Single-commodity network design with random edge capacities

Biju K. Thapalia[*] Teodor Gabriel Crainic[†] Michal Kaut[‡]
Stein W. Wallace[§]

July 2010

Abstract

This paper examines the single-commodity network design problem with stochastic edge capacities. We characterize the structures of the optimal designs and compare with the deterministic counterparts. We do this primarily to understand what constitutes robust network designs, but hope that the results can be used also to develop better heuristics than those available today.

Keywords: Network flows, Single-commodity network design, Survivable networks, Edge failure, Correlations, Robustness

1 Introduction

Many important physical networks, such as distribution networks for water, oil and gas pipelines, road system, or distribution channels are integral parts of our lives. These networks are made to last for a long time and are often subjected to daily routine operational decisions. If any parts of these networks are down, major portion of society will be affected. Focus on cost savings tends to make these networks sparser, and hence also more vulnerable to any kind of disruption, failure, maintenance, congestion, etc that may occur (see discussions in Ball et al. (1995) and Balakrishnan et al. (1998)). The owners of these networks must therefore design and maintain them, often under strict budgetary regimes, so that they work well even in the case of reduced capacities or broken links in the network. Hence, in our view, there is an increased need to understand what constitutes a good design in light of random capacities. In particular, we wish to see if it is important to use models explicitly expressing the randomness in capacities when designing the networks. And if the answer is yes, we would like to understand in what ways the designs from deterministic design models fall short of designs from models explicitly considering the uncertainty.

Two terms which are used frequently in case of network failures are reliability and survivability. Reliability is the probability that a network functions according to a specification. Survivability is the ability of a network to perform according to a specification after it has been damaged. Reliability is a measure of the service provided by the network and survivability is a measure of the network itself. Hence, higher level of reliability depends upon higher degree of survivability.

[*]Molde University College, Norway
[†]Université du Québec à Montréal (UQAM), Canada
[‡]Norwegian University of Science and Technology (NTNU), Norway; michal.kaut@iot.ntnu.no
[§]Lancaster University Management School, England

Good robust designs trade off deterministic initial costs versus expected future costs in a good way. Most often, by increasing initial costs, the network is provided with more operational flexibility, and expected future costs decrease. Optimality is achieved when additional initial investments cost more than what is gained operationally. (Note though, that at times uncertainty induces lower initial investments. This may for example take the form of postponements of decisions.) The best way to have a high degree of survivability in a network is to have many links between the nodes of the network, i.e., a dense network, but this increases the cost to built the network. So it is expensive to built a network with a high level of survivability, but of course, the operational costs in light of disturbances will decrease. More important to this paper, though, is that many networks, having the same initial investment costs, may react very differently to disturbances. So even though it is clear that many links will increase the quality of a network, we would like to understand what characterizes a good way to increase the number of links. And, again, we would like to understand if deterministic models will guide us well in designing the networks, or if they will lead us astray.

Stochastics in a network may arise both in terms of supply/demand and edge capacities. The latter, which is what we discuss in this paper, may be in the form of edges being on/off or in terms of capacities being a random fraction of the maximal capacity, representing such as broken cables and damaged roads. The on/off situation may cover both capacitated and uncapacitated edges. Clearly, for capacitated edges, the on/off situation is a special case of the random capacity case where there are only two possibilities, no capacity or full capacity. The latter is sometimes referred to as a binary state system, while with many possible capacities it is referred to as a multi-state system. For more information about the multi-state system refer to the book by Lisnianski and Levitin (2003). We consider the case of capacitated edges with random edge capacities, and hence includes also the on/off capacitated case.

There are many application areas where network design with random edge capacities are important. Possibly the most famous one is the design of survivable networks in telecommunications. Much of this work originates with Suurballe and Tarjan (1984). Refer to Balakrishnan et al. (1998), Clarke and Anandalingam (1995) and Myung et al. (1999) for a more detailed understanding and many variations and extensions. Water, oil, and gas distribution system are other central cases. If we drop the pressure constraints, these pipeline design problems simplify and can be expressed as single commodity design problems with underlying linear single commodity flow problems (Brimberg et al., 2003).

A water pipeline network spans different consumers, of which some are very sensitive to disruption in supply, such as hospitals and certain industries. Disruption in the supply arise due to pipeline ruptures, leakages or blockage, which affect the overall flow in the network. Many studies are done on designing optimal and reliable water distribution systems (Chung et al., 2009; di Pierro et al., 2009; Montalvo et al., 2008). A seminal paper in gas distribution is by Rothfarb et al. (1970). The papers by Wolf and Smeers (1996), Martin et al. (2006) and Brimberg et al. (2003) show the design issues of oil\gas distribution networks. The work by Midthun et al. (2009) shows how the network structure and the physical properties effects the operation and development of natural gas transportation networks. Logistics network design with robustness consideration (Meepetchdee and Shah, 2007) is another application area. The paper by Chen et al. (1999) discuss the reliability of transport networks with random link capacity. Many applications are also seen in network interdiction problems (Cormican et al., 1998; Ramirez-Marquez and Rocco S., 2009; Smith et al., 2007), as well as the protection of network systems from natural disasters. Examples of the latter are retrofit of highway bridges for increased robustness of the transportation system (Liu et al., 2009), and evacuation planning for emergencies (Andreas and Smith, 2009).

In their work on fleet management (see for example Cheung and Powell (1996)) Powell and his co-authors use random edge capacities to represent random demand. Bounds on the recourse problem in this situation is discussed in Wallace (1987).

It is evident that a network with random edge capacities must function reasonably well in many situations with partial or full breakdown of capacities. A common way to investigate this situation is to perform single- or multi-parameter sensitivity analysis in order to understand how the optimal solution changes as a function of these breakdowns. This approach might seem appropriate, but in fact it is not. This is outlined in detail in Wallace (2000) and Higle and Wallace (2003). Logically, when performing sensitivity analysis one is assuming that the design can be postponed until after breakdowns have become known. This is hardly ever pointed out, though. So, whether sensitivity analysis is performed or not, one ends up with a solution not created for robustness, and hence, may have to face difficult operational decisions when breakdowns occur.

Much of the literature in the field of survivable networks discuss different heuristics but do not address the resulting network structure. We know that a deterministic solution might perform very badly when used in a stochastic environment, that is, when subjected to the uncertainties that were suppressed when the deterministic model was solved, see for example Thapalia et al. (2010a). The reason is simply that it is not made to handle uncertainties in a good way. This paper studies the structural differences between stochastic and deterministic designs, in order to understand what flexibility means in the optimal network structure for a single commodity flow problem with single or multiple sources and sinks. We also hope that this can be used to develop heuristics for the stochastic case.

The remainder of the paper is organized as follows. Section 2 explains the problem in detail with the mathematical formulation. Section 3 explains the experimentation set-ups and scenario generation. Section 4 lists the computational results with discussions and finally Section 5 will conclude the paper.

2 Problem description and Modeling issues

Given a set of nodes (divided into source nodes, demand nodes, and transshipment nodes) and a set of potential edges connecting these nodes, the single-commodity network design problem with random edge capacities is the problem of determining a subset of the edges to open (including the edges' capacities), so as to fulfill the demand at the demand nodes at minimal cost, taking into account capacities of the source nodes and the potential failures of the edges.

The design is based on minimizing the sum of the fixed costs of selecting edges connecting the nodes; linear costs to open capacities in the edges; per unit flow costs on the edges; and per unit penalty costs for not satisfying demand. Not satisfying demand can have many interpretations, such as sending the flow at a later point in time, with another mode, or a straightforward rejection. In any case, in the model, it takes the form of a penalty cost per unit of unsatisfied demand. We find it crucial to include the possibility of not satisfying all the demand, as it is unlikely to have a network which satisfies all demand in all situations (see Thapalia et al. (2010a) for more discussion). The same formulation is used in both the stochastic and deterministic models, to make the results comparable. We view supply as a capacity, and hence, do not consider unused supply as a problem.

