RUTCOR RESEARCH REPORT

Single commodity stochastic network design under probabilistic constraint with discrete random variables

András Prékopa [a] Merve Unuvar [b]

RRR 9-2012, February 2012

RUTCOR
Rutgers Center for
Operations Research
Rutgers University
640 Bartholomew Road
Piscataway, New Jersey
08854-8003
Telephone: 732-445-3804
Telefax: 732-445-5472
Email: rrr@rutcor.rutgers.edu
http://rutcor.rutgers.edu/∼rrr

[a] RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway, NJ, 08854-8003; prekopa@rutcor.rutgers.edu
[b] RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway, NJ, 08854-8003; unuvar@rci.rutgers.edu

Rutcor Research Report
RRR 9-2012, February 2012

Single commodity stochastic network design under probabilistic constraint with discrete random variables

András Prékopa Merve Unuvar

Abstract.
Single commodity networks are considered, where demands at the nodes are random. The problem is to find minimum cost optimal capacities at the nodes and arcs subject to the constraint that all demands should be met on a prescribed probability level (reliability constraint) and some constraints on the capacities should be satisfied. The reliability constraint is formulated in terms of the Gale–Hoffman feasibility inequalities but their number is reduced by elimination technique. The concept of a p-efficient point is used in a smart way to convert and then relax the problem into an LP. Two solution techniques are presented depending on if all p-efficient points are known or are simultaneously generated with the solution of the LP. The joint distribution of the demands is used to obtain the p-efficient points for all non-eliminated stochastic inequalities and the solution of a multiple choice knapsack problem is used to generate new p-efficient points. The model can be applied to planning in interconnected power systems, flood control networks, design of shelter and road capacitates in evacuation, parking lot capacities, financial networks, etc. Numerical examples are presented.

Acknowledgements: This research was supported by National Science Foundation (Grant CMMI-0856663, 2009-2012)

1 Introduction

Stochastic network design problems are in the center of interest. Many of them concern multi-commodity networks but important problems can be formulated in connection with cooperating power systems, water resources, road traffic, home security (evacuation), finance, etc., where the network is of a single-commodity type.

In a network design problem the topology of the network is frequently given and we are looking for optimal node and arc capacities which are decision variables in the problem. Given the initial investments, i.e., the capacities, the network is then operated in time which may be considered continuous or is subdivided into discrete periods. The model may focus on the investments: we may want to find only the optimal capacities and disregard the operational cost. In that case, we are dealing with a static model construction whereas the model is dynamic if operational policies are determined for the subsequent periods and the operational cost is also taken into account in the investment problem.

In this paper we propose a static model, under the assumption that the system is influenced by randomness and we have to decide in the face of uncertainty. Thus, we are dealing with stochastic networks. Our main concern is the handling of reliability, by the use of probabilistic constraint that we include in the model. Research is underway to formulate and solve the dynamic type network design problems, where reliability is taken care of in the same way as it is worked out in this paper.

A static model typically has the decision-observation scheme:

decision on a vector x

observation of the value of a random vector ξ.

We will distinguish between local demands and system demands. The local demands are the ζ_i, where i designate a node. For example, in case of an interconnected power system ζ_i may represent the local demand for power in area i and x_i the local generating capacity. However, x_i may be reduced and the available generating capacity is $x_i - \eta_i$. The system demand is $\eta_i + \zeta_i - x_i = \xi_i - x_i$, by definition. Thus, ξ_i represents the local demand plus the deficiency in the generation capacity. In what follows we will simply call ξ_i equal to the local demand. Deficiency may exist in the arc capacities as we discuss it later.

Our models are of stochastic programming type. In particular, we use the programming under probabilistic constraint model, also called chance constrained model.

The first paper on programming under probabilistic constraints was published by Charnes, Cooper and Symonds (1958). However, they cannot take the whole credit for initiation because they proposed the use of individual chance constraints which is rarely legitimate from the point of view of probability theory and statistics. Miller and Wagner (1965) have used joint probabilistic constraint but only in case of independent random variables when the convexity and algorithmic problems are not difficult to handle. Models, theories and algorithms for joint probabilistic constraints, where random variables may also be stochastically dependent, were introduced and developed by Prékopa in a series of publications (1970, 1971, 1973, 1980, 1995, etc.)

Since many papers, especially in engineering literature, do not distinguish between individual and joint probabilistic constraints, it is not superfluous to clarify the difference. As an example, consider an interconnected power system consisting of two areas and a transmission line (Figure 1).

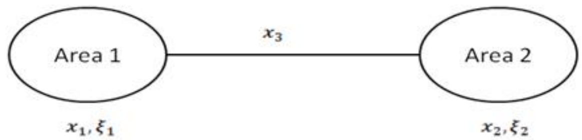

Figure 1: Interconnected power system consisting of two areas. The generating capacities are x_1, x_2, the local demands are ξ_1, ξ_2, respectively, and x_3 is the capacity of the transmission line.

Looking at the system at one designated time, we can formulate linear inequalities in terms of x_1, x_2, x_3, ξ_1, ξ_2 that provide us with necessary and sufficient condition that all demands can be satisfied. These inequalities are the following:

$$\begin{array}{ccccccccc} \xi_1 & & & \leq & x_1 & & & + & x_3 \\ & \xi_2 & & \leq & & & x_2 & + & x_3 \\ & \xi_2 & + \ \xi_3 & \leq & x_1 & + & x_2. & & \end{array} \qquad (1)$$

If ξ_1 and ξ_2 are random variables, then the power system reliability is the probability that all three inequalities are simultaneously satisfied. The individual probabilistic constraints

$$\begin{aligned} P(\xi_1 \leq x_1 + x_3) &\geq p_1 \\ P(\xi_2 \leq x_2 + x_3) &\geq p_2 \\ P(\xi_1 + \xi_2 \leq x_1 + x_2) &\geq p_3 \end{aligned} \qquad (2)$$

have no interpretation for system reliability. Only a crude lower bound for the joint probability of (1), based on (2), can be derived which is obtained by Boole's inequality and is equal to: $p_1 + p_2 + p_3 - 2$. However, it is possible to efficiently use joint probabilistic constraints in practical problem solutions (see Prékopa, 1995) and this paper provides us with further examples in this respect.

A stochastic network design problem with probabilistic constraint was introduced in Prékopa (1980). It is a two-stage, dynamic type model but no solution method was proposed. In this paper we look at a related static problem and propose an elegant and efficient solution method for it. We will use the method of p-efficient points (see Prékopa 1990 a, Prékopa et al. 1998, Dentcheva et al. 2000). We will also use preprocessing and reliability results that have been presented by Prékopa, Boros (1991) and Wallace, Wets (1993). See also the presentation of the network reliability calculation and network design model construction in Prékopa (1995). Recently, the method of p-efficient points captured great interest in the civil engineering literature, see, e.g., Yazici, Ozbay (2007). We also mention recent papers by Thapalia, Crainic, Kaut, Wallace (2010) and Thapalia, Kaut, Wallace, Crainic (2010), where the reader can find ideas in connection with single commodity stochastic network design, even though their main interest of these authors is different from ours.

The organization of the paper is as follows.

In section 2 we give definitions for networks, network flows and demands. Then we present the Gale–Hoffman theorem and its refinements by Prékopa, Boros (1991) and Wallace, Wets (1993) for feasibility of a demand function. Then we describe how we can eliminate redundant inequalities from the feasibility condition. In section 3 we present examples for interconnected power system, water reservoir system and road network used for evacuation. In section 4 the notion of a p-efficient point is recalled and an important theorem is proved that makes possible to write up the joint probabilistic constraint for the demand feasibility inequalities, using the p-efficient points of a much smaller system of inequalities. In section 5 we formulate and in section 6 solve a static stochastic network design problem. Section 7 is devoted to the case where the demands corresponding to the nodes are independent. In this case the p-efficient points are generated by solutions of multiple choice knapsack problems. In section 8 we summarize the solution algorithm for the static problem and in section 9 we present two numerical examples for an 8-node network. Finally, in section 10 we summarize our conclusions and mention further research directions.

1.1 Network Concepts and Feasibility Conditions

Below we give the definitions for the network and network flow, suitable for our problem.

Definition A network $[N, y]$ is a pair of a finite set of nodes N and a capacity function $y(i, j)$ on the arcs $(i, j) \in N \times N$ assumed to have nonnegative values or $+\infty$.

While we speak about network flows, we typically have source, terminal and intermediate nodes. Sometimes we know exactly which node is of which type but sometimes the nodes randomly become sources or terminals and we cannot categorize them in advance. This is the case, for example, in interconnected power systems, where some of the nodes in the network represent areas that may have surplus generating capacities or may need assistance from other nodes to meet the demand, and it happens randomly.

If we know in advance the types of the nodes, then we can write up equations/inequalities for each type to set up the constraints that the flow values have to satisfy. If, however, we do not know the types in advance, then it is more convenient to define the flow differently, e.g., the way Gale (1957) and Ford, Fulkerson (1962) have done it. This definition is presented below.

Definition A flow or feasible flow in the network is a function $f(i, j), (i, j) \in N \times N$ such that

$$\begin{aligned} f(i,j) + f(j,i) &= 0 \\ f(i,j) \le y(i,j) &\text{ for all } (i,j) \in N \times N. \end{aligned} \tag{3}$$

The first line in condition (3) expresses a general convention that the flow from j to i is the negative of the flow from i to j. In view of the equalites in (3) there is no need to write up any further flow conservation equations because they are satisfied in a trivial way.

1.2 Feasible Demands

If the network flow is defined as in Definition 1.1, then we have to introduce additionally, the notion of a system demand and a feasible system demand.

In what follows, we use the following notations:

$$f(A,B) = \sum_{i\in A,\ j\in B} f(i,j)$$
$$y(A,B) = \sum_{i\in A,\ j\in B} y(i,j) \quad \text{for } A,B \subset N,\ \ AB = \emptyset$$

Definition A system demand is a function $d(j)$, $j \in N$. In what follows we use the notation $d(A) = \sum_{i\in A} d(i)$ for $A \subset N$. A system demand is said to be feasible if there exists a flow f satisfying Definition 1.1 and the relations:

$$f(N,j) \geq d(j),\ \ j \in N. \tag{4}$$

Relations (3), (4) can be thought of as a system of homogenous linear inequalities, if we take $f(i,j)$, $y(i,j)$, $d(j)$, $j \in N$, $(i,j) \in N \times N$ as variables. An important question is: what are the conditions on $d(j)$, $y(i,j)$, $j \in N$, $(i,j) \in N \times N$ that ensure the existence of a feasible flow such that the inequalities (4) are satisfied? The question was answered by Gale (1957) and Hoffman (1960) and their result is contained in the theorem below.

Theorem 1.1. *The system demand $d(i)$, $i \in N$ is feasible iff the following inequalities hold:*

$$d(S) \leq y(\bar{S},S),\ \ S \subset N, \quad \textit{where } \bar{S} = N \setminus S. \tag{5}$$

In what follows we call relations (5) Gale–Hoffman inequalities.

Let $|N| = n$. The number of Gale–Hoffman inequalities is $2^n - 1$, the case where $S = \emptyset$ being trivial. Since it is a large number, also if n is relatively small, we are looking for reduction of the Gale–Hoffman inequalities by eliminating redundant ones. A theorem, first proved by Prékopa and Boros (1991) and later by Wallace and Wets (1993), states the following:

Theorem 1.2. *The inequality* (5) *is redundant among the Gale–Hoffman inequalities if and only if at least one of the graphs $G(S), G(\bar{S})$ is not connected. In that case the inequality $d(S) \leq y(\bar{S},S)$ is the sum of other Gale–Hoffman inequalities.*

Remark 1.3. *Prékopa and Boros (1991) stated only the sufficiency part of the theorem but their proof contains also the proof of the necessary part.*

Prékopa and Boros (1991) worked out a variety of elimination procedures in connection with the Gale–Hoffman inequalities so that at the end only the non-redundant ones remain. In their paper, however, only the system demands are variables while the arc capacities are assumed to be constant. Since we look at the arc capacities also as variables, we have to slightly rework the elimination procedure. Below, we present the new version.

1.3 Elimination by Network Topology

Based on Theorem 1.2, in this procedure we eliminate those inequalities in (5) which are sums of others. We subsequently enumerate the sets $S \subset N$ according to their cardinalities, and look for sets $S, \bar{S}$ such that at least one of $G(S), G(\bar{S})$ is not connected. Then, eliminate the corresponding inequality among those in (5). In what follows we assume that there are known lower and upper bounds on the variables $d(i), y(i,j)$:

$$\begin{aligned} l(i) \le d(i) \le u(i), &\qquad i \in N \\ l(i,j) \le d(i,j) \le u(i,j), &\qquad (i,j) \in N \times N, \end{aligned}$$

where

$$l(i), l(i,j) \in R \cup \{-\infty\}, u(i), u(i,j) \in R \cup \{+\infty\}.$$

We define $l(A)$, $u(A)$, $A \subset N$, $l(A,B)$, $u(A,B)$, where, $A, B \subset N$, $AB = \emptyset$, in a similar way as we have defined $d(A)$, $f(A,B)$, $y(A,B)$.

1.4 Elimination by Upper Bounds on $d(S)$ and Lower Bounds on $y(S, \bar{S})$

If for an S we have the inequality $u(S) \le l(S, \bar{S})$, then clearly the Gale–Hoffman inequality $d(S) \le y(S, \bar{S})$ is redundant.

1.5 Elimination by Lower Bounds on the Demands and Lower and Upper Bounds on the Arc Capacities

If $S \subset N$ and we have the inequality $y(S, \bar{S}) \ge d(S)$, further, $T \subset S$ and we have the inequality,

$$l(T, \bar{T}) - l(T) \ge u(S) - l(S, \bar{S}), \tag{6}$$

then the inequality,

$$y(\bar{T}, T) \ge d(T)$$

is redundant. In fact, if we subtract $l(S)$ on both sides of the inequality $y(S,\bar{S}) \geq d(S)$, we obtain:

$$y(S,\bar{S}) - l(S) \geq d(S) - l(S). \tag{7}$$

On the other hand, relation $T \subset S$ implies that:

$$d(S) - l(S) \geq d(T) - l(T). \tag{8}$$

Using (6), (7), and (8), we derive

$$\begin{gathered} y(\bar{T},T) - l(T) \geq l(\bar{T},T) - l(T) \\ \geq u(S,\bar{S}) - l(S) \geq y(S,\bar{S}) - l(S) \\ \geq d(S) - l(S) \geq d(T) - l(T) \end{gathered}$$

which implies $y(T,\bar{T}) \geq d(T)$.

