other feasible solution $s - t$ path Q so that

$$T_\sigma(Q) \leqslant T_\sigma(P) \quad \text{and} \quad R_\sigma(Q) \geqslant R_\sigma(P)$$

with at least one strict inequality. Otherwise, P is dominated by Q. If P is an efficient solution, it will be called an $s-t$ efficient path and $(T_\sigma(P), R_\sigma(P))$ will be a non-dominated point.

Two efficient solutions P and Q are called equivalent when corresponding to a unique non-dominated point. A complete set of efficient solutions is a set of efficient solutions $\mathcal{P}_e$ such that every feasible solution not in $\mathcal{P}_e$ is either dominated or equivalent to at least one feasible solution in $\mathcal{P}_e$. The set of all efficient solutions is called the maximal complete set. A set that contains a single solution from any set of equivalent solutions (corresponding to a unique non-dominated point) is called a minimal complete set.

In order to solve the msmm-EQPP, we consider again the networks $\mathcal{G}_j$ defined in (8) and associated with each arc $(u, v) \in \mathcal{A}_j$ the delay time $l(u, v)$. We consider the delay time of path P, $l(P)$, and the minimum residual energy, $R^j_\sigma(P)$, and define the following minsum-maxmin bi-objective path problem in $\mathcal{G}_j$:

$$\begin{aligned} \text{msmm-PP}_j : \quad & \min_P \quad l(P) \\ & \max_P \quad R^j_\sigma(P) \\ & \text{s.t.} \\ & \qquad\quad P \text{ is an } s-t \text{ path in } \mathcal{G}_j \end{aligned} \tag{13}$$

The main conclusion in this section is that the maximal complete set of efficient solutions of the msmm-EQPP can be obtained by solving problem (13) for $j = 1, \ldots, r$.

Theorem 8. *Let $\widetilde{P}$ be an $s-t$ efficient path for the msmm-EQPP with capacity $c(\widetilde{P}) = c_h$. Then, $\widetilde{P}$ is an $s-t$ efficient path for the msmm-PP$_h$.*

Proof. By construction of the network, the path $\widetilde{P}$ is an $s-t$ path in $\mathcal{G}_h$. Let us consider the msmm-PP$_h$ and assume that there is an $s-t$ path Q in $\mathcal{G}_h$ which dominates $\widetilde{P}$ with respect to the msmm-PP$_h$. Then,

$$l(Q) \leqslant l(\widetilde{P}) \qquad \text{and} \qquad R^h_\sigma(Q) \geqslant R^h_\sigma(\widetilde{P})$$

with at least one strict inequality. Let us assume for the time being that $l(Q) < l(\widetilde{P})$. Then, taking into account that $c(Q) \geqslant c_h = c(\widetilde{P})$

$$T_\sigma(Q) = l(Q) + \frac{\sigma}{c(Q)} \leqslant l(Q) + \frac{\sigma}{c(\widetilde{P})} < l(\widetilde{P}) + \frac{\sigma}{c(\widetilde{P})} = T_\sigma(\widetilde{P})$$

and

$$R_\sigma(Q) \geqslant R_\sigma^h(Q) \geqslant R_\sigma^h(\widetilde{P}) = R_\sigma(\widetilde{P})$$

which contradicts that $\widetilde{P}$ is an $s-t$ efficient path for the msmm-EQPP. The case $R_\sigma^h(Q) > R_\sigma^h(\widetilde{P})$ is analogous. □

Corollary 9. *Let $\widetilde{P}$ be an $s-t$ efficient path for the msmm-PP$_j$ so that $c(\widetilde{P}) = c_j$. Then, considering the msmm-EQPP, there is no $s-t$ feasible path in $\mathcal{G}$ with capacity c_j that dominates $\widetilde{P}$.*

Proof. Let Q be an $s-t$ feasible path in $\mathcal{G}$ with capacity $c(Q) = c_j$. Then Q is an $s-t$ path in $\mathcal{G}_j$. If Q dominates $\widetilde{P}$ with respect to the msmm-EQPP:

$$T_\sigma(Q) \leqslant T_\sigma(\widetilde{P}) \quad \text{and} \quad R_\sigma(Q) \geqslant R_\sigma(\widetilde{P})$$

with at least one strict inequality. If $T_\sigma(Q) < T_\sigma(\widetilde{P})$, then

$$l(Q) + \frac{\sigma}{c_j} = T_\sigma(Q) < T_\sigma(\widetilde{P}) = l(\widetilde{P}) + \frac{\sigma}{c_j} \quad \Longrightarrow \quad l(Q) < l(\widetilde{P})$$

and

$$R_\sigma^j(Q) = R_\sigma(Q) \geqslant R_\sigma(\widetilde{P}) = R_\sigma^j(\widetilde{P})$$

Therefore $\widetilde{P}$ would not be an $s-t$ efficient path for the msmm-PP$_j$, which contradicts the hypothesis of the corollary. The proof in the case $R_\sigma(Q) > R_\sigma(\widetilde{P})$ is analogous. □

As a consequence of both results, solving the msmm-EQPP amounts to solving the msmm-PP$_j$ (13) for all networks $\mathcal{G}_j$, $j = 1, \ldots, r$. The minsum-maxmin bi-objective path problem has been addressed in the literature when each arc has associated to it a nonnegative parameter which refers to length and a positive parameter which refers to capacity. For this problem, several polynomial algorithms have been developed with time complexity $O(m^2 + mn\log(n))$ [12, 19, 24]. We use the ideas of the algorithm by Pelegrín and Fernández [24] to develop a polynomial time algorithm which provides a minimal complete set of efficient solutions for the msmm-EQPP.

The algorithm is based on considering in each iteration the network $\mathcal{G}_j$ with the parameters delay time $l(u,v)$ and residual energy $b^j_\sigma(u,v)$ associated to the arcs. Then, the candidates to be efficient solutions of the msmm-EQPP are obtained by solving shortest path problems with respect to $l(u,v)$ in subnetworks of $\mathcal{G}_j$ with progressively fewer arcs. These subnetworks are constructed by removing arcs taking into account the minimum residual energy of the last path computed. Therefore, by embedding subnetworks in networks, eventually only shortest path problems are solved. The description of the algorithm is as follows:

The algorithm msmm-EQPA

Step 0.

Set $j = 1$, $\mathcal{E} = \emptyset$

Step 1.

Find P, an $s - t$ shortest path with respect to $l(u,v)$ in $\mathcal{G}_j$.

If there is no $s - t$ path in $\mathcal{G}_j$, go to Step 2.

If $c(P) = c_j$, compute $(T_\sigma(P), R_\sigma(P))$ and set $\mathcal{E} = \text{Merge}(\mathcal{E}, \{P\})$.

Update $\mathcal{G}_j$ by removing from $\mathcal{A}_j$ all arcs (u,v) such that $b^j_\sigma(u,v) \leqslant R^j_\sigma(P)$.

Go to Step 1.

Step 2.

If $j < r$, set $j = j + 1$ and go to Step 1.

Step 3.

If $\mathcal{E} = \emptyset$, the msmm-EQPP is not feasible.

Otherwise, $\mathcal{E}$ solves the msmm-EQPP.

The operation Merge is defined as follows:

$$\text{Merge}(\mathcal{E}, \{P\}) = \{Q \in \mathcal{E} \cup \{P\} : \text{There is no } \widetilde{Q} \in \mathcal{E} \cup \{P\} \text{ such that } \widetilde{Q}$$

$$\text{dominates } Q \text{ with respect to the bi-objective function } (T_\sigma, R_\sigma)\}$$

Set $\mathcal{E}$ contains a minimal complete set of efficient paths for the msmm-EQPP. The Pareto front is formed by the images of the set $\mathcal{E}$.

Remark 10.

At the iteration j, the msmm-PP$_j$ problem is analyzed. In Step 1, after obtaining a shortest path P with respect to $l(u, v)$, the information about its minimum residual energy $R^j_\sigma(P)$ is used to reduce the incumbent network so that feasible paths which are dominated by P are removed. Notice that removing from $\mathcal{A}_j$ all arcs (u, v) such that $b^j_\sigma(u, v) \leqslant R^j_\sigma(P)$, the paths dominated by P with respect to the bi-objective function (l, R^j_σ) are eliminated. Therefore, when the iteration j terminates, we have identified a set of $s - t$ efficient paths with respect to (l, R^j_σ) which are candidates to be $s - t$ efficient paths for the msmm-EQPP.

Remark 11.

At the iteration j it is only necessary to keep an $s - t$ shortest path if its capacity coincides with c_j, otherwise it is of no interest. Indeed, assume for the time being that the algorithm provides Q, a shortest path in Step 1 with respect to the delay time l with $c(Q) > c_j$. Let $R^j_\sigma(Q)$ be its minimum residual energy. Notice that Q is a shortest path in the network $\mathcal{G}_h$ with $c_h = c(Q)$ and so its potential interest will be analyzed in a posterior iteration.

On the other hand, assume that at this point in the algorithm we are skipping the shortest path P with $c(P) = c_j$. Let $R^j_\sigma(P)$ be its minimum residual energy. Since $l(Q) = l(P)$

$$T_\sigma(Q) = l(Q) + \frac{\sigma}{c(Q)} < l(P) + \frac{\sigma}{c(P)} = T_\sigma(P)$$

Let us assume for the time being that $R^j_\sigma(P) \leqslant R^j_\sigma(Q)$. Then

$$R_\sigma(Q) > R^j_\sigma(Q) \geqslant R^j_\sigma(P) = R_\sigma(P)$$

Hence, P is dominated by Q with respect to the objectives (l, R^j_σ) and so it cannot be an $s-t$ efficient path for the msmm-EQPP.

Otherwise, i.e. if $R^j_\sigma(P) > R^j_\sigma(Q)$, P will be a path in the updated network $\mathcal{G}_j$ formed after removing from $\mathcal{A}_j$ all arcs (u, v) such that $b^j_\sigma(u, v) \leqslant R^j_\sigma(Q)$. Thus, its potential interest will then be analyzed.

Theorem 12. *The time complexity of the Algorithm msmm-EQPA is* $O(rm(m+n\log(n)))$.

Proof. The number of different residual energy values $b^j_\sigma(u, v)$ in $\mathcal{G}_j$ is at most m. Hence, the algorithm amounts to solving rm times a shortest path problem each running in $O(m + n\log(n))$ time [10]. Moreover, computing the set of non-dominated points in $\mathcal{E}$ runs in $O(rm\log(m))$ time. Hence the time complexity is $O(rm(m + n\log(n)))$. □

5. The energy-constrained quickest path problem with at least a fixed residual energy at the nodes

In this section, we are interested in computing an $s-t$ path in $\mathcal{G}$ which minimizes the transmission time in the set of paths which guarantee a certain residual energy at the nodes. Let R be this residual energy. The problem of getting a restricted energy-constrained quickest path can be formulated as:

$$\begin{aligned} \text{REQPP}: \quad & \min_P \quad T_\sigma(P) \\ & \text{s.t.} \\ & \qquad b_u(\sigma, P) \geqslant 0, u \in \mathcal{N} \\ & \qquad R_\sigma(P) \geqslant R \\ & \qquad P \text{ is an } s-t \text{ path in } \mathcal{G} \end{aligned} \tag{14}$$

Notice that $R \geqslant R^*_\sigma$ is required, where R^*_σ refers to the optimal value of the mm-EPP. Otherwise, the REQPP is not feasible. Moreover, if $R = R^*_\sigma$, problem (14) provides a quickest path in the set of maxmin paths and so an efficient solution of the msmm-EQPP.

Lemma 13. *Let $P = (s = u_1, u_2, \ldots, u_k = t)$ be an $s - t$ path in the network $\mathcal{G}$. Then, P is an $s - t$ feasible path for the REQPP if and only if $\tilde{b}_u(\sigma, P) \geqslant 0$ for all $u \in \mathcal{N}$, where*

$$\tilde{b}_u(\sigma, P) = \begin{cases} b_u - \omega(u, u_{i+1})\dfrac{\sigma}{c(P)} - R & \text{if } u = u_i, i = 1, \ldots, k-1 \\ b_u & \text{otherwise} \end{cases}$$

Proof. If $\tilde{b}_u(\sigma, P) \geqslant 0$ for all $u \in \mathcal{N}$, then $b_u(\sigma, P) \geqslant 0$ for all $u \in \mathcal{N}$ and $b_{u_i}(\sigma, P) \geqslant R$, $i = 1, \ldots, k-1$. Hence,

$$R_\sigma(P) = \min_{i=1,\ldots,k-1} b_{u_i}(\sigma, P) \geqslant R$$

Analogously, if P is an $s - t$ feasible path for the REQPP,

$$R_\sigma(P) = \min_{i=1,\ldots,k-1} b_{u_i}(\sigma, P) \geqslant R \quad \Longrightarrow \quad b_{u_i}(\sigma, P) \geqslant R, i = 1, \ldots, k-1$$

Therefore, $b_u(\sigma, P) \geqslant 0$ for all $u \in \mathcal{N}$. □

As a consequence, the REQPP can be reformulated as:

$$\begin{aligned} \min_{P} \quad & T_\sigma(P) \\ \text{s.t.} \quad & \\ & \tilde{b}_u(\sigma, P) \geqslant 0, u \in \mathcal{N} \\ & P \text{ is an } s-t \text{ path in } \mathcal{G} \end{aligned}$$

This formulation corresponds to an energy-constrained quickest path problem as introduced in [4], where $\tilde{b}_u(\sigma, P)$ plays the role of the residual energy at the nodes, and so it can be solved in $O(r(m + n\log(n)))$ time. Notice that this is the time complexity of the algorithms developed in the literature for solving the QPP.

6. Computational experience

In order to analyze the performance of the algorithms mm-EPA and msmm-EQPA, we have considered the sets of test problems used in [4] which are based on the benchmark

Table 1: Parameters of test problems Set 1 and Set 2

	n	m	r
Set 1	10,000, 20,000, 30,000, 40,000	$10n, 20n, 30n, 40n, 50n$	10, 100, 1000
Set 2	20,000, 40,000, 60,000, 80,000, 100,000	$10n, 20n, 30n, 40n, 50n$	10, 100, 1000

instances proposed in [28]. The following subsections describe the characteristics of the test problems and the results obtained. The numerical experiments have been performed on a PC Intel® Core™ I7-3820 CPU at 3.6 GHz × 8 having 32 GB of RAM under Ubuntu Linux 14.04 LTS. Although we had a multi-processor computer at hand, only one processor was used in our tests. The code has been written in C++, GCC 4.8.2. Both algorithms consist of solving shortest path problems and so involve Dijkstra's algorithm whose implementation is based on a min-priority queue implemented using a binary heap data structure. It is worth mentioning that the performance of both algorithms depends very much on the performance of the algorithm used for solving the shortest path problem.

6.1. Problem characteristics

We have considered three different sets of test problems. Set 1 uses the network generator NETGEN [15] to provide the skeleton of the network. Set 2 is based on the network generator GRIDGEN, which is able to provide larger networks. This has been obtained from `http://dimacs.rutgers.edu/pub/netflow/generators/network/gridgen/gridgen.c`. Table 1 displays the parameters n, m and r of the networks in Sets 1 and 2. There are 60 problem groups defined by the number of nodes n, the number of arcs m and the number of distinct capacities r in Set 1, and 75 problems in Set 2. For each problem group, 10 instances have been generated. Delay time and capacity coefficients have been generated from uniform distributions in the range $[10, 10{,}000]$. To obtain problems with a fixed number of capacities, first the required number of capacities is generated from the corresponding uniform distribution. Then, each arc is assigned one of the capacities generated with a uniform probability. The energy rate of the arc (u, v) is computed as $\omega(u, v) = 10^{-5}c(u, v)l^2(u, v)$. The power at the nodes has been fixed at 3×10^8, 6×10^8 and 15×10^8.