When we wish to compare a stochastic network design model with its deterministic counterpart, we need to be careful about how we define the deterministic model. For random demand, this is not so difficult. If historical data is available, for example, demand will usually be the average observed demand (or possibly some other forecasted demand based on the history). Hence, it is not unreasonable to compare the stochastic model with a deterministic model where all demands are replaced with their mean values.

It is not quite as easy for the case of random edge capacities. If the starting point is the stochastic model, and we ask "What is the natural deterministic counterpart?", the answer is most likely a model where edge capacities are replaced by their means. But if the starting point is that of setting up a deterministic network design model (possibly realizing that edges might fail, but not wanting to model it), it is rather likely that edges will be treated with capacities equal to their capacities when they are fully operational, that is, their maximal capacities from a stochastic perspective. One will argue: This edge costs a\$ and has a capacity of b. One will not use expected capacity taking possible failures into account. So in what follows, for each stochastic case, we shall consider two deterministic cases: average capacity and maximal (design) capacity.

We shall let all source nodes (in the case of multiple sources) have the same capacity. Hence, our assumption is that a set of demand nodes will have their demands satisfied from a set of equally-sized source nodes through edges with random capacities. We have chosen this approach to prevent our optimal designs from being affected by variations (across nodes) in a parameter which is not the primary focus of the paper. So, the first stage decisions in this problem are to decide which edges to open and what capacities to install. The second stage decisions are the flow decisions in the given network. The recourse action here is described by a penalty cost incurred for not satisfying demand.

A word of warning might be in place here. If a stochastic optimization problem, as well as its deterministic counterpart (where all random variables are replaced by their means or some other related values), use hard constraints in the formulation (for example requiring that demand *must* be met), the optimal design from the deterministic program will normally be infeasible in the stochastic program, and hence, its expected cost be infinitely large. In our context that will be caused by the network not having enough capacity to satisfy all demand in scenarios with many edges at low capacity. On the other hand, if soft constraints are used (allowing demand to be rejected at a cost), the deterministic solution will normally be feasible in the stochastic model, but its expected performance can be made arbitrarily bad by choosing large penalties on the soft constraints (unsatisfied demand). Hence, if the *goal* is to make the deterministic solution look bad, that is easy to achieve. However, that is not our goal. So we set the penalties at reasonable levels, and our goal is not to (again) show how bad the deterministic solution is, but to *understand* its relationship to its stochastic counterpart. So, we shall certainly present numbers, and we do believe the numbers are informative. However, there will never be really objective results in this setting. There will always be a subjective element.

2.1 Mathematical formulation

Let $G = (\mathcal{N}, \mathcal{E})$ be a network defined by a set $\mathcal{N}$ of n nodes and set $\mathcal{E}$ of m edges (undirected arcs), where

$$\mathcal{E} \subset \{k = (i, j) : i \in \mathcal{N}, j \in \mathcal{N} \text{ and } i < j\}.$$

Each edge is indexed either by i, j or by k.

The randomness in the edges is described by a set of scenarios $\mathcal{S}$, where each individual scenario $s \in \mathcal{S}$ has one capacity realization for each edge. We shall discuss in Section 3.2 how the scenarios were generated. The notations for the sets, parameters, and variables associated with this problem are as follows:

Sets:

$\mathcal{C}$	set of all source nodes;
$\mathcal{D}$	set of all demand nodes;
$\mathcal{T}$	set of all transshipment nodes; $\mathcal{T} = \mathcal{N} \setminus (\mathcal{C} \bigcup \mathcal{D})$;
$\mathcal{S}$	set of all scenarios s.

Parameters:

M	"big M";
R	unit cost of unsatisfied demand;
P^s	probability of scenario $s \in \mathcal{S}$;
C_k	flow cost on edge $k \in \mathcal{E}$;
G_k	fixed setup cost for edge $k \in \mathcal{E}$;
H_k	variable setup cost; cost for adding one unit of cap. to edge $k \in \mathcal{E}$;
V_k	initial/ existing capacity on edge $k \in \mathcal{E}$, if any;
D_i	demand ($D_i < 0$) in node $i \in \mathcal{D}$;
D	supply in each source node, $D > 0$;
Δ_k^s	the portion of capacity on edge $k \in \mathcal{E}$ that works in scenario s.

Variables:

$x_k^s = x_{ij}^s$	flow on edge $k = (i, j) \in \mathcal{E}$ going in direction $i \to j$, in scenario $s \in \mathcal{S}$;
$z_k^s = z_{ij}^s$	flow on edge $k = (i, j) \in \mathcal{E}$ going in direction $j \to i$, in scenario $s \in \mathcal{S}$;
u_k	new capacity that is developed on edge $k \in \mathcal{E}$;
e_i^s	for $i \in \mathcal{D}$, this is the unsatisfied/lost demand in node i in scen. $s \in \mathcal{S}$;
	for $i \in \mathcal{C}$, this is the unused capacity of source node i in scen. $s \in \mathcal{S}$;
y_k	1 if edge $k \in \mathcal{E}$ is developed, 0 otherwise.

We assume that total supply, coming from equally-sized source nodes equals maximal demand in the network, so that

$$D = \{-\sum_{j \in \mathcal{D}} \{D_j\}\}/|\mathcal{C}| \tag{1}$$

where $|\mathcal{C}|$ is the number of source nodes.

Our overall problem is hence:

$$\min \sum_k G_k y_k + \sum_k H_k u_k + \sum_s P^s \left\{ \sum_k C_k \left(x_k^s + z_k^s \right) + R \sum_{i \in \mathcal{D}} e_i^s \right\} \tag{2}$$

Subject to:

$$\sum_{j:\,(ij)\in\mathcal{E}} \left(x_{ij}^s - z_{ij}^s\right) - \sum_{j:\,(ji)\in\mathcal{E}} \left(x_{ji}^s - z_{ji}^s\right) = \begin{cases} 0 & \forall i \in \mathcal{T}, \forall s \in \mathcal{S} \\ D - e_i^s & \forall i \in \mathcal{C}, \forall s \in \mathcal{S} \\ D_i + e_i^s & \forall i \in \mathcal{D}, \forall s \in \mathcal{S} \end{cases} \tag{3}$$

$$x_k^s + z_k^s \le \Delta_k^s \left(u_k + V_k\right) \quad \forall k \in \mathcal{E} \quad \forall s \in \mathcal{S} \tag{4}$$

$$u_k \le M y_k \qquad \forall k \tag{5}$$

$$0 \le e_i^s \le -D_i \qquad \forall i \in \mathcal{D}; \forall s \tag{6}$$

$$x_k^s, z_k^s, u_k, e_i^s \ge 0 \text{ and } y_k \in \{0, 1\} \qquad \forall k; \forall i; \forall s \tag{7}$$

The objective function (2) minimizes the total costs of the network. The first part is the costs of constructing all new edges, the second part the costs of building all the new capacities, the third part the expected flow costs through all the edges and the fourth part is the expected penalty costs of not fulfilling demand. Constraints (3) model conservation of flow at nodes. The left-hand side is the net outflow from node i, which must be zero for all the transshipment nodes $i \in \mathcal{T}$ and is equal to the unused capacity for source node $i \in \mathcal{C}$. For the demand nodes, the net outflow must be equal to the satisfied demand; since D_i is negative in this case, the right-hand side is the a difference between the scenario demand D_i and the (positive) unsatisfied demand e_i^s.

Constraints (4) represent the flow limit in each edge. The left hand side of the equation is the net flow on the edge k which should be less then or equal to the total capacity of the edge. Since we do not start with any initial/existing capacity, we always have $V_k = 0$. Note that in an optimal solution, an edge will never have flow in both directions. Constraints (5) show that new capacity u_k can be developed only if edge k is built. Constraints (6) give bounds for the rejection amount and finally, (7) insure that all variables are non-negative and the edge constructions binary.