The elimination works in such a way that we start by $S = N$, eliminate all inequalities corresponding to $T \subset S$ for which (6) is satisfied, then decrease the cardinality of S, etc.

1.6 Elimination by Linear Programming

Consider the inequalities that have not been eliminated. Let S_0 be one of them and $S_1,\dots,S_m$ be the remaining ones. Then we formulate the LP:

$$\begin{gathered} \max\left\{d(S_0) - y(\bar{S}_0,S)\right\} \\ \text{subject to} \\ d(S_i) - y(\bar{S}_i,S) \leq 0,\ \ i=1,\dots,m \\ l(i) \leq d(i) \leq u(i),\ \ i \in N \\ l(i,j) \leq y(i.j) \leq u(i,j),\ \ (i,j) \in N \times N. \end{gathered} \tag{9}$$

The inequality $d(S_0) - y(\bar{S}_0,S) \leq 0$ is redundant if and only if the optimum value of problem (9) is nonpositive. Problem (9) takes a more convenient form if we subtract the lower bound from each variable. Let

$$\begin{aligned} x(i) &= d(i) - l(i),\ \ i \in N \\ x(S_i) &= d(S_i) - l(S_i),\ \ i=1,\dots,m \\ x(i,j) &= y(i,j) - l(i,j),\ \ (i,j) \in N \times N \\ x(\bar{S}_i,S_i) &= y(\bar{S}_i,S_i) - l(\bar{S}_i,S_i),\ \ i=1,\dots,m. \end{aligned}$$

Then problem (9) takes the form:

$$\begin{gathered} \max\left\{x(S_0) - x(\bar{S}_0,S) + l(S_0) - l(\bar{S}_0,S)\right\} \\ \text{subject to} \\ x(S_i) - x(\bar{S}_i,S_i) \leq l(\bar{S}_i,S_i) - l(S_i),\ \ i=1,\dots,S \\ 0 \leq x(i) \leq u(i) - l(i),\ \ i \in N \\ 0 \leq x(i.j) \leq u(i,j) - l(i,j),\ \ (i,j) \in N \times N. \end{gathered} \tag{10}$$

If we remove the constant term $l(S_0) - l(\bar{S}_0 - S_0)$ from the objective function, then we can state that the inequality $d(S_0) - y(\bar{S}_0 - S_0) \leq 0$ is redundant if the optimum value of problem (10) is smaller than or equal to $l(S_0) - l(\bar{S}_0 - S_0)$.

In problem (10) we may have too many constraints; therefore it may be more convenient to work with the dual. Let $z(S_i)$, $w(i)$, $w(i,j)$ be the dual variables corresponding to the constraints involving $x(S_i)$, $x(i)$, $x(i,j)$, respectively. Then the dual of problem (10) can be written as follows:

$$\min \left\{ \sum_{i=1}^{m} (l(\bar{S}_i, S_i) - l(S_i)) z(S_i) + \sum_{i \in N} (u(i) - l(i)) w(i) + \sum_{(i,j) \in NxN} (u(i,j) - l(i,j)) w(i,j) \right\}$$

$$\begin{aligned} &\text{subject to} \\ w(i) + \sum_{j: S_j \in i} z(S_j) &\geq \ 1, i \in S_0 \\ w(i,k) + \sum_{\bar{S}_j \in i, S_j \in k} z(S_j) &\geq -1, \ \ i \in \bar{S}_0, k \in S_0 \\ z(S_i) &\geq 0, \ \ i = 1, \ldots, m \\ w(i) &\geq 0, \ \ i \in N \\ w(i,k) &\geq 0, \ \ (i,k) \in N \times N. \end{aligned} \tag{11}$$

If the optimum value is smaller than or equal to $l(S_0) - l(\bar{S}_0 - S_0)$, then the inequality $d(S_0) - y(\bar{S}_0 - S_0) \leq 0$ is redundant.

Note that we do not need to solve optimally problem (11). In fact, if in the course of the optimization procedure we find that the current objective function value is less than or equal to $l(S_0) - l(\bar{S}_0 - S_0)$, then we may stop and declare that $d(S_0) - y(\bar{S}_0 - S_0) \leq 0$ is a redundant inequality. We may also simply try to find feasible solution to the constraints of problem (11) supplemented by the additional constraint that the objective function is less than or equal to $l(\bar{S}_0 - S_0) - l(S_0)$. In what follows the local demands $\xi_1, \ldots, \xi_n$ will be assumed to be random variables and the system demand will be the function $d(i) = \xi_i - x_i, \ i \in N$.

2 Examples for Networks

In this section we present three examples. The topology of the first one is taken from Prékopa and Boros (1991) where we have in mind power networks. The second one is a flood control reservoir system, and the third one is a road network intended to be used in evacuation. In these examples and in further parts of the paper we use the notation x_i for node capacities and y_{ij} for arc capacities.

Examle 2.1. *We look at the 8-node network in the mentioned paper, where the network topology is depicted in Figure 2. It may represent an interconnected power system, where the nodes are the areas and the arcs the transmission system. At the nodes we have x_i generating capacities and on the arcs $y_{ij} = y_{ji}$ transmission capacities. At the nodes there are ξ_i random local demands and, by the use of them we define the system demand function $d(i) = \xi_i - x_i$, $1 \leq i \leq 8$. At each node we have both power generation capacity and demand. There may also be random deficiencies in the generation capacities but we have assumed that they are* *already combined with the demands. Flows on the arcs can take place in both directions. The arc capacities are the same in both directions, on each arc. In the paper by Prékopa and Boros the arc capacities are assumed to be constant. In this illustration we keep this assumption and indicate the numerical values of the arc capacities in Figure 2.*

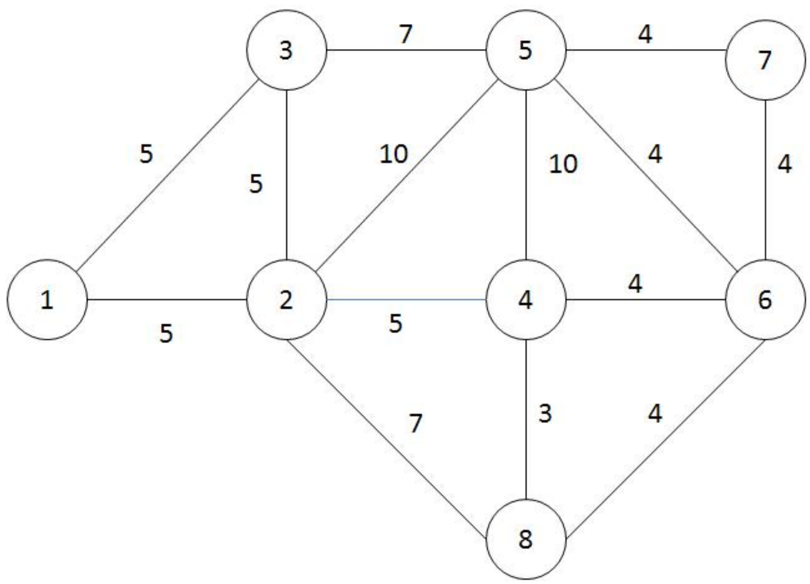

Figure 2: Eight-node network with arc capacities.

Examle 2.2. *The next example is a flood control reservoir system depicted in Figure 3. The system demand function is* $d(i) = \xi_i - x_i$, $i = 1, 2, 3, 4, 5$ *but at each node either* $\xi_i = 0$ *or* $x_i = 0$ *or both of them are 0. In fact, in our case* $\xi_2 = \xi_3 = \xi_4 = 0$ *and* $x_1 = x_3 = x_5 = 0$. *The demand now is not for water but for freeboard to retain. The water is coming from nodes 1 and 5, the arc capacities are* y_1, y_2, y_3, y_5 *in the direction indicated in Figure 3 and are* 0 *in the opposite directions. The Gale–Hoffman inequalities are presented below. Since the number of nodes is* 5, *the number of Gale–Hoffman inequalities is* $2^5 - 1 = 31$.

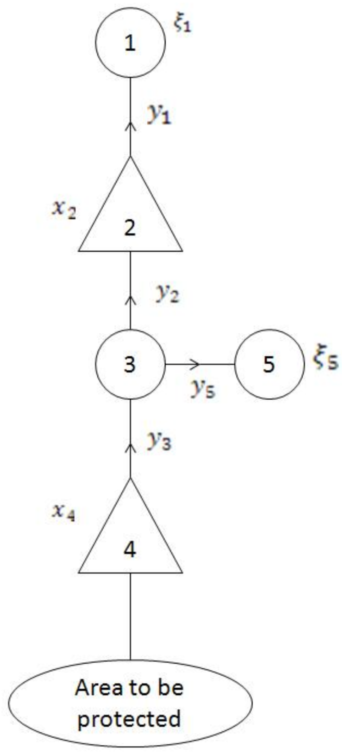

Figure 3: Flood control reservoir system.

1. $\bar{S} = \emptyset$, trivial
2. $\bar{S} = N,\ \xi_1 + \xi_5 \le x_2 + x_4$
3. $\bar{S} = 1,\ \xi_1 \le y_1$
4. $\bar{S} = 2,3,4,5\ ,\xi_5 - x_2 - x_4 \le 0$
5. $\bar{S} = 2,\ -x_2 \le y_2$
6. $\bar{S} = 1,3,4,5,\ \xi_1 + \xi_5 - x_4 \le y_1$
7. $\bar{S} = 3,\ 0 \le y_3$
8. $\bar{S} = 1,2,4,5,\ \xi_1 + \xi_5 - x_2 - x_4 \le y_2 + y_5$
9. $\bar{S} = 4,\ -x_4 \le 0$
10. $\bar{S} = 1,2,3,5,\ \xi_1 + \xi_5 - x_2 \le y_3$
11. $\bar{S} = 5,\ \xi_5 \le y_5$
12. $\bar{S} = 1,2,3,4,\ \xi_1 - x_2 - x_4 \le 0$
13. $\bar{S} = 1,2,\ \xi_1 - x_2 \le y_2$
14. $\bar{S} = 3,4,5,\ \xi_5 - x_4 \le 0$
15. $\bar{S} = 1,3,\ \xi_1 \le y_1 + y_3$
16. $\bar{S} = 2,4,5,\ \xi_5 - x_2 - x_4 \le y_2 + y_5$

$$
\begin{aligned}
17.\ \bar{S} &= 1,4,\ \xi_1 - x_4 \le y_1 \\
18.\ \bar{S} &= 2,3,5,\ \xi_5 - x_2 \le y_2 + y_5 \\
19.\ \bar{S} &= 1,5,\ \xi_1 + \xi_5 \le y_1 + y_5 \\
20.\ \bar{S} &= 2,3,4,\ -x_2 - x_4 \le 0 \\
21.\ \bar{S} &= 2,3,\ -x_2 \le y_3 \\
22.\ \bar{S} &= 1,4,5,\ \xi_1 + \xi_5 - x_4 \le y_1 + y_5 \\
23.\ \bar{S} &= 2,4,\ -x_2 - x_4 \le y_2 \\
24.\ \bar{S} &= 1,3,5,\ \xi_1 + \xi_5 \le y_1 + y_3 \\
25.\ \bar{S} &= 2,5,\ \xi_5 - x_2 \le y_2 + y_5 \\
26.\ \bar{S} &= 3,4,\ -x_4 \le 0 \\
27.\ \bar{S} &= 1,2,5,\ \xi_1 + \xi_5 - x_2 \le y_2 + y_5 \\
28.\ \bar{S} &= 3,5,\ \xi_5 \le y_3 \\
29.\ \bar{S} &= 1,2,4,\ \xi_1 - x_2 - x_4 \le y_2 \\
30.\ \bar{S} &= 4,5,\ \xi_5 - x_4 \le y_5 \\
31.\ \bar{S} &= 1,2,3,\ \xi_1 - x_2 \le y_3.
\end{aligned}
$$

The remaining inequalities after elimination by graph structure are:

$$
\begin{aligned}
2.&\ \xi_1 + \xi_5 \le x_2 + x_4 \\
3.&\ \xi_1 \le y_1 \\
10.&\ \xi_1 + \xi_5 \le y_3 + x_2 \\
11.&\ \xi_5 \le y_5 \\
13.&\ \xi_1 \le x_2 + y_2 \\
14.&\ \xi_5 \le x_4 \\
28.&\ \xi_5 \le y_3.
\end{aligned}
$$

A more compact form of the remaining inequalities is as follows:

$$
\begin{aligned}
\xi_1 &\le \min(y_1, x_2 + y_2) \\
\xi_5 &\le \min(y_3, y_5, x_4) \\
\xi_1 + \xi_5 &\le \min(x_2 + x_4, y_3 + x_2).
\end{aligned}
$$

If $y_1 = y_3 = y_5 = \infty$, then we have only two remaining inequalities:

$$
\begin{aligned}
\xi_1 + \xi_5 &\le x_2 + x_4 \\
\xi_5 &\le x_4.
\end{aligned}
$$

Under the assumptions on the arc capacities, these inequalities provide us with a necessary and sufficient condition that the total flood will be retained.