Table 2: Dimension and destination nodes of the network in Set 3

Road network	n	m	Dest. 1	Dest. 2	Dest. 3	Dest. 4
NY	264,346	733,846	264,346	132,173	857	20
BAY	321,270	800,172	321,270	160,635	567	18
COL	435,666	1,057,066	435,666	217,833	660	19
FLA	1,070,376	2,712,798	1,070,376	535,188	1035	21
NE	1,524,453	3,897,636	1,542,453	762,227	1235	21
CAL	1,890,815	4,657,742	1,890,815	945,408	1375	21
LKS	2,758,119	6,885,658	2,758,119	1,379,060	1661	23

Set 3 [28] is based on seven USA road networks which have been obtained from `http://www.dis.uniroma1.it/challenge9/download.shtml`. Table 2 shows the characteristics of the networks: name of the network, number of nodes and arcs, and the destination node t. In all cases, the node origin is $s = 1$. The energy rate of the arcs and the power of the nodes is the same as in Sets 1 and 2. Based on these networks, two different groups of test problems have been generated. In the first group, the delay is taken as the parameter distance of the road network [28]. The capacity is computed from the parameter time of the road network and problems with 100 distinct capacities are constructed. For this purpose, the range of the arc times is partitioned in 100 intervals of equal length. In order to have integer capacities, the intervals are rounded off by applying the ceiling function to the upper endpoint and properly adjusting the intervals. For instance, if $(a_1, a_2]$, $(a_2, a_3]$ are the first two intervals of the partition, the resulting intervals would be $(a_1, \lceil a_2 \rceil]$, $(\lceil a_2 \rceil, \lceil a_3 \rceil]$. Then, if an arc time is in the interval $(a, b]$, the arc capacity is b. The second group of instances takes the arc delay and capacity from the empirical distributions proposed in [7], which are displayed in Tables 3 and 4. For this group, 10 instances have been generated for each problem.

For evaluating the effect of the number of items which are sent, in all the test instances the data units to be transmitted is taken to be $\sigma_1 = 100$, $\sigma_2 = 10{,}000$ and $\sigma_3 = 1{,}000{,}000$.

Table 3: Arcs delay empirical distribution

$l(u,v)$	11	16	25	42	73	128	227	410	744	1365	2520	4681	8700
%	2.3	5.4	8.5	10.0	12.0	11.0	10.0	11.0	8.5	7.0	5.0	5.0	4.3

Table 4: Arcs capacity empirical distribution

$c(u,v)$	1360	64	128	256	800	1680	2640	4000	8000
%	51.30	7.15	5.30	0.88	4.40	19.47	4.40	2.70	4.40

6.2. The mm-EPA performance evaluation

In this section the mm-EPA is evaluated. Tables 5 and 6 refer to Sets 1 and 2, respectively. Their format is similar. The first to third columns show the value of the parameters r, n and m. Then, there are three blocks of six columns, one for each value of the power. The first three columns of each block display the mean in the 10 runs of the number of the $s-t$ shortest paths computed by the algorithm which are candidates to be an optimal solution of the mm-EPP, depending on the size of σ. The following three columns show the mean CPU time in seconds of the 10 runs depending on the value of σ. Table 7 summarizes the results.

The main characteristic to be emphasized is that, in general, the computing times are small. As expected, CPU time increases with the number of capacities and the size of the network, but looking at Table 7 we see that, on average, this time is almost negligible when the number of different capacities is small and it is less than two minutes when the number of different capacities is 1000. The networks in Set 2 are larger and so the CPU times are longer. We can also appreciate that CPU times are very similar when the power is 3×10^8 and 6×10^8, but there is a perceptible decrease when the power equals 15×10^8. Figures 1 and 2 display the boxplot of the CPU time for each number of capacities, each value of σ and each value of power, depending on the type of network generator. Every boxplot summarizes the information of 200 problems when using Set 1 and 250 problems when using Set 2. Note that in both groups the variability increases when the number of capacities increases. But, this variability is smaller when the power is 15×10^8. Another aspect to be emphasized is that the algorithm solves the shortest path problem in as many networks as the number of different capacities r. Thus, we could expect the number of candidate $s - t$ shortest paths to be close to that number. Nevertheless, from Table 7 we see that the number of candidate shortest paths is substantially lower than r, especially when the number of distinct capacities increases. In addition, the number of candidate $s - t$ shortest paths is very similar when the power values are 3×10^8 and 6×10^8, but increases when the power equals 15×10^8.

Tables 8 and 9 show the results of the first group and the second group of Set 3, respectively. Both have the same format. The first column displays the name of the network and the second column shows the destination node. The other columns display, for the first group, the number of candidate shortest paths and the CPU time depending on the size σ and the power. For the second group, the columns which contain the number of candidate shortest paths and CPU time provide an average of the 10 instances. A significant characteristic of the USA road networks considered is that they are sparse. In fact, the average node degree is 2.6. Hence, a poor number of candidate $s - t$ shortest paths could be expected, and this is indeed the case. Hence, although the networks are large, the CPU time involved in solving the problems is not very long and is very similar regardless of the power value.

Table 5: mm-EPA test results: Set 1. Mean of the number of candidate $s-t$ shortest paths P_j and mean of the computing time (CPU time in seconds)

			Power= 3×10^8						Power= 6×10^8						Power= 15×10^8					
			# Shortest paths			CPU time			# Shortest paths			CPU time			# Shortest paths			CPU time		
r	n	m	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3
10	10,000	100,000	8.6	8.2	4.7	0.05	0.04	0.04	8.6	8.3	7.5	0.04	0.04	0.04	8.1	8.1	8.5	0.03	0.03	0.03
		200,000	9.4	9.3	8.4	0.09	0.09	0.08	9.4	9.3	8.7	0.09	0.09	0.09	8.5	8.5	9.1	0.06	0.06	0.06
		300,000	9.5	9.3	8.8	0.13	0.12	0.13	9.5	9.3	9.1	0.13	0.12	0.13	8.9	8.9	9.0	0.09	0.09	0.10
		400,000	8.7	8.2	7.7	0.15	0.15	0.15	8.7	8.3	8.2	0.15	0.15	0.16	7.6	7.8	8.5	0.12	0.12	0.13
		500,000	9.9	9.5	9.8	0.20	0.19	0.20	9.9	9.5	9.8	0.20	0.19	0.21	9.0	9.0	9.1	0.14	0.14	0.15
	20,000	200,000	8.3	8.0	5.1	0.10	0.11	0.09	8.3	8.0	6.6	0.10	0.11	0.10	8.4	8.4	8.5	0.08	0.08	0.09
		400,000	9.3	9.0	7.9	0.21	0.21	0.19	9.3	8.9	8.5	0.21	0.21	0.19	8.9	8.9	9.1	0.15	0.15	0.16
		600,000	9.0	8.7	7.6	0.26	0.27	0.25	9.0	8.7	8.2	0.26	0.26	0.27	8.9	8.9	8.6	0.19	0.19	0.21
		800,000	9.9	9.8	9.3	0.39	0.35	0.39	9.9	9.8	9.6	0.39	0.36	0.40	9.5	9.5	9.4	0.25	0.25	0.27
		1,000,000	9.8	9.6	9.5	0.45	0.41	0.44	9.8	9.6	9.6	0.45	0.41	0.43	9.2	9.2	9.4	0.29	0.30	0.32
	30,000	300,000	8.1	8.2	6.1	0.18	0.18	0.16	8.1	8.2	7.1	0.18	0.18	0.18	8.0	8.0	8.2	0.13	0.13	0.14
		600,000	9.6	9.4	8.2	0.36	0.31	0.33	9.6	9.4	9.1	0.36	0.31	0.36	9.6	9.6	9.5	0.23	0.23	0.25
		900,000	9.8	9.4	9.2	0.45	0.45	0.42	9.8	9.4	9.5	0.45	0.45	0.46	9.4	9.4	9.7	0.31	0.31	0.33
		1,200,000	9.7	9.2	9.6	0.63	0.55	0.54	9.7	9.2	9.5	0.63	0.55	0.55	9.4	9.4	9.7	0.38	0.38	0.41
		1,500,000	9.7	9.6	9.5	0.78	0.69	0.67	9.7	9.6	9.7	0.78	0.69	0.71	9.8	9.8	9.7	0.45	0.45	0.49
	40,000	400,000	8.4	8.1	5.3	0.24	0.26	0.24	8.4	8.0	7.1	0.24	0.26	0.26	8.6	8.6	8.5	0.20	0.20	0.21
		800,000	9.3	8.9	7.7	0.47	0.46	0.45	9.3	9.0	8.4	0.47	0.47	0.49	9.3	9.3	9.3	0.32	0.32	0.34
		1,200,000	9.7	9.7	8.4	0.64	0.59	0.58	9.7	9.7	9.2	0.64	0.58	0.61	9.5	9.5	9.5	0.40	0.40	0.44
		1,600,000	10.0	9.7	9.4	0.83	0.80	0.78	10.0	9.7	9.9	0.83	0.80	0.80	9.8	9.8	9.9	0.52	0.52	0.55
		2,000,000	9.9	9.9	9.9	0.97	0.87	0.97	9.9	9.9	9.9	0.97	0.87	0.96	9.8	9.8	9.8	0.62	0.62	0.66
100	10,000	100,000	53.2	40.7	25.6	0.32	0.26	0.18	53.2	40.4	35.8	0.26	0.21	0.19	57.4	57.4	63.0	0.12	0.12	0.14
		200,000	59.1	47.5	55.4	0.41	0.38	0.35	59.1	47.5	54.0	0.41	0.38	0.37	48.8	48.9	58.0	0.15	0.15	0.17
		300,000	63.8	53.6	60.9	0.57	0.45	0.49	63.8	53.4	59.8	0.57	0.44	0.51	52.3	52.2	58.8	0.19	0.19	0.21
		400,000	52.3	48.2	50.4	0.60	0.82	0.53	52.3	48.3	52.0	0.60	0.82	0.52	41.8	41.7	47.5	0.22	0.22	0.25
		500,000	65.9	60.3	61.9	0.72	0.73	0.67	65.9	60.1	60.5	0.72	0.73	0.69	46.6	46.6	52.0	0.26	0.26	0.30
	20,000	200,000	59.4	47.0	32.8	0.61	0.59	0.40	59.4	46.7	49.0	0.60	0.59	0.45	65.3	65.2	70.1	0.32	0.32	0.36
		400,000	62.7	53.5	47.7	1.00	1.07	0.78	62.7	53.6	57.9	1.01	1.08	0.87	65.6	65.5	71.9	0.41	0.41	0.47
		600,000	58.9	55.3	50.4	1.27	1.34	1.22	58.9	55.6	60.3	1.27	1.33	1.33	60.1	60.1	66.3	0.44	0.44	0.52
		800,000	75.7	62.8	69.6	1.94	1.86	1.76	75.7	62.5	68.9	1.94	1.86	2.05	63.1	63.1	73.5	0.63	0.63	0.74
		1,000,000	74.9	64.2	71.2	2.27	1.91	1.95	74.9	64.5	69.5	2.29	1.92	2.16	61.1	61.4	64.6	0.62	0.62	0.73
	30,000	300,000	62.9	53.7	41.9	1.08	0.97	0.85	62.9	53.8	56.9	1.08	0.97	0.91	68.6	68.6	75.1	0.58	0.58	0.67
		600,000	73.7	61.2	62.1	1.97	1.71	1.36	73.7	61.2	63.8	1.96	1.71	1.55	71.0	71.0	76.4	0.72	0.72	0.82
		900,000	73.5	60.1	64.2	2.47	2.03	2.05	73.5	60.2	68.8	2.48	2.03	2.27	64.7	64.7	73.4	0.86	0.86	1.02
		1,200,000	76.7	65.4	69.7	3.22	3.07	2.59	76.7	65.1	72.3	3.23	3.08	2.58	66.5	66.7	74.8	0.97	0.98	1.17
		1,500,000	75.7	63.9	70.6	4.18	3.12	3.08	75.7	64.0	74.6	4.20	3.11	3.29	68.0	68.1	74.2	1.16	1.17	1.40
	40,000	400,000	58.9	50.3	31.3	1.43	1.59	0.92	58.9	50.7	43.8	1.44	1.60	1.10	69.7	69.8	69.5	0.94	0.94	0.99
		800,000	67.6	63.2	58.9	2.50	2.54	1.77	67.6	63.2	65.8	2.51	2.55	2.20	71.5	71.5	78.4	1.14	1.15	1.32
		1,200,000	69.0	58.3	61.7	3.52	2.76	3.00	69.0	58.4	64.6	3.53	2.75	3.14	60.5	60.5	68.3	1.20	1.21	1.39
		1,600,000	83.0	71.5	76.6	4.62	4.44	4.09	83.0	71.4	79.8	4.64	4.46	4.42	69.8	70.0	79.2	1.49	1.51	1.73
		2,000,000	82.7	75.7	75.4	4.89	5.06	4.18	82.7	75.4	77.9	4.90	5.06	4.32	65.6	65.8	74.3	1.58	1.58	1.84
1000	10,000	100,000	102.7	75.8	47.2	2.40	2.12	1.48	102.7	75.4	83.4	2.32	2.08	1.71	161.5	161.5	202.8	0.99	0.99	1.10
		200,000	132.6	98.9	103.1	3.61	3.01	2.51	132.6	98.8	122.7	3.61	3.01	2.76	160.5	160.1	183.6	1.04	1.04	1.11
		300,000	137.8	113.3	117.6	5.47	4.87	4.28	137.8	113.3	142.7	5.46	4.88	4.86	156.4	156.6	179.3	1.15	1.14	1.23
		400,000	136.0	111.1	111.0	4.94	5.39	3.97	136.0	110.6	124.8	4.92	5.38	4.19	143.6	143.6	162.7	1.30	1.31	1.40
		500,000	164.8	127.4	140.0	7.15	6.57	5.27	164.8	127.8	151.1	7.14	6.57	5.23	133.6	133.7	151.4	1.31	1.31	1.45
	20,000	200,000	136.9	104.3	84.9	4.92	4.58	2.93	136.9	104.2	106.2	4.95	4.60	3.95	244.2	244.5	287.2	2.65	2.65	2.91
		400,000	167.6	128.7	130.3	8.22	10.86	7.71	167.6	128.7	148.0	8.27	10.91	8.89	263.3	263.4	308.1	3.52	3.52	3.87
		600,000	151.0	111.6	124.3	10.93	7.24	8.62	151.0	111.5	145.7	10.97	7.24	8.72	187.6	188.0	215.3	2.83	2.84	3.08
		800,000	204.0	175.2	181.7	19.72	16.53	13.53	204.0	175.1	203.4	19.69	16.49	14.79	225.5	225.7	265.4	3.82	3.82	4.25
		1,000,000	221.8	183.8	178.9	21.36	16.45	15.22	221.8	183.4	199.4	21.44	16.49	16.76	190.3	190.5	220.0	3.32	3.34	3.80
	30,000	300,000	192.1	138.4	98.5	10.85	8.99	7.30	192.1	137.5	155.5	10.87	8.95	8.35	316.4	316.8	372.2	4.74	4.78	5.20
		600,000	234.6	153.2	157.1	19.20	16.61	14.13	234.6	152.9	203.4	19.18	16.60	13.92	297.0	297.4	355.6	5.82	5.83	6.57
		900,000	263.2	182.5	198.1	20.45	23.66	18.16	263.2	182.4	216.4	20.41	23.62	19.54	294.4	294.0	345.6	6.31	6.31	7.05
		1,200,000	255.5	209.0	198.5	29.76	25.40	20.03	255.5	208.3	222.7	29.85	25.52	21.88	251.2	251.1	288.4	5.88	5.89	6.55
		1,500,000	228.0	186.5	168.2	31.10	26.03	25.44	228.0	186.2	228.6	31.19	26.10	27.08	252.8	252.9	290.7	6.54	6.56	7.55
	40,000	400,000	185.8	135.2	87.1	11.83	15.41	7.96	185.8	134.8	128.7	11.88	15.46	10.16	352.4	353.0	421.8	7.70	7.68	8.27
		800,000	232.7	177.0	195.7	19.72	22.99	17.07	232.7	176.5	190.4	19.76	23.05	17.61	346.1	346.3	409.7	8.75	8.80	9.71
		1,200,000	256.2	202.0	202.5	34.02	31.04	27.57	256.2	201.4	230.4	34.01	31.11	29.38	315.5	315.2	352.7	9.09	9.09	9.92
		1,600,000	296.2	235.3	248.2	32.78	35.67	32.17	296.2	235.3	276.6	32.88	35.72	36.90	303.4	303.2	347.3	9.44	9.47	10.47
		2,000,000	315.8	255.3	260.1	40.54	39.64	37.56	315.8	255.3	276.5	40.52	39.48	41.23	284.8	284.4	324.5	9.88	9.93	10.74