For the deterministic counterpart (as mentioned in Section 2) we replace the stochastic edge capacities by their expectations and their maximal values, resulting in two separate deterministic cases.

We model the problem in AMPL and solve it to optimality using CPLEX 9.0. The solution time varies from few seconds to 5 hours depending on the case, on a PC with 3 GHz Intel® CPU and 8 GB of RAM.

3 Experimentation and Scenario generation

In this section we first discuss the test cases and their sources before turning to scenario generation and the question of stability relative to the chosen scenarios. Our tests are designed to achieve two goals: Firstly, we wish to understand how deterministic designs perform in stochastic environments and to what extent information from deterministic designs are useful for the stochastic problem. Secondly, our goal with these tests is to be able to characterize the stochastic designs, so we can qualitatively describe good designs and use this knowledge to evaluate a given design without making any serious calculations.

3.1 Test instance generation

We took five different networks used in Thapalia et al. (2010a) and Thapalia et al. (2010b). The networks named Germany, Nobel-EU, France, and Pdh are telecommunication examples from the SNDlib library (Orlowski et al., 2010), and Montreal_r06 is obtained from CIRRELT (Interuniversity Research Centre on Enterprise Networks, Logistics and Transport), Montreal. The names, as such, of the test networks do not mean anything in our computational setup.

In total 76 test instances are constructed using the above five networks. These test instances are created in the following way: for each of the five networks, we created single-source and multi-source test cases by selecting one or multiple source nodes. This way we created 38 test instances of which 20 are single-source and 18 are multi-source. Since we know that correlations may play important roles in the design of a network, we created positive correlated and uncorrelated cases for each instance. In the case of positive correlation, adjacent edges are given correlations of 0.5, while edges which are separated by one edge are given correlations of 0.2. Edges which are separated by two edges have a correlation of 0.1. This is a natural setting for natural calamities. Whenever one edge is hit hard, there is a chance that also nearby edges are hit, see for example Che et al. (2002). In this way we get 76 test instances, of which 40 are for the single-source case and 36 are for the multi-source case.

It is worth noting that all these networks are multi-commodity network design problems, so to suit our problem only some parameter values are used. We only kept the coordinates (where available) for the nodes and the fixed setup cost G_k for the edges. The values for the other parameters – variable setup costs H_k and flow costs C_k – are all chosen proportional to the Euclidean distance between the node pairs. The cost of unfulfilled demand R is derived for each test cases using some multiple of the highest value of the fixed setup cost, variable setup cost, and unit flow cost together for an edge in the network. We made sure that R is not driving the solution. The results in the first part of section 4.1 are based on test cases with these cost structures.

The Montreal test instance does not have node coordinates, so we used Graphviz (Gansner and North, 2000) to draw the graph using fixed setup cost as distance measure. The graphs of the test instances coming from Nobel-EU are planar whereas the other graphs are non-planar. A closer description of the test instances are found in Table 1.

In the case of single-source networks, we selected four potential source nodes. When one of them is source node, the rest are transshipment nodes. And for the cases of multi-source networks we selected four (or three for Montreal_r06 and France) sets of source nodes to make four (or three for Montreal_r06 and France) test instances from each network. The number of source nodes for each case is listed in the fifth column of Table 1. In the pictures that will follow, the distance between two nodes reflects not only the actual distance, but also the levels of the variable setup costs and flow costs. If an edge is twice as long as another, it is also twice as costly with respect to these two costs.

Given the difficulty of solving the stochastic network design problem to optimality we kept n (the number of nodes) below 30 and m (the number of edges) below 50 for the cases.

3.2 Scenario generation and Stability test

Stochastic programs need discrete probability distributions. A scenario is a vector of length m containing a possible capacity for each edge. We have created scenarios with equal prob-

Table 1: The different test cases. The basic names are kept as in the source, even though the cases are adjusted to our needs.

Problem name	# nodes	# edges	# dem. n.	# src. n.	# tests
Germany_SS	29	48	10	1	4
Germany_MS	29	48	10	3	4
France_SS	16	30	10	1	4
France_MS	16	30	10	3	3
Montreal_r06.1_SS	10	38	5	1	4
Montreal_r06.1_MS	10	38	7	3	3
Pdh_SS	11	30	7	1	4
Pdh_MS	11	30	7	4	4
Nobel-EU_SS	28	38	8	1	4
Nobel-EU_MS	28	38	8	4	4

abilities of occurring, using a variant of the moment-matching method from Høyland et al. (2003).

Lacking specific knowledge, we have chosen a triangular distribution on the $[0, 1]$ interval, with mode at one, which gives an expected value of 0.67. Note that a mode below one would imply that the edge (in the continuous case) has an extremely low probability of being close to fully operational. We feel the chosen distribution is a reasonable description of edge failures.

The decision on the number of scenarios used to represent the stochastics is critical as we want to be sure we study the effects of randomness on our model, and not some random side-effect of the scenario generating procedure. For a given scenario generation procedure, there is normally a trade-off between the number of scenarios representing the underlying distribution and the time needed to solve the stochastic program to optimality. As we increase the number of scenarios, we increase the quality of the representation of the distribution, but also decrease the chance to solve the model to optimality within a manageable time. The task is thus to find the smallest number of scenarios that still gives solutions that are both in- and out-of-sample stable, in the sense described in Kaut and Wallace (2007).

We ran our in-sample stability test with different numbers of scenarios and ended up with 200, considering the solution time and stability. The deviation (measured by standard deviation of the objective values of all runs divided by the mean of the objective values) in all cases are less then 1% except for the case of Montreal_r06, where it is 1.5% for single-source cases and 2% for multi-source cases. Out-of-sample stability tests, using a reference tree with 2000 scenarios, are all within 1%. With these values, we are satisfied that we have stability.

3.3 Comparison Tests

As outlined in the Introduction, the deterministic solution, by construction, has a worse expected behavior than its stochastic counterpart. However, we would like to understand more about why this is the case, and in what sense it is worse. This is partly motivated by what we found in Thapalia et al. (2010a) and Thapalia et al. (2010b) where we discussed random demand: The edges (if not their capacities) from the deterministic solution provided a good starting point for the stochastic case. This is unusual.

In order to check the quality of the deterministic designs, as well as comparing them to the stochastic ones, we have set up two tests, named *comparisons*. Whenever a comparison is performed, we take the deterministic and stochastic designs – or parts thereof – (i.e. the first-stage solutions) and evaluate them using reference trees – in our case trees with 2000 scenarios, to make sure we have good approximations of the true distributions. The costs from the design and evaluation phases are added up, making the reported costs comparable across all tests.

A The classical test where the whole first-stage solution is evaluated out-of-sample. This amounts to solving a 2000-scenario stochastic program with all first-stage variables (designs and capacities) fixed, so in fact this equals the solution of 2000 independent second-stage problems. Since the second stage does not involve any integer variables, this is very fast.

B Only information on which edges should be opened is imported from the first stage. So, in a 2000-scenario stochastic program, all discrete variables y describing opened and closed edges—we call it a *skeleton*—are fixed and the stochastic program is run. So the model is allowed to install any capacity on the opened edges (also lower than in the deterministic case), but not to open new ones.

Applied on the deterministic solution, Comparison A is the classical test of the quality of the deterministic solution. The purpose of Comparisons B is to check if the deterministic solution possibly has a good structure, but badly chosen capacities (typically too low). If this is the case, the skeleton can be found using a deterministic model, and then capacities set in a stochastic *linear* model. Algorithmically, this is much simpler than solving a stochastic mixed-integer model. Of course, if the skeleton is good, it also provides information about the relationship between the stochastic and deterministic models of the same problem.

In what follows, we discuss the major findings, details are given in the Appendix. Our first need is to understand the relationship between the stochastic and deterministic solutions.