Examle 2.3. *The last example is an evacuation network of Cape May, NJ. The Category 4 Cape May hurricane in 1821 was the last major hurricane to make direct landfall in New Jersey. During evacuation, vehicles cannot maintain everyday free-flow velocity because of heavy congestion. In the network depicted in Figure 4, nodes represent the connection points to the highways and the directed arcs represent evacuation roads.*

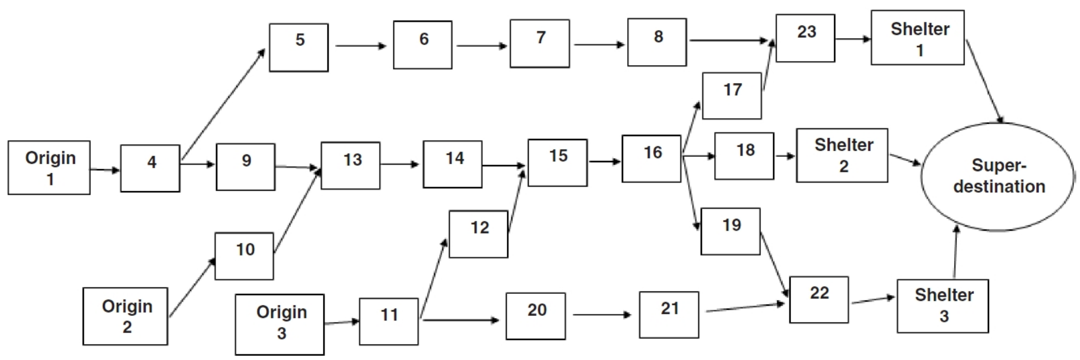

Figure 4: Simplified cell representation of Cape May evacuation network.

If we want to write up necessary and sufficient condition that the shelters have enough capacities to accommodate the evacuees, we may use the method presented in Example 2.2.

3 p-level Efficient Points

The concept of a p-level efficient point or briefly p-efficient point was introduced in Prékopa (1990). It was further studied and used to solve probabilistic constrained stochastic programming problem with discrete random variables by Prékopa, Vizvári, Badics (1998). The new results in that paper include an algorithmic enumeration method of the p-efficient points. Another algorithm is proposed by Boros, Elbassioni, Gurvich, Khachiyan and Makino (2003). Dentcheva, Prékopa and Ruszczyński (2000) gave another solution method for the same problem that generates the p-efficient points simultaneously with the solution algorithm if the random variables are independent.

In the present paper our optimization problem is of probabilistic constrained type. If, for example, the arc capacities in a network are constants but the demands are random, then the $d(S)$ symbols on the left hand sides in (4) are random variables while the right hand sides are constants and a probabilistic constraint in an optimization problem may take the form:

$$P(d(S) \leq y((S, \bar{S}), S \subset N)) \geq p. \tag{12}$$

The inequalities in (4), however, include a number of redundant ones that first we eliminate and it is sufficient to impose probabilistic constraint on those inequalities that are not deleted in the course of the eliminations. Still, in many cases quite a few inequalities remain after the elimination (as it can be seen in example 1 of section 3; see Appendix A), hence it is reasonable to look for further simplifications in the enumeration of the set of p-efficient points. Fortunately, the random variables $d(S)$ in (4) allow for such simplification. We formulate it in more general terms. First, for the reader's convenience, we recall the definition of a p-efficient point.

Let $\xi = (\xi_1, \dots, \xi_n)$ be a discrete random vector, where the supports of the random variables $\xi_1, \dots, \xi_n$ are the finite sets $Z_1, \dots, Z_n$, respectively. Introduce the notation:

$$Z = Z_1 \times Z_2 \times \cdots \times Z_n. \tag{13}$$

Let $Z_i = \{z_{i1}, \dots, z_{ik_i}\}$, where $z_{i1} \leq \cdots \leq z_{ik_i}$, $i = 1, \dots, n$.

Definition A point $z \in Z$ is a p-efficient point ($0 \leq p \leq 1$) of the probability distribution function F of ξ if $F(z) \geq p$ and there is no $y < z$ such that $F(y) \geq p$. ($y < z$ means $y \leq z$, $y \neq z$). The next theorem tells us that if we know the p-efficient points of a random vector ξ, then, under some conditions, we can at once obtain the p-efficient points of a random vector consisting of all components of ξ and some others that are linear combinations of ξ with nonnegative coefficients.

Theorem 3.1. *Let $\xi \in Z$ be a random vector and $B \geq 0$ a matrix with n columns and an arbitrary number of rows such that in each row there is at least one positive element. Suppose that the p-efficient points of ξ are $z^{(1)}, \dots, z^{(M)}$ and the following condition holds for every $i = 1, \dots, M$: $P(\{z \in Z \mid z \leq z^{(i)}\} \backslash \{z^{(i)}\}) < p$. Then the p-efficient points of the random vector $\binom{\xi}{B\xi}$ are:*

$$\begin{pmatrix} z^{(1)} \\ Bz^{(1)} \end{pmatrix}, \dots, \begin{pmatrix} z^{(M)} \\ Bz^{(M)} \end{pmatrix}. \tag{14}$$

Note that $P(\{z \mid z \leq z^{(i)}\}) \geq p$ for every $i = 1, \dots, M$, hence the condition in Theorem 3.1 implies that every p-efficient vector has positive probability.

Proof. Proof of Theorem 3.1. For every i, $1 \leq i \leq M$, the inequality $B\xi \leq Bz^{(i)}$ is a consequence of the inequality $\xi \leq z^{(i)}$. It follows that

$$P\left(\begin{pmatrix} \xi \\ B\xi \end{pmatrix} \leq \begin{pmatrix} z^{(i)} \\ Bz^{(i)} \end{pmatrix}\right) \geq p,\ \ i = 1, \dots, M. \tag{15}$$

We have to show that if we decrease the value of at least one of the components of $\binom{z^{(i)}}{Bz^{(i)}}$ within the support of $\binom{\xi}{B\xi}$ then the inequality (15) is no longer valid for the given i. Obviously, if the decrease happens among the first n components, $z^{(i)}$ decreases to $w^{(i)}$, then $P(\xi \leq w^{(i)}) < p$ and also $P(\xi \leq w^{(i)},\ B\xi \leq z^{(i)}) < p$. If, on the other hand, one component of $Bz^{(i)}$ decreases to $Bw^{(i)}$, then, because B has at least one positive entry in each row, the point $z^{(i)}$ is excluded. In view of our assumption the probability decreases to $P(\xi \leq z^{(i)}, \xi \neq z^{(i)}, B\xi \leq Bw^{(i)}) < p$. □

Remark 3.2. *While in practice most frequently we have $P(\xi = z^{(i)}) > 0,\ i = 1, \dots, M$, the condition that $P(\xi \le z^{(i)},\ \xi \ne z^{(i)}) < p$ may not hold. Still, we advise to use the set of vectors (14) as an approximation of the set of p-efficient points of $\binom{\xi}{B\xi}$. The reason is that the probability distribution of ξ can slightly be perturbed (at least in most practical problems) in such a way that the p-efficient points of the perturbed distribution are those in (14). In fact, if we add to each $p_{jk} = P(\xi = z_{jk}),\ z_{jk} \in \bigcup_{i=1}^{M} (z^{(i)} + R^n_-) \cap Z$ a value $\varepsilon_{jk} \ge 0$ and subtract $\varepsilon_{lt} \ge 0$ from each $p_{lt} = P(\xi = z_{lt}),\ z_{lt} \notin \bigcup_{i=1}^{M} (z^{(i)} + R^n_-) \cap Z$, keeping the probabilities nonnegative and their sum equal to 1, then under suitable choices of the ε's we may accomplish the task.*

An important special case of Theorem 3.1 is the following. Let $I_1, \dots, I_l$ be non-empty subsets of the set $\{1, \dots, l\}$ and consider the random vector

$$\left(\xi_1, \dots, \xi_n, \sum_{i \in I_1} \xi_i, \dots, \sum_{i \in I_l} \xi_i\right)^T. \tag{16}$$

Theorem 3.3. *If the p-efficient points of ξ are $\{z^{(1)}, \dots, z^{(M)}\}$ and $P(\xi \le z^{(k)},\ \xi \ne z^{(k)}) < p,\ k = 1, \dots, M$, then the p-efficient points of the random vector* (16) *are:*

$$\left(z_1^{(k)}, \dots, z_n^{(k)}, \sum_{i \in I_1} z_i^{(k)}, \dots, \sum_{i \in I_l} z_i^{(k)}\right)^T,\ \ k = 1, \dots, M. \tag{17}$$

Remark 3.4. *The condition in Theorem 3.3 holds true if $P(\xi \le z^{(k)}) = p$ and $P(\xi = z^{(k)}) > 0,\ \ k = 1, \dots, M$.*

Examle 3.1. *Let $(\xi = \xi_1,\ \xi_2,\ \xi_3,\ \xi_4)^T$ and consider the random vector:*

$$(\xi_1,\ \xi_2,\ \xi_3,\ \xi_4,\ \xi_1 + \xi_2,\ \xi_1 + \xi_3 + \xi_4,\ \xi_2 + \xi_4)^T. \tag{18}$$

If the condition mentioned in Theorem 3.3 holds true, then the p-efficient points of the random vector (18) are:

$$\begin{pmatrix} z_1^{(k)} \\ z_2^{(k)} \\ z_3^{(k)} \\ z_4^{(k)} \\ z_1^{(k)} + z_2^{(k)} \\ z_1^{(k)} + z_3^{(k)} + z_4^{(k)} \\ z_2^{(k)} + z_4^{(k)} \end{pmatrix},\ \ k = 1, \dots, M.$$

If each random variable is uniformly distributed in the same support set $\{1,2,3,4,5\}$ and $p = 0.8$ then the p-efficient points of ξ are:

$$\begin{pmatrix}4\\5\\5\\5\end{pmatrix},\begin{pmatrix}5\\4\\5\\5\end{pmatrix},\begin{pmatrix}5\\5\\4\\5\end{pmatrix},\begin{pmatrix}5\\5\\5\\4\end{pmatrix}.$$

The value of the joint c.d.f. is equal to 0.8 at each of these points, hence the condition mentioned in Theorem 3.3 is satisfied. It follow that the p-efficient points of the random vector (18) are:

$$\begin{pmatrix}4\\5\\5\\5\\9\\14\\10\end{pmatrix},\begin{pmatrix}5\\4\\5\\5\\9\\15\\9\end{pmatrix},\begin{pmatrix}5\\5\\4\\5\\10\\14\\10\end{pmatrix},\begin{pmatrix}5\\5\\5\\4\\10\\14\\9\end{pmatrix}.$$

Examle 3.2. *We show that the statement of Theorem 3.1, may not valid without the assumption. Consider the random vector* $\xi = (\xi_1, \xi_2)^T$ *and suppose that* ξ *has the following probability distribution:* $P(\xi = (0,0)) = 0.4,\ P(\xi = (0,1)) = P(\xi = (1,0)) = 0.2,\ P(\xi = (0,2)) = P(\xi = (2,0)) = 0.1.$

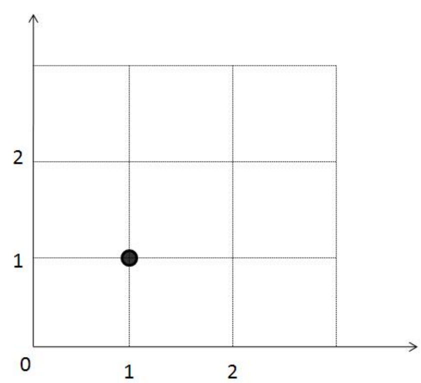

Figure 5: The condition in Theorem 3.3 is not satisfied for the marked point.

The marked point $(1,1)$ *in Figure 5 is the only* 0.8*-efficient point of* ξ *and it has* 0 *probability. If we consider the random vector* $\xi = (\xi_1,\ \xi_2,\ \xi_1+\xi_2)^T$*, then we can see that its only* 0.8*-efficient point is* $(1,\ 1,\ 1)$ *and not* $(1,\ 1,\ 2)$ *as it would be the case under the condition of Theorem 3.1.*

For the sake of completeness we present an algorithm that generates all p-efficient points of the random vector (16). The p-efficient points of (16) will be called network p-efficient points.

If for a k the condition of Theorem 3.1 is satisfied, i.e., $P(\xi \leq z^{(k)},\ \xi \neq z^{(k)}) < p$, then the vector (17) is a network p-efficient point. If this is not the case, then we define the set

$$K_h^{(k)} = \left\{ z \in Z \mid z \leq z^{(k)},\ \sum_{i \in I_j} (z_i^{(k)} - z_i) \geq h_j,\ j = 1, \ldots, t \right\}$$
$$h_j \text{ integer},\ 0 \leq h_j \leq |I_j|, \quad j = 1, \ldots, t; \quad k = 1, \ldots, M. \tag{19}$$

For a given k, for which $P(\xi \leq z^{(k)},\ \xi \neq z^{(k)}) < p$, we want to find all $h^* = (h_1^*, \ldots, h_t^*)$ vectors that satisfy the condition in the second line of (19) and

$$\begin{aligned} F(z^{(k)}) - P(K_h^{(k)}) &\geq p \\ F(z^{(k)}) - P(K_{h^*}^{(k)}) &< p \end{aligned} \tag{20}$$

for $h \leq h^*$. If we rewrite (20) as follows:

$$\begin{aligned} P(K_h^{(k)}) &\leq F(z^{(k)}) - p \\ P(K_{h^*}^{(k)}) &> F(z^{(k)}) - p \end{aligned} \tag{21}$$

then the problem is to find all $F(z^{(k)}) - p$-efficient points in the integer lattice of the cube $\{h \mid h_j \leq |I_j|,\ j = 1, \ldots, t\}$. The efficiency is now defined in the sense of (21). To find all efficient h^*, in the sense of (21), any of the existing algorithms, to find all p-efficient points, can be used with obvious modification.

If $M = 1$, i.e., there is only one p-efficient point of ξ and the condition of Theorem 3.1 is satisfied, then we are done, the corresponding vector in (18) is the only network p-efficient point. If $M = 1$ and the condition of Theorem 3.1 is not satisfied, then we generate all h^* and the obtained $F(z^{(1)}) - p$-efficient points simultaneously provide us with the set of all network p-eficient points. If $M > 1$ and there is at least one k $(1 \leq k \leq M)$ for which the condition of Theorem 3.1 is not satisfied, then we use an algorithm to generate the set of network p-efficient points. Note that it is not enough to find the $F(z^{(k)}) - p$-efficient points for every k for which $P(\xi \leq z^{(k)},\ \xi \neq z^{(k)}) < p$, because some of them may be dominated by others, corresponding to different k values and therefore an elimination procedure has to be included.