Table 6: mm-EPA test results: Set 2. Mean of the number of candidate $s-t$ shortest paths P_j and mean of the computing time (CPU time in seconds)

			Power= 3×10^8						Power= 6×10^8						Power= 15×10^8					
			# Shortest paths			CPU time			# Shortest paths			CPU time			# Shortest paths			CPU time		
r	n	m	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3
10	20,000	200,000	7.3	8.3	6.0	0.14	0.14	0.12	7.3	8.0	8.3	0.14	0.14	0.13	6.5	6.1	8.6	0.08	0.07	0.09
		400,000	8.0	9.3	8.8	0.28	0.30	0.28	8.0	9.3	9.6	0.28	0.30	0.33	8.1	6.9	9.7	0.15	0.16	0.17
		600,000	9.4	8.9	8.8	0.50	0.45	0.45	9.4	8.9	9.2	0.50	0.46	0.45	8.0	8.5	9.7	0.22	0.21	0.24
		800,000	8.5	9.3	9.3	0.56	0.67	0.57	8.5	9.2	9.7	0.56	0.67	0.57	6.3	7.7	9.6	0.26	0.27	0.28
		1,000,000	8.5	9.4	9.8	0.69	0.62	0.77	8.5	9.4	9.9	0.68	0.62	0.77	7.3	7.5	9.8	0.29	0.31	0.34
	40,000	400,000	6.9	8.6	6.0	0.29	0.32	0.29	6.9	8.6	7.3	0.29	0.32	0.32	4.8	6.7	8.5	0.17	0.19	0.23
		800,000	7.8	9.1	8.4	0.60	0.69	0.62	7.8	9.1	9.5	0.60	0.69	0.68	7.7	6.1	9.7	0.33	0.34	0.37
		1,200,000	8.6	9.4	9.2	0.87	0.93	0.90	8.6	9.3	9.9	0.86	0.92	0.93	8.0	8.6	10.0	0.42	0.42	0.48
		1,600,000	6.6	9.7	9.6	1.08	1.24	1.20	6.6	9.7	9.7	1.07	1.25	1.28	7.2	7.3	9.6	0.53	0.53	0.61
		2,000,000	10.0	9.8	9.5	1.53	1.63	1.59	10.0	9.9	9.9	1.53	1.61	1.61	9.5	7.4	9.7	0.65	0.60	0.72
	60,000	600,000	7.5	8.8	7.1	0.46	0.56	0.45	7.5	8.8	8.2	0.46	0.57	0.50	6.5	6.9	8.6	0.26	0.26	0.34
		1,200,000	7.4	8.7	8.0	0.90	0.99	0.95	7.4	8.7	8.9	0.90	1.00	1.03	6.9	7.5	9.6	0.46	0.47	0.59
		1,800,000	9.2	9.3	9.4	1.30	1.68	1.37	9.2	9.3	9.6	1.30	1.68	1.41	8.4	8.0	9.7	0.64	0.73	0.77
		2,400,000	8.0	9.7	9.9	1.72	2.01	2.08	8.0	9.7	10.0	1.71	2.00	1.92	7.0	7.3	9.9	0.77	0.77	0.85
		3,000,000	9.4	9.8	9.9	2.14	2.32	2.28	9.4	9.8	9.9	2.14	2.29	2.42	8.4	7.4	9.7	0.96	0.93	1.02
	80,000	800,000	7.7	8.4	6.3	0.83	0.75	0.60	7.7	8.4	7.6	0.83	0.75	0.72	7.0	6.3	8.6	0.39	0.38	0.48
		1,600,000	9.0	8.9	8.3	1.44	1.21	1.16	9.0	8.9	9.0	1.44	1.20	1.37	7.9	6.9	9.5	0.66	0.62	0.80
		2,400,000	9.0	9.9	9.1	1.90	2.04	1.85	9.0	9.9	9.6	1.88	2.03	1.80	7.6	8.4	9.7	0.91	0.83	0.90
		3,200,000	7.9	9.1	9.7	2.32	2.16	2.51	7.9	9.0	9.9	2.32	2.11	2.42	7.3	8.4	10.0	1.05	1.05	1.19
		4,000,000	8.1	9.6	9.8	2.43	3.69	2.80	8.1	9.6	9.9	2.41	3.69	3.01	6.9	8.2	9.9	1.23	1.31	1.50
	100,000	1,000,000	8.2	7.7	5.5	0.94	0.94	0.74	8.2	7.7	6.8	0.94	0.94	0.81	6.6	5.7	8.9	0.52	0.52	0.63
		2,000,000	9.3	9.0	8.7	1.88	1.62	1.59	9.3	8.9	9.1	1.90	1.59	1.82	8.4	7.5	9.4	0.88	0.75	0.97
		3,000,000	9.1	9.6	8.9	2.95	2.21	2.33	9.1	9.6	9.3	2.97	2.19	2.61	7.6	7.9	9.7	1.14	1.06	1.22
		4,000,000	9.1	9.6	9.9	3.31	3.62	3.50	9.1	9.6	9.9	3.31	3.64	3.46	7.4	6.4	9.9	1.36	1.33	1.58
		5,000,000	8.5	9.6	9.8	4.20	4.64	4.39	8.5	9.7	9.9	4.22	4.59	4.44	8.3	7.6	10.0	1.51	1.54	1.74
100	20,000	200,000	46.7	53.2	44.7	0.73	0.99	0.58	46.7	52.9	54.1	0.73	1.00	0.66	43.1	56.4	70.9	0.31	0.37	0.47
		400,000	58.9	57.3	58.6	1.64	2.06	1.80	58.9	57.3	63.7	1.64	2.06	2.16	54.3	52.2	74.9	0.60	0.57	0.74
		600,000	53.9	62.1	68.5	2.35	3.05	2.65	53.9	62.4	73.0	2.35	3.05	3.11	46.7	55.3	74.3	0.69	0.74	0.97
		800,000	76.5	59.7	70.2	4.30	3.16	3.49	76.5	59.7	72.5	4.29	3.16	3.95	57.9	48.1	68.7	0.95	0.94	1.05
		1,000,000	64.6	65.7	73.4	4.23	4.91	5.36	64.6	66.1	78.0	4.18	4.91	5.15	44.7	45.5	71.1	1.00	0.88	1.27
	40,000	400,000	54.8	60.8	43.1	2.27	2.08	2.06	54.8	60.4	57.5	2.26	2.08	2.36	53.8	67.1	80.0	0.88	1.12	1.36
		800,000	71.6	67.9	64.6	5.14	5.01	4.38	71.6	68.2	71.1	5.15	4.97	4.70	60.5	58.9	80.6	1.39	1.29	1.94
		1,200,000	83.2	69.6	76.2	6.94	8.38	6.97	83.2	68.9	79.8	6.98	8.37	7.22	71.6	63.3	83.0	1.89	1.70	2.00
		1,600,000	77.6	72.2	74.9	7.82	7.81	7.66	77.6	72.1	80.4	7.86	7.84	8.05	59.8	59.9	80.8	1.91	1.91	2.34
		2,000,000	64.7	76.6	77.3	7.63	11.06	11.27	64.7	75.8	79.4	7.67	11.01	11.90	53.7	58.8	75.9	2.22	2.28	2.71
	60,000	600,000	49.0	64.6	57.7	2.43	4.11	2.04	49.0	64.2	66.4	2.43	4.12	2.56	46.4	54.5	76.1	1.32	1.47	1.89
		1,200,000	59.5	70.2	74.1	6.54	7.64	5.99	59.5	69.9	75.7	6.52	7.69	7.01	56.2	71.9	84.6	2.06	2.47	2.79
		1,800,000	49.8	77.0	78.7	5.79	12.04	8.92	49.8	77.4	82.1	5.79	12.08	9.85	54.7	74.5	87.4	2.30	3.12	3.21
		2,400,000	80.7	74.0	80.2	13.70	12.68	13.66	80.7	74.1	81.8	13.61	12.80	13.80	60.8	61.4	86.1	3.05	3.41	3.48
		3,000,000	65.7	76.6	84.7	12.64	15.44	15.72	65.6	77.3	87.0	12.67	15.25	15.50	57.4	60.2	82.0	3.11	3.26	4.10
	80,000	800,000	55.5	56.4	49.2	4.23	6.64	4.48	55.5	55.9	64.5	4.26	6.64	4.96	51.0	57.0	77.6	2.04	2.81	2.95
		1,600,000	71.7	72.3	72.4	9.11	10.32	8.22	71.7	72.2	74.5	9.06	10.24	8.91	65.6	66.6	85.5	2.99	2.62	3.52
		2,400,000	69.7	74.4	76.0	11.94	16.49	14.27	69.7	74.4	79.6	11.98	16.81	14.40	65.7	69.3	86.0	3.49	3.23	4.29
		3,200,000	66.0	82.4	84.1	12.57	20.15	17.21	66.0	82.1	83.5	12.69	19.95	18.15	63.0	67.7	87.9	3.93	4.12	4.67
		4,000,000	56.5	81.3	81.9	12.08	23.93	22.00	56.5	81.5	86.6	12.10	23.94	21.97	54.4	56.4	87.3	4.16	4.12	5.44
	100,000	1,000,000	71.3	63.6	40.0	7.55	7.17	4.47	71.3	63.2	57.3	7.52	7.11	5.98	57.2	58.3	78.0	3.21	3.76	4.39
		2,000,000	58.1	73.3	72.1	9.26	13.66	11.43	58.1	73.9	81.3	9.27	13.62	12.07	58.8	68.8	87.3	4.30	4.23	4.63
		3,000,000	78.4	78.7	82.0	19.81	17.81	15.36	78.4	78.7	84.1	19.87	17.82	14.69	63.6	73.3	87.8	4.85	5.38	5.47
		4,000,000	65.8	82.7	85.2	18.44	24.07	21.73	65.8	82.6	85.1	18.41	24.08	22.13	72.5	66.2	86.7	5.83	4.97	6.25
		5,000,000	69.5	80.2	85.0	28.13	20.68	26.76	69.5	80.0	86.7	28.34	20.61	28.27	63.4	55.7	87.2	5.39	5.11	7.24
1000	20,000	200,000	120.1	112.5	95.0	7.35	6.53	6.05	120.0	112.3	126.3	7.36	6.53	7.24	212.3	247.2	312.0	2.69	2.57	3.76
		400,000	186.4	159.8	152.0	17.69	20.35	13.23	186.5	159.3	178.2	17.70	20.44	14.89	259.8	233.6	309.9	5.65	5.07	6.17
		600,000	183.6	173.6	154.5	27.13	28.44	23.69	183.5	174.2	187.0	27.12	28.39	25.79	234.5	233.2	261.5	6.36	5.78	7.48
		800,000	209.0	169.0	186.9	34.23	29.31	36.72	208.9	168.9	204.7	34.12	29.26	33.43	201.7	188.7	258.3	5.85	7.13	8.68
		1,000,000	212.8	190.6	205.3	37.22	41.59	39.34	212.7	190.5	212.2	37.09	40.99	44.88	187.5	212.1	233.9	6.72	7.15	7.65
	40,000	400,000	200.7	147.1	96.9	24.37	23.97	17.00	200.7	147.4	148.0	24.39	23.95	21.62	331.7	307.7	465.2	9.38	8.98	13.35
		800,000	195.6	198.5	198.1	40.43	44.89	37.53	195.6	197.9	210.2	40.16	44.81	42.65	294.0	341.9	423.1	10.60	12.68	18.91
		1,200,000	240.9	235.2	211.9	43.20	70.86	57.54	240.9	233.3	256.8	43.20	71.09	66.45	313.1	329.0	399.5	13.54	15.05	17.63
		1,600,000	303.4	255.6	253.8	85.42	95.07	66.39	303.4	256.1	303.5	85.39	95.36	74.43	295.8	284.6	371.5	16.84	13.61	18.75
		2,000,000	279.3	262.1	275.1	79.06	100.19	101.28	279.3	263.5	284.2	78.26	100.51	106.03	245.8	256.5	339.1	13.46	15.84	19.23
	60,000	600,000	198.4	151.2	122.1	28.99	34.35	23.46	198.4	150.9	188.7	28.99	34.45	31.87	290.5	295.6	561.8	14.83	15.97	19.33
		1,200,000	276.4	243.4	229.6	60.21	68.60	58.27	276.4	243.9	268.1	60.82	68.74	65.22	392.9	365.4	507.2	19.59	19.27	23.81
		1,800,000	290.1	275.7	285.8	72.91	101.05	77.30	290.0	275.0	310.7	71.89	101.37	84.03	330.5	351.3	474.4	20.26	23.24	26.70
		2,400,000	331.4	288.5	286.1	121.13	114.45	115.69	331.4	288.7	339.2	119.84	113.20	121.04	334.8	340.6	422.0	24.53	23.83	31.33
		3,000,000	332.8	287.5	324.1	128.83	180.90	138.31	332.8	287.4	369.4	128.29	179.90	142.08	316.8	309.6	376.5	25.71	27.83	28.59
	80,000	800,000	281.0	190.6	166.7	63.75	55.97	41.25	281.0	189.6	220.1	63.51	54.99	45.52	493.8	386.2	568.3	24.25	20.67	26.70
		1,600,000	325.4	276.9	277.5	86.21	81.52	69.67	325.4	275.8	320.1	86.31	81.31	83.45	381.7	372.9	589.7	23.09	23.91	31.63
		2,400,000	369.9	301.9	302.5	136.08	135.21	118.86	369.9	301.1	343.9	137.18	138.25	112.85	400.9	341.5	477.6	28.75	23.88	33.74
		3,200,000	380.9	341.4	357.3	139.46	197.56	135.77	380.9	341.2	393.2	139.26	197.72	156.41	353.3	357.9	451.7	28.18	31.17	39.33
		4,000,000	292.3	355.3	377.4	140.68	230.85	213.38	292.2	355.8	373.8	144.19	229.62	225.14	344.1	348.9	463.6	34.81	35.34	45.36
	100,000	1,000,000	262.9	214.9	153.2	64.11	64.92	41.30	262.9	215.6	232.3	64.36	65.01	55.61	427.1	429.3	588.2	35.41	28.97	38.71
		2,000,000	326.8	293.1	259.6	105.39	119.10	112.24	326.8	291.6	296.4	104.67	118.75	126.68	456.0	412.4	629.6	37.99	31.82	47.14
		3,000,000	402.8	318.6	332.0	201.47	167.09	158.85	402.8	319.4	377.7	199.55	165.70	172.05	423.5	407.7	565.9	37.62	38.72	48.43
		4,000,000	394.9	358.1	407.8	200.32	257.36	204.34	394.9	359.1	401.8	195.82	257.77	229.24	393.2	394.8	524.6	45.28	44.50	56.56
		5,000,000	437.9	363.2	379.3	302.46	242.13	223.96	437.9	363.1	405.7	300.31	242.00	224.15	352.4	374.2	479.7	38.44	51.40	48.95

