IMA Journal of Management Mathematics Advance Access published March 6, 2007

IMA Journal of Management Mathematics Page 1 of 31
doi:10.1093/imaman/dpm007

The SMPS format explained

HORAND I. GASSMANN†
School of Business Administration, Dalhousie University, Halifax, Canada

AND

BJARNI KRISTJÁNSSON‡
Maximal Software, Inc., 2111 Wilson Boulevard, Suite 700, Arlington, VA 22201, USA

[Accepted on 31 January 2007]

Recent extensions to the SMPS format have vastly increased the range of stochastic linear programs that can be expressed within the format. This paper illustrates some of the features of SMPS using sample problems from the literature. For each problem, we give the general mathematical formulation, a small illustrative instance and the SMPS core, time and stoch files.

Keywords: SMPS format; stochastic programming; input format; examples.

1. Introduction

Stochastic programming is an active area of research due to recent advances in computing power. However, for benchmarking and comparison of algorithms it is essential to have a way of exchanging test problem sets. The situation was best described by Klingman *et al.* (1974) who write: “One of the problems . . . in trying to benchmark codes based on different methodologies . . . [is] their lack of uniformity for input specification. This nonstandardization of problem specification . . . is most frustrating and has hampered benchmarking since researchers are reluctant to recode their input routines.” Klingman *et al.* (1974) were writing about network problems, but their remarks could justifiably be applied to stochastic programming as well.

The SMPS format is available to describe stochastic linear and quadratic programs (LP and QP, respectively). It is based on the well-known MPS format (Argonne National Laboratory, 1996), the *de facto* standard for linear programs, and has gone through several revisions (see Birge *et al.*, 1987; Edwards, 1988; Gassmann, 2005; Gassmann & Schweitzer, 2001).

While it is widely used, SMPS does not enjoy universal acceptance. Part of the reason is that the record-based structure of MPS is deemed to be overly rigid and limiting, but we feel that at least in part the reason is a lack of examples that describe the many and varied constructs of the SMPS format.

We will explain most of the current features of SMPS using sample problems. Many of these problems have appeared in the literature. Length restrictions prohibit the inclusion of complete examples in every instance, so in places we will only show the most salient features of a model. Data files giving the full examples can be downloaded from the first author’s web site (myweb.dal.ca/gassmann/RESEARCH.html).

†Email: hgassman@mgmt.dal.ca
‡Email: bjarni@maximalsoftware.com

Unlike other collections of stochastic programs (see, e.g. Felt *et al.*, 2001; Holmes), whose main focus is archiving and benchmarking, the problems offered in this paper have been selected mainly for variety, both of application area and of model type and features.

The structure of the paper is as follows: A brief introduction to stochastic programs and the SMPS format is presented in Section 2. The subsequent sections highlight different modelling possibilities. Sections 3 and 4 deal with explicit event trees, having deterministic and stochastic problem dimensions, respectively. Sections 5 and 6 illustrate the work with implicit event trees; Section 5 deals with both discrete and continuous univariate distributions and Section 6 features multidimensional blocks. The SMPS format also supports a network option; a problem of this type is shown in Section 7. The mixing of LP and network structure is explained in Section 8, and a simple recourse problem is constructed in Section 9. Section 10 shows probabilistic constraints and objectives. We end with some concluding remarks.

2. Problem formulation and a brief introduction to SMPS

There are many different types of stochastic programs. We repeat here a taxonomy (see Fig. 1) that was first given in Gassmann & Ireland (1996).

For our purposes, the most general formulation of the problem is as follows:

$$
\begin{aligned}
\text{Opt}\ & (c_1, c_2, \ldots, c_T)'(x_1, x_2, \ldots, x_T) \\
& + \frac{1}{2}(x_1, x_2, \ldots, x_T)' \begin{bmatrix} Q_{11} & Q_{12} & \ldots & Q_{1T} \\ Q_{21} & Q_{22} & \ldots & Q_{2T} \\ \vdots & \vdots & \ddots & \vdots \\ Q_{T1} & Q_{T2} & \ldots & Q_{TT} \end{bmatrix} (x_1, x_2, \ldots, x_T) \\
\text{s.t.}\ & A_{11}x_1 + A_{12}x_2 + \cdots + A_{1T}x_T \sim b_1, \\
& A_{21}x_1 + A_{22}x_2 + \cdots + A_{2T}x_T \triangle b_2, \\
& \quad \vdots \qquad\quad \vdots \qquad\qquad\qquad \vdots \qquad\quad \vdots \\
& A_{T1}x_1 + A_{T2}x_2 + \cdots + A_{TT}x_T \triangle b_T, \\
& l_i \leqslant x_i \leqslant u_i, \quad i = 1, \ldots, T, \\
& x_i \in R^{n_i^1} \times Z^{n_i^2}, \quad i = 1, \ldots, T.
\end{aligned}
$$