3.1 Algorithm to Find all Network p-efficient Points

Step 1. Find all p-efficient points of ξ and designate them by $z^{(k)}$, $k = 1, \ldots, M$.

Step 2. Initialize $J = 0$ and let $H^{(J)}$ be the current set of network p-efficient points.

Step 3. Set $J = J + 1$. If $J > M + 1$, then go to Step 6. Otherwise go to Step 4.

Step 4. If for $z^{(J)}$ we have $P(\xi \leq z^{(J)},\ \xi \neq z^{(J)}) < p$, then include $z^{(J)}$ into $H^{(J)}$ and go to Step 6. Otherwise go to Step 5.

Step 5. Using $z^{(J)}$, generate all $F(z^J) - p$-efficient points and form the new (18)-type vectors. Eliminate those which are dominated by vectors in $H^{(J)}$. Include into $H^{(J)}$ the remaining ones. Go to Step 3.

Step 6. Stop, $H^{(M)}$ is the set of all network p-efficient points.

4 Static Stochastic Network Design Problem Using Probabilistic Constraint

Our stochastic network design problem can be formulated in following way.

Corresponding to each node i in the network a capacity x_i and a random demand ξ_i are associated. Bearing in mind application to interconnected power systems, we call x_i generating capacity and the capacity y_{ij}, corresponding to arc (i, j), transmission capacity. If the system demand $\xi_i - x_i$ at node i is positive, then the local generating capacity is not enough to meet the local demand ξ_i and assistance is needed from other nodes. If, however, $\xi_i - x_i < 0$, then there is surplus generating capacity at node i and the node can assist others.

The unknown decision variables in our optimization problem are the node capacities $x_i,\ i \in N$ and the arc capacities $y_{ij},\ (i, j) \in N \times N$. The static formulation of the problem is the following:

$$
\begin{gathered}
\min \left\{ \sum_{i \in N} c_i(x_i) + \sum_{(i,j) \in N \times N} c_{ij}(y_{ij}) \right\} \\
\text{subject to} \\
P(d(S) \leq y((S, \bar{S}), S \subset N)) \geq p \\
A_1 x + A_2 y \geq b \\
x \geq 0, y \geq 0.
\end{gathered} \tag{22}
$$

The constraint $A_1 x + A_2 y \geq b$ may simply mean lower and upper bounds for the decision variables $x_i,\ y_{ij}$. In that case, we write them up as follows:

$$
\begin{gathered}
l_i \leq x_i \leq u_i,\ \ i \in N \\
l_{ij} \leq y_{ij} \leq u_{ij},\ \ (i, j) \in N \times N \quad .
\end{gathered} \tag{23}
$$

The static stochastic programming problems can be of probabilistic constrained, recourse (penalty) or hybrid type. Instead of problem (22) we may easily construct a hybrid type model, where the expectation of the measure of violation of the stochastic constraints is incorporated into the objective function. Since we have discrete random variables, the inclusion of penalty terms into the objective function does not change the type of the problem. The objective function is extended by linear terms and new linear constraints are incorporated (see Prékopa, 1995, Chapter 9). However, our main concern is the handling of the probabilistic constraint, therefore we disregard the formulation of a hybrid model. In the probabilistic constraint of problem (22) we have the Gale–Hoffman inequalities: $d(S) \leq y(S, \bar{S})$, $S \subset N$. If we apply the elimination procedure described in the introduction, then we can significantly reduce the number of them. After the elimination the problem takes the form:

$$
\begin{gathered}
\min\left\{\sum_{i\in N} c_i(x_i) + \sum_{(i,j)\in NxN} c_{ij}(y_{ij})\right\} \\
\text{subject to} \\
P\left(\begin{array}{c} \xi_k \le x_k + \sum_{(j,k)\in N\times N} y_{jk},\ \ k=1,\dots,n \\ \sum_{k\in I_j}\xi_k \le \sum_{k\in I_j} x_k + \sum_{k\in I_j}\sum_{(j,k)\in N\times N} y_{jk},\ \ j=1,\dots,t \end{array}\right) \ge p \\
A_1 x + A_2 y \ge b \\
x \ge 0, y \ge 0,
\end{gathered}
\tag{24}
$$

where $n = \mid N \mid$. Inside the parentheses in the probabilistic constraint the non-eliminated feasibility inequalities are listed. It is essential, from our point of view, that all individual stochastic constraints, i.e., those that contain a single component of the random vector $\xi = (\xi_1, \dots, \xi_n)^T$ appear among the stochastic constraints. We need them in order to be able to apply the methodology of Theorem 3.1. The requirement that the individual stochastic constraints should not be eliminated is not a restriction, however, from the practical point of view.

In what follows we will be looking at the random vector:

$$
\left(\begin{array}{c} \xi \\ \sum_{k\in I_j}\xi_k \\ j=1,\dots,t \end{array}\right)
\tag{25}
$$

and assume that $\{z^{(1)}, \dots, z^{(M)}\}$ is the set of p-efficient points of ξ. Then, we use the vectors

$$
\left(\begin{array}{c} z^{(i)} \\ \sum_{k\in I_j} z_k^{(i)} \\ j=1,\dots,t \end{array}\right),\ \ i=1,\dots,M
\tag{26}
$$

as the correct or approximate set of network p-efficient points. With these vectors our network design problem can be formulated in the following way:

$$
\begin{gathered}
\min\left\{\sum_{i\in N} c_i(x_i) + \sum_{(i,j)\in N\times N} c_{ij}(y_{ij})\right\} \\
\text{subject to the constraints that for at least one } i=1,\dots,M \text{ we have} \\
x_k + \sum_{(j,k)\in N\times N} y_{jk} \ge z_k^{(i)},\ \ k=1,\dots,n \\
\sum_{k\in I_l} x_k + \sum_{k\in I_l}\sum_{(j,k)\in N\times N} y_{jk} \ge \sum_{k\in I_l} z_k^{(i)}, 0\ \ l=1,\dots,t \\
\text{and} \\
A_1 x + A_2 y \ge b \\
x \ge 0,\ \ y \ge 0.
\end{gathered}
\tag{27}
$$

Problem (27) is a disjunctive optimization problem that we relax by a standard convexification procedure: we take the convex combination of the upper $M(n+t)$ inequalities. The new problem is:

$$
\begin{gathered}
\min\left\{\sum_{i\in N} c_i(x_i) + \sum_{(i,j)\in N\times N} c_{ij}(y_{ij})\right\} \\
\text{subject to} \\
\begin{pmatrix} x_k + \sum_{(j,k)\in N\times N} y_{jk} \geq z_k^{(i)},\ \ k=1,\dots,n \\ \sum_{k\in I_l} x_k + \sum_{k\in I_l}\sum_{(j,k)\in N\times N} y_{jk} \geq \sum_{k\in I_l} z_k^{(i)},\ \ l=1,\dots,t \end{pmatrix} \geq \\
\sum_{i=1}^M \lambda_i \begin{pmatrix} z_k^{(i)},\ \ k=1,\dots,n \\ \sum_{k\in I_l} z_k^{(i)},\ \ l=1,\dots,t \end{pmatrix} \\
A_1x + A_2y \geq b \\
\sum_{i=1}^M \lambda_i = 1 \\
x\geq 0,\ \ y\geq 0,\ \ \lambda \geq 0.
\end{gathered} \tag{28}
$$

In what follows, we assume that the cost functions $c_k(x_k)$, $c_{jk}(y_{jk})$ are linear. If these functions are nonlinear but convex, then we approximate them by piecewise linear functions and again the problem is an LP.

5 Solution of the Problem Presented in Section 5

For simplicity we assume that the objective function is linear, but it can be of a very large size, hence a special algorithm may be more efficient than the use of a general purpose LP package. There are two algorithms available that offer solutions for our problem:

I. The Prékopa–Vizvári-Badics (PVB) algorithm (1998), where the p-efficient points are first enumerated or they are all known from another source.

II. The Dentcheva–Prékopa–Ruszczyński (DPR) algorithm (2000) that generates the p-efficient points simultaneously with the solution algorithm.

The PVB algorithm is described in a somewhat more complete way in Prékopa (2006). We will comment on it in the next section.

We propose the use of the DPR algorithm with an important improvement regarding the calculation of the new p-efficient points in the course of the iteration. We also use ideas from Vizvári (2002), where the DPR algorithm is presented in a slightly different way. First we rewrite problem (28) in the following form, where $J = \{1,\dots,M\}$:

$$
\begin{gathered}
\min\left\{c_1^T x + c_2^T y\right\} \\
\text{subject to} \\
T_1x + T_2y \geq \sum_{j\in J} \lambda_j v^{(j)} \\
A_1x + A_2y \geq b \\
\sum_{i=1}^M \lambda_i = 1 \\
x\geq 0,\ \ y\geq 0,\ \ \lambda\geq 0.
\end{gathered} \tag{29}
$$

where $v^{(1)}, \ldots, v^{(M)}$ are the network p-efficient points (26).
If we introduce slack variables u,w in the inequality constraints, then the problem can be written as:

$$\min \left\{ c_1^T x + c_2^T y + 0^T u + 0^T w + 0^T \lambda \right\}$$

$$\text{(P)} \qquad \text{subject to} \quad \begin{pmatrix} T_1 & T_2 & -E & 0 & 0 \\ A_1 & A_2 & 0 & -E & -V \\ 0^T & 0^T & 0^T & 0^T & e^T \end{pmatrix} \begin{pmatrix} x \\ y \\ u \\ w \\ \lambda \end{pmatrix} = \begin{pmatrix} 0 \\ b \\ 1 \end{pmatrix} \tag{30}$$

$$x \geq 0, \ y \geq 0, \ u \geq 0, \ w \geq 0, \ \lambda \geq 0,$$

where $V = \left(v^{(j)}, \ j \in J\right)$, is the $(n+t) \times M$ matrix and $e^T = (1, \ldots, 1)$. We subsequently generate the columns of V. Let J_h designate the subscript set of the available p-efficient points and $V_h = \left(v^{(j)}, \ j \in J_h\right)$. In iteration h we have a problem that differs from (P) in such a way that we replace V_h for V. Let (P_h) designate that LP.

Solve (P_h) by a method that produces an optimal basis satisfying the optimality condition and let α be the optimal dual vector. Partition α into α_1, α_2, α_3, consistent with the partitioning of the rows of the matrix in problem (P_h).

If α is an optimal dual vector in problem (P) too, then we are done, the current problem (P_h) provides us with the optimal solution of problem (P). Otherwise there exists a column in the matrix of problem (P) that has a scalar product with α greater than the corresponding objective function coefficient. This may happen to a column that belongs to the last block of the matrix because the other columns are the same as those in (P_h). The column in the last block we are referring to is unknown but we know that its transpose has the form: $(0^T, -v^T, 1)$, where v is a column in V but not in V_h. Writing up the scalar product we obtain:

$$\left(0^T, -v^T, 1\right) \alpha = \alpha_3 - v^T \alpha_2 > 0. \tag{31}$$

On the other hand, if we look at the columns in problem (P_h), then we observe that at least one component of λ must be basic. If the corresponding column in the problem is $\left(0^T, -v^{(h)}, 1\right)$, then we have the equation:

$$\left(0^T, -\left(v^{(h)}\right)^T, 1\right) \alpha = 0$$

which implies that

$$\alpha_3 - \left(v^{(h)}\right)^T \alpha_2 = 0. \tag{32}$$

Relations (31) and (32) tell us that a new column (and variable) can enter the basis in problem (P_h) iff

$$\min_{i \in J_h} \alpha_2^T v^{(i)} > \min_{i \in J} \alpha_2^T v^{(i)}. \tag{33}$$

The new column and variable will be supplied by the solution of the problem:

$$\min_{i \in J} \alpha_2^T v^{(i)} \tag{34}$$

If the two values in (33) are equal, then the prcedure terminates. Note that if we take the scalar product with the columns in (P_h) that belong to the second to the last block, then we obtain the inequality $\alpha_2 \geq 0$. On the other hand, if the optimum value of problem (30) is different from 0, then (32) implies that $\alpha_2 > 0$. In fact, if $\alpha_2 = 0$, then $\alpha_3 = 0$ and the optimum value of the dual of problem (30) would be 0, contrary to the assumption.

In the next section we need the stronger inequality: $\alpha_2 \gg 0$. To ensure it, we need same condition in connection with the p-efficient points.

Regularity condition. Let $v^{(i)}$, $i \in L$ be a collection of linearly independent p-efficient points. Then there exists a $v^{(j)}$, $j \in L$ such that the intersection of the linear subspace spanned by $v^{(i)} - v^{(j)}$, $i \in L$, $i \neq j$ and the nonnegative orthant R_+^n has the 0 vector in common.

Theorem 5.1. *If the regularity condition holds true, then at any iteration of the solution algorithm of Section 6 the dual vector can be chosen in such a way that* $\alpha_2 \gg 0$.

Proof. Proof of Theorem 5.1. Let L be the subscript set of the basic vectors from the last block in problem (30). The regularity condition tells us that there is a $j \in L$ such that there are real values y_i, $i \in L \setminus \{j\}$ satisfying

$$\sum_{i \in L \setminus \{i\}} y_i (v^{(i)} - v^{(j)}) > 0. \tag{35}$$

Then, by the theorem of Stiemke ($Ax = 0$ has a solution $x \gg 0$ iff there is no y such that $y^T A > 0$) we have that

$$(v^{(i)} - v^{(j)})^T \alpha_2 = 0, \;\; i \in L \setminus \{j\} \tag{36}$$

has a solution $\alpha_2 \gg 0$. Since $\alpha_3 = (v^{(j)})^T \alpha_2$, (36) implies that

$$\alpha_3 - (v^{(i)})^T \alpha_2 = 0, \;\; i \in L.$$

□

6 Finding new *p*-efficient Point

6.1 The General Case

The solution of the problem (34) can be carried out by solving another problem, where the unknown vector is of much smaller size. Let $F(z)$ be the c.d.f. of ξ.