Table 7: Summarized mm-EPA test results of Sets 1 and 2

Mean of the number of candidate $s-t$ shortest paths

		Set 1						Set 2					
		Power= 3×10^8		Power= 6×10^8		Power= 15×10^8		Power= 3×10^8		Power= 6×10^8		Power= 15×10^8	
	r	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev
σ_1	10	9.33	1.00	9.33	1.00	9.01	1.08	8.36	2.59	8.36	2.59	7.42	2.01
	100	67.48	14.16	67.48	14.16	61.90	12.29	64.79	27.12	64.78	27.13	57.47	16.86
	1000	200.77	72.10	200.77	72.10	244.03	83.52	281.43	121.23	281.41	121.27	330.71	113.39
σ_2	10	9.09	1.17	9.09	1.16	9.02	1.07	9.18	1.00	9.16	1.06	7.33	1.94
	100	57.82	13.44	57.80	13.44	61.94	12.26	70.11	11.23	70.05	11.40	61.09	16.44
	1000	155.23	60.79	154.97	60.73	244.10	83.55	246.57	82.32	246.46	82.44	324.91	108.75
σ_3	10	8.11	2.26	8.76	1.73	9.15	0.99	8.63	1.72	9.22	1.23	9.52	0.71
	100	56.92	19.30	61.80	15.73	68.47	12.37	70.19	16.39	75.43	12.37	81.11	8.50
	1000	151.65	69.75	177.83	71.53	284.22	96.37	243.62	98.69	278.09	91.68	442.19	123.60

Mean of the CPU time in seconds

		Set 1						Set 2					
		Power= 3×10^8		Power= 6×10^8		Power= 15×10^8		Power= 3×10^8		Power= 6×10^8		Power= 15×10^8	
	r	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev
	r	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev
σ_1	10	0.38	0.28	0.38	0.28	0.25	0.16	1.41	1.18	1.41	1.18	0.63	0.41
	100	1.98	1.61	1.98	1.62	0.70	0.46	8.69	8.24	8.70	8.27	2.55	1.69
	1000	16.95	13.52	16.97	13.54	4.80	3.19	89.92	85.16	89.59	84.52	21.19	14.59
σ_2	10	0.36	0.25	0.36	0.25	0.25	0.16	1.50	1.24	1.49	1.23	0.63	0.40
	100	1.84	1.51	1.83	1.52	0.70	0.46	10.45	8.34	10.45	8.32	2.64	1.67
	1000	16.15	13.72	16.16	13.73	4.82	3.19	100.49	83.63	100.40	83.73	21.38	14.77
σ_3	10	0.36	0.26	0.37	0.26	0.27	0.17	1.42	1.12	1.47	1.13	0.72	0.46
	100	1.61	1.34	1.75	1.40	0.81	0.53	9.54	7.91	9.98	8.07	3.17	1.91
	1000	13.65	12.00	14.90	12.88	5.31	3.48	85.26	72.89	92.51	75.88	26.72	16.51

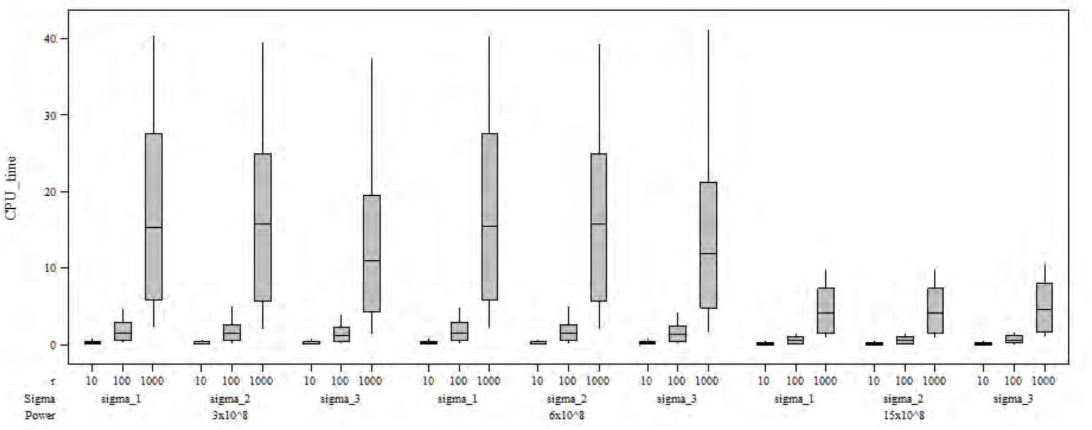

Figure 1: Set 1: Boxplots of the mm-EPA computing time depending on the number of capacities, the value of σ and the power of the nodes

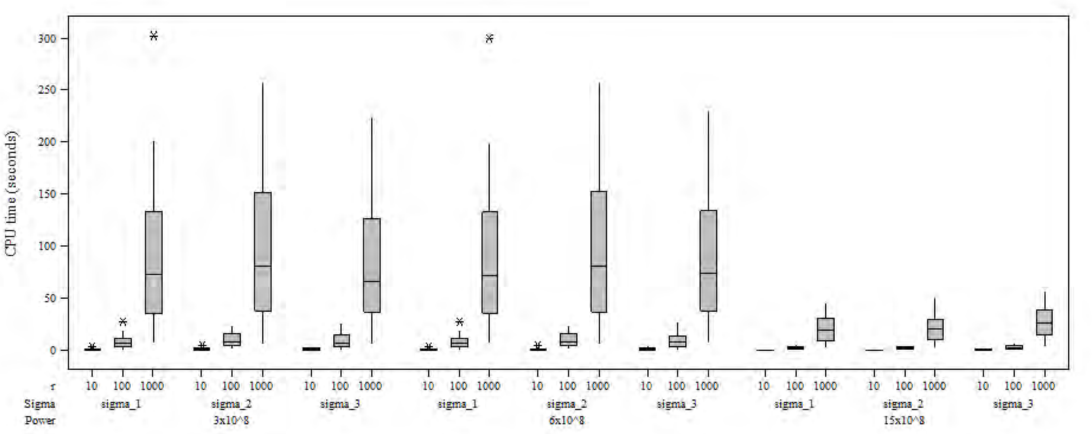

Figure 2: Set 2: Boxplots of the mm-EPA computing time depending on the number of capacities, the value of σ and the power of the nodes

Table 8: mm-EPA test results: Set 3, first group. Number of candidate $s-t$ shortest paths P_j and computing time (CPU time in seconds)

		Power= 3×10^8						Power= 6×10^8						Power= 15×10^8					
		# Shortest paths			CPU time			# Shortest paths			CPU time			# Shortest paths			CPU time		
	Dest.	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3
NY	4	2	2	0	1.47	1.45	1.38	2	2	0	1.47	1.45	1.41	2	2	0	1.38	1.43	1.41
	3	2	2	1	1.38	1.38	1.38	2	2	2	1.38	1.38	1.38	2	2	2	1.38	1.38	1.38
	2	1	1	0	1.45	1.45	1.38	1	1	0	1.44	1.44	1.41	1	1	0	1.47	1.43	1.41
	1	1	1	0	1.43	1.47	1.38	1	1	0	1.43	1.47	1.40	1	1	0	1.45	1.45	1.41
BAY	4	1	1	0	1.50	1.54	1.47	1	1	0	1.51	1.56	1.48	1	1	0	1.50	1.56	1.48
	3	1	1	0	1.54	1.48	1.47	1	1	0	1.55	1.50	1.48	1	1	0	1.54	1.55	1.48
	2	1	1	0	1.49	1.49	1.48	1	1	0	1.49	1.51	1.48	1	1	0	1.52	1.52	1.48
	1	1	1	0	1.52	1.54	1.48	1	1	0	1.52	1.54	1.48	1	1	0	1.55	1.55	1.48
COL	4	1	1	0	2.08	2.07	2.07	1	1	0	2.07	2.06	2.07	1	1	0	2.19	2.18	2.08
	3	1	1	0	2.15	2.07	2.07	1	1	0	2.14	2.44	2.05	1	1	0	2.07	2.08	2.07
	2	1	0	0	2.08	2.07	2.07	1	1	0	2.08	2.10	2.07	1	1	0	2.10	2.17	2.07
	1	1	0	0	2.19	2.07	2.07	1	0	0	2.19	2.14	2.07	1	1	0	2.16	2.17	2.07
FLA	4	1	1	0	3.94	3.94	3.96	1	1	0	3.95	4.22	3.97	1	1	0	4.29	4.26	3.96
	3	1	1	0	4.16	3.95	3.96	1	1	0	4.19	4.21	3.96	1	1	0	4.28	4.28	3.97
	2	1	1	0	4.02	3.99	3.95	1	1	0	4.12	4.03	3.97	1	1	0	4.08	4.18	3.96
	1	1	1	0	4.25	3.97	3.95	1	1	0	4.18	4.11	3.96	1	1	0	4.15	4.21	3.98
NE	4	1	1	0	9.78	9.81	9.77	1	1	0	9.81	9.86	9.86	1	1	1	10.20	9.85	9.85
	3	1	1	0	10.16	9.76	9.75	1	1	0	10.21	9.78	9.77	1	1	0	10.27	10.24	9.85
	2	1	1	0	9.94	10.17	9.76	1	1	0	10.00	10.22	9.82	1	1	0	10.26	10.20	9.89
	1	1	1	0	10.15	10.20	9.78	1	1	0	10.20	10.17	9.85	1	1	0	10.13	10.23	9.88
CAL	4	1	1	0	10.17	10.14	10.23	1	1	0	10.20	10.30	10.23	1	1	0	10.82	10.77	10.26
	3	1	0	0	10.69	10.18	10.17	1	0	0	10.75	10.21	10.22	1	1	0	10.80	10.77	10.23
	2	1	0	0	10.54	10.18	10.20	1	0	0	10.58	10.21	10.22	1	1	0	10.72	10.64	10.22
	1	1	0	0	10.76	10.13	10.19	1	0	0	10.80	10.22	10.23	1	1	0	10.40	10.66	10.23
LKS	4	1	1	0	15.34	15.32	15.37	1	1	0	15.39	15.82	15.37	1	1	0	16.12	16.04	15.41
	3	1	1	0	15.54	15.34	15.36	1	1	0	15.52	15.36	15.37	1	1	0	16.13	16.07	15.40
	2	1	1	0	16.22	16.08	15.37	1	1	0	16.26	15.60	15.46	1	1	0	15.81	15.74	15.40
	1	1	1	0	15.42	15.88	15.37	1	1	0	15.38	15.57	15.40	1	1	0	15.95	15.92	15.43

Table 9: mm-EPA test results: Set 3, second group. Mean of the number of candidate $s-t$ shortest paths P_j and mean of the computing time (CPU time in seconds)

		Power= 3×10^8						Power= 6×10^8						Power= 15×10^8					
		# Shortest paths			CPU time			# Shortest paths			CPU time			# Shortest paths			CPU time		
	Dest.	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3
NY	4	2.2	2.2	0.0	0.55	0.52	0.33	2.2	2.2	0.2	0.55	0.51	0.38	2.2	2.3	1.0	0.47	0.43	0.44
	3	3.2	3.2	2.3	0.32	0.32	0.33	3.2	3.2	3.0	0.32	0.31	0.33	3.5	3.5	3.9	0.32	0.33	0.36
	2	3.2	3.7	0.1	0.46	0.45	0.33	3.2	3.7	0.5	0.46	0.44	0.34	3.2	3.7	2.0	0.47	0.47	0.39
	1	2.2	2.7	0.0	0.52	0.53	0.33	2.2	2.7	0.0	0.52	0.52	0.37	2.2	2.7	0.8	0.47	0.49	0.45
BAY	4	1.7	1.6	0.0	0.45	0.41	0.37	1.7	1.7	0.2	0.45	0.42	0.37	1.8	1.7	0.5	0.43	0.42	0.39
	3	1.9	1.6	0.0	0.43	0.44	0.37	1.9	1.6	0.0	0.43	0.44	0.37	1.9	1.6	0.1	0.46	0.46	0.38
	2	1.8	2.2	0.0	0.43	0.46	0.37	1.8	2.2	0.0	0.43	0.46	0.37	1.8	2.2	0.2	0.44	0.45	0.38
	1	1.7	1.6	0.0	0.47	0.45	0.37	1.7	1.7	0.0	0.47	0.44	0.37	1.7	1.7	0.3	0.47	0.47	0.39
COL	4	1.8	2.4	0.3	0.55	0.52	0.50	1.8	2.4	0.4	0.55	0.52	0.51	1.6	1.9	1.0	0.63	0.64	0.52
	3	3.1	3.2	0.0	0.53	0.53	0.50	3.1	3.2	0.3	0.54	0.53	0.51	2.9	3.6	0.9	0.55	0.53	0.51
	2	2.8	3.0	0.0	0.52	0.53	0.50	2.8	3.0	0.5	0.52	0.52	0.50	2.8	3.0	1.8	0.54	0.55	0.51
	1	1.0	0.9	0.0	0.68	0.69	0.50	1.0	0.9	0.0	0.68	0.69	0.50	1.0	1.0	0.0	0.65	0.65	0.54
FLA	4	1.2	1.1	0.0	1.49	1.43	1.27	1.2	1.1	0.0	1.49	1.46	1.26	1.2	1.1	0.0	1.61	1.55	1.25
	3	1.0	1.2	0.0	1.46	1.55	1.27	1.0	1.2	0.0	1.46	1.56	1.26	1.0	1.2	0.0	1.61	1.59	1.25
	2	1.2	1.0	0.0	1.37	1.36	1.27	1.2	1.1	0.0	1.37	1.41	1.26	1.2	1.1	0.0	1.34	1.34	1.25
	1	1.3	1.1	0.0	1.40	1.34	1.27	1.3	1.1	0.0	1.41	1.33	1.26	1.3	1.1	0.0	1.41	1.37	1.25
NE	4	3.8	4.1	0.3	2.52	2.51	2.09	3.8	4.1	1.0	2.52	2.59	2.14	3.8	4.1	2.3	2.71	2.80	2.44
	3	2.8	3.3	0.2	2.54	2.46	2.09	2.8	3.3	0.7	2.55	2.49	2.22	3.1	3.4	2.1	3.05	2.99	2.50
	2	1.6	1.4	0.0	3.39	3.35	2.09	1.6	1.5	0.0	3.39	3.36	2.22	1.6	1.5	0.1	3.16	3.20	2.59
	1	1.1	0.9	0.0	3.38	3.42	2.09	1.1	1.0	0.0	3.40	3.42	2.29	1.1	1.0	0.0	3.17	3.22	2.70
CAL	4	2.3	2.1	0.0	2.42	2.43	2.53	2.3	2.2	0.2	2.43	2.43	2.51	2.1	2.1	0.6	2.99	2.99	2.48
	3	1.0	1.0	0.0	2.96	2.86	2.53	1.0	1.0	0.0	2.97	2.82	2.51	1.0	1.0	0.0	2.50	2.53	2.49
	2	1.0	1.0	0.0	2.90	2.97	2.53	1.0	1.0	0.0	2.91	2.97	2.50	1.0	1.0	0.0	2.78	2.78	2.49
	1	1.0	1.0	0.0	3.00	2.90	2.53	1.0	1.0	0.0	3.00	2.86	2.51	1.0	1.0	0.0	2.79	2.75	2.49
LKS	4	2.3	2.0	0.9	3.95	3.82	3.86	2.3	1.9	1.1	3.97	3.81	3.91	2.5	2.4	1.6	4.55	4.63	4.16
	3	3.5	2.9	0.0	4.23	3.99	3.86	3.5	2.9	0.2	4.23	3.90	3.87	3.5	2.9	1.4	5.26	5.05	4.21
	2	2.7	2.6	0.0	5.13	5.03	3.86	2.7	2.6	0.0	5.13	5.17	3.92	2.7	2.6	0.8	4.94	4.90	4.37
	1	2.6	2.4	0.0	4.87	4.88	3.86	2.6	2.4	0.2	4.89	4.82	3.87	2.6	2.4	0.8	5.00	5.12	4.19

6.3. The msmm-EQPA performance evaluation

In this section we have used the same instances to evaluate the msmm-EQPA. However, it is worth mentioning that the bi-objective energy-constrained quickest path problem is more difficult to solve. Thus, longer CPU times can be expected. This is confirmed below. In fact, solving the instances of Set 2 (GRIDGEN networks) with 1000 capacities involved, in general, very long CPU times (more than two hours). So we have eliminated these instances from the study.