For each of the 38 test instances, we solve two deterministic problems, one with full capacity available and the other with the mean (in our case 67% of full) capacity on each edge. These choices were motivated in Section 2. In addition, we solve two stochastic versions of each instance, one with uncorrelated and one with positively correlated edge failures. For each stochastic versions, we take the solution of the stochastic model and the two deterministic solutions and evaluate them out-of-sample on the reference tree, i.e. a tree with 2000 scenarios and the same correlations as those used to solve the stochastic programs.

Our measure of the quality of a solution (or a partial solution like a skeleton) will be the ratio between the expected costs using the deterministic solution and the expected costs using the stochastic solution. As the expected costs are never close to zero in our problems, there is no danger of running into problems amounting to a division by zero. Note that since both the stochastic and deterministic solutions are evaluated out-of-sample, the ratio might become slightly smaller than 1.

We also want to explore the relationship between the variable setup cost and the performance of the deterministic *skeleton* in the stochastic environment. The hypothesis is that as the variable setup costs increase, the stochastic skeleton will look increasingly like the deterministic one (which is a tree) due to the cost of opening more capacity than what is absolutely necessary. Similarly, as the variable setup costs increase, the deterministic *skeleton* will tend toward the stochastic one, in terms of number of open edges, as it is (relatively speaking) governed more and more by installed capacity and less and less by the number of opened edges.

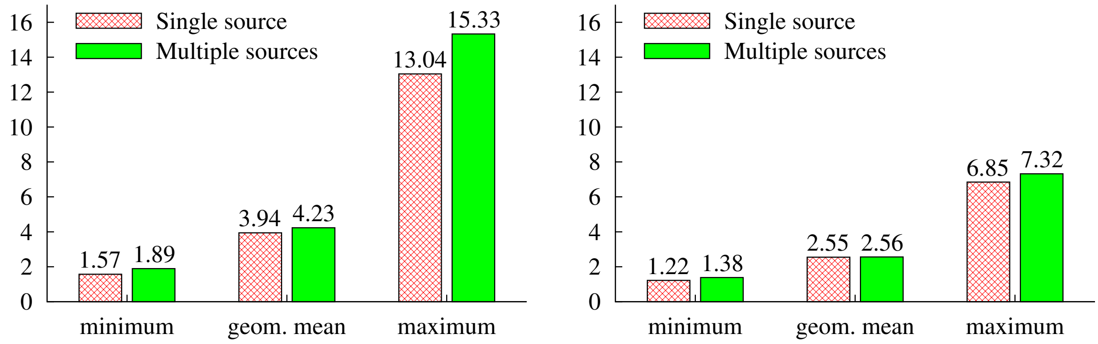

Figure 1: Results of the Comparison A tests. Quality of the solutions to the deterministic problems with full (*left*) and mean (*right*) edge capacities, measured as a ratio of their out-of-sample costs relative to the costs of the corresponding stochastic solutions.

In other words, we postulate that as variable setup costs decrease, the expected behavior of the deterministic *skeleton* in a stochastic setting becomes increasingly bad, and this is true for both multi- and single-source cases. And as the variable setup costs increase the deterministic *skeleton* will perform better, more so for multi-source case than the single-source case.

For this, we start with the above discussed test cases which we define as base cases. From each base case we make five additional test cases by taking 33%, 66%, 133%, 166% and 200% of the variable setup cost of the base case.

Certainly, the mean-capacity deterministic problem is equivalent to the full-capacity deterministic problem with $1/0.67 = 1.5$ times higher variable setup costs. So this parametric analysis of variable setup costs contains the analysis of the relationship between the two deterministic cases. However, to keep the interpretations apart, we have chosen this approach instead of reading one set of deterministic results from within the results of the other.

4 Computational Results

We present our computational results with discussion.

4.1 Inheritance from the deterministic solutions

The deterministic solution behaves badly in the stochastic environment, while inheriting the deterministic structures, *the skeletons*, is rather good for both the single- and multi-source cases, see Figures 1 and 2. In Figure 1 we see that both deterministic designs are bad but there are some differences. We observe clearly that when mean capacities are used, results are better (around 2.5 times higher on average for both the single- and multi-source cases) than when the full capacities are used (around 4 to 4.25 times higher on average). The reason is simply that when mean capacities are used, the edges seem to have less capacity, and hence more is installed. And as we have observed in earlier papers—and that will be confirmed here—deterministic designs do not only suffer in terms of structure (skeletons) but also in terms of too low capacities. So doing what many practitioners do—run deterministic models with a pessimistic view on edge capacities—is indeed a good idea.

From Figure 1, we also observe that the single-source case performs better than the multi-source one for Comparison A for both deterministic versions. This observation is in line with

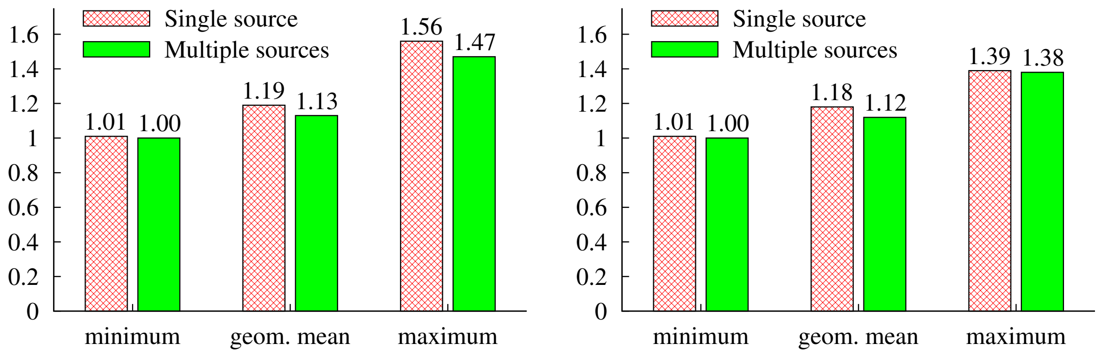

Figure 2: Results of the Comparison B tests. Quality of the solutions to the deterministic problems with full (*left*) and mean (*right*) edge capacities, measured as a ratio of their out-of-sample costs relative to the costs of the corresponding stochastic solutions.

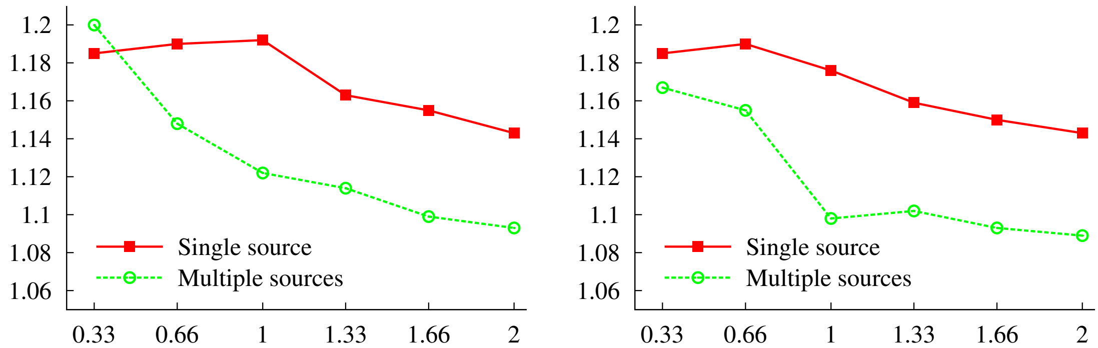

Figure 3: Graph of Comparison B test values for different variable setup cost. The X-axis shows the percentage of variable setup cost as compared to the base case and the Y-axis the quality of the solutions to the deterministic problems with full (*left*) and mean (*right*) edge capacities, measured as a ratio of their out-of-sample costs relative to the costs of the corresponding stochastic solutions.

our previous work on random demand (Thapalia et al., 2010b), where we found that as the number of source nodes increases, the deterministic network designs behave steadily worse. The reason is that while the deterministic skeleton is a tree for the single-source case, it is generally a forest with rather shallow trees in the multi-source case. And in a stochastic environment the forest simply does not provide enough connections to satisfy demand in the case of highly variable capacities. The negative effects of a forest are less pronounced than in the case of random demands, though, as the demand within a tree does not change.