If we take into account the p-efficient point $v^{(i)}$, $i = 1, \dots, M$ are those in (26), then we can derive an expression for $\alpha_2^T v^{(i)}$ by the use of $z^{(i)}$ which is an efficient point of ξ. In fact,

$$\begin{aligned} \alpha_2^T v^{(i)} &= \sum_{j=1}^{n} \alpha_{2j} z_j^{(i)} + \sum_{h=n+1}^{r} \alpha_{2h} \sum_{j \in I_h} z_j^{(i)} \\ &= \sum_{j=1}^{n} \alpha_{2j} z_j^{(i)} + \sum_{j=1}^{n} z_j^{(i)} \sum_{I_h \ni j} \alpha_{2h} \\ &= \sum_{j=1}^{n} \left(\alpha_{2j} + \sum_{I_h \ni j} \alpha_{2h} \right) z_j^{(i)}, \;\; i = 1, \dots, n, \end{aligned} \tag{37}$$

where $r = n + t$. Introducing the notation:

$$\begin{aligned} \gamma_j &= \alpha_{2j} + \sum_{I_h \ni j} \alpha_{2h} \\ \gamma &= (\gamma_1, \dots, \gamma_n)^T, \end{aligned}$$

equation (37) can be written in the form:

$$\alpha_2^T v^{(i)} = \gamma^T z^{(i)}, i = 1, \dots, n. \tag{38}$$

There is a one-to-one correspondence between the p-efficient points $z^{(i)}$, $i = 1, \dots, M$, such that $v^{(i)} \leftrightarrow z^{(i)}$ and (38) holds true. Since $\alpha_2 \gg 0$, this implies that problem (34) can be solved in such a way that we solve the smaller size problem:

$$\begin{aligned} \min \gamma^T z \\ \text{subject to} \\ F(z) \geq p \\ z \in Z. \end{aligned} \tag{39}$$

In most cases we know lower and upper bounds on the components of ξ, which, in turn, can be prescribed for the components of z. At this point we just supplement the constraint $z \in D$ to problem (39) with the remark that it may mean the mentioned lower and upper bounds on z. The solution of problem (39), in that general form, may still be computationally intensive because the number of p-efficient points of F may be very large. There is no need, however, to solve problem (39) optimally. It is enough to enumerate the p-efficient points (e.g., by the use of the PVB algorithm), until a $z \in J$, $z \notin J_h$ is found for which

$$\min_{i \in J_h} \alpha_2^T v^{(i)} > \gamma_2^T z.$$

Let z be the new p-efficient point.

Problem (39) can be reformulated as a discrete optimization problem as follows:

$$\begin{gathered}\min \sum_i \gamma^T u_i \varepsilon_i \\ \text{subject to} \\ \sum_i F(u_i)\varepsilon_i \geq p \\ \sum_i \varepsilon_i = 1 \\ u_i \in Z, u_i \in D, \varepsilon_i \in \{0,1\}, \text{all } i.\end{gathered} \tag{40}$$

Another simple reformulation is possible if the probability function values, rather than the c.d.f. values of ξ are available.

Let $p_i = p(\xi = u_i)$, for $u_i \in Z$, $u_i \in D$. Then the problem is:

$$\begin{gathered}\min \sum_i \gamma^T u_i \varepsilon_i \\ \text{subject to} \\ \sum_i p_i \varepsilon_i \geq p \\ \varepsilon_i \leq \varepsilon_j \text{ for } u_j \leq u_i \\ u_i \in Z, u_i \in D, \varepsilon_i \in \{0,1\}, \text{all } i.\end{gathered} \tag{41}$$

6.2 Specialization and Relaxation of Problem (40)

In this section we present our version of the greedy algorithm (see Pisinger, 1995) for the solution of the knapsack problem and its application to solve problem (39) in case of independent random variables with strictly logconcave univariate marginal c.d.f.'s are also logconcave.

Assume that $\xi_1, \ldots, \xi_n$ are independent, integer valued and let F_i be the c.d.f. of ξ_i, $i = 1, \ldots, n$. Assume, further, that $\xi_i \in [l_i, u_i]$ and F_i is strictly logconcave in $[l_i, u_i]$, $i = 1, \ldots, n$. Then problem (39) an be written in the form:

$$\begin{gathered}\min \sum_{i=1}^n \sum_{k=l_i}^{u_i} \gamma_i k \delta_{ik} \\ \text{subject to} \\ \sum_{i=1}^n \sum_{k=l_i}^{u_i} a_{ik}\delta_{ik} \leq d \\ z \in D \\ \sum_{k=l_i}^{u_i} \delta_{ik} = 1, \;\; i = 1, \ldots, n \\ \delta_{ik} \in \{0,1\}, \;\; \text{all } i, k,\end{gathered} \tag{42}$$

where $a_{ik} = -\log F_i(k)$ and $d = -\log p$. The problem is a special case of the Multiple Choice Knapsack Problem (MCKP). In the general case we have h_{ik} instead of $k\alpha_{2i}$.

For the solution of problem (42) we use a greedy method in Pisinger (1995), where the first step is the solution of a relaxed LP called Linear Multiple Choice Knapsack Problem (LMCKP). We relax problem (42) in such a way that we allow the δ_{ik} variables to move freely in the interval $[0,1]$:

$$\begin{array}{c} \min \sum_{i=1}^{n} \sum_{k=l_i}^{u_i} \gamma_i k \delta_{ik} \\ \text{subject to} \\ \sum_{i=1}^{n} \sum_{k=l_i}^{u_i} a_{ik}\delta_{ik} \le d \\ z \in D \\ \sum_{k=l_i}^{n_i} \delta_{ik} = 1, \ \ i = 1,\dots,n \\ 0 \le \delta_{ik}, \ \ \text{all } i,k. \end{array} \tag{43}$$

To solve problem (43) we use a special algorithm. We introduce slack variable u in the inequality constraint in problem (43), then split the sum into n terms, each term corresponds to a component of ξ. It will be more convenient in the new problem to use slightly different notations. We change the range of the second subscripts so that the summation should go from 1 to m_i and designate the coefficient of δ_{ik} in the objective function by h_{ik}. Note that for every i, the discrete function h_{ik} is linear in k with coefficient $\delta_i > 0$. Then the new problem is:

$$\min \left\{ 0u + 0u_1 + \cdots + 0u_n + h_{11}\delta_{11} + \cdots + h_{1m_1}\delta_{1m_1} + \cdots + h_{n1}\delta_{n1} + \cdots + h_{nm_n}\delta_{nm_n} \right\}$$

subject to

$$\begin{array}{lllllll} u + & u_1 & + \cdots + u_n & & & = & d \\ & u_1 & & -a_{11}\delta_{11} - \cdots - a_{1m_1}\delta_{1m_1} & & = & 0 \\ & & \ddots & & & & \vdots \\ & & u_n & & -a_{n1}\delta_{n1} - \cdots - a_{nm_n}\delta_{nm_n} & = & 0 \\ & & & \delta_{11} + \cdots + \delta_{n1} & & = & 1 \\ & & & \ddots & & & \vdots \\ & & & & \delta_{n1} + \cdots + \delta_{nm_n} & = & 1 \end{array} \tag{44}$$

$$u \ge 0, \ \ u_i \ge 0, \ \ i = 1,\dots,n, \ \ \delta_{ik} \ge 0, \quad \text{all } i,k.$$

Problem (44) is related to the simple recourse problem in stochastic programming, when we apply the λ-representations for the piecewise linear separable functions in the objective, for the case of discrete random variables (see Prékopa, 1990 a, 1995, Chapter 9). The matrix of the equality constraints, together with the coefficient sequences in the objective function, can be partitioned into $n+1$ blocks and labeled by $0, 1, \dots, n$ respectively. The matrices taken from blocks $1, \dots, n$,

$$\begin{pmatrix} -a_{i1} & \dots & -a_{im_i} \\ 1 & \dots & 1 \end{pmatrix}, \ \begin{pmatrix} h_{i1} & \dots & h_{im_i} \\ -a_{i1} & \dots & -a_{im_i} \\ 1 & \dots & 1 \end{pmatrix} \tag{45}$$

have a property enjoyed by the corresponding matrices in the simple recourse problem as formulated by Prékopa.

Theorem 6.1. *All* 2×2 *minors of the first and all* 3×3 *minors of the second matrices in (45) are nonnegative.*

Proof. Proof of Theorem 6.1. The sequence $-a_{i1}, \dots, -a_{im_i}$ is non-decreasing, hence any 2×2 minor of the first matrix is nonnegative. As regards the second matrix, if we pick three columns from it, corresponding to $j < k < l$, then its determinant is γ_i times the second order divided difference of $-a_{ij}, -a_{ik}, -a_{il}$, where $\gamma_i > 0$. Here we took into account the convexity of the sequence $-a_{i1}, \dots, -a_{im_i}$ which is a consequence of the strict logconcavity of the c.d.f. F_i and the strict convexity of the sequences $-a_{i1}, \dots, -a_{im_i}$). It follows that, any dual feasible basis of problem (44) has two consecutive columns from each block $1, \dots, n$. (see Prékopa 1990 a, 1995).

An LP: $\min(\max) c^T x$ subject to $Ax = b,\ x \geq 0$ is called totally positive (in Prékopa, 1990 b) if all $m \times m$ minors of A and all $(m+1) \times (m+1)$ minors of $\binom{c^T}{A}$ are positive, where A is a $m \times n$ matrix, $n \geq m + 1$. In the above mentioned paper Prékopa proved the following

Theorem 6.2. *The dual feasible basis of a totally positive LP have the following structure, presented in terms of the basis subscripts:*

	m even	*m odd*
min problem	$i, i+1, \dots, j, j+1$	$i, i+1, \dots, j, j+1, n$
max problem	$1, i, i+1, \dots, j, j+1, n$	$1, i, i+1, \dots, j, j+1,$

where the subscripts are arranged in increasing order.

If we specialize this theorem for the LP:

$$\begin{gathered} \min\ \sum_{j=1}^{n_i} h_{ij}\delta_{ij} \\ \text{subject to} \\ \sum_{j=1}^{n_i}(-a_{ij})\delta_{ij} = -u_i \\ \sum_{j=1}^{u_i} \delta_{ij} = 1 \\ \delta_{ij} = 1,\ \ j = 1, \dots, n_i, \end{gathered} \tag{46}$$

then we derive the consequence that all dual feasible basis of the problem are consecutive pairs of columns of the matrix of the equality constraints.

6.3 Solution of Problem (44)

An efficient dual type algorithm for the solution of the simple recourse problem is presented in Prékopa (1990 a, 1995) and further developed by Fábián, Prékopa, Ruff-Fiedler (1995). The same method solves efficiently problem (44) too. Here we present only the construction of the initial dual feasible basis, because it is particularly simple in this case and mention how we can obtain fast and very good bounds for the optimum value.

Finding initial dual feasible basis
Pick arbitrary two consecutive columns from each of the blocks $1, \ldots, n$ in problem (44), as part of a dual feasible basis of the entire problem. Let v_i, w_i be the dual variables corresponding to problem (46), $i = 1, \ldots, n$. Since the rows of blocks $1, \ldots, n$ are disjoint, the $v_1, \ldots, v_n,\ w_1, \ldots, w_n$ can be regarded as dual variables corresponding to problem (46), where, however, one column and one dual variable is further to be chosen. Let y designate the last dual variable. This and the final column of the dual feasible basis can be found by the solution of the LP:

$$\begin{gathered} \min \sum_{i=1}^{n} (-v_i) u_i \\ \text{subject to} \\ \sum_{i=1}^{n} u_i = d \\ u \geq 0. \end{gathered} \tag{47}$$

The optimal solution is $u_i = d$, $u_i = 0$, for $i \neq j$, where $j = \text{argmin}(-v_i)$. The column of u_i in block 0 is the final one to form a dual feasible basis B_0 with the already chosen consecutive pairs from blocks $1, \ldots, n$. The final consecutive of the corresponding dual vector is $y = -v_j$.
Fast bounds for the optimal value
Having dual feasible basis for the minimization problem we also have a lower bound for the optimum value. The basic solution, corresponding to the dual feasible basis is not necessarily primal feasible, but we can easily create a primal feasible basis in the following way. Keep the vector that has been obtained as the optimal solution of problem (46) and $j_1, \ldots, j_n$ in such a way that the solution for $\delta_{ij}, \delta_{ij_{i+1}}$ of the equation:

$$\begin{aligned} -a_{ij_i}\delta_{ij_i} + (-a_{ij_{i+1}})\delta_{ij_{i+1}} &= -u_i \\ \delta_{ij_i} + \delta_{ij_{i+1}} &= 1, \quad i = 1, \ldots, n \end{aligned}$$

be nonnegative. Then the new basis B_1, consisting of the columns subscripted by j, from block 0, and j_i, j_{i+1}, from block i, $i = a, \ldots, n$, is primal feasible and provides us with an upper bound for the optimum value of problem (44).

The bounding procedure can be continued. Keeping the conseutive pairs from blocks $1, \ldots, n$ we can construct a further dual feasible basis B_2 in the same way as we have constructed B_0 etc. The lower bounds may not be increasing and the upper bounds may not be decreasing. In addition, the bounding procedure may not provide us with the exact optimum value but we choose the best bounds after a finite number of steps. Having a close bound, corresponding to a primal feasible basis we may pass to a feasible solution where any one of the $\lambda_{ij_i}, \lambda_{ij_{i+1}}$ is positive, for every $i = 1, \ldots, n$, in a cost efficient way. If the obtained p-efficient point is not good enough, then we solve problem (44) optimally and only then pass to a p-efficient point in a cost efficient way.