The tables presented in this part of the computational study are very similar to those presented in Section 6.2. Table 10 refers to Set 1. Besides the value of the parameters r, n and m shown in the first to third columns, the table displays three blocks of nine columns, one for each value of the power. The first three columns of every block display the mean of the number of candidate $s-t$ paths computed by the algorithm, depending on the size of σ. The second three columns of each block show the mean of the cardinality of the minimal complete set of efficient paths of the msmm-EQPP computed by the algorithm in the 10 runs, depending on the size of σ. Finally, the last three columns of the block display the mean CPU time in seconds of the 10 runs for the different values of σ. Table 11 has the same format, but refers to Set 2. Table 12 and Figures 3 and 4 summarize the results. Notice that Figure 4 only displays the results when $r = 10$ and $r = 100$, because the case $r = 1000$ has not been considered. We can see that the number of candidate shortest paths is small, although the algorithm can solve up to $m \times r$ shortest path problems. Also, the cardinality of the minimal complete set is very low, less than six efficient paths on average. However, as we could expect, msmm-EQPA computing times are much longer than mm-EPA computing times. Although for the smallest problems the times are almost negligible, for the larger instances in Set 1, CPU time is almost one hour. Another point to note is that the power value does not seem to affect very much either the number of paths or the CPU times involved.

Tables 13 and 14 show the results of the first group and the second group of Set 3,

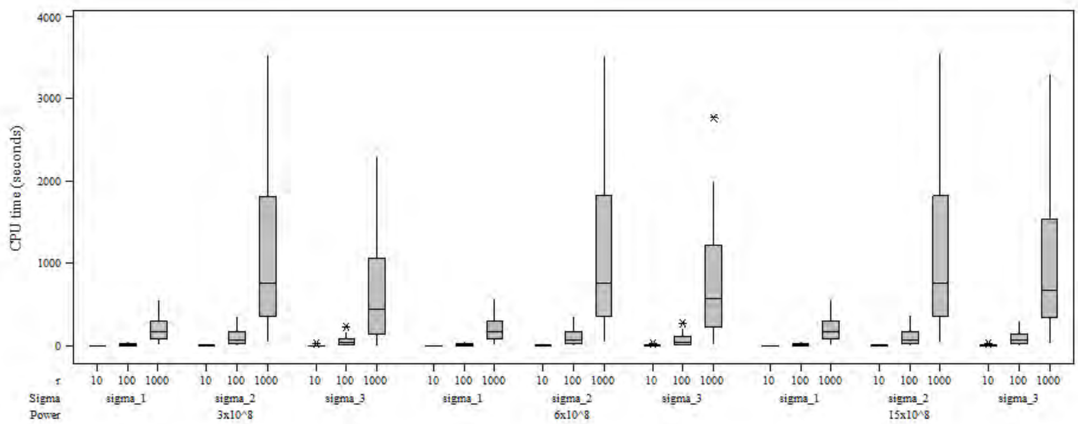

Figure 3: Set 1: Boxplots of the msmm-EQPA computing time depending on the number of capacities, the value of σ and the power of the nodes

respectively. Their format is similar to that of the previous tables. However, it is worth pointing out that, for the first group, the columns display the number of candidate $s-t$ paths, the cardinality of the minimal complete set of efficient paths and CPU time, whereas for the second group the columns show the average values of the 10 instances. We can see that the number of candidate paths and efficient paths is very small. Moreover, as occurred in Sets 1 and 2, the power value does not seem to affect very much either these values or the CPU times involved.

Table 10: msmm-EQPA test results: Set 1. Mean of the number of candidate $s-t$ paths P, mean of the cardinality of the minimal complete set of efficient paths of the msmm-EQPP and mean of the computing time (CPU time in seconds)

Power= 3×10^8

r	n	m	# Candidate paths P σ_1	σ_2	σ_3	# Efficient paths σ_1	σ_2	σ_3	CPU time σ_1	σ_2	σ_3
10	10,000	100,000	32.3	85.8	34.7	2.1	1.5	1.3	0.17	0.49	0.16
		200,000	37.3	136.4	84.0	4.0	1.4	2.0	0.46	1.99	1.06
		300,000	38.1	163.7	107.8	3.3	1.6	2.5	0.78	3.63	2.08
		400,000	29.9	155.2	95.6	3.3	2.2	2.2	0.78	4.33	2.45
		500,000	35.8	195.4	144.9	4.2	2.6	2.1	1.10	6.71	4.50
	20,000	200,000	34.9	107.2	40.5	2.6	1.5	1.9	0.59	2.12	0.64
		400,000	41.2	141.9	77.7	3.4	1.9	2.3	1.30	4.95	2.44
		600,000	33.4	148.4	88.6	2.4	1.3	1.9	1.52	7.31	4.20
		800,000	50.2	193.0	133.9	5.4	2.7	2.3	2.64	11.66	8.35
		1,000,000	51.1	200.8	173.3	3.7	2.2	2.2	3.16	14.41	11.37
	30,000	300,000	37.2	90.0	40.1	2.9	1.6	1.8	1.13	2.91	1.18
		600,000	43.2	153.4	84.6	3.8	1.8	2.0	2.14	8.40	4.48
		900,000	43.8	183.3	115.8	4.1	1.7	2.5	2.85	14.52	8.19
		1,200,000	47.3	204.8	150.8	4.5	2.2	2.1	3.91	19.98	13.15
		1,500,000	26.5	216.5	152.4	4.5	2.5	2.2	3.08	24.41	17.05
	40,000	400,000	35.2	100.4	39.8	2.1	1.4	1.5	1.52	4.51	1.65
		800,000	34.0	163.8	89.4	4.0	1.7	1.8	2.47	12.67	6.05
		1,200,000	40.8	170.4	111.0	4.4	2.0	2.0	3.96	18.33	10.45
		1,600,000	38.7	211.6	135.7	4.6	2.3	1.8	4.36	28.18	16.11
		2,000,000	38.8	236.0	191.3	5.4	2.9	2.0	5.49	36.10	27.26
100	10,000	100,000	86.7	170.4	75.7	3.0	2.3	2.2	1.51	4.50	1.16
		200,000	129.0	300.3	242.7	4.0	2.2	3.6	4.63	17.98	10.93
		300,000	150.0	427.3	324.0	6.4	2.9	3.2	6.95	35.51	21.17
		400,000	105.1	395.3	256.6	4.4	2.3	4.3	8.10	47.05	25.96
		500,000	151.8	484.8	368.6	7.6	2.8	4.1	10.84	64.48	41.87
	20,000	200,000	124.7	241.8	97.0	3.2	2.6	2.7	5.91	18.14	4.50
		400,000	113.0	386.2	164.3	3.0	2.0	3.3	10.53	53.60	17.33
		600,000	107.5	383.4	250.4	3.9	1.6	2.3	12.99	83.80	40.93
		800,000	171.2	544.3	398.5	8.0	3.3	3.9	25.22	129.00	74.60
		1,000,000	163.1	601.3	382.7	6.9	2.9	5.0	27.65	167.20	92.83
	30,000	300,000	130.4	293.0	144.8	2.2	2.4	3.2	10.75	36.30	11.16
		600,000	143.2	389.8	319.8	3.1	2.1	3.6	18.59	81.35	48.37
		900,000	157.8	409.6	347.8	4.1	2.6	4.7	27.31	132.83	84.92
		1,200,000	149.9	549.8	372.9	8.0	3.6	4.4	34.49	196.90	112.49
		1,500,000	128.7	560.9	418.7	6.3	2.6	3.9	31.69	260.40	153.79
	40,000	400,000	97.1	262.2	87.1	1.9	2.2	2.6	13.06	43.94	9.78
		800,000	139.0	387.5	264.3	3.8	2.9	4.3	23.99	117.19	51.49
		1,200,000	149.0	487.8	262.0	4.3	2.1	3.6	36.17	187.90	85.27
		1,600,000	210.0	600.6	472.7	5.5	1.8	3.8	50.90	284.80	173.90
		2,000,000	209.7	730.4	534.5	6.1	2.0	4.4	63.56	375.40	242.20
1000	10,000	100,000	157.8	268.5	78.4	2.3	2.4	3.0	18.69	50.36	9.47
		200,000	210.7	432.6	301.3	4.0	2.2	4.1	50.91	210.14	102.77
		300,000	179.9	501.7	371.3	6.5	3.6	4.5	71.19	355.30	197.64
		400,000	191.5	475.4	368.7	4.8	2.6	4.5	84.24	482.99	272.88
		500,000	185.7	625.9	560.8	7.8	3.6	4.3	117.32	666.90	495.10
	20,000	200,000	187.7	348.8	189.2	2.3	2.5	3.1	62.44	200.00	64.55
		400,000	198.3	392.7	273.5	5.3	2.6	3.1	121.31	476.30	251.60
		600,000	137.6	417.0	252.9	4.3	3.1	3.4	126.83	666.90	316.12
		800,000	275.3	725.2	560.4	6.6	3.1	4.5	250.20	1263.70	775.80
		1,000,000	248.7	852.9	529.1	6.8	3.0	6.0	253.42	1584.70	996.60
	30,000	300,000	211.3	341.9	210.7	2.9	2.4	3.8	122.53	341.70	128.22
		600,000	226.7	441.9	310.8	3.1	2.8	4.6	219.80	853.40	420.20
		900,000	264.9	567.7	445.3	5.8	2.2	4.3	283.10	1400.50	769.30
		1,200,000	268.3	702.9	526.5	5.8	2.2	4.7	373.10	1889.20	1242.10
		1,500,000	168.5	689.5	398.7	6.8	2.1	5.2	326.90	2328.80	1326.30
	40,000	400,000	143.2	287.0	136.9	2.6	2.2	3.7	145.53	434.20	121.72
		800,000	231.1	519.4	333.7	4.2	2.2	4.1	259.60	1085.80	551.38
		1,200,000	256.8	682.2	454.6	3.5	2.2	4.2	376.69	1940.10	1085.10
		1,600,000	295.8	725.0	569.9	8.3	2.9	4.7	464.00	2623.90	1629.80
		2,000,000	267.8	966.8	616.2	8.6	2.9	4.2	565.90	3545.90	2309.20

Power= 6×10^8

r	n	m	# Candidate paths P σ_1	σ_2	σ_3	# Efficient paths σ_1	σ_2	σ_3	CPU time σ_1	σ_2	σ_3
10	10,000	100,000	32.3	85.8	67.0	2.1	1.5	1.5	0.17	0.49	0.30
		200,000	37.3	136.4	104.7	4.0	1.4	2.0	0.46	1.99	1.33
		300,000	38.1	163.7	130.9	3.3	1.6	2.4	0.78	3.63	2.59
		400,000	29.9	155.2	120.9	3.3	2.2	2.3	0.79	4.37	3.02
		500,000	35.8	195.4	165.7	4.2	2.6	2.1	1.11	6.76	5.23
	20,000	200,000	34.9	107.2	63.9	2.6	1.5	1.9	0.59	2.12	1.01
		400,000	41.2	142.0	103.1	3.4	1.9	2.4	1.30	4.98	3.30
		600,000	33.4	148.4	105.9	2.4	1.3	2.0	1.53	7.35	5.08
		800,000	50.2	192.8	164.5	5.4	2.7	2.2	2.66	11.72	10.09
		1,000,000	51.1	200.5	195.5	3.7	2.2	2.2	3.18	14.62	12.88
	30,000	300,000	37.2	89.8	57.9	2.9	1.6	1.9	1.14	2.92	1.66
		600,000	43.2	153.4	112.2	3.8	1.8	1.7	2.14	8.41	6.00
		900,000	43.8	183.3	146.3	4.1	1.7	2.5	2.85	14.53	9.95
		1,200,000	47.3	204.4	178.3	4.5	2.2	2.2	3.91	19.89	15.74
		1,500,000	26.5	216.5	177.9	4.5	2.5	2.2	3.08	24.43	19.78
	40,000	400,000	35.2	101.9	68.4	2.1	1.5	1.6	1.52	4.59	2.83
		800,000	34.0	164.1	106.2	4.0	1.7	2.1	2.48	12.80	7.49
		1,200,000	40.8	170.5	141.5	4.4	2.0	1.9	3.97	18.35	13.01
		1,600,000	38.7	211.1	174.1	4.6	2.3	1.8	4.36	28.09	20.31
		2,000,000	38.8	236.0	221.3	5.4	2.9	2.0	5.48	36.03	31.68
100	10,000	100,000	86.7	169.9	99.6	3.0	2.1	3.9	1.57	4.15	1.65
		200,000	129.0	300.0	267.5	4.0	2.2	3.2	4.66	17.97	13.37
		300,000	150.0	427.0	346.7	6.4	2.9	2.7	6.99	35.91	24.34
		400,000	105.1	395.8	292.6	4.4	2.3	4.4	8.22	48.00	31.40
		500,000	151.8	485.2	423.5	7.6	2.8	3.6	10.97	65.89	48.67
	20,000	200,000	124.7	243.3	165.0	3.2	2.5	3.5	5.97	18.50	8.14
		400,000	113.0	385.4	229.7	3.0	2.0	3.4	10.65	54.15	26.44
		600,000	107.5	384.5	300.5	3.9	1.8	2.4	13.07	84.54	51.90
		800,000	171.2	545.2	461.8	8.0	3.3	4.1	25.46	129.90	91.10
		1,000,000	163.1	600.4	479.2	6.9	2.8	5.0	27.79	168.70	114.90
	30,000	300,000	130.4	292.4	222.2	2.2	2.2	3.3	10.83	36.75	19.87
		600,000	143.2	389.4	360.9	3.1	2.0	3.4	18.70	81.93	60.27
		900,000	157.8	409.7	418.7	4.1	2.6	4.4	27.32	133.65	105.30
		1,200,000	149.9	550.5	480.2	8.0	3.7	4.0	34.53	198.20	137.20
		1,500,000	128.7	560.8	509.8	6.3	2.7	3.9	31.70	261.40	190.70
	40,000	400,000	97.1	262.1	148.6	1.9	2.2	3.0	13.20	44.60	19.14
		800,000	139.0	388.2	359.4	3.8	2.5	4.7	24.22	119.38	71.69
		1,200,000	149.0	486.7	352.6	4.3	2.1	3.9	36.33	188.80	115.83
		1,600,000	210.0	599.3	553.7	5.5	1.8	4.0	50.98	284.50	217.20
		2,000,000	209.7	729.8	605.8	6.1	2.0	4.6	63.36	375.90	278.00
1000	10,000	100,000	157.8	268.2	188.1	2.3	2.2	3.7	17.75	51.16	23.49
		200,000	210.7	433.0	375.5	4.0	2.2	3.8	50.14	205.65	135.51
		300,000	179.9	502.2	448.3	6.5	3.6	4.6	71.21	357.10	254.30
		400,000	191.5	475.6	436.3	4.8	2.7	4.8	84.49	490.98	318.10
		500,000	185.7	625.6	628.7	7.8	3.7	4.7	117.86	669.40	578.60
	20,000	200,000	187.7	348.0	269.3	2.3	2.4	3.2	62.38	201.50	104.33
		400,000	198.3	392.9	357.4	5.3	2.3	4.0	121.67	473.40	337.70
		600,000	137.6	416.5	336.7	4.3	3.2	3.2	126.44	666.60	413.69
		800,000	275.3	725.2	682.0	6.6	3.1	4.4	250.80	1275.00	947.10
		1,000,000	248.7	851.8	686.8	6.8	2.8	6.0	253.33	1611.10	1175.40
	30,000	300,000	211.3	340.9	386.7	2.9	2.4	5.1	122.86	346.60	227.90
		600,000	226.7	441.0	457.0	3.1	2.8	4.5	218.70	864.10	579.90
		900,000	264.9	569.0	520.6	5.8	2.2	4.0	279.10	1416.60	950.00
		1,200,000	268.3	703.2	680.1	5.8	2.3	4.9	373.50	1902.10	1498.10
		1,500,000	168.5	689.2	621.7	6.8	2.2	5.1	327.30	2337.50	1714.60
	40,000	400,000	143.2	287.0	254.3	2.6	2.1	3.8	146.16	432.20	231.68
		800,000	231.1	519.2	454.6	4.2	2.4	4.8	261.60	1098.90	762.40
		1,200,000	256.8	682.7	484.3	3.5	2.4	4.3	374.78	1947.50	1250.30
		1,600,000	295.8	723.9	654.6	8.3	3.0	4.5	465.90	2630.90	1997.60
		2,000,000	267.8	966.5	807.0	8.6	3.0	4.3	566.90	3533.90	2768.80