Constraint matrices A_{ij} with $i > j$ define so-called global (or linking) constraints; special algorithms are necessary to deal with them. If $A_{ij} = 0$ for $j > i + 1$, the problem is said to possess ‘staircase structure’.

All the data items except A_{11}, Q_{11}, c_1, b_1, l_1 and u_1 can be stochastic. The symbol ‘$\sim$’ stands for an arbitrary relation ($\leqslant$, =, $\geqslant$) and $\triangle$ indicates that constraints involving random variables may be required to hold for every possible realization, or with probability one, or subject to a probabilistic constraint. The last item includes 1D and multidimensional constraints of the form

$$\Pr\{B_{t1}x_1 + B_{t2}x_2 + \cdots + B_{tt}x_t \sim b_t\} \sim \alpha_t.$$

If all the ‘$\triangle$-type’ constraints are of this type, then the problem is termed a ‘chance-constrained problem’. Individual chance constraint problems are characterized by having only 1D chance constraints; a single multidimensional chance constraint (usually with $T = 2$ and no second-stage variables) defines a joint chance constraint problem.

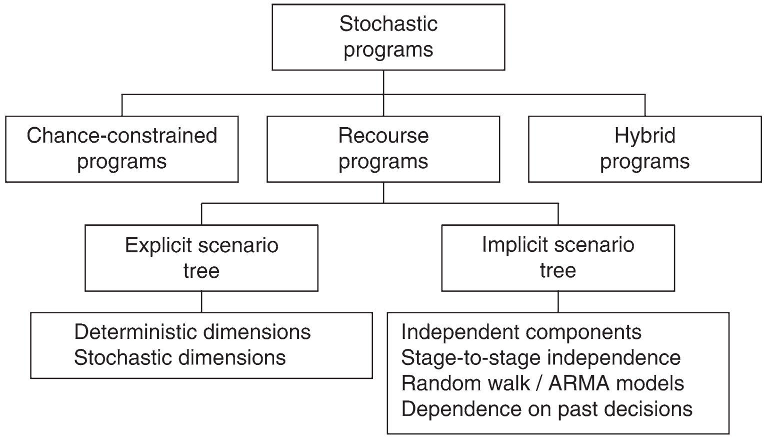

FIG. 1. A taxonomy of stochastic programming problems.

If on the other hand all the △-type constraints must hold surely or almost surely (and if $T > 1$), then the problem is termed a 'recourse problem'. In this case, at least one recourse decision follows each observation of random variables to allow the decision maker to correct any adverse effect of randomness—usually at a cost. Two-stage recourse problems in which the second-stage constraint matrix A_{22} is of the form $A_{22} = [I \ \ -I]$ are said to have 'simple recourse'. (Simple recourse can also be thought of as a penalty on the violation of a constraint in the presence of uncertainty.)

There is no enough space for a full description of the SMPS format here; for that the user is directed to Gassmann & Schweitzer (2001) or Gassmann (2005). However, we will present a very brief introduction.

The SMPS format makes use of three text files. All have set record structures, alternating header records that mark the start of various sections (in fixed order) with data records that hold the data for each section.

The core file fixes the problem dimensions and deterministic coefficients as well as the locations of all the stochastic coefficients. The core file may be in the usual MPS format (Argonne National Laboratory, 1996) or it may use a network format similar to Klingman *et al.* (1974). It is also possible to mix the two formats.

The time file describes the dynamic structure of the problem and breaks the data into stages. If the core file is given in time-ordered fashion, then this is a simple matter of recording the first row and column of each stage, otherwise a full list of rows and columns must be given along with the stage to which each of them belongs.