An efficient algorithm for the solution of a problem of which (44) is a special case, is presented in Prékopa (1990, 1995) and further developed by Fábián, Prékopa, Ruff-Friedler (1995). The application of it to problem (44) is straightforward and will not be detailed. We note, however, that the specialized algorithm is very simple because of the simplicity of that part of the matrix which constitutes block 0. In the optimal solution, we need exactly one argument z_i of each F_i so that $z = (z_1, \ldots, z_n)$ is an optimal solution to the problem (44). However, at the end of the algorithm there may be blocks, among these labeled by $1, \ldots, n$, which have two columns in the optimal basis. The final step is to remove one out of each consecutive pairs in a cost efficient way. The obtained z solves problem (44).

Once we have the new p-efficient point for the distribution of ξ, we create the new p-efficient point for the random vector (25) and enter it into problem (P_h) to obtain (P_{h+1}).

6.4 Methods for the Case of Stochastically Dependent $\xi_1, \ldots, \xi_n$

In this section we propose two methods to use, in the solution of the network design problem, for the case of stochastically dependent random variables $\xi_1, \ldots, \xi_n$. Both are based on the assumption that $\xi_1, \ldots, \xi_n$ are positively correlated but we need more than that.

The first method consists in the assumption that $\xi_1, \ldots, \xi_n$ are associated. Following Esary, Proschan, Walkup (1967) we say that the components of the random vector $\xi = (\xi_1, \ldots, \xi_n)$ are associated if

$$Cov(f(\xi), g(\xi)) \geq 0 \tag{48}$$

for all nondecreasing functions f, g for which $E(f(\xi))$, $E(g(\xi))$, $E(f(\xi)g(\xi))$ exist. Esary, Proschan, Walkup have proved (1967, Theorem 5.1) that if the components of ξ are associated then

$$P(\xi_i \leq z_i, \; i = 1, \ldots, n) \geq \prod_{i=1}^{n} P(\xi_i \leq z_i) \tag{49}$$

Thus, under the assumption that $\xi_1, \ldots, \xi_n$ are associated, if we replace the constraint $F(z) \geq p$ by the constraint $\prod_{i=1}^{n} F_i(z_i) \geq p$, then the constraints of problem (22) and those of the relaxed problem (28) will be satisfied. The optimum values, however, will be larger because the sets of feasible solutions will be smaller.

A second possibility is applicable for a few probability distribution only. For simplicity, first we illustrate the method on the Poisson distribution and then mention what other distributions we can use the same way.

Let H, be the $n \times (2^n - 1)$ matrix the columns of which are all 0, 1 component n-vectors, except for the 0 vector. Let H_2 be the $\binom{n}{2} \times (2^n - 1)$ matrix rows of which are counterwise products of the rows of H_1 and let $H = \binom{H_1}{H_2}$ which is an $\left(n + \binom{n}{2}\right) \times (2^n - 1)$ matrix. Let $\kappa_1 \ldots, \kappa_k$ be independent, Poisson distributed random variables with parameters $\upsilon_1, \ldots, \upsilon_k$ respectively, where $k = 2^n - 1$. Suppose that $\upsilon_1, \ldots, \upsilon_k$ are such that $\xi = H_1 \kappa$. Then the expectation and the covariance of ξ and $H_1 \kappa$ coincide. For example, if $m = 4$ then this means

$$
\begin{array}{lllllllllllllllll}
\lambda_1 & = & \upsilon_1 & & & & +\upsilon_5 & +\upsilon_6 & +\upsilon_7 & & & & +\upsilon_{11} & +\upsilon_{12} & +\upsilon_{13} & & +\upsilon_{15} \\
\lambda_2 & = & & \upsilon_2 & & & +\upsilon_5 & & & +\upsilon_8 & +\upsilon_9 & & +\upsilon_{11} & +\upsilon_{12} & & +\upsilon_{14} & +\upsilon_{15} \\
\lambda_3 & = & & & \upsilon_3 & & & +\upsilon_6 & & +\upsilon_8 & & +\upsilon_{10} & +\upsilon_{11} & & +\upsilon_{13} & +\upsilon_{14} & +\upsilon_{15} \\
\lambda_4 & = & & & & \upsilon_4 & & & +\upsilon_7 & & +\upsilon_9 & +\upsilon_{10} & & +\upsilon_{12} & +\upsilon_{13} & +\upsilon_{14} & +\upsilon_{15} \\
c_{12} & = & & & & & \upsilon_5 & & & & & & +\upsilon_{11} & +\upsilon_{12} & & & +\upsilon_{15} \\
c_{13} & = & & & & & & \upsilon_6 & & & & & +\upsilon_{11} & & +\upsilon_{13} & & +\upsilon_{15} \\
c_{14} & = & & & & & & & \upsilon_7 & & & & & +\upsilon_{12} & +\upsilon_{13} & & +\upsilon_{15} \\
c_{23} & = & & & & & & & & \upsilon_8 & & & +\upsilon_{11} & & & +\upsilon_{14} & +\upsilon_{15} \\
c_{24} & = & & & & & & & & & \upsilon_9 & & & +\upsilon_{12} & & +\upsilon_{14} & +\upsilon_{15} \\
c_{34} & = & & & & & & & & & & \upsilon_{10} & & & +\upsilon_{13} & +\upsilon_{14} & +\upsilon_{15}
\end{array} \tag{50}
$$

We can approximate the distribution of ξ in such a way that we take $\upsilon_1, \ldots, \upsilon_{15}$ as nonnegative variables and find a feasible solution to the LP: min $\sum_{1=1}^{15} \upsilon_i$ subject to (50) and $\upsilon \geq 0$. The general approximation scheme is:

$$
\begin{array}{c}
\min \sum_{i=1}^{k} \upsilon_i \\
\text{subject to} \\
\lambda = E(H_1 \kappa) \\
C = E(H_2 \kappa),
\end{array} \tag{51}
$$

where $\lambda = (\lambda_1 \ldots, \lambda_n)^T$ and $C = (c_{12} \ldots, c_{1n}, c_{23}, \ldots, c_{2n}, \ldots, c_{n-1,n})^T$. Note that equations correspoding to $c_{11}, c_{22}, \ldots, c_{nn}$ are unnecesary removed from (51) because those are the same as the equations in $\lambda = E(H, \upsilon)$.

Since all ξ_i components are partial sums of the components of $(\kappa_1, \ldots, \kappa_k)$, we can use the technique of Section 7.2 to find the new p-efficient point in the solution of algorithm of the stochastic network design problem.

The same technique can be used for any probability distribution that depends on a single parameter and has the property that the distribution of the sum of independent random variables is of the same type and the parameter of the sum is equal to the sum of the parameters of the terms. Examples of such distributions other than Poisson are: (a) the bionomial distribution, where $\kappa_i \upsilon \left\{ \binom{u_i}{j} p^j q^{u_i - j} \right\}_{j=0}^{u_i}$, $i = 1, \ldots, k$, where $q = 1 - p$ and p is fixed.

The above mentioned technique, to represent positivity correlated random variables as partial sums of independent random variables was first used in Prékopa, Szántai (1978), in connection with water resources and gamma distributed random variables. The matrix H appears also in Boolean probability bounding scheme (see Hailperin, 1965).

7 Summary of the Solution Algorithm

In this section we summarize the solution algorithm of the stochastic network design problem (29).

Table 1: Possible Values of the Random Demands

ξ_2	33	38	43	48	53	58	63	68	73	78
ξ_5	15	20	25	30	35	40	45	50	55	60

It consists of the following steps.

Step 1. Rewrite problem (29) in the form of (30).

Step 2. Generate a few p-efficient points for ξ and create the corresponding p-efficient points of the random vector (25). Initialize J_0 as the subscript set of these p-effcient points.

Step 3. Set up and solve problem (P_h) by a method that produces primal-dual feasible (optimal) basis. Let α designate the optimal dual vector.

Step 4. Solve problem (39) to check if an entering variable to (P_h) exists, i.e. (33) holds. If it is not the case, then go to Step 5. If (33) holds then we find a new p-effcient point, form the union of j_h and the new p-efficient point, to obtain j_{h+1} and define (P_{h+1}). Go to Step 3.

Step 5. Stop, the optimal solution of problem (P_h) is the optimal solution of problem (P).

Finding a new p-efficient point means the solution of problem (40), if the components of ξ, are stochastically dependent. If the components of ξ are independent, then to find new p-efficient point for ξ in problem (25) is a multiple choice knapsack problem that we solve by the algorithm in Section 7.2.

8 Illustrative Example

Examle 8.1. *This example is an 8 node network with only 2 random demand nodes (Node 2 and Node 5) which are both binomially distributed on arithmetic sequences. The demands are assumed to be independent. Among 161 non-eliminated inequalities (see Appendix A) only 136 include at least one of the two demands. The 136 inequalities are stochastic and the remaining 25 are deterministic constraints. Table 1 provides us with the possible values of the random demands at Nodes 2 and 5. The associated probability distributions are presented in Table 2.*

Table 2: Associated probabilities (Binomial), cdf

F_2 $(n=9,\ p=0.47)$	0.004	0.038	0.149	0.361	0.621	0.83	0.950	0. 990	0.999	1
F_5 $(n=9,\ p=0.47)$	0.003	0.029	0.123	0.316	0.573	0.80	0.936	0. 987	0.998	1

Since the bionomial probability function is logconcave, both F_2 and F_5 are logconcave discrete function. The Stochastic Programming Problem to be solved is: ((S) means the feasibility inequality corresponding to $S \subset N$):

$$\min c^T x$$
$$\text{subject to}$$
$$P(y(S,\bar{S}) \geq d(S),\ S \subset N_1,\ (S) \text{ non-eliminated}) \geq 0.95,$$
$$\text{where } N_1 \text{ is the collection of the nodes with random demand,}$$
$$y(S,\bar{S}) \geq d(S),\ S \subset N_2,\ N_2 = N \setminus N_1$$
$$d(S) = \textstyle\sum_{i \in S}(\xi_i - x_i)$$
$$l_i \leq x_i \leq u_i \text{ for } i = 1, \ldots, 8,$$

$$\text{where } l_i = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},\ u_i = \begin{bmatrix} 100 \\ 100 \\ 100 \\ 100 \\ 100 \\ 100 \\ 100 \\ 100 \end{bmatrix},\ c = \begin{bmatrix} 2 \\ 3 \\ 4 \\ 2 \\ 1 \\ 1 \\ 7 \\ 4 \end{bmatrix}$$

and $x = (x_1,\ x_2,\ x_3,\ x_4, x_5,\ x_6,\ x_7,\ x_8)^T$ is the decision vector. The solution steps are the following.

Step 1. Rewrite problem in the form of (30).

Step 2. $V_1 = (z^{(1)}) = \binom{78}{45}$.

Step 3. Set up and solve problem (P_1) by a method that produces primal-dual feasible (optimal) basis. The optimal solution is: $x = \begin{bmatrix} 89 & 100 & 58 & 100 & 100 & 87 & 34 & 47 \end{bmatrix}^T$ with optimum value= 1523. Let α designate the optimal dual vector.

Step 4.

Iteration 1: Solve problem (37). The optimal solution is $z^{(2)} = \binom{73}{55}$. Since (33) holds, include the new p-efficient point (26) that we obtain for the random vector (25), into (P_1), define (P_2) with $V_2 = (V_1, \binom{73}{55})$.

Iteration 2: Solve problem (37). The optimal solution is $z^{(3)} = \binom{68}{60}$. Since (33) holds, include the new p-efficient point into (P_1), define (P_2) with (P_2) with $V_3 = (V_2, \binom{68}{60})$.

Iteration 3: Solve problem (37). The optimal solution is $z^{(4)} = \binom{63}{60}$. Since (33) holds, include the new p-efficient point into (P_1), define (P_2) with (P_2) with $V_4 = (V_3, \binom{63}{60})$.

Iteration 4: Solve problem (37). The optimal solution is $z^{(5)} = \binom{63}{45}$. Since (33) holds, include the new p-efficient point into (P_1), define (P_2) with (P_2) with $V_5 = (V_4, \binom{63}{45})$.

Iteration 5: Solve problem (37). The optimal solution is $z^{(6)} = \binom{68}{55}$. Since (33) holds, include the new p-efficient point into (P_1), define (P_2) with (P_2) with $V_6 = (V_5, \binom{68}{55})$.

Iteration 6: Solve problem (37). The optimal solution is $z^{(7)} = \binom{78}{50}$. Since (33) holds, include the new p-efficient point into (P_1), define (P_2) with (P_2) with $V_7 = (V_6, \binom{78}{50})$.

Iteration 7: Solve problem (37). The optimal solution is $z^{(8)} = \binom{68}{45}$. Since (33) holds, include the new p-efficient point into (P_1), define (P_2) with (P_2) with $V_8 = (V_7, \binom{68}{45})$.

Iteration 8: Solve problem (37). The optimal solution is $z^{(9)} = \binom{73}{60}$. Since (33) holds, include the new p-efficient point into (P_1), define (P_2) with (P_2) with $V_9 = (V_8, \binom{73}{60})$.

Iteration 9: Solve problem (37). Solve problem (37). The optimal solution is $z^{(10)} = \binom{58}{60}$. Since (33) holds, include the new p-efficient point into (P_1), define (P_2) with (P_2) with $V_10 = (V_9, \binom{58}{60})$.

Step 5. Equation (33) has equal values on both sides of the inequality therefore algorithm terminates. Optimal solution is obtained; $x = \begin{bmatrix} 89 & 100 & 44.785 & 100 & 100 & 87 & 34 & 45.215 \end{bmatrix}^T$ with optimum value= 1463.

If we solve the same problem by using existing muliptle choice knapsack solution algorithms, we obtaine the same optimal soltuion with following pleps: $\binom{78}{45}, \binom{63}{50}, \binom{68}{45}, \binom{78}{40}, \binom{63}{45}, \binom{63}{60}, \binom{58}{50}, \binom{58}{60}$.