Power= 15×10^8

r	n	m	# Candidate paths P σ_1	σ_2	σ_3	# Efficient paths σ_1	σ_2	σ_3	CPU time σ_1	σ_2	σ_3
10	10,000	100,000	32.3	85.8	86.9	2.1	1.5	1.6	0.19	0.49	0.43
		200,000	37.3	136.4	125.1	4.0	1.4	2.0	0.47	2.01	1.66
		300,000	38.1	163.7	162.9	3.3	1.6	2.5	0.79	3.66	3.22
		400,000	29.9	154.8	134.4	3.3	2.2	2.2	0.79	4.41	3.60
		500,000	35.8	195.4	198.1	4.2	2.6	2.1	1.11	6.81	6.24
	20,000	200,000	34.9	107.2	90.5	2.6	1.5	1.9	0.60	2.14	1.55
		400,000	41.2	142.3	116.0	3.4	1.9	2.2	1.32	5.08	3.93
		600,000	33.4	150.1	149.6	2.4	1.3	2.0	1.53	7.52	6.90
		800,000	50.2	192.8	213.3	5.4	2.7	2.2	2.68	11.82	12.50
		1,000,000	51.1	200.5	215.6	3.7	2.2	2.2	3.20	14.64	14.91
	30,000	300,000	37.2	89.8	84.2	2.9	1.5	1.9	1.16	2.96	2.50
		600,000	43.2	153.4	144.1	3.8	1.8	1.8	2.17	8.53	7.66
		900,000	43.8	183.3	166.6	4.1	1.7	2.5	2.87	14.62	11.60
		1,200,000	47.3	204.5	191.5	4.5	2.2	2.2	3.94	19.99	17.69
		1,500,000	26.5	216.5	215.1	4.5	2.5	2.2	3.10	24.61	23.94
	40,000	400,000	35.2	101.9	112.4	2.1	1.5	2.0	1.53	4.60	4.74
		800,000	34.0	163.8	132.8	4.0	1.7	2.3	2.48	12.89	9.92
		1,200,000	40.8	170.5	175.7	4.4	2.0	2.0	3.99	18.42	16.73
		1,600,000	38.7	211.1	199.3	4.6	2.3	1.8	4.38	28.21	24.80
		2,000,000	38.8	236.0	245.3	5.4	2.9	2.0	5.50	36.17	37.22
100	10,000	100,000	86.7	170.1	179.7	3.0	2.2	2.8	1.55	4.08	3.35
		200,000	129.0	300.2	353.6	4.0	2.2	3.4	4.68	18.14	18.25
		300,000	150.0	426.7	370.0	6.4	2.9	2.9	7.09	36.51	29.81
		400,000	105.1	395.6	331.2	4.4	2.3	4.1	8.31	48.34	38.40
		500,000	151.8	485.2	498.6	7.6	2.8	3.4	11.13	66.90	59.27
	20,000	200,000	124.7	240.7	230.1	3.2	2.4	2.8	6.00	18.72	13.18
		400,000	113.0	383.8	324.1	3.0	1.9	3.3	10.76	54.81	38.18
		600,000	107.5	384.9	343.8	3.9	1.9	2.8	13.22	86.04	66.09
		800,000	171.2	545.2	549.3	8.0	3.3	4.2	25.92	131.90	113.42
		1,000,000	163.1	600.4	523.4	6.9	2.8	4.9	28.23	170.80	139.00
	30,000	300,000	130.4	293.0	316.9	2.2	2.2	2.6	10.99	37.38	30.65
		600,000	143.2	389.1	448.0	3.1	2.0	3.5	19.10	83.81	80.00
		900,000	157.8	409.7	483.0	4.1	2.6	4.1	27.57	134.94	130.36
		1,200,000	149.9	550.4	576.6	8.0	3.7	4.0	34.82	199.90	165.17
		1,500,000	128.7	560.4	578.3	6.3	2.7	3.8	31.84	262.00	226.30
	40,000	400,000	97.1	262.1	256.1	1.9	2.2	2.9	13.18	44.41	34.42
		800,000	139.0	388.5	448.1	3.8	2.6	4.2	24.25	119.08	98.59
		1,200,000	149.0	486.6	454.1	4.3	2.2	3.8	36.36	187.33	154.99
		1,600,000	210.0	599.3	624.5	5.5	1.8	4.1	51.16	282.60	252.70
		2,000,000	209.7	729.8	635.0	6.1	2.0	4.6	63.63	377.60	316.20
1000	10,000	100,000	157.8	267.7	275.8	2.3	2.1	4.2	17.28	51.01	40.81
		200,000	210.7	433.1	474.0	4.0	2.2	4.7	50.35	207.69	178.70
		300,000	179.9	502.1	531.0	6.5	3.6	3.8	71.83	356.40	334.10
		400,000	191.5	475.8	555.6	4.8	2.7	4.7	84.09	478.82	397.60
		500,000	185.7	625.4	675.5	7.8	3.7	4.7	119.20	667.70	649.90
	20,000	200,000	187.7	347.8	339.7	2.3	2.5	3.3	62.55	204.00	158.10
		400,000	198.3	392.6	467.8	5.3	2.3	4.4	121.04	474.30	464.40
		600,000	137.6	416.7	413.0	4.3	3.2	3.3	126.43	674.00	554.80
		800,000	275.3	725.5	638.8	6.6	3.1	4.6	251.10	1268.60	1088.30
		1,000,000	248.7	851.8	761.2	6.8	2.8	5.8	249.17	1601.20	1394.90
	30,000	300,000	211.3	341.1	398.7	2.9	2.4	4.7	123.20	341.80	294.30
		600,000	226.7	440.2	478.6	3.1	2.7	4.7	220.60	864.50	694.10
		900,000	264.9	567.7	624.3	5.8	2.1	4.0	279.60	1403.10	1183.50
		1,200,000	268.3	703.3	666.8	5.8	2.4	5.1	371.30	1910.30	1601.80
		1,500,000	168.5	689.2	729.4	6.8	2.2	4.7	320.70	2339.80	2219.40
	40,000	400,000	143.2	286.6	364.0	2.6	2.1	4.0	145.53	429.20	400.30
		800,000	231.1	518.3	543.5	4.2	2.4	4.2	260.90	1095.90	961.40
		1,200,000	256.8	683.2	638.0	3.5	2.4	4.0	374.40	1949.70	1632.30
		1,600,000	295.8	723.9	705.0	8.3	2.9	4.5	462.70	2642.90	2336.10
		2,000,000	267.8	966.5	986.9	8.6	3.0	4.0	557.30	3557.70	3318.10

Table 11: msmm-EQPA test results: Set 2. Mean of the number of candidate $s-t$ paths P, mean of the cardinality of the minimal complete set of efficient paths of the msmm-EQPP and mean of the computing time (CPU time in seconds)

			Power= 3×10^8									Power= 6×10^8									Power= 15×10^8								
			# Candidate paths P			# Efficient paths			CPU time			# Candidate paths P			# Efficient paths			CPU time			# Candidate paths P			# Efficient paths			CPU time		
r	n	m	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3
10	20,000	200,000	37.8	114.7	52.4	2.8	1.5	2.1	0.83	2.64	0.93	37.8	112.8	86.4	2.8	1.5	2.4	0.84	2.61	1.66	37.8	112.6	118.5	2.8	1.5	2.1	0.84	2.61	2.43
		400,000	46.3	155.5	100.5	4.0	2.2	1.7	1.96	7.95	4.42	46.3	155.5	128.5	4.0	2.2	1.6	1.97	7.97	5.54	46.3	155.5	138.1	4.0	2.2	1.6	1.97	7.98	6.32
		600,000	42.4	193.2	129.3	4.4	1.5	1.9	2.98	14.69	9.00	42.4	193.2	149.9	4.4	1.5	1.9	2.99	14.87	10.88	42.4	193.2	173.7	4.4	1.5	2.0	2.99	14.90	13.04
		800,000	46.2	192.0	150.5	4.6	2.9	2.5	3.84	20.30	13.35	46.2	192.0	168.0	4.6	2.9	2.5	3.84	20.18	14.88	46.2	192.0	196.9	4.6	2.9	2.4	3.84	20.24	17.80
		1,000,000	45.7	237.6	182.9	4.9	1.9	2.7	4.60	28.74	21.89	45.7	237.7	210.4	4.9	1.9	2.7	4.60	28.85	25.21	45.7	237.7	231.0	4.9	1.9	2.7	4.59	28.89	28.12
	40,000	400,000	36.6	135.4	50.4	2.9	1.5	1.8	1.93	7.71	2.53	36.6	135.5	91.7	2.9	1.5	2.0	1.93	7.73	4.51	36.6	135.5	127.9	2.9	1.5	2.1	1.93	7.75	6.78
		800,000	50.1	182.8	118.1	4.3	1.4	1.6	4.49	19.77	11.68	50.1	182.9	144.9	4.3	1.4	1.6	4.49	19.85	14.58	50.1	182.9	188.5	4.3	1.4	1.5	4.51	19.92	18.37
		1,200,000	48.3	195.5	146.2	3.7	1.5	2.2	6.76	30.01	21.46	48.3	195.4	169.8	3.7	1.5	2.2	6.79	30.18	25.70	48.3	195.4	219.7	3.7	1.5	2.2	6.77	30.28	32.11
		1,600,000	47.6	237.3	168.0	4.4	1.8	2.5	8.65	48.72	31.37	47.6	237.0	200.9	4.4	1.8	2.5	8.65	48.52	38.12	47.6	236.9	221.8	4.4	1.8	2.5	8.65	48.76	43.60
		2,000,000	53.3	235.8	171.4	3.2	1.8	2.2	11.48	60.76	42.12	53.3	235.6	213.3	3.2	1.8	2.3	11.48	61.00	50.79	53.3	235.6	244.8	3.2	1.8	2.3	11.49	60.94	59.13
	60,000	600,000	57.8	128.1	56.8	1.9	1.5	1.6	4.00	11.52	4.39	57.8	128.0	84.0	1.9	1.5	2.0	4.01	11.52	6.47	57.8	128.0	118.1	1.9	1.5	2.1	4.01	11.55	9.61
		1,200,000	53.7	176.1	101.9	3.7	1.3	2.5	7.98	32.08	15.02	53.7	176.0	140.9	3.7	1.3	2.5	7.99	32.41	21.48	53.7	176.0	172.1	3.7	1.3	2.4	8.00	32.41	26.63
		1,800,000	55.7	211.3	159.0	4.3	1.7	2.1	10.90	54.01	31.89	55.7	211.3	173.2	4.3	1.7	2.1	10.88	53.48	37.17	55.7	211.3	194.0	4.3	1.7	2.1	10.91	53.76	43.44
		2,400,000	55.8	230.1	156.0	4.0	1.3	2.4	14.96	75.73	45.43	55.8	230.1	210.3	4.0	1.3	2.4	15.01	76.29	60.12	55.8	230.1	232.6	4.0	1.3	2.4	15.00	76.03	69.36
		3,000,000	61.8	272.1	194.9	3.8	1.8	3.0	19.06	100.53	67.99	61.8	272.1	220.2	3.8	1.8	3.0	19.08	100.87	79.70	61.8	272.1	244.1	3.8	1.8	3.0	19.06	101.12	89.94
	80,000	800,000	47.3	127.7	55.3	2.7	1.5	2.0	5.67	15.17	5.76	47.3	127.7	82.6	2.7	1.5	2.0	5.67	15.23	8.60	47.3	127.7	122.7	2.7	1.5	2.3	5.65	15.18	13.24
		1,600,000	57.7	185.3	119.8	3.2	1.7	1.9	11.23	42.62	22.90	57.7	185.3	142.8	3.2	1.7	2.1	11.25	42.58	29.03	57.7	185.3	156.7	3.2	1.7	2.2	11.22	42.50	33.05
		2,400,000	51.1	236.8	157.7	4.5	1.9	1.9	13.57	76.23	45.21	51.1	236.8	208.3	4.5	1.9	1.8	13.60	76.65	58.35	51.1	236.8	237.7	4.5	1.9	1.7	13.59	76.41	66.93
		3,200,000	69.3	242.0	191.8	4.6	2.2	1.6	24.53	100.16	69.67	69.3	241.7	221.6	4.6	2.2	1.6	24.59	100.32	81.79	69.3	241.8	246.8	4.6	2.2	1.6	24.63	100.01	92.74
		4,000,000	62.3	265.1	209.9	3.4	2.2	2.2	23.88	141.41	92.30	62.3	265.1	236.5	3.4	2.2	2.2	23.92	141.73	104.90	62.3	265.1	246.9	3.4	2.2	2.2	23.86	141.54	114.94
	100,000	1,000,000	46.1	115.2	57.2	3.1	1.4	1.5	6.06	17.92	7.30	46.1	115.9	76.0	3.1	1.4	1.7	6.08	18.01	10.35	46.1	115.0	114.8	3.1	1.4	1.7	6.05	17.78	15.96
		2,000,000	56.9	207.1	116.2	4.3	1.3	2.1	13.12	56.95	28.93	56.9	205.5	152.7	4.3	1.3	2.0	13.13	56.37	38.23	56.9	205.3	194.4	4.3	1.3	2.0	13.10	56.55	50.06
		3,000,000	53.4	236.6	148.4	4.4	1.7	1.9	19.27	95.00	54.03	53.4	236.0	166.4	4.4	1.7	1.9	19.23	95.15	61.32	53.4	236.0	208.6	4.4	1.7	1.9	19.19	94.94	76.29
		4,000,000	53.6	247.5	184.7	3.8	1.6	3.0	25.09	129.80	84.04	53.6	247.5	214.5	3.8	1.6	3.0	25.01	131.10	100.70	53.6	247.5	236.7	3.8	1.6	3.0	24.99	130.70	117.60
		5,000,000	59.5	264.3	186.7	2.9	2.5	2.4	33.85	183.10	118.50	59.5	264.4	218.2	2.9	2.5	2.4	33.87	184.60	138.30	59.5	264.4	260.6	2.9	2.5	2.4	33.78	184.50	167.20
100	20,000	200,000	141.0	279.8	179.7	4.0	2.2	2.9	7.99	27.03	8.12	141.0	279.0	239.5	4.0	2.4	3.5	8.09	27.54	13.40	141.0	278.7	274.1	4.0	2.3	3.3	8.13	27.67	18.95
		400,000	173.7	370.9	232.6	5.2	2.0	2.9	20.62	71.83	34.53	173.7	370.3	281.3	5.2	2.0	3.0	20.79	72.66	45.32	173.7	370.5	384.9	5.2	2.0	2.9	20.74	72.28	62.63
		600,000	188.9	489.7	370.2	6.4	2.6	3.1	27.91	133.31	82.89	188.9	489.7	459.5	6.4	2.6	3.0	27.99	133.67	103.44	188.9	489.5	511.4	6.4	2.6	2.8	27.97	134.01	123.65
		800,000	187.2	572.8	390.2	6.6	3.2	3.9	38.27	198.40	127.80	187.2	572.4	414.7	6.6	3.2	4.0	38.23	200.00	145.70	187.2	572.3	536.1	6.6	3.2	3.7	38.31	198.70	183.40
		1,000,000	197.8	613.2	532.1	6.8	2.2	5.0	48.54	284.80	193.30	197.8	613.0	611.9	6.8	2.2	4.9	48.50	284.80	232.10	197.8	613.0	666.0	6.8	2.2	4.9	48.64	281.80	262.20
	40,000	400,000	124.3	363.4	145.9	2.4	2.4	2.7	20.23	74.61	18.70	124.3	363.3	249.9	2.4	2.4	3.3	20.30	75.29	34.22	124.3	362.8	322.0	2.4	2.4	3.7	20.31	75.61	52.43
		800,000	175.7	434.7	323.0	3.6	2.2	2.9	43.00	182.20	94.14	175.7	434.3	443.1	3.6	2.2	2.9	42.94	183.90	134.17	175.7	433.8	531.2	3.6	2.2	3.2	42.93	182.90	173.60
		1,200,000	224.8	568.2	410.4	6.4	2.4	3.7	66.39	314.90	199.50	224.8	567.5	501.2	6.4	2.4	3.9	66.51	315.40	245.90	224.8	567.2	614.3	6.4	2.4	3.7	66.72	318.90	311.00
		1,600,000	208.5	626.9	427.0	4.9	2.9	4.4	90.72	444.10	261.50	208.5	625.4	542.8	4.9	2.9	4.4	90.50	444.00	314.80	208.5	625.5	665.2	4.9	2.9	4.1	90.62	443.50	408.70
		2,000,000	189.1	757.8	456.0	7.9	2.1	4.8	99.25	618.20	371.80	189.1	757.8	596.2	7.9	2.1	4.4	99.21	609.80	455.50	189.1	756.5	615.6	7.9	2.1	4.5	99.40	620.80	513.30
	60,000	600,000	203.1	359.8	247.5	2.7	1.8	2.8	39.26	104.60	40.52	203.1	357.8	307.4	2.7	1.8	3.1	39.25	104.40	61.20	203.1	356.8	383.1	2.7	1.8	3.5	39.42	104.37	86.37
		1,200,000	200.5	486.1	384.6	5.6	2.9	4.7	70.64	281.80	153.50	200.5	485.2	455.9	5.6	3.0	4.1	70.94	282.60	198.60	200.5	484.5	523.1	5.6	3.0	3.8	70.97	284.50	240.80
		1,800,000	215.6	599.5	478.2	5.8	2.9	4.1	97.46	519.00	295.90	215.6	600.1	616.4	5.8	2.9	4.2	97.32	511.20	379.70	215.6	599.9	644.3	5.8	2.9	3.8	97.89	521.70	453.10
		2,400,000	223.4	706.3	450.8	7.7	2.5	4.2	134.00	730.60	414.90	223.4	706.0	580.4	7.7	2.5	4.2	134.08	719.60	499.60	223.4	705.5	670.5	7.7	2.4	4.2	134.18	739.10	614.80
		3,000,000	210.2	814.3	576.4	7.1	2.5	5.4	149.63	998.20	628.20	210.2	813.9	662.0	7.1	2.5	5.4	149.54	1011.20	755.70	210.2	813.5	751.2	7.1	2.5	5.1	149.50	992.20	849.20
	80,000	800,000	186.9	319.3	159.6	2.9	1.7	3.6	52.50	150.60	43.81	186.9	317.4	268.3	2.9	1.7	4.1	52.61	152.06	75.05	186.9	317.6	365.6	2.9	1.7	3.6	52.41	150.06	118.94
		1,600,000	229.8	573.6	347.1	4.0	2.4	4.2	98.99	426.60	198.00	229.8	573.4	482.6	4.0	2.4	4.3	98.60	430.00	276.10	229.8	573.1	534.0	4.0	2.4	4.2	98.83	423.60	346.90
		2,400,000	265.5	611.3	443.2	5.9	2.6	4.0	154.07	648.60	426.70	265.5	611.2	561.3	5.9	2.6	4.4	154.71	655.50	532.10	265.5	611.0	728.5	5.9	2.6	4.1	154.24	645.30	663.50
		3,200,000	296.1	731.0	518.4	5.6	2.8	5.0	212.50	1003.60	585.20	296.1	731.0	613.7	5.6	2.8	4.9	211.40	972.20	699.90	296.1	731.0	656.8	5.6	2.8	4.6	211.60	975.20	804.80
		4,000,000	256.2	774.2	579.3	6.7	2.4	4.0	239.60	1266.30	841.40	256.2	773.6	652.1	6.7	2.4	4.1	239.20	1264.90	948.90	256.2	773.5	745.6	6.7	2.4	3.9	238.70	1264.30	1115.20
	100,000	1,000,000	157.9	401.5	171.3	2.2	1.7	3.5	56.04	204.60	51.36	157.9	398.5	237.4	2.2	1.7	3.6	56.00	203.10	82.62	157.9	398.5	322.3	2.2	1.8	3.4	55.88	202.70	141.63
		2,000,000	228.1	554.2	412.2	4.7	2.6	4.4	130.56	535.90	274.80	228.1	554.3	523.8	4.7	2.6	4.2	129.17	534.10	369.90	228.1	554.0	586.0	4.7	2.6	3.7	129.31	535.90	459.00
		3,000,000	224.9	708.5	512.2	5.1	2.2	4.7	188.70	981.00	513.70	224.9	708.6	594.0	5.1	2.2	4.7	187.00	992.60	617.50	224.9	708.6	583.5	5.1	2.2	4.7	188.20	994.20	738.30
		4,000,000	238.2	817.4	571.8	6.1	2.5	4.9	229.10	1282.30	758.40	238.2	818.1	628.4	6.1	2.5	5.3	231.30	1291.70	899.90	238.2	818.0	657.7	6.1	2.5	5.3	229.80	1290.40	1019.40
		5,000,000	247.3	880.3	637.6	7.6	3.1	4.5	310.70	1607.80	1118.50	247.3	879.9	726.7	7.6	3.1	4.9	307.90	1620.90	1297.60	247.3	880.1	801.4	7.6	3.1	4.9	309.20	1632.20	1509.10