Finally, the stoch file gives the stochastic data. There are many different ways to present this information. The ultimate goal is to produce an event tree, and the two major ways this can be done use implicit and explicit constructions. (The two are mutually exclusive.) Other features of the stoch file include linear and quadratic penalties for violating a stochastic constraint, probabilistic constraints and objectives and integrated chance constraints.

This paper does not concern itself with the construction of appropriate scenarios, a topic treated, for instance, in Dupačová *et al.* (2000, 2003) and Pflug (2001). We assume that the relevant stochastic structure has been prepared beforehand, so that the entire stochastic program is ready to be cast in the SMPS format.

3. Scenarios

This is the most frequently used format in practice, due to its flexibility, which permits modelling of a variety of dependencies, both within and across time periods. This form of the stoch file can be used for recourse problems with fixed problem dimensions and leads to an explicit formulation of the event tree.

Every scenario is a path from the root of the event tree to one of the leaves. However, scenarios may share data items for several stages and thus be indistinguishable until the first data item is encountered. It is customary in this instance to specify only one set of data items (and only one set of induced decisions) until the branch point occurs. One 'parent' scenario holds the information, while the 'children' branching from it are thought to spring into existence only after the branch point. This idea of labelling the data is quite old and goes back to some work by Lane & Hutchinson (1980).

Each child scenario inherits all the parent scenario's values unless specifically replaced in the stoch file. Hence, the stoch file needs to record only those values that differ from the parent scenario.

We illustrate the use of the 'SCENARIOS' format with an asset management problem taken from the text by Birge & Louveaux (1997). A decision maker has to determine the optimal investment levels in various investment opportunities subject to uncertain returns. At predetermined intervals, the assets can be redistributed, based on the returns realized to date. The objective is to meet a certain investment goal at the end of the planning horizon; falling short of the financial goal carries a penalty. The mathematical formulation of this problem is as follows:

$$
\begin{aligned}
\min \sum_{s\in S_T} & p_s[4w(s) - y(s)] \\
\text{s.t.} \sum_{i\in I} & x_{1i} = b, \\
& -\sum_{i\in I} r_2(s)x_{1i} + \sum_{i\in I} x_{2i}(s) = 0, \quad s_2 \in S_2, \\
& -\sum_{i\in I} r_{t+1}(s_{t+1})x_{ti}(s_t) + \sum_{i\in I} x_{t+1,i}(s_{t+1}) = 0, \quad s_t \in S_t,\ \ s_{t+1} \in \sigma(s_{t+1}),\ t = 2,\dots,T-1, \\
& \sum_{i\in I} r_T(s_T)x_{T-1,i}(s_{T-1}) - y(s_T) + w(s_T) = C, \quad s_{T-1} \in S_{T-1},\ \ s_T \in \sigma(s_{T-1}), \\
& x_{ti} \geqslant 0,\ \ t = 1,\dots,T,\ i \in I, \quad y \geqslant 0, \quad w \geqslant 0,
\end{aligned}
$$

where b is the initial budget, C is the capital target at the end of the planning horizon, T is the number of stages considered, I is the set of investment opportunities, S_t is the set of scenario bundles indistinguishable at time t (where each scenario bundle is defined as the set of scenarios that share the same history up to and including stage t—see Rockafellar & Wets, 1991) and $\sigma_{t+1}(s_t)$ is the set of branches that occur in scenario bundle s_t after time t. (Each element of $\sigma_{t+1}(s_t)$ is another scenario bundle s_{t+1}, which is a subset of s_t. Technically, the sets S_t form a 'filtration' of the probability space spanned by the set of scenarios. We will identify each scenario bundle with the scenario of the lowest number that is contained in it.)

The decision variables $x_{ti}(s_t)$ represent the amount of money invested in instrument i at the beginning of stage t under scenario bundle s_t; $r_{t+1,i}(s_{t+1})$ represents the return on this investment (per dollar invested) during stage t if scenario s_{t+1} occurs by the end of stage t; $y(s_T)$ is the amount of terminal wealth in excess of the target C and $w(s_T)$ is the amount by which the terminal wealth falls short of the target if scenario s_T is observed.