Examle 8.2. *This example is an 8 node network, where all nodes have random demands with binomial distribution on arithmetic sequences. The number of non-eliminated 161 inequalities is (see Appendix A), those are the stochastic constraints. Table 3 provides us with the possible values of the random demands of Node 1 thorough Node 8. The associated probability distributions can be found in Table 4.*

Table 3: Possible Values of the Random Demands

ξ_1	34	39	44	49	54	59	64	69	74	79
ξ_2	33	38	43	48	53	58	63	68	73	78
ξ_3	17	22	27	32	37	42	47	52	57	62
ξ_4	33	38	43	48	53	58	63	68	73	78
ξ_5	15	20	25	30	35	40	45	50	55	60
ξ_6	10	15	20	25	30	35	40	45	50	55
ξ_7	15	20	25	30	35	40	45	50	55	60
ξ_8	25	30	35	40	45	50	55	60	65	70

Table 4: Associated probabilities (Binomial probability functions)

$p(1)$ $(n = 9,\ p = 0.4)$	0.01	0.0704	0.2316	0.4824	0.7332	0.9004	0.9747	0.9959	0.9994	1
$p(2)$ $(n = 9,\ p = 0.45)$	0.0046	0.0385	0.1494	0.3612	0.6212	0.8339	0.9499	0.9905	0.9988	1
$p(3)$ $(n = 9,\ p = 0.5)$	0.0019	0.0194	0.0897	0.2537	0.4997	0.7457	0.9097	0.98	0.9975	1
$p(4)$ $(n = 9,\ p = 0.6)$	0.0002	0.0037	0.0249	0.0992	0.2664	0.5172	0.768	0.9292	0.9896	1
$p(5)$ $(n = 9,\ p = 0.48)$	0.0027	0.0257	0.1109	0.2945	0.5488	0.7835	0.9279	0.985	0.9981	1
$p(6)$ $(n = 9,\ p = 0.35)$	0.0207	0.121	0.3371	0.6087	0.828	0.9461	0.9885	0.9982	0.9995	1
$p(7)$ $(n = 9,\ p = 0.42)$	0.0074	0.0558	0.196	0.4329	0.6902	0.8765	0.9664	0.9943	0.9993	1
$p(8)$ $(n = 9,\ p = 0.38)$	0.0135	0.0881	0.2711	0.5329	0.7735	0.921	0.9812	0.997	0.9994	1

All eight distributions are binomial hence all discrete functions F_i, $i = 1, 2, 3, 4, 5, 6, 7, 8$ are logconcave in the supports of ξ_i, $i = 1, 2, 3, 4, 5, 6, 7, 8$, respectively. The Stochastic

Programming Problem to be solved is:

$$\min c^T x$$
$$\text{subject to}$$
$$P(y(S,\bar{S})) \geq d(S),\ (S) \text{ non-eliminated}) \geq 0.95,\ \ S \subset N,$$
$$\text{where } N \text{ is the collection of the nodes with random demand,}$$
$$\text{and } d(S) = \textstyle\sum_{i \in S}(\xi_i - x_i),\ \text{ for } S \subset N.$$

$$l_i = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},\quad u_i = \begin{bmatrix} 100 \\ 100 \\ 100 \\ 100 \\ 100 \\ 100 \\ 100 \\ 100 \end{bmatrix},\quad c = \begin{bmatrix} 2 \\ 3 \\ 4 \\ 2 \\ 1 \\ 1 \\ 7 \\ 4 \end{bmatrix}$$

and $x = (x_1,\ x_2,\ x_3,\ x_4, x_5,\ x_6,\ x_7,\ x_8)^T$ is the decision vector. The solution steps are the following.

Step 1. Rewrite problem in the form of (6.2).

Step 2. $V_1 = (z^{(1)}) = \begin{pmatrix} 64 \\ 68 \\ 57 \\ 73 \\ 55 \\ 45 \\ 55 \\ 65 \end{pmatrix}$.

Step 3. Set up and solve problem (P_0) by a method that produces primal-dual feasible (optimal) basis. Optimal basis$x = \begin{bmatrix} 74 & 60 & 40 & 67 & 86.1799 & 56.8201 & 47 & 51 \end{bmatrix}^T$ with optimal value= 1298.

Let α designate the optimal dual vector.

Step 4.

Iteration 1:Solve problem (37). The optimal solution is $z^{(2)} = \begin{pmatrix} 64 \\ 73 \\ 62 \\ 78 \\ 50 \\ 45 \\ 50 \\ 65 \end{pmatrix}$. Since (33) holds, include the new p-efficient point (26) that we obtain for the random vector (25), into (P_1), define (P_2) with $V_2 = \left(V_1, \begin{pmatrix} 64 \\ 73 \\ 62 \\ 78 \\ 50 \\ 45 \\ 50 \\ 65 \end{pmatrix} \right)$.

Iteration 2: Solve problem (37). The optimal solution is $z^{(3)} = \begin{pmatrix} 69 \\ 68 \\ 57 \\ 78 \\ 50 \\ 40 \\ 55 \\ 60 \end{pmatrix}$. Since (33) holds, include the new p-efficient point (26) that we obtain for the random vector (25), into (P_1), define (P_2) with $V_3 = \left(V_2, \begin{pmatrix} 69 \\ 68 \\ 57 \\ 78 \\ 50 \\ 40 \\ 55 \\ 60 \end{pmatrix} \right)$.

Iteration 3: Solve problem (37). The optimal solution is $z^{(4)} = \begin{pmatrix} 59 \\ 73 \\ 62 \\ 73 \\ 55 \\ 40 \\ 55 \\ 55 \end{pmatrix}$. Since (33) holds, include the new p-efficient point (26) that we obtain for the random vector (25), into (P_1), define (P_2) with $V_4 = \left(V_3, \begin{pmatrix} 59 \\ 73 \\ 62 \\ 73 \\ 55 \\ 40 \\ 55 \\ 55 \end{pmatrix} \right)$.

Iteration 4:Solve problem (37). The optimal solution is $z^{(5)} = \begin{pmatrix} 74 \\ 73 \\ 62 \\ 73 \\ 55 \\ 40 \\ 55 \\ 55 \end{pmatrix}$. Since (33) holds, include the new p-efficient point (26) that we obtain for the random vector (25), into (P_1), define (P_2) with $V_5 = \left(V_4, \begin{pmatrix} 74 \\ 73 \\ 62 \\ 73 \\ 55 \\ 40 \\ 55 \\ 55 \end{pmatrix} \right)$.

Iteration 5: :Solve problem (37). The optimal solution is $z^{(6)} = \begin{pmatrix} 79 \\ 73 \\ 52 \\ 73 \\ 60 \\ 40 \\ 50 \\ 60 \end{pmatrix}$. Since (33) holds, include the new p-efficient point (26) that we obtain for the random vector (25), into (P_1), define (P_2) with $V_6 = \left(V_5, \begin{pmatrix} 79 \\ 73 \\ 52 \\ 73 \\ 60 \\ 40 \\ 50 \\ 60 \end{pmatrix} \right)$.

Iteration 6: :Solve problem (37). The optimal solution is $z^{(7)} = \begin{pmatrix} 74 \\ 68 \\ 57 \\ 73 \\ 50 \\ 50 \\ 50 \\ 60 \end{pmatrix}$. Since (33) holds, include the new p-efficient point (26) that we obtain for the random vector (25), into (P_1), define (P_2) with $V_7 = \left(V_6, \begin{pmatrix} 74 \\ 68 \\ 57 \\ 73 \\ 50 \\ 50 \\ 50 \\ 60 \end{pmatrix} \right)$.

Step 5.Equation (33) has equal values on both sides of the inequality therefore algorithm terminates. Optimal solution is obtained; $x = \begin{bmatrix} 69 & 60 & 40 & 67 & 81.0192 & 61.9808 & 42 & 46 \end{bmatrix}^T$ with optimal value= 1233.

If we solve the same problem by using existing muliptle choice knapsack solution algorithms, we obtaine the same optimal soltuion with following pleps: $\begin{pmatrix} 64\\68\\57\\73\\55\\45\\55\\65 \end{pmatrix}$, $\begin{pmatrix} 69\\68\\57\\78\\50\\40\\55\\60 \end{pmatrix}$, $\begin{pmatrix} 79\\78\\52\\73\\60\\40\\50\\60 \end{pmatrix}$, $\begin{pmatrix} 74\\73\\62\\73\\55\\40\\55\\55 \end{pmatrix}$, $\begin{pmatrix} 69\\68\\62\\78\\60\\40\\50\\55 \end{pmatrix}$, $\begin{pmatrix} 74\\68\\57\\73\\50\\50\\50\\60 \end{pmatrix}$.

Remark 8.1. *The condition in Theorem 3.1 holds true for the p-efficient points encountered in the solution algorithm in examples 1 and 2.*

9 Conclusion and Further Research

We have formulated a static, one-stage stochastic network design problem, where joint probabilistic constraint is used to ensure network reliability. It means that all demands should be met on a prescribed probability level that is near 1 in practice. The probabilistic constraint is based on the probability that the Gale–Hoffman feasibility inequalities are satisfied but elimination technique is applied to remove the redundant ones. The concept of a p-efficient point is used to reformulate the reliability constraint. A novel generation technique, that can be used in the network design context, of the p-efficient points is presented, where the much smaller size p-efficient points of the random demand vector provides us with that correspond to the non-eliminated feasibility inequalities.

Two algorithms are proposed for the solution of the network design problem, depending on if all p-efficient points are available right at the beginning of the solution algorithm or they are generated simultaneously with it. If the random demands at the nodes are independent, then the solution of a multiple choice knapsack problem (MCKP) provides us with the new p-efficient point needed in the algorithm. There are several methods to solve the MCKP but we propose to use our version that is an improvement of the well-known greedy algorithm. Two numerical examples are presented in connection with an 8-node network. In the first one only two demands are random, in the second one all are random. Our stochastic network design problem is applicable for various stochastic networks: power, water supply, transportation,financial and other networks. Our future plans include a reformulation of the problem as a two-stage programming under uncertanity stochastic programming problem with network recourse, where a probabilistic constraint ensures the reliability of the second stage problem. The new problem will include long term operational cost in the objective function and will allow for accounting for losses on the arcs which are the transmission lines in case of a power system expansion problem. Even though the problem is significantly more complex than the original two-stage problem without probabilistic constraint, our methodology will make it possible to solve it by the use of Bender's decomposition.

References

[Charles et al.(1958)] A. Charnes, W. W. Cooper, G. H. Symonds 1958. Cost horizons and certainty equivalents: an approach to stochastic programming of heating oil. *Management Science* **4** , 235–263.

[Dentcheva et al.(2000)] Dentcheva, D., A. Prékopa, A. Ruszczyński. 2000. Concavity and efficient points of discrete distributions in probabilistic programming. *Mathematical Programming* **Ser. A 89:** 66–77.

[Earey(1967)] Esary, J. D., F. Porschan, D. W. Walkup. 1967. Association of random variables with applications. *The Annals of Mathematical Statistics* **38**, 1466–1474.

[Fábián et al.(2002)] Fábián, C., A. Prékopa, O. Ruff-Fiedler. 2002. On a dual method for a specially structured linear programming problem with application to stochastic programming. *Optimization Methods and Software* **17**, 445–492.

[Ford(1962)] Ford, Jr., L.R., D.R. Fulkerson. 1962. *Flows in Networks.* Princeton University Press, Princeton, N.J.

[Friedler et al.(1995)] Fiedler, O., A. Prékopa, Cs. Fábián 2002. On a dual method for a specially structured linear programming problem with application to stochastic programming. *Optimization Methods and Software* **17**, 445–492.

[Gale(1957)] Gale, D. 1957. A theorem on flows in network. *Pacific J. Math.* **7,** 1073–1082

[Hailperin(1960)] Hailperin, Th. 1965. Best possible inequalities for the probability of a logical function of events. *The American Mathematical Monthly* **72**, 343–359.

[Hoffman(1960)] Hoffman, A.J. 1960. Some recent applications of the theory of linear inequalities to external combinatorial analysis. Proceedings of Symposia in Applied Mathematics, Vol. X. Combinatorial Analysis *American Mathematical Society* 113–127.

[Bruce L. Miller, Harvey M. Wagner] Bruce L. Miller, Harvey M. Wagner. 1965. Chance constrained programming with joint constraints. *Opeartions Research* **13**, 930–945.

[Pisinger (1995)] Pisinger, D. 1995. A Minimal algorithm for the multiple-choice knapsack problem. *European Journal of Operations Research* **83** 394–410.

[Prékopa(1970)] Prékopa, A. 1970. On probabilistic constrained programming *Princetion University Press* , 113–138.

[Prékopa(1971)] Prékopa, A. 1971. Logarithmic concave measures with applications to stochastic programming. *Acta Sci. Math. (Szeged)* **32**, 301–316.

[Prékopa(1973)] Prékopa, A. 1973 a. Contributions to the theory of stochastic programming. *Mathematical Programming* **4**, 202–221.

[Prékopa(1973)] Prékopa, A. 1973 b. On logarithmic concave measures and functions. *Acta Sci. Math. (Szeged)* **34**, 339–343.

[Prékopa(1978)] Prékopa, A., T. Szántai. 1978. A new multivariate gamma distribution and its fitting to empirical streamflow data. *Water Resources Research* **14**, 19–24.

[Prékopa(1980)] Prékopa, A. 1980. Network planning using two-stage programming under uncertainty. *Recent Results in Stochastic Programming. Lecture Notes in Economics and Mathematical Systems 179* (P. Kall, A. Prkopa, eds, Springer, Berlin), 216–237.

[Prékopa(1990 a)] Prékopa, A. 1990 a. Dual method for the solution of a one-stage stochastic programming problem with random RHS obeying a discrete probability distribution. *ZOR- Methods and Models of Operations Research* 34, 441–461.

[Prékopa(1990 b)] Prékopa, A. 1990 b. Totally positive linear programming problems. *Functional Analysis, Optimization and Mathematical Economics. A Collection of Papers Dedicated to the Memory of L. V. Kantorovich (L. J. Leifman, et. al).* (Oxford University Press), New York, Oxford, 197–207.

[Prékopa(1995)] Prékopa, A. 1995. Stochastic programming. *Kluwer Scientific Publishers*, Boston, 84, 107, 180, 210, 315, 363, 364, 368, 369, 375, 387, 504, 507, 510.