Figure 2 shows that when we use the deterministic *skeleton* and apply a stochastic program to set capacities, the results are rather good, implying that the skeletons perform quite well. We can also see that the multi-source cases do better than the single-source cases. This observation holds for most of the tested levels of variable costs, as shown in Figure 3. There, we can also observe that when variable setup costs are only 33% of the base case costs, the losses are comparable and higher than for higher variable setup costs. As the variable setup costs increase, the loss of using a deterministic skeleton decreases, and more so for the multi-source case. This is true for both maximal and mean value edge capacities. This observation confirms that the quality of a design is a function of both connectivity and capacity. When the variable setup cost is low, the deterministic designs are guided by the shortest routes in terms of fixed costs of opening the edges, and the resulting *skeletons* perform relatively badly in a stochastic environment. But when variable setup costs increase the deterministic designs change. They are no longer primarily guided by the shortest routes in terms of fixed setup costs (mostly implying as few edges as possible), but also edge capacity costs. This results in more edges being opened as total installed capacity, rather than the number of edges with capacity is the primary driver of costs. These *skeletons*, when used in a stochastic environment, perform better as they contain more connections. In the multi-source case this happens more quickly because there are more edges than in the single-source case. Again, since the demand does not vary within each small tree in the forest, contrary to the case with random demand, the fact that the trees effectively cut the design into smaller parts is not a problem. So what is needed here is high density (many paths) and enough installed capacity within each tree to make sure the demand within the tree is satisfied with a high probability so as to achieve low penalty costs.

4.2 Structural Characteristics

This section examines the structures of the deterministic and stochastic network designs from the tests mentioned in Section 3.1 and focuses on a few important observations which shed light on the characteristics of the stochastic designs under random edge capacities.

Deterministic structures with maximal and mean value capacities

Though not very surprising, it is interesting to note that there are some differences between the skeletons based on maximal edge capacity and those based on mean capacity. The designs with mean capacity have higher installed capacities. This is natural since the reduction in actual edge capacities causes higher installed capacities.

As mentioned before, the mean-capacity deterministic problem is equivalent to the full-capacity deterministic problem with $1/0.67 = 1.5$ times higher variable setup costs. Hence, also Figure 3 sheds light on the relationship between the two deterministic solutions.

Network Density

We observe that stochastic designs (multi-source as well as single-source cases) have more edges than their deterministic counterparts and also have higher installed capacities per open edge. In Table 2, the second, third, and fourth columns show that stochastic designs have more edges than their deterministic counterparts. Similarly, columns five, six, and seven show that installed capacity *per open edge* is higher in the stochastic designs except for the cases of Germany (both for multi- and single-source case) and Nobel-EU_SS. However, in all cases we find a higher total installed capacity in the stochastic designs.

The reasons for this is firstly that with more edges in the network, there are alternative ways to reach demand nodes in the event of reduced edge capacities. And secondly, higher installed capacities insure that there are reasonably high capacities reaching the demand nodes even when there are faults in the edges.

Also we can see from Table 2 (third and fourth columns), that designs for the uncorrelated cases generally have more edges than the corresponding positively correlated cases. This can be explained by the fact that in the uncorrelated cases it is very useful to have alternate paths by setting up extra edges. So when one path leading to a demand node has reduced

Table 2: Average number of edges and average capacities installed per open edge for all tests in each case. SS and MS denotes single-source and multi-source cases. The parameter ρ refers to correlations in stochastic cases.

	Number of edges			Capacity per edge		
	det.	$\rho\geq\mathbf{0}$	$\rho=\mathbf{0}$	det.	$\rho\geq\mathbf{0}$	$\rho=\mathbf{0}$
France_SS	11.75	16.25	18	624	1096	843
Germany_SS	16.75	28.75	32.75	7	5	4
Montreal_r06_SS	5.5	9.5	10	104	162	111
Nobel-EU_SS	13.62	25.25	29	28	24	18
Pdh_SS	7.37	9.5	10.5	131	186	154
France_MS	13	17	17.67	475	745	646
Germany_MS	19	32.25	36	5	4	3
Montreal_r06_MS	7	9	9.33	53	188	130
Nobel-EU_MS	15.75	23	25.5	13	19	15
Pdh_MS	7.89	8.75	9	87	170	159

capacity, another might work well — capacities are uncorrelated. In the positive correlation case, all edges incident to a node are positively correlated, so even though there may be many paths, they would tend to have difficulties at the same time. This reduces the value of alternative paths. But still some alternate paths are present, which is discussed later. The positive correlation cases compensate by installing more capacities on the edges (see sixth and seventh columns of Table 2). The latter is of course a function of how we defined edge failures – as a percentage of installed capacity. These effects can be seen in Figure 4, where the uncorrelated case has more edges than the positively correlated one and the positively correlated case generally has higher installed capacities on the edges. In the figure, solid (blue) edges are installed with the given capacities, dotted edges are not installed. The dark (red) nodes are the source nodes, the white ones transshipment nodes. The shaded (yellow) nodes are demand nodes. The same color scheme is followed throughout the paper.

Alternative paths

In the stochastic structures we observe the creation of alternative paths, even for the case of positively correlated edge failures. This is due to the fact that with alternative paths, the network increases the chance that demand is at least partly satisfied even when one of the paths fails or works at low capacity. We can see this aspect in Figure 5. Demand node 16 in the stochastic structure is served by three paths, one approaching via transshipment node 19, one via transshipment node 6 and finally one via a collection of other demand nodes (through node 17). In this way alternative paths help fulfill demand when one of the paths may be down or have low capacities. Even though this is less useful when failures are positively correlated, the alternative paths still provide some extra chance of reaching demand node 16.

When studying optimal designs under random demand (see Thapalia et al. (2010b)) we observed that consolidation was the major tool for hedging against uncertainty. By having several demand nodes share paths, one node could use the path when the others did not need it. With random edge capacities, this need does not emerge since demand is known.

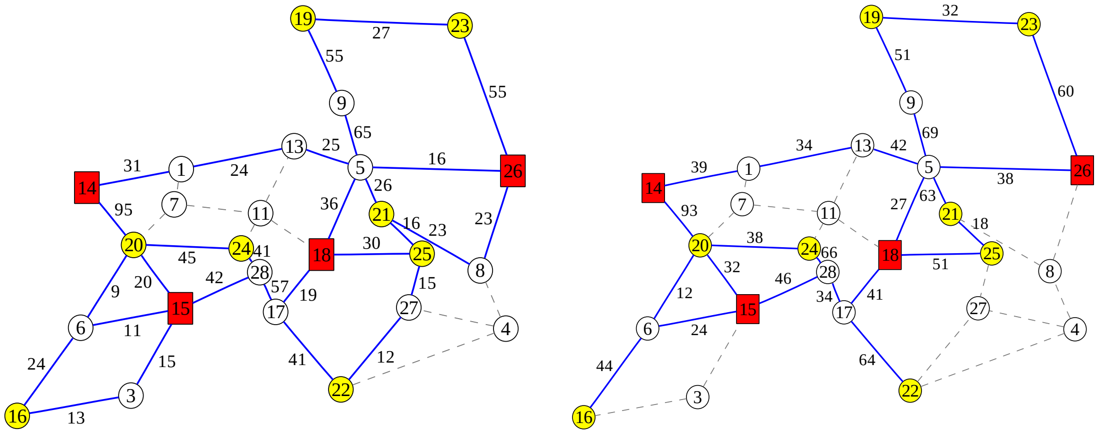

Figure 4: Network Density. Uncorrelated (left) and positively correlated (right) stochastic structure of Nobel-EU showing higher number of edges for the uncorrelated case and generally higher installed capacities on edges for the positively correlated case.