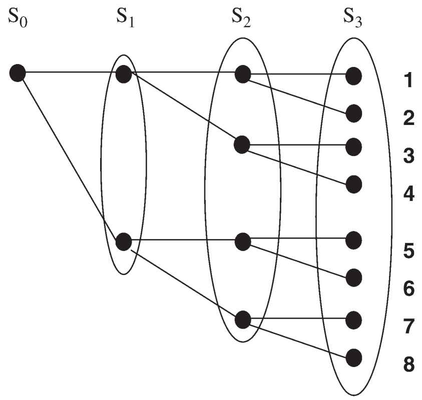

FIG. 2. A scenario tree for the investment problem in Birge and Louveaux.

The data given in Birge & Louveaux (1997) provide two rebalancing points (after 5 and 10 years), which, together with the initial decision point and the final valuation point at the horizon, define four distinct stages. There are two investments, stocks and bonds, and the stochastic data provided set up the event tree of Fig. 2, leading to the following scenario bundles: $S_2 = \{\{1, 2, 3, 4\}, \{5, 6, 7, 8\}\}$, $\sigma_3(\{1, 2, 3, 4\}) = \{\{1, 2\}, \{3, 4\}\}$, $\sigma_3(\{5, 6, 7, 8\}) = \{\{5, 6\}, \{7, 8\}\}$, $S_3 = \{\{1, 2\}, \{3, 4\}, \{5, 6\}, \{7, 8\}\}$ and $S_4 = \{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}, \{7\}, \{8\}\}$.

We next give the SMPS representation of this instance of the asset management problem, starting with the core file.

```
NAME          Asset Mgt
ROWS
 N  WEALTH
 E  BUDGET
 E  BAL1
 E  BAL2
 E  BAL3
COLUMNS
    STOCK1    BUDGET        1.00     BAL1        -1.15
    BONDS1    BUDGET        1.00     BAL1        -1.13
    STOCK2    BAL1          1.00     BAL2        -1.15
    BONDS2    BAL1          1.00     BAL2        -1.13
    STOCK3    BAL2          1.00     BAL3        -1.15
    BONDS3    BAL2          1.00     BAL3        -1.13
    SHORT     BAL3         -1.00     WEALTH       1.00
    OVER      BAL3          1.00     WEALTH      -1.00
RHS
    RHS       BUDGET       55.00     BAL3        80.00
ENDATA
```

The time file provides the markers to split this core file into four stages, named 'TODAY', 'YEAR_5', 'YEAR_10' and 'HORIZON'. Since the rows and columns were given in the core file in temporal order, only the first row and column in each stage are recorded and the stage information for the remaining rows and columns can be inferred by the system.

```
TIME           Asset Mgt
PERIODS
    STOCK1     BUDGET                   TODAY
    STOCK2     BAL1                     YEAR_5
    STOCK3     BAL2                     YEAR_10
    SHORT      BAL3                     HORIZON
ENDATA
```

The stoch file defines eight scenarios and has the following form. Each record marked ‘SC’ denotes the start of a new scenario, its (path) probability, which scenario it branches from and the stage in which the branch occurs, i.e. the first stage for which the information of the descendant scenario differs from that of the parent.

The actual data are given on the records following the ‘SC’ record. For instance, scenario ‘SCEN_2’ shares the return of scenario ‘SCEN_1’ for the first 10 years, but over the last 5 years SCEN_1 has good performance of stocks, while SCEN_2 represents a scenario where stocks fare badly in the last stage.

Only those data items that differ from the parent scenario must be present; data items not referenced are inherited from the parent scenario. (In this problem, all the records containing the values −1.25 and −1.14 are redundant; their values could have been inferred from the core file and previously recorded data. We chose to include them for somewhat easier reading.)

```
STOCH          Asset Mgt
SCEN
 SC SCEN_1     'ROOT'         0.125     TODAY
    STOCK1     BAL1          -1.25
    BONDS1     BAL1          -1.14
    STOCK2     BAL2          -1.25
    BONDS2     BAL2          -1.14
    STOCK3     BAL3          -1.25
    BONDS3     BAL3          -1.14
 SC SCEN_2     SCEN_1         0.125     HORIZON
    STOCK3     BAL3          -1.06
    BONDS3     BAL3          -1.12
 SC SCEN_3     SCEN_1         0.125     YEAR_10
    STOCK2     BAL2          -1.06
    BONDS2     BAL2          -1.12
    STOCK3     BAL3          -1.25
    BONDS3     BAL3          -1.14
 SC SCEN_4     SCEN_3         0.125     HORIZON
    STOCK3     BAL3          -1.06
    BONDS3     BAL3          -1.12
 SC SCEN_5     SCEN_1         0.125     YEAR_5
    STOCK1     BAL1          -1.06
    BONDS1     BAL1          -1.12
    STOCK2     BAL2          -1.25
    BONDS2     BAL2          -1.14
    STOCK3     BAL3          -1.25
    BONDS3     BAL3          -1.14
 SC SCEN_6     SCEN_5         0.125     HORIZON
    STOCK3     BAL3          -1.06
    BONDS3     BAL3          -1.12
```

```
 SC SCEN_7    SCEN_5          0.125     YEAR_10
    STOCK2    BAL2           -1.06
    BONDS2    BAL2           -1.12
    STOCK3    BAL3           -1.25
    BONDS3    BAL3           -1.14
 SC SCEN_8    SCEN_7          0.125     HORIZON
    STOCK3    BAL3           -1.06
    BONDS3    BAL3           -1.12
ENDATA
```