Table 12: Summarized msmm-EQPA test results of Sets 1 and 2

Mean of the number of candidate $s-t$ shortest paths

		Set 1						Set 2					
		Power= 3×10^8		Power= 6×10^8		Power= 15×10^8		Power= 3×10^8		Power= 6×10^8		Power= 15×10^8	
	r	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev
σ_1	10	38.49	17.45	38.49	17.45	38.49	17.45	51.85	18.50	51.85	18.50	51.85	18.50
	100	140.85	72.64	140.85	72.64	140.85	72.64	207.79	90.67	207.79	90.67	207.79	90.67
	1000	215.38	120.02	215.38	120.02	215.38	120.02						
σ_2	10	162.90	59.22	162.92	59.10	162.99	59.09	201.00	59.92	200.84	60.08	200.79	60.14
	100	430.34	179.06	430.28	178.85	430.09	178.88	576.59	205.53	576.07	205.92	575.82	206.00
	1000	548.25	257.21	548.08	257.03	547.93	257.07						
σ_3	10	104.60	56.64	130.31	59.00	157.97	60.51	134.64	57.33	164.48	59.87	193.91	59.19
	100	289.26	167.03	353.90	182.59	426.22	185.12	398.29	172.25	490.02	184.09	562.98	187.88
	1000	374.45	224.46	486.50	246.68	563.38	237.14						

Mean of the number of efficient paths

		Set 1						Set 2					
		Power= 3×10^8		Power= 6×10^8		Power= 15×10^8		Power= 3×10^8		Power= 6×10^8		Power= 15×10^8	
	r	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev
	r	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev
σ_1	10	3.74	1.77	3.74	1.77	3.74	1.77	3.75	1.62	3.75	1.62	3.75	1.62
	100	4.79	3.00	4.79	3.00	4.79	3.00	5.36	2.89	5.36	2.89	5.36	2.89
	1000	5.12	3.27	5.12	3.27	5.12	3.27						
σ_2	10	1.95	1.07	1.96	1.07	1.95	1.07	1.74	0.93	1.74	0.93	1.74	0.93
	100	2.46	1.40	2.43	1.37	2.44	1.37	2.43	1.23	2.44	1.24	2.44	1.23
	1000	2.64	1.34	2.65	1.36	2.64	1.37						
σ_3	10	2.02	0.89	2.05	0.88	2.08	0.86	2.13	0.88	2.18	0.88	2.18	0.88
	100	3.66	1.63	3.77	1.65	3.61	1.56	4.01	1.65	4.11	1.57	3.98	1.53
	1000	4.20	1.78	4.39	1.89	4.37	1.59						

Mean of the CPU time in seconds

		Set 1						Set 2					
		Power= 3×10^8		Power= 6×10^8		Power= 15×10^8		Power= 3×10^8		Power= 6×10^8		Power= 15×10^8	
	r	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev
	r	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev
σ_1	10	2.17	1.66	2.18	1.66	2.19	1.66	11.23	9.09	11.24	9.09	11.22	9.07
	100	21.24	18.03	21.33	18.01	21.49	18.09	105.07	84.42	104.88	83.92	104.96	84.14
	1000	214.69	168.33	214.64	167.92	213.46	165.52						
σ_2	10	11.38	9.99	11.40	9.97	11.48	10.01	54.94	47.83	55.12	48.18	55.09	48.14
	100	116.91	101.98	117.64	102.11	118.26	102.20	523.64	441.44	523.72	442.83	524.48	443.33
	1000	1120.04	967.68	1125.61	968.41	1125.93	973.54						
σ_3	10	7.14	7.27	8.66	8.36	10.59	9.73	34.08	31.79	41.14	36.94	48.59	42.59
	100	65.23	66.71	81.36	78.01	100.42	88.83	309.49	292.75	376.76	336.87	450.84	385.43
	1000	653.29	641.72	813.47	753.08	995.15	876.58						

Table 13: msmm-EQPA test results: Set 3, first group. Number of candidate $s-t$ paths P, cardinality of the minimal complete set of efficient paths of the msmm-EQPP and computing time (CPU in seconds)

		Power= 3×10^8									Power= 6×10^8									Power= 15×10^8								
		# Candidate paths P			# Efficient paths			CPU time			# Candidate paths P			# Efficient paths			CPU time			# Candidate paths P			# Efficient paths			CPU time		
	Dest.	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3
NY	4	13	30	0	1	1	0	2.36	3.76	1.57	13	36	0	1	1	0	2.36	4.13	1.59	13	28	0	1	1	0	2.37	3.23	1.59
	3	11	16	6	2	1	1	2.07	2.33	1.85	11	16	17	2	1	2	2.08	2.32	2.36	11	16	21	2	1	1	2.07	2.32	2.56
	2	16	65	0	1	1	0	3.28	8.49	1.56	16	65	0	1	1	0	3.27	8.52	1.59	16	64	0	1	1	0	3.27	8.38	1.60
	1	27	35	0	1	1	0	3.30	3.95	1.57	27	31	0	1	1	0	3.30	3.71	1.59	27	40	0	1	1	0	3.29	4.30	1.60
BAY	4	3	2	0	1	1	0	1.82	1.85	1.67	3	2	0	1	1	0	1.83	1.86	1.69	3	7	0	1	1	0	1.81	2.13	1.67
	3	7	8	0	1	1	0	2.28	2.24	1.68	7	8	0	1	1	0	2.30	2.30	1.68	7	8	0	1	1	0	2.28	2.33	1.69
	2	6	21	0	1	1	0	2.30	3.93	1.66	6	7	0	1	1	0	2.31	2.33	1.67	6	7	0	1	1	0	2.32	2.36	1.66
	1	14	14	0	1	1	0	3.78	3.67	1.69	14	24	0	1	1	0	3.79	5.13	1.66	14	18	0	1	1	0	3.76	4.33	1.67
COL	4	1	1	0	1	1	0	2.40	2.39	2.33	1	1	0	1	1	0	2.41	2.39	2.34	1	1	0	1	1	0	2.40	2.38	2.34
	3	6	4	0	1	1	0	2.73	2.60	2.35	6	7	0	1	1	0	2.71	2.79	2.34	6	5	0	1	1	0	2.71	2.66	2.33
	2	11	0	0	1	0	0	3.31	2.34	2.33	11	19	0	1	1	0	3.34	4.03	2.34	11	23	0	1	1	0	3.33	4.48	2.34
	1	19	0	0	1	0	0	6.22	2.34	2.33	19	0	0	1	0	0	6.22	2.42	2.34	19	23	0	1	1	0	6.20	6.65	2.35
FLA	4	11	6	0	1	1	0	6.02	5.34	4.51	11	7	0	1	1	0	6.03	5.80	4.51	11	7	0	1	1	0	6.01	5.80	4.50
	3	1	8	0	1	1	0	4.87	5.60	4.52	1	9	0	1	1	0	4.88	6.03	4.49	1	9	0	1	1	0	4.88	6.06	4.48
	2	90	43	0	1	1	0	27.17	15.38	4.48	90	117	0	1	1	0	27.19	33.30	4.52	90	108	0	1	1	0	27.25	31.48	4.49
	1	73	43	0	1	1	0	19.51	13.10	4.54	73	99	0	1	1	0	19.49	24.47	4.50	73	83	0	1	1	0	19.50	20.94	4.49
NE	4	15	9	0	1	1	0	15.52	13.96	11.21	15	8	0	1	1	0	15.53	13.78	11.32	15	8	10	1	1	1	15.65	13.66	14.23
	3	2	4	0	1	1	0	12.34	12.32	11.21	2	2	0	1	1	0	12.33	11.92	11.26	2	2	0	1	1	0	12.39	11.84	11.27
	2	60	54	0	1	1	0	52.79	50.02	11.23	60	45	0	1	1	0	52.73	43.90	11.32	60	46	0	1	1	0	52.61	45.08	11.34
	1	47	31	0	1	1	0	41.98	31.89	11.29	47	54	0	1	1	0	41.97	48.77	11.24	47	55	0	1	1	0	42.00	46.86	11.23
CAL	4	7	1	0	1	1	0	13.52	11.79	11.57	7	1	0	1	1	0	13.54	11.86	11.62	7	4	0	1	1	0	13.56	12.73	11.57
	3	14	0	0	1	0	0	16.27	11.60	11.57	14	0	0	1	0	0	16.34	11.64	11.61	14	7	0	1	1	0	16.30	14.32	11.63
	2	62	0	0	1	0	0	58.60	11.56	11.59	61	0	0	1	0	0	57.94	11.61	11.60	61	20	0	1	1	0	57.94	22.55	11.61
	1	20	0	0	1	0	0	23.44	11.65	11.57	20	0	0	1	0	0	23.44	11.63	11.60	20	23	0	1	1	0	23.37	28.01	11.57
LKS	4	2	1	0	1	1	0	18.52	17.99	17.59	2	1	0	1	1	0	18.50	18.51	17.52	2	1	0	1	1	0	18.49	18.53	17.43
	3	52	72	0	1	1	0	46.81	55.76	17.56	52	98	0	1	1	0	46.91	68.97	17.59	52	96	0	1	1	0	46.90	68.30	17.51
	2	38	162	0	1	1	0	61.77	193.00	17.55	38	375	0	1	1	0	61.78	402.00	17.57	38	359	0	1	1	0	61.89	392.00	17.50
	1	55	85	0	1	1	0	50.06	71.22	17.58	55	185	0	1	1	0	50.23	148.00	17.57	55	179	0	1	1	0	50.13	145.00	17.56

Table 14: msmm-EQPA test results: Set 3, second group. Mean of the number of candidate $s-t$ paths P, mean of the cardinality of the minimal complete set of efficient paths of the msmm-EQPP and mean of the computing time (CPU time in seconds)