Instead hedging comes from having alternative paths, and of course, generally higher installed capacities since edges may fail. Consolidation-like phenomena are hence observed in some cases, but they come from the same phenomena as in deterministic cases: Paths to demand nodes share edges (even if the paths become slightly longer) so that fewer fixed setup costs need to be paid, and, of course, two shortest paths (including both variable and fixed setup costs) may simply happen to share edges.

Loops

The formation of loops is quite visible in networks designed for stochastic edge capacities. Loops are formed among the demand nodes, including or excluding the source nodes, or by joining the leaves of the trees. When we compare this with the networks for stochastic demand (Thapalia et al., 2010b), it is far more prominent here. The main reason for loop formation is the need to provide alternate paths to fulfill demand when some edges are (partly) down. Loops have the advantage that they can be used both ways. So somewhat high capacities (which characterizes the stochastic designs) combined with loops provide alternative paths to demand nodes. We can see this in Figures 5. Here loops are seen in the stochastic network design which is not in the deterministic structures.

Removing edges

It is observed from the test results that in almost all cases, the stochastic skeleton contains the deterministic one. The additional edges are providing flexibility to the network structure. But as we increase the fixed or variable setup costs compared to the base case, the additional edges which were seen in the stochastic skeletons disappear and finally very few are left. We can observe this in Figure 6. When fixed costs are increased, keeping the rest of the cost the same (left column figures), we see that edges or partial paths which were seen only in the stochastic design, but with rather low capacities, like 16-17, 18-10-21, or 3-27-19-16 disappear, making it similar to the deterministic design (top structure of Figure 5). Also we see this effect when we increase the variable setup costs (right column figures). This points

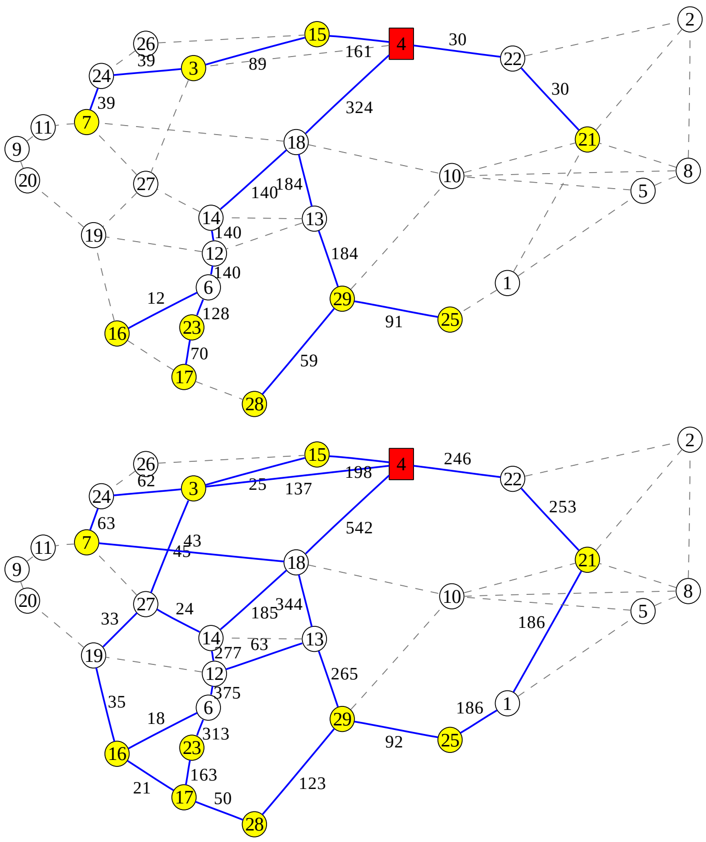

Figure 5: Alternate Paths. Deterministic (top) and stochastic structures(bottom) of Germany_SS_04 showing the alternative paths in the stochastic structure.

toward the suggestion that the deterministic structures are the base for the stochastic network structures. The reason behind this phenomenon is that as the fixed or variable setup costs increase, maintaining the same structure is no longer optimal given the cost of opening the edges and building the capacities compared to the rejection cost. Thus as the fixed or variable setup costs increase, it becomes profitable to remove the edges, which are not core to the basic *backbone*. The edges which are removed are the ones which are not seen in the deterministic structures and\or with lower capacities or the ones which are far from the source nodes.

5 Conclusion

We have seen that optimal stochastic designs for both the single- and multi-source cases differ from the deterministic ones in both skeletons and capacities. The flexibility, which gives the stochastic designs better expected performance, comes from a higher number of edges and higher installed capacities. With a higher number of edges there exist more paths to demand nodes and hence it also becomes easier to find alternative routes to the demand nodes in the case of edge failures. So while the sharing of paths is the main vehicle for hedging in the case of random demand, here it comes from providing alternative paths from sources to demand nodes.

The deterministic structures as such are not good in the stochastic setting as their expected

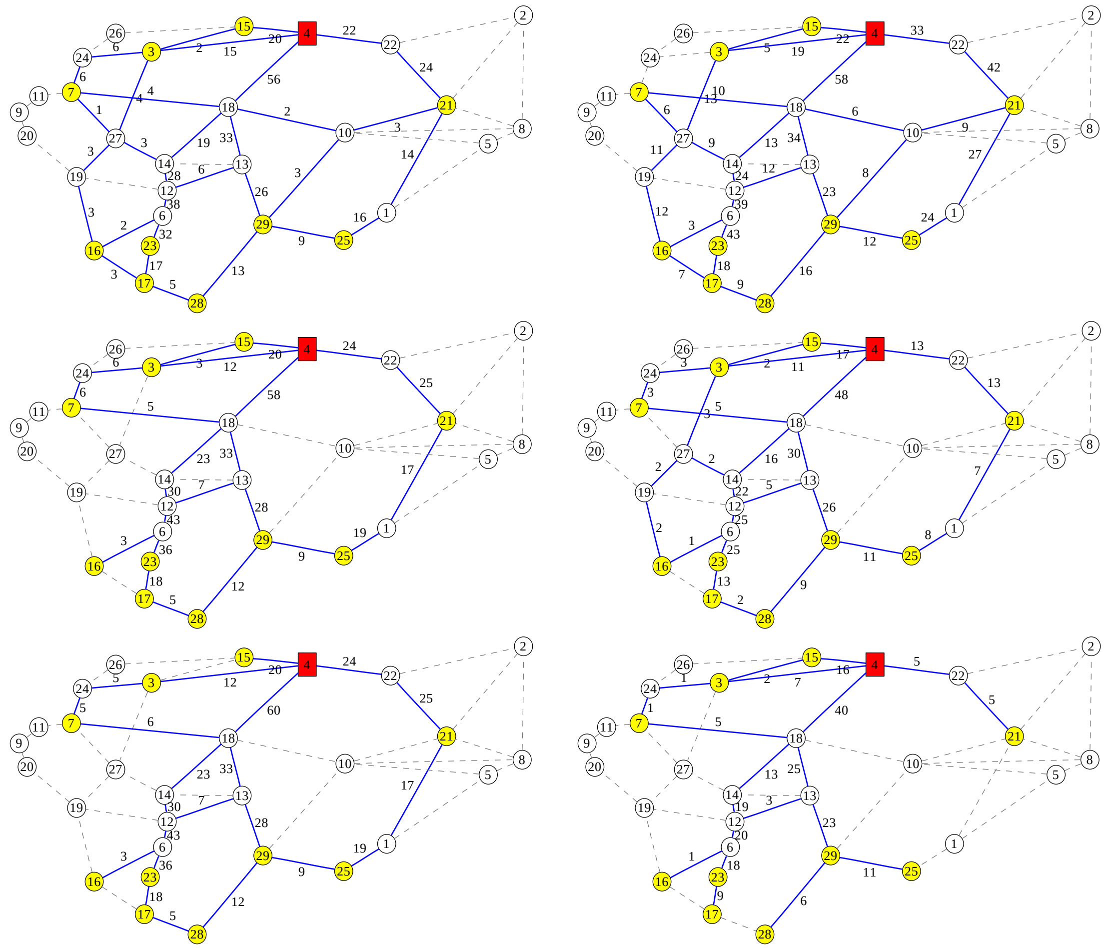

Figure 6: Disappearing Edges. Stochastic structures of Germany_SS with increasing fixed setup costs (left column) and with increasing variable setup costs (right column) showing that stochastic the stochastic designs eliminate edges to emerge as similar to deterministic ones.