4. Nodes

If the problem dimensions may depend on past events, then the SCENARIOS format of Section 3 becomes cumbersome and potentially wasteful. SMPS allows an explicit node by node construction of the event tree. Stochastic problem dimensions are not used very often in practice. We are aware of only one example (Fleten *et al.*, 2002) where they appeared in the literature. However, this example is too large to be reproduced here. Instead, we reprise an artificial example that first appeared in Gassmann & Schweitzer (2001).

The example describes a three-stage production problem. At the start, only a single item ('widgets') is produced. If the demand in stage 1 is low, production ceases entirely (the company goes out of business); if the demand in stage 1 is medium, production of the single item continues; and if demand in stage 1 is high, a second item ('gadgets') is introduced. The mathematical formulation of this problem is as follows:

$$\begin{aligned}
\min\ & c_{01}p_{01} + \sum_{s\in S_1} \pi_{1s}\left(k_{1s}x_{1s} + g_{1s}y_{1s} - r_{1s}z_{1s}\right) + \pi_{12}c_{12}p_{12} + \pi_{14}\left(c_{14}p_{14} + l_{14}q_{14}\right)\\
&+ \sum_{t\in S_2(2)} \pi_{12}\pi_{2t}\left(g_{2t}y_{2t} + h_{2t}x_{2t} - r_{2t}z_{2t}\right)\\
&+ \sum_{t\in S_2(4)} \pi_{14}\pi_{2t}\left(g_{2t}y_{2t} + h_{2t}x_{2t} - r_{2t}z_{2t} + f_{2s}v_{2s} + n_{2s}u_{2s} - m_{2s}w_{2s}\right)
\end{aligned}$$

$$\begin{aligned}
\text{s.t. } & p_{01} \leqslant K_1,\\
& p_{01} - x_{1s} - z_{1s} = 0, \quad s \in S_1,\\
& -p_{01} + z_{1s} \leqslant 0, \quad s \in S_1,\\
& y_{1s} + z_{1s} = d_{1s}, \quad s \in S_1,\\
& x_{1s} + p_{1s} - x_{2t} - z_{2t} = 0, \quad s = 2, 4, \quad t \in S_2(s),\\
& -p_{1s} + z_{2t} \leqslant 0, \quad s = 2, 4, \quad t \in S_2(s),\\
& y_{2t} + z_{2t} = d_{2t}, \quad s = 2, 4, \quad t \in S_2(s),\\
& p_{12} \leqslant K_1,\\
& q_{14} - u_{2t} - w_{2t} = 0, \quad t \in S_2(4),\\
& ap_{14} + bq_{14} \leqslant K_2,\\
& -q_{14} + w_{2t} \leqslant 0, \quad t \in S_2(4),\\
& v_{2t} + w_{2t} = e_{2t}, \quad t \in S_2(4),\\
& p_{ij}, q_{ij}, x_{ij}, y_{ij}, z_{ij}, u_{ij}, v_{ij}, w_{ij} \geqslant 0, \quad i = 0, \ldots, 2, \quad j \in S_i,
\end{aligned}$$