[Prékopa(2007)] Prékopa, A. 2007. On the relationship between probabilistic constrained, disjunctive and multiobjective programming. *RUTCOR Research Report* 07–2007.

[Prékopa et al.(1991)] Prékopa, A., E. Boros 1991. On the existence of a feasible flow in a stochastic transportation network. *Operations Research* **39,** 119–129.

[Prékopa et al.(1998)] Prékopa, A., B. Vizvári, T. Badics 1998. Programming under probabilistic constraint with discrete random variable. *New Trends in Mathematical Programming* 235–255.

[1] Stiemke, E. 1915. Über positive lösungen homogener linearer gleichungen. *Mathematische Annalen* **76,** 340-342.

[Thapalia et al.(2010)] Thapalia, B.K., T.G. Crainic, M. Kaut, S.W. Wallace 2010. Single-commodity stochastic network design with multiple sources and sinks. *Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation* CIRRELT-2010-59.

[Thapalia et al.(2010)] Thapalia, B.K., M. Kaut, S.W. Wallace, T.G. Crainic 2010. Single source single-commodity stochastic network design. *Computational Management Science*, Springer, Berlin, 1619.

[Vizvári(2002)] Vizvári, B., 2002. The Integer programming background of a stochastic integer programming algorithm of Dentcheva-Prèkopa-Ruszczyński. *Optimization Methods and Software* **17,** 543–559

[Wallace et al.(1993)] Wallace, S.W., R.J.-B. Wets 1993. The facets of the polyhedral set determined by the Gale- Hoffman inequalities. *Mathematical Programming* **62,** 215–222.

[Yazici et al.(2007)] Yazici, M.A., K. Ozbay 2007. Impact of Probabilistic Road Capacity Constraints on the Spatial Distribution of Hurricane Evacuation Shelter Capacities. *Transportation Research Record* Vol.2022, pp55–62.

A

The number of eliminated inequalities by network topology: 94. Their numbers are: 11, 12, 13, 14, 15, 19, 20, 22, 24, 25, 26, 29, 33, 36, 40, 41, 43, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 60, 61, 65, 70, 72, 73, 74, 75, 76, 79, 80, 81, 83, 91, 98, 99, 102, 103, 104, 112, 118, 120, 125, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 144, 145, 146, 148, 150, 155, 159, 160, 165, 178, 184, 185, 186, 187, 188, 189, 190, 193, 194, 195, 203, 209, 211, 216, 226, 228, 241.

Table 5: Gale–Hoffman inequalities for the 8-node network ($2^8 - 1 = 255$ inequalities)

Number	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Node 8
1	1	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0
3	0	0	1	0	0	0	0	0
4	0	0	0	1	0	0	0	0
5	0	0	0	0	1	0	0	0
6	0	0	0	0	0	1	0	0
7	0	0	0	0	0	0	1	0
8	0	0	0	0	0	0	0	1
9	1	1	0	0	0	0	0	0
10	1	0	1	0	0	0	0	0
11	1	0	0	1	0	0	0	0
12	1	0	0	0	1	0	0	0
13	1	0	0	0	0	1	0	0
14	1	0	0	0	0	0	1	0
15	1	0	0	0	0	0	0	1
16	0	1	1	0	0	0	0	0
17	0	1	0	1	0	0	0	0
18	0	1	0	0	1	0	0	0
19	0	1	0	0	0	1	0	0
20	0	1	0	0	0	0	1	0
21	0	1	0	0	0	0	0	1
22	0	0	1	1	0	0	0	0
23	0	0	1	0	1	0	0	0
24	0	0	1	0	0	1	0	0
25	0	0	1	0	0	0	1	0
26	0	0	1	0	0	0	0	1
27	0	0	0	1	1	0	0	0
28	0	0	0	1	0	1	0	0
29	0	0	0	1	0	0	1	0
30	0	0	0	1	0	0	0	1
31	0	0	0	0	1	1	0	0

Continued on next page

Number	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Node 8
32	0	0	0	0	1	0	1	0
33	0	0	0	0	1	0	0	1
34	0	0	0	0	0	1	1	0
35	0	0	0	0	0	1	0	1
36	0	0	0	0	0	0	1	1
37	1	1	1	0	0	0	0	0
38	1	1	0	1	0	0	0	0
39	1	1	0	0	1	0	0	0
40	1	1	0	0	0	1	0	0
41	1	1	0	0	0	0	1	0
42	1	1	0	0	0	0	0	1
43	1	0	1	1	0	0	0	0
44	1	0	1	0	1	0	0	0
45	1	0	1	0	0	1	0	0
46	1	0	1	0	0	0	1	0
47	1	0	1	0	0	0	0	1
48	1	0	0	1	1	0	0	0
49	1	0	0	1	0	1	0	0
50	1	0	0	1	0	0	1	0
51	1	0	0	1	0	0	0	1
52	1	0	0	0	1	1	0	0
53	1	0	0	0	1	0	1	0
54	1	0	0	0	1	0	0	1
55	1	0	0	0	0	1	1	0
56	1	0	0	0	0	1	0	1
57	1	0	0	0	0	0	1	1
58	0	1	1	1	0	0	0	0
59	0	1	1	0	1	0	0	0
60	0	1	1	0	0	1	0	0
61	0	1	1	0	0	0	1	0
62	0	1	1	0	0	0	0	1
63	0	1	0	1	1	0	0	0

Continued on next page

Number	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Node 8
64	0	1	0	1	0	1	0	0
65	0	1	0	1	0	0	1	0
66	0	1	0	1	0	0	0	1
67	0	1	0	0	1	1	0	0
68	0	1	0	0	1	0	1	0
69	0	1	0	0	1	0	0	1
70	0	1	0	0	0	1	1	0
71	0	1	0	0	0	1	0	1
72	0	1	0	0	0	0	1	1
73	0	0	1	0	0	0	1	1
74	0	0	1	0	0	1	0	1
75	0	0	1	0	0	1	1	0
76	0	0	1	0	1	0	0	1
77	0	0	1	0	1	0	1	0
78	0	0	1	0	1	1	0	0
79	0	0	1	1	0	0	0	1
80	0	0	1	1	0	0	1	0
81	0	0	1	1	0	1	0	0
82	0	0	1	1	1	0	0	0
83	0	0	0	1	0	0	1	1
84	0	0	0	1	0	1	0	1
85	0	0	0	1	0	1	1	0
86	0	0	0	1	1	0	0	1
87	0	0	0	1	1	0	1	0
88	0	0	0	1	1	1	0	0
89	0	0	0	0	1	1	1	0
90	0	0	0	0	1	1	0	1
91	0	0	0	0	1	0	1	1
92	0	0	0	0	0	1	1	1
93	0	0	0	0	1	1	1	1
94	0	0	0	1	0	1	1	1
95	0	0	0	1	1	0	1	1

Continued on next page

Number	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Node 8
96	0	0	0	1	1	1	0	1
97	0	0	0	1	1	1	1	0
98	0	0	1	0	0	1	1	1
99	0	0	1	0	1	0	1	1
100	0	0	1	0	1	1	0	1
101	0	0	1	0	1	1	1	0
102	0	0	1	1	0	0	1	1
103	0	0	1	1	0	1	0	1
104	0	0	1	1	0	1	1	0
105	0	0	1	1	1	0	0	1
106	0	0	1	1	1	0	1	0
107	0	0	1	1	1	1	0	0
108	0	1	0	0	0	1	1	1
109	0	1	0	0	1	0	1	1
110	0	1	0	0	1	1	0	1
111	0	1	0	0	1	1	1	0
112	0	1	0	1	0	0	1	1
113	0	1	0	1	0	1	0	1
114	0	1	0	1	0	1	1	0
115	0	1	0	1	1	0	0	1
116	0	1	0	1	1	0	1	0
117	0	1	0	1	1	1	0	0
118	0	1	1	0	0	0	1	1
119	0	1	1	0	0	1	0	1
120	0	1	1	0	0	1	1	0
121	0	1	1	0	1	0	0	1
122	0	1	1	0	1	0	1	0
123	0	1	1	0	1	1	0	0
124	0	1	1	1	0	0	0	1
125	0	1	1	1	0	0	1	0
126	0	1	1	1	0	1	0	0
127	0	1	1	1	1	0	0	0

Continued on next page

Number	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Node 8
128	1	0	0	0	0	1	1	1
129	1	0	0	0	1	0	1	1
130	1	0	0	0	1	1	0	1
131	1	0	0	0	1	1	1	0
132	1	0	0	1	0	0	1	1
133	1	0	0	1	0	1	0	1
134	1	0	0	1	0	1	1	0
135	1	0	0	1	1	0	0	1
136	1	0	0	1	1	0	1	0
137	1	0	0	1	1	1	0	0
138	1	0	1	0	0	0	1	1
139	1	0	1	0	0	1	0	1
140	1	0	1	0	0	1	1	0
141	1	0	1	0	1	0	0	1
142	1	0	1	0	1	0	1	0
143	1	0	1	0	1	1	0	0
144	1	0	1	1	0	0	0	1
145	1	0	1	1	0	0	1	0
146	1	0	1	1	0	1	0	0
147	1	0	1	1	1	0	0	0
148	1	1	0	0	0	0	1	1
149	1	1	0	0	0	1	0	1
150	1	1	0	0	0	1	1	0
151	1	1	0	0	1	0	0	1
152	1	1	0	0	1	0	1	0
153	1	1	0	0	1	1	0	0
154	1	1	0	1	0	0	0	1
155	1	1	0	1	0	0	1	0
156	1	1	0	1	0	1	0	0
157	1	1	0	1	1	0	0	0
158	1	1	1	0	0	0	0	1
159	1	1	1	0	0	0	1	0

Continued on next page

Number	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Node 8
160	1	1	1	0	0	1	0	0
161	1	1	1	0	1	0	0	0
162	1	1	1	1	0	0	0	0
163	0	0	0	1	1	1	1	1
164	0	0	1	0	1	1	1	1
165	0	0	1	1	0	1	1	1
166	0	0	1	1	1	0	1	1
167	0	0	1	1	1	1	0	1
168	0	0	1	1	1	1	1	0
169	0	1	0	0	1	1	1	1
170	0	1	0	1	0	1	1	1
171	0	1	0	1	1	0	1	1
172	0	1	0	1	1	1	0	1
173	0	1	0	1	1	1	1	0
174	0	1	1	0	0	1	1	1
175	0	1	1	0	1	0	1	1
176	0	1	1	0	1	1	0	1
177	0	1	1	0	1	1	1	0
178	0	1	1	1	0	0	1	1
179	0	1	1	1	0	1	0	1
180	0	1	1	1	0	1	1	0
181	0	1	1	1	1	0	0	1
182	0	1	1	1	1	0	1	0
183	0	1	1	1	1	1	0	0
184	1	0	0	0	1	1	1	1
185	1	0	0	1	0	1	1	1
186	1	0	0	1	1	0	1	1
187	1	0	0	1	1	1	0	1
188	1	0	0	1	1	1	1	0
189	1	0	1	0	0	1	1	1
190	1	0	1	0	1	0	1	1
191	1	0	1	0	1	1	0	1

Continued on next page

Number	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Node 8
192	1	0	1	0	1	1	1	0
193	1	0	1	1	0	0	1	1
194	1	0	1	1	0	1	0	1
195	1	0	1	1	0	1	1	0
196	1	0	1	1	1	0	0	1
197	1	0	1	1	1	0	1	0
198	1	0	1	1	1	1	0	0
199	1	1	0	0	0	1	1	1
200	1	1	0	0	1	0	1	1
201	1	1	0	0	1	1	0	1
202	1	1	0	0	1	1	1	0
203	1	1	0	1	0	0	1	1
204	1	1	0	1	0	1	0	1
205	1	1	0	1	0	1	1	0
206	1	1	0	1	1	0	0	1
207	1	1	0	1	1	0	1	0
208	1	1	0	1	1	1	0	0
209	1	1	1	0	0	0	1	1
210	1	1	1	0	0	1	0	1
211	1	1	1	0	0	1	1	0
212	1	1	1	0	1	0	0	1
213	1	1	1	0	1	0	1	0
214	1	1	1	0	1	1	0	0
215	1	1	1	1	0	0	0	1
216	1	1	1	1	0	0	1	0
217	1	1	1	1	0	1	0	0
218	1	1	1	1	1	0	0	0
219	0	0	1	1	1	1	1	1
220	0	1	0	1	1	1	1	1
221	0	1	1	0	1	1	1	1
222	0	1	1	1	0	1	1	1
223	0	1	1	1	1	0	1	1

Continued on next page

Number	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Node 8
224	0	1	1	1	1	1	0	1
225	0	1	1	1	1	1	1	0
226	1	0	0	1	1	1	1	1
227	1	0	1	0	1	1	1	1
228	1	0	1	1	0	1	1	1
229	1	0	1	1	1	0	1	1
230	1	0	1	1	1	1	0	1
231	1	0	1	1	1	1	1	0
232	1	1	0	0	1	1	1	1
233	1	1	0	1	0	1	1	1
234	1	1	0	1	1	0	1	1
235	1	1	0	1	1	1	0	1
236	1	1	0	1	1	1	1	0
237	1	1	1	0	0	1	1	1
238	1	1	1	0	1	0	1	1
239	1	1	1	0	1	1	0	1
240	1	1	1	0	1	1	1	0
241	1	1	1	1	0	0	1	1
242	1	1	1	1	0	1	0	1
243	1	1	1	1	0	1	1	0
244	1	1	1	1	1	0	0	1
245	1	1	1	1	1	0	1	0
246	1	1	1	1	1	1	0	0
247	1	1	1	1	1	1	1	0
248	1	1	1	1	1	1	0	1
249	1	1	1	1	1	0	1	1
250	1	1	1	1	0	1	1	1
251	1	1	1	0	1	1	1	1
252	1	1	0	1	1	1	1	1
253	1	0	1	1	1	1	1	1
254	0	1	1	1	1	1	1	1
255	1	1	1	1	1	1	1	1