performance is bad. But borrowing the skeleton from the deterministic structure is rather good. The reason is that the deterministic structure often forms a backbone in the stochastic one. With increased costs for adding edges or for adding capacities in the edges, the stochastic skeletons start to look more and more like the deterministic ones. This happens by shedding installed capacities and edges which are not observed in the deterministic structures. Thus it seems that for cases of this type, using a deterministic method to set the skeleton, and solving a stochastic *linear* program to set capacities is a very promising approach. Naturally, we cannot test this for larger cases, since we cannot solve the stochastic versions to optimality.

Using the deterministic skeleton is slightly better if based on average edge capacities rather than maximal ones. The reason is somewhat subtle: Using average rather than maximal edge capacities is equivalent to increasing variable setup costs. That reduces the importance of the fixed setup costs, generally leading to more edges being opened, and hence a better starting point for the stochastic linear program.

Correlations have important effects on the structure of the design. With uncorrelated edge failures, the stochastic designs have more edges than when edge failures are positively correlated. With positively correlated edge failures, the networks have higher installed capacities. The reason is simply that with positively correlated failures, all paths to a node tend to have difficulties at the same time, providing less hedging from multiple paths.

Loops are present in the stochastic networks due to a combination of two phenomena. The first is the one we observe for consolidation in the deterministic problem: avoid paying too many fixed setup costs. The second is the characteristics of a ring network. It provides two connections between any pair of nodes in the ring, and the ring can be used in both directions. For these reasons, loops are much more prominent here than with random demand.

So, network designs for stochastic edge capacities are fundamentally different from network designs for stochastic demand. With stochastic edge capacities there are more edges, more loops, and more installed capacities as compared to the design for stochastic demand. A major reason is that there is less consolidation. For stochastic edge capacities we only see consolidation of the type we see in deterministic designs, mostly caused by savings in the fixed setup costs. Instead, alternative connections become more important as a hedge against edges having reduced capacities. Skeletons generally do better here than with random demand as trees in the forests – typical for deterministic designs – no longer have the need to contact each other when randomness strikes, as each tree has enough supply.

Future work. We have now studies random demand and random arc capacities separately. A potential future project is to understand how they interact.

Acknowledgments

This project was supported in part by grant 171007/V30 from The Research Council of Norway. While working on this project, T.G. Crainic was the NSERC Industrial Research Chair on Logistics Management, ESG UQAM, and Adjunct Professor with the Department of Computer Science and Operations Research, Université de Montréal, and the Department of Economics and Business Administration, Molde University College, Norway. Partial funding for this project has been provided by the Natural Sciences and Engineering Council of Canada (NSERC), through its Industrial Research Chair and Discovery Grants programs.

References

April K. Andreas and J. Cole Smith. Decomposition algorithms for the design of a nonsimultaneous capacitated evacuation tree network. *Networks*, 53(2):91–103, 2009.

A. Balakrishnan, T.L. Magnanti, and P. Mirchandani. Designing hierarchical survivable networks. *Operations Research*, 46(1):116–136, 1998.

Michael O. Ball, Charles J. Colbourn, and J. Scott Provan. Network reliability. In M. O. Ball, T. L. Magnanti, C. L. Monma, and G. L. Nemhauser, editors, *Network Models*, volume 7 of *Handbooks in Operation Research & Management Science*, chapter 11. North-Holland, Amsterdam, 1995.

Jack Brimberg, Pierre Hansen, Keh-Wei Lih, Nenad Maldenović, and Michéle Breton. An oil pipeline design problem. *Oper. Res.*, 51(2):228–239, March-April 2003.

Anthony Che, Hai Yang, Hong K. Lo, and Wilson H. Tang. Capacity relaibility of a road network: an assessment methodology and numerical results. *Trans. Research. -B.*, 36: 225–252, 2002.

Anthony Chen, Hai Yang, Hong K. Lo, and Wilson H. Tang. A capacity related relaibility for transportation networks. *Journal of Advanced Transportation*, 33(2):183–200, 1999.

R.K.-M. Cheung and W.B. Powell. An algorithm for multistage dynamic networks with random arc capacities, with an application to dynamic fleet management. *Operations Research*, 44(6):951–963, 1996.

G. Chung, K. Lansey, and G. Bayraksan. Reliable water supply system design under uncertainty. *Environmental Modelling & Software*, 24:449–462, 2009.

L. W. Clarke and G. Anandalingam. A bootstrap heuristic for designing minimum cost survivable networks. *Computers Ops Res.*, 22(9):921–934, 1995.

K. Cormican, D.P. Morton, and R.K. Wood. Stochastic network interdiction. *Operations Research*, 46:184–197, 1998.

Francesco di Pierro, Soon-Thaim Khu, Dragan Savić, and Luigi Berardi. Efficient multi-objective optimal design of water distribution networks on a budget of simulations using hybrid algorithms. *Environmental Modelling & Software*, 24:202–213, 2009.

Emden R. Gansner and Stephen C. North. An open graph visualization system and its applications to software engineering. *Software: Practice and Experience*, 30(11):1203–1233, 2000. doi: 10.1002/1097-024X(200009)30:11⟨1203::AID-SPE338⟩3.0.CO;2-N.

J. L. Higle and S. W. Wallace. Sensitivity analysis and uncertainty in linear programming. *Interfaces*, 33:53–60, 2003.

K. Høyland, M. Kaut, and S.W. Wallace. A heuristic for moment-matching scenario generation. *Computational Optimization and Applications*, 24(2–3):169–185, 2003.

ILOG CPLEX 9.0 User's Manual. ILOG, S. A., 2003.

Michal Kaut and Stein W. Wallace. Evaluation of scenario-generation methods for stochastic programming. *Pacific Journal of Optimization*, 3(2):257–271, 2007.

Anatoly Lisnianski and Gregory Levitin. *Multi-state system reliability-:assessment, optimization and applications*, volume 6. World Scienetific Publishing Co. Pte. Ltd., 2003.

Changzheng Liu, Yueyue Fan, and Fernando Ordonez. A two-stage sytochastic programming model for transportation network protection. *Computers & Operations Research*, 36:1582–1590, 2009.

Alexander Martin, Markus Möller, and Susanne Moritz. Mixed integer models for the stationary case of gas network optimization. *Math Programming*, 105(2-3):563–582, Feb 2006.

Youngyut Meepetchdee and Nilay Shah. Logistical network design with robustness and complexity considerations. *International Journal of Physical Distribution & Logistics Management*, 37(3):201–222, 2007.

Kjetil T. Midthun, Mette Bjørndal, and Asgeir Tomasgard. Modelling optimal economic dispatch and system effects in natural gas networks. *The Energy Journal*, 30(4):155, 2009.

Idel Montalvo, Joaquin Izquíerdo, Rafael Pérez, and Michael M. Tung. Particle swarm optimization applied to the design of water supply systems. *Computers and Mathematics with Applications*, 56:769–776, 2008.

Young-Soo. Myung, Kim Hyun-joon., and Tch Dong-wan. Design of communication networks with survivability constraints. *Management Science*, 45(2):238–252, 1999.