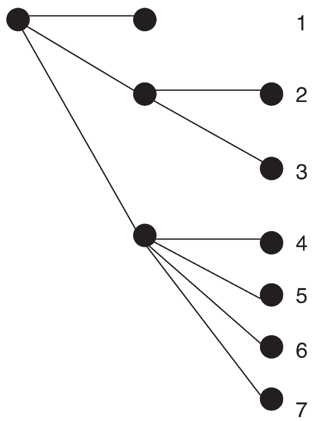

FIG. 3. An event tree with a coffin state.

where

c_{ij} is the cost of producing one widget in stage i under scenario j,

p_{ij} is the number of widgets produced in stage i under scenario j,

g_{ij} is the unit shortage cost of widgets in stage i under scenario j,

y_{ij} is the number of widgets short in stage i under scenario j,

h_{ij} is the cost of holding one widget in storage in stage i under scenario j,

k_{ij} is the cost of holding one widget in storage and disposing it if production is stopped in stage i under scenario j,

x_{ij} is the number of widgets stored from stage $i-1$ to stage i under scenario j,

r_{ij} is the revenue from selling one widget in stage i under scenario j,

z_{ij} is the number of widgets sold in stage i under scenario j,

d_{ij} is the number of widgets demanded in stage i under scenario j,

l_{ij} is the cost of producing one gadget in stage i under scenario j,

q_{ij} is the number of gadgets produced in stage i under scenario j,

f_{ij} is the unit shortage cost of gadgets in stage i under scenario j,

υ_{ij} is the number of gadgets short in stage i under scenario j,

n_{ij} is the cost of holding one gadget in storage in stage i under scenario j,

u_{ij} is the number of gadgets stored from stage $i-1$ to stage i under scenario j,

m_{ij} is the revenue from selling one gadget in stage i under scenario j,

ω_{ij} is the number of gadgets sold in stage i under scenario j,

e_{ij} is the number of gadgets demanded in stage i under scenario j,

K_1 is the production limit of widgets,

K_2 is the resource availability for the production of widgets and gadgets,

a is the amount of the resource needed in the production of one widget,

b is the amount of the resource needed in the production of one gadget,

π_{ij} is the conditional probability of reaching scenario j in stage i given the parent scenario in stage $i-1$.

The corresponding event tree with coffin state is given in Fig. 3. The scenario bundles associated with this tree are $S_1 = \{\{1\}, \{2, 3\}, \{4, 5, 6, 7\}\}$, $S_2(2) = \{2, 3\}$ and $S_2(4) = \{4, 5, 6, 7\}$.

The core file for this problem is optional. It may contain data for one complete or partial scenario—typically including the first stage—and can be referenced node by node from the stoch file. In this example, we set up the first-stage production decision in the core file and leave all the rest to the stoch file.

```
NAME          ProdDemo
ROWS
 N  COST
 L  CAP0
COLUMNS
    PROD0     CAP0              1.0    COST              2.0
RHS
    RHS       CAP0           5000.0
ENDATA
```

The time file does not have to be present and it does not have to describe the full temporal structure. It must contain enough information to separate those nodes of the core file that are referenced explicitly in the stoch file.

```
TIME          ProdDemo
PERIODS
    PROD0     COST                     STAGE0
ENDATA
```

The stoch file builds the event tree. In this case, the root node is taken from the core file; the second-stage nodes are created from scratch, as are the third-stage nodes for scenarios 2 and 4, and the remaining third-stage nodes are copied and modified appropriately.

```
STOCH         ProdDemo
NODES
 CP NODE01    'ROOT'            1.0    'CORFIL'
*
* Node11 has low demand, so the production is shut down
*-------------------------------------------------------
 MK NODE11    NODE01            0.3
ROWS
 N  COST
 E  BALAN1
 E  DEMD1
 L  AVAIL1
COLUMNS
    PROD0     BALAN1            1.0    AVAIL1           -1.0
    HOLD1     BALAN1           -1.0    COST              0.6
    SHORT1    DEMD1             1.0    COST              0.1
    SELL1     BALAN1           -1.0    COST             -4.0
    SELL1     AVAIL1            1.0    DEMD1             1.0
RHS
    RHS       DEMD1          1000.0
ENDATA
```

```
*
* Node12 has medium demand; continue operations
*-----------------------------------------------
 MK NODE12    NODE01            0.5
ROWS
 N  COST
 E  BALAN1
 E  DEMD1
 L  AVAIL1
 L  CAP1
COLUMNS
    PROD0     BALAN1            1.0    AVAIL1