		Power= 3×10^8									Power= 6×10^8									Power= 15×10^8								
		# Candidate paths P			# Efficient paths			CPU time			# Candidate paths P			# Efficient paths			CPU time			# Candidate paths P			# Efficient paths			CPU time		
	Dest.	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3
NY	4	6.0	6.0	0.0	1.7	1.0	0.0	0.92	0.87	0.34	6.0	6.2	0.6	1.7	1.0	0.2	0.92	0.88	0.43	6.0	6.2	3.4	1.7	1.0	0.6	0.92	0.88	0.73
	3	6.2	8.3	5.0	1.7	1.4	1.5	0.63	0.76	0.58	6.2	8.3	7.2	1.7	1.4	1.8	0.63	0.76	0.70	6.2	8.3	10.6	1.7	1.4	2.1	0.63	0.76	0.85
	2	12.0	16.5	0.2	1.5	1.2	0.1	1.55	1.99	0.35	12.0	16.7	2.5	1.5	1.2	0.4	1.55	2.01	0.56	12.0	16.7	10.4	1.5	1.2	0.8	1.55	2.02	1.31
	1	7.3	9.1	0.0	1.6	1.0	0.0	1.10	1.22	0.33	7.3	9.4	0.0	1.6	1.0	0.0	1.10	1.24	0.38	7.3	9.4	2.6	1.6	1.0	0.5	1.10	1.23	0.65
BAY	4	2.8	4.1	0.0	1.1	1.0	0.0	0.67	0.67	0.38	2.8	4.4	0.4	1.1	1.1	0.2	0.67	0.68	0.40	2.8	4.5	0.8	1.1	1.1	0.5	0.67	0.68	0.44
	3	5.4	6.0	0.0	1.6	1.0	0.0	0.90	0.94	0.38	5.4	6.5	0.0	1.6	1.0	0.0	0.90	0.98	0.38	5.4	6.6	0.3	1.6	1.0	0.1	0.90	1.00	0.41
	2	7.7	7.2	0.0	1.3	1.1	0.0	1.34	1.29	0.38	7.7	7.4	0.0	1.3	1.1	0.0	1.35	1.31	0.37	7.7	7.7	0.8	1.3	1.1	0.2	1.35	1.35	0.48
	1	4.2	8.4	0.0	1.3	1.0	0.0	1.07	1.52	0.38	4.2	9.3	0.0	1.3	1.0	0.0	1.07	1.62	0.38	4.2	9.3	1.4	1.3	1.0	0.3	1.07	1.65	0.52
COL	4	2.5	4.0	0.4	1.5	1.9	0.2	0.82	0.94	0.54	2.5	4.4	0.5	1.5	1.9	0.3	0.82	0.99	0.59	2.5	4.5	1.9	1.5	1.9	0.9	0.82	0.99	0.78
	3	6.3	7.3	0.0	1.8	1.2	0.0	1.03	1.11	0.50	6.3	7.4	0.5	1.8	1.2	0.3	1.03	1.11	0.55	6.3	7.4	1.8	1.8	1.2	0.8	1.03	1.11	0.64
	2	5.8	9.4	0.0	1.3	1.1	0.0	1.04	1.36	0.50	5.8	9.5	1.6	1.3	1.1	0.5	1.04	1.37	0.63	5.8	9.5	6.4	1.3	1.1	1.0	1.04	1.37	1.06
	1	3.5	2.3	0.0	1.0	0.9	0.0	1.47	1.21	0.50	3.5	2.6	0.0	1.0	0.9	0.0	1.47	1.28	0.51	3.5	2.9	0.0	1.0	1.0	0.0	1.47	1.34	0.56
FLA	4	2.0	1.6	0.0	1.1	1.0	0.0	1.95	1.81	1.28	2.0	1.6	0.0	1.1	1.0	0.0	1.95	1.80	1.27	2.0	1.6	0.0	1.1	1.0	0.0	1.95	1.80	1.26
	3	1.5	1.3	0.0	1.0	1.0	0.0	1.82	1.87	1.29	1.5	1.3	0.0	1.0	1.0	0.0	1.82	1.87	1.27	1.5	1.3	0.0	1.0	1.0	0.0	1.82	1.87	1.25
	2	5.5	5.1	0.0	1.2	1.0	0.0	2.97	2.84	1.29	5.5	5.9	0.0	1.2	1.0	0.0	2.97	3.05	1.27	5.5	6.1	0.0	1.2	1.0	0.0	2.97	3.12	1.26
	1	4.6	6.1	0.0	1.1	1.0	0.0	2.43	2.81	1.28	4.6	7.7	0.0	1.1	1.0	0.0	2.43	3.21	1.27	4.6	7.8	0.0	1.1	1.0	0.0	2.43	3.23	1.26
NE	4	9.9	10.8	0.5	1.3	1.3	0.3	5.59	5.72	2.26	9.9	11.0	2.5	1.3	1.3	1.0	5.59	5.81	2.94	9.9	11.0	7.2	1.3	1.3	1.4	5.59	5.79	4.48
	3	5.0	8.1	0.3	1.4	1.4	0.2	4.30	4.96	2.20	5.0	8.1	1.2	1.4	1.4	0.6	4.30	4.95	2.64	5.0	8.1	5.7	1.4	1.4	1.3	4.29	4.95	4.23
	2	4.4	5.2	0.0	1.1	1.0	0.0	7.44	7.85	2.12	4.4	5.6	0.0	1.1	1.0	0.0	7.44	8.19	2.31	4.4	5.6	0.3	1.1	1.0	0.1	7.44	8.20	3.06
	1	2.9	2.7	0.0	1.0	0.9	0.0	5.74	5.56	2.12	2.9	3.0	0.0	1.0	1.0	0.0	5.74	5.76	2.40	2.9	3.1	0.0	1.0	1.0	0.0	5.74	5.83	2.98
CAL	4	3.7	4.9	0.0	1.3	1.3	0.0	3.87	4.17	2.55	3.7	5.1	0.2	1.3	1.3	0.2	3.87	4.22	2.60	3.7	5.4	1.2	1.3	1.3	0.6	3.87	4.32	2.92
	3	2.2	3.0	0.0	1.0	1.0	0.0	3.89	3.81	2.55	2.2	3.1	0.0	1.0	1.0	0.0	3.89	3.86	2.53	2.2	3.1	0.0	1.0	1.0	0.0	3.88	3.88	2.49
	2	4.3	2.5	0.0	1.0	1.0	0.0	7.19	5.53	2.55	4.3	2.9	0.0	1.0	1.0	0.0	7.19	5.91	2.52	4.3	2.7	0.0	1.0	1.0	0.0	7.19	5.73	2.49
	1	3.3	3.2	0.0	1.0	1.0	0.0	5.36	5.21	2.55	3.3	3.3	0.0	1.0	1.0	0.0	5.36	5.24	2.52	3.3	3.4	0.0	1.0	1.0	0.0	5.36	5.47	2.49
LKS	4	3.3	2.6	0.9	1.5	1.3	0.9	7.10	6.70	5.05	3.3	2.6	1.1	1.5	1.3	1.0	7.09	6.69	5.34	3.3	2.6	1.3	1.5	1.3	1.2	7.10	6.66	5.76
	3	9.8	11.5	0.0	2.0	1.1	0.0	9.49	10.44	3.94	9.8	11.9	0.9	2.0	1.1	0.2	9.49	10.57	4.47	9.8	11.9	6.6	2.0	1.1	1.0	9.50	10.57	7.56
	2	12.5	13.8	0.0	2.0	1.0	0.0	18.53	19.95	3.93	12.5	14.0	0.0	2.0	1.0	0.0	18.53	20.23	4.01	12.5	14.1	5.5	2.0	1.0	0.8	18.53	20.37	10.23
	1	11.2	15.9	0.0	1.4	1.0	0.0	13.25	16.87	3.93	11.2	15.6	1.0	1.4	1.0	0.2	13.26	16.53	4.75	11.2	15.6	4.8	1.4	1.0	0.6	13.26	16.53	7.74

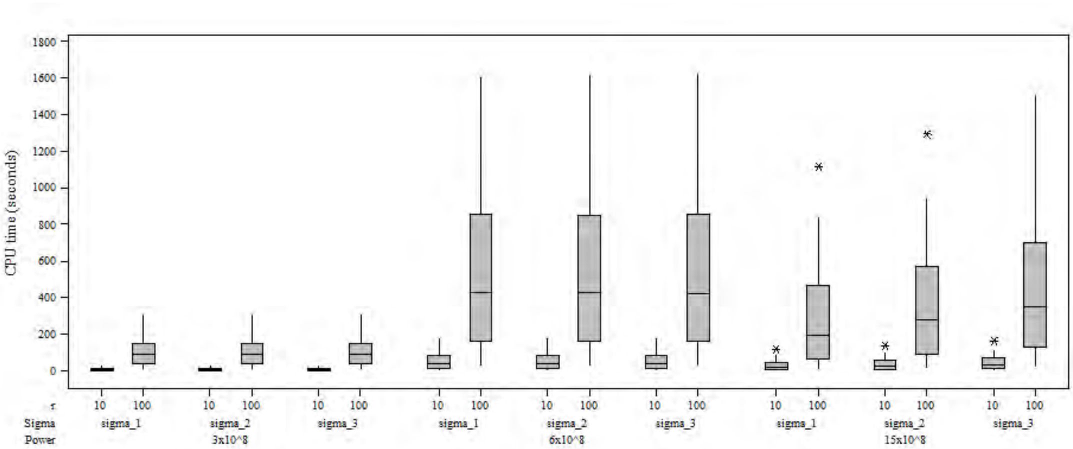

Figure 4: Set 2: Boxplots of the msmm-EQPA computing time depending on the number of capacities, the value of σ and the power of the nodes

7. Conclusions

In this paper we have addressed several problems relating to residual energy in energy-constrained capacitated networks. First, we have analyzed the problem of maximizing the residual energy at the nodes after transmitting σ data units. The polynomial algorithm developed to solve this problem is based on computing maxmin problems with respect to a defined residual energy at arcs in subnetworks which are in a sense associated with the different capacities. These subnetworks satisfy, by construction, that there is energy available at the nodes for transmitting the given data units. Second, the bi-objective problem is analyzed in which transmission time is minimized and residual energy at the nodes is maximized. This problem is solved in polynomial time with an algorithm which is reminiscent of the algorithms developed for solving the QPP, since eventually only SPPs with respect to the delay time are solved. However, it is worth mentioning that the algorithm proposed in this paper is more sophisticated as it involves the consideration of subnetworks embedded in networks. The results of the computational study carried out demonstrate the excellent performance of both algorithms proposed.

Finally, taking into account the optimal solution of the mm-EPP, the problem is then studied of obtaining an energy-constrained quickest path restricted to leave a certain residual energy at the nodes. This problem generalizes the energy-constrained quickest path problem introduced in [4] and can be solved by slightly modifying the polynomial algorithm proposed there to take into account a new definition of an $s-t$ feasible path.

8. Acknowledgements

The authors would like to thank the two anonymous referees whose comments have helped to improve the presentation of the paper.

9. References

References

[1] H.I. Calvete. The quickest path problem with interval lead times. *Computers and Operations Research*, 31(3):383–395, 2004.

[2] H.I. Calvete and L. del-Pozo. The quickest path problem with batch constraints. *Operations Research Letters*, 31(4):277–284, 2003.

[3] H.I. Calvete, L. del-Pozo, and J.A. Iranzo. Algorithms for the quickest path problem and the reliable quickest path problem. *Computational Management Science*, 9(2):255–272, 2012.

[4] H.I. Calvete, L. del-Pozo, and J.A. Iranzo. The energy-constrained quickest path problem. *Optimization Letters*, 11:1319–1339, 2017.

[5] G.H. Chen and Y.C. Hung. Algorithms for the constrained quickest path problem and the enumeration of quickest paths. *Computers and Operations Research*, 21:113–118, 1994.

[6] Y.L. Chen and Y.H. Chin. The quickest path problem. *Computers and Operations Research*, 17(2): 153–161, 1990.

[7] J.C.N. Clímaco, M.M.B. Pascoal, J.M.F. Craveirinha, and M.E.V. Captivo. Internet packet routing: Application of a k-quickest path algorithm. *European Journal of Operational Research*, 181:1045–1054, 2007.

[8] M. Ehrgott. *Multicriteria Optimization*. Springer, Berlin, Heildeberg, 2nd edition, 2005.

[9] Z. Fei, B. Li, S. Yang, C. Xing, H. Chen, and L. Hanzo. A survey of multi-objective optimization in wireless sensor networks: Metrics, algorithms, and open problems. *IEEE Communications Surveys and Tutorials*, 19:550–586, 2017.

[10] M.L. Fredman and R.E. Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of the Association for Computing Machinery*, 34(3):596–615, 1987.

[11] H.W. Hamacher and S.A. Tjandra. Mathematical modelling of evacuation problems: A state of the art. In M. Schreckenberg and S.D. Sharma, editors, *Pedestrian and Evacuation Dynamics*, pages 227–266. Springer, Berlin, 2002.

[12] P. Hansen. Bicriterion path problems. In G. Fandel and T. Gal, editors, *Multiple criteria decision making: Theory and applications*, volume 177 of *Lecture Notes in Economics and Mathematical Systems*, pages 109–127. Springer, Heidelberg Berlin, 1980.

[13] M.E. Keskin. A column generation heuristic for optimal wireless sensor network design with mobile sinks. *European Journal of Operational Research*, 260:291–304, 2017.

[14] M. El Khadiri and W.C. Yeh. An efficient alternative to the exact evaluation of the quickest path flow network reliability problem. *Computers and Operations Research*, 76:22–32, 2016.

[15] D. Klingman, A. Napier, and J. Stutz. Netgen: A program for generating large scale capacitated assignment, transportation, and minimum cost flow network problems. *Management Science*, 20(5): 814–821, 1974.

[16] C. Lersteau, A. Rossi, and M. Sevaux. Minimum energy target tracking with coverage guarantee in wireless sensor networks. *European Journal of Operational Research*, 265:882–894, 2018.

[17] Y.K. Lin. Extend the quickest path problem to the system reliability evaluation for a stochastic-flow network. *Computers and Operations Research*, 30:567–575, 2003.

[18] Y.K. Lin. Calculation of minimal capacity vectors through k minimal paths under budget and time constraints. *European Journal of Operational Research*, 200:160–169, 2010.

[19] E.Q.V. Martins. On a special class of bicriterion path problems. *European Journal of Operational Research*, 17:85–94, 1984.

[20] E.Q.V. Martins and J.L.E. Santos. An algorithm for the quickest path problem. *Operations Research Letters*, 20(4):195–198, 1997.

[21] M.H. Moore. On the fastest route for convoy-type traffic in flowrate-constrained networks. *Transportation Science*, 10(2):113–124, 1976.

[22] C-K. Park, S. Lee, and S. Park. A label-setting algorithm for finding a quickest path. *Computers and Operations Research*, 31(14):2405–2418, 2004.

[23] M.M.B. Pascoal, M.E.V. Captivo, and J.C.N. Clímaco. A comprehensive survey on the quickest path problem. *Annals of Operations Research*, 147(1):5–21, 2006.

[24] B. Pelegrín and P. Fernández. On the sum-max bicriterion path problem. *Computers and Operations Research*, 25(12):1043–1054, 1998.

[25] M. Pollack. Letter to the editor - the maximum capacity through a network. *Operations Research*, 8 (5):733–736, 1960.

[26] N.S.V. Rao, W.C. Grimmell, S. Radhakrishan, Y.C. Bang, and N. Manickam. Quickest paths for different network router mechanisms. *Proceedings of ninth International Conference on Advanced Computing and Communications*, 2001.

[27] J.B. Rosen, S.Z. Sun, and G.L. Xue. Algorithms for the quickest path problem and the enumeration of quickest paths. *Computers and Operations Research*, 18(6):579–584, 1991.

[28] A. Sedeño-Noda and J.D. González-Barrera. Fast and fine quickest path algorithm. *European Journal of Operational Research*, 238(2):596–606, 2014.

[29] B. Tang, R. Bagai, F.N.U. Nilofar, and M.B. Yildirim. A generalized data preservation problem in sensor networks - a network flow perspective. In M. García-Pineda, J. Lloret, S. Papavassiliou, S. Ruehrup, and C.B. Westphall, editors, *Ad-Hoc Networks and Wireless. Lecture Notes in Computer Science*, pages 275–289. Springer, Berlin, 2015.