S. Orlowski, M. Pióro, A. Tomaszewski, and R. Wessäly. SNDlib 1.0 – Survivable Network Design library. *Networks*, 55(3):276–286, 2010. doi: 10.1002/net.20371.

José Emmanuel Ramirez-Marquez and Claudio M. Rocco S. Stochastic network interdiction optimization via capacitated network reliability modeling and probabilistic solution discovery. *Reliability Engineering and System Safety*, 94:913–921, 2009.

B. Rothfarb, H. Frank, D. M. Rosenbaum, K. Steiglitz, and D. J. Kleitman. Optimal design of offshore natural-gas pipeline system. *Operations Research*, 18:992–1020, 1970.

J. Cole Smith, Churlzu Lim, and Fransisca Sudargho. Survivable network design under optimal and heuristic interdiction scenarios. *Journal of Global Optimization*, 38:181–199, 2007.

J.W. Suurballe and Robert E. Tarjan. A quick method for finding shortest pairs of paths. *Networks*, 14:325–336, 1984.

Biju K. Thapalia. *Stochastic Single-Commodity Network Design*. Phd thesis, Molde University College, Molde, Norway, 2010.

Biju K. Thapalia, Teodor G. Crainic, Michal Kaut, and Stein W. Wallace. Single source single-commodity stochastic network design. In *Stochastic Single-Commodity Network Design* Thapalia (2010), pages 33–55.

Biju K. Thapalia, Teodor G. Crainic, Michal Kaut, and Stein W. Wallace. Single-commodity stochastic network design with multiple sources and sinks. In *Stochastic Single-Commodity Network Design* Thapalia (2010), pages 61–88.

S. W. Wallace. A piecewise linear upper bound on the network recourse function. *Mathematical Programming*, 38:133–146, 1987.

Stein W. Wallace. Decision making under uncertainty: Is sensitivity analysis of any use? *Operations Research*, 48(1):20–25, 2000.

Daniel De Wolf and Yves Smeers. Optimal dimensioning of pipe networks with application to gas transmission networks. *Oper. Res.*, 44(4):596, 1996.

A Results of the numerical tests

This appendix provides detailed results from the tests in Section 3.1.

Table 3: The ratios for the single source cases corresponding to Figures 1 and 2, split by correlation structure.

	Full capacity				**Mean value capacity**			
	comp. A		**comp. B**		**comp. A**		**comp. B**	
Test Name	$\rho = \mathbf{0}$	$\rho \geq \mathbf{0}$	$\rho = \mathbf{0}$	$\rho \geq \mathbf{0}$	$\rho = \mathbf{0}$	$\rho \geq \mathbf{0}$	$\rho = \mathbf{0}$	$\rho \geq \mathbf{0}$
Germany_SS_04	1.76	1.62	1.20	1.11	1.39	1.26	1.18	1.09
Germany_SS_10	1.71	1.57	1.17	1.08	1.34	1.22	1.17	1.08
Germany_SS_13	2.12	1.92	1.20	1.10	1.57	1.39	1.20	1.10
Germany_SS_27	1.99	1.81	1.21	1.10	1.55	1.37	1.21	1.10
France_SS_06	6.31	5.46	1.27	1.13	3.15	3.23	1.27	1.13
France_SS_10	4.82	4.41	1.56	1.43	2.74	2.66	1.39	1.28
France_SS_13	5.04	4.42	1.31	1.18	3.44	2.74	1.31	1.18
France_SS_16	5.91	5.30	1.22	1.11	3.98	2.91	1.22	1.11
Montreal_r06_SS_01	12.29	11.45	1.24	1.17	6.17	5.71	1.24	1.17
Montreal_r06_SS_03	11.16	9.35	1.30	1.10	5.92	4.89	1.30	1.10
Montreal_r06_SS_04	12.62	11.17	1.33	1.15	6.61	5.69	1.33	1.15
Montreal_r06_SS_08	13.04	10.84	1.32	1.14	6.85	5.65	1.32	1.14
Pdh_01_SS	1.97	1.81	1.10	1.03	1.49	1.33	1.10	1.03
Pdh_02_SS	2.48	2.36	1.07	1.03	1.67	1.56	1.07	1.03
Pdh_04_SS	2.36	2.25	1.04	1.01	1.62	1.51	1.04	1.01
Pdh_08_SS	2.15	2.00	1.13	1.07	1.53	1.40	1.13	1.07
Nobel_EU_SS_04	3.88	3.56	1.42	1.32	2.81	2.54	1.39	1.29
Nobel_EU_SS_05	4.86	4.24	1.34	1.19	3.28	2.79	1.34	1.19
Nobel_EU_SS_15	4.02	3.53	1.33	1.18	2.87	2.45	1.37	1.21
Nobel_EU_SS_18	4.83	4.27	1.31	1.18	3.14	2.74	1.31	1.18

Table 4: The ratios for the multi-source cases corresponding to Figures 1 and 2, split by correlation structure.

	Full capacity				**Mean value capacity**			
	comp. A		**comp. B**		**comp. A**		**comp. B**	
Test Name	$\rho = \mathbf{0}$	$\rho \geq \mathbf{0}$	$\rho = \mathbf{0}$	$\rho \geq \mathbf{0}$	$\rho = \mathbf{0}$	$\rho \geq \mathbf{0}$	$\rho = \mathbf{0}$	$\rho \geq \mathbf{0}$
Germany_1_MS	2.05	1.89	1.14	1.07	1.54	1.40	1.14	1.07
Germany_2_MS	2.23	2.00	1.22	1.12	1.63	1.43	1.22	1.12
Germany_3_MS	2.10	1.95	1.17	1.09	1.52	1.38	1.17	1.09
Germany_4_MS	2.15	1.96	1.12	1.04	1.57	1.41	1.12	1.04
France_MS_1	6.57	5.98	1.17	1.08	3.56	3.20	1.17	1.08
France_MS_2	5.83	5.35	1.38	1.26	3.59	3.30	1.38	1.26
France_MS_3	6.33	5.72	1.47	1.35	3.48	3.10	1.22	1.13
Montreal_r06_MS_1	15.33	14.78	1.09	1.04	6.90	6.76	1.09	1.04
Montreal_r06_MS_2	14.29	13.23	1.20	1.10	6.88	6.36	1.20	1.10
Montreal_r06_MS_3	15.31	14.04	1.14	1.02	7.32	6.69	1.14	1.02
Pdh_1_MS	2.50	2.42	1.02	1.00	1.60	1.52	1.02	1.00
Pdh_2_MS	2.43	2.30	1.03	1.00	1.58	1.48	1.03	1.00
Pdh_3_MS	2.65	2.58	1.02	1.00	1.65	1.59	1.02	1.00
Pdh_4_MS	2.65	2.49	1.08	1.03	1.74	1.60	1.08	1.03
Nobel_EU_MS_plus_01	4.50	4.21	1.28	1.18	2.88	2.70	1.17	1.10
Nobel_EU_MS_plus_02	4.61	4.32	1.20	1.13	2.72	2.54	1.20	1.13
Nobel_EU_MS_plus_03	5.05	4.67	1.26	1.17	2.94	2.70	1.26	1.17
Nobel_EU_MS_plus_04	4.89	4.52	1.23	1.13	3.08	2.77	1.23	1.13

Table 5: The ratios for Comparison B corresponding to Figure 3, for full capacities and mean capacities at different variable setup costs.

	Single source		**Multi-source**	
Var. setup cost	**Full Cap.**	**Mean Cap.**	**Full Cap.**	**Mean Cap.**
0.33	1.18	1.18	1.2	1.17
0.66	1.19	1.19	1.15	1.16
1	1.19	1.18	1.12	1.1
1.33	1.16	1.16	1.11	1.1
1.66	1.16	1.15	1.1	1.09
2	1.14	1.14	1.09	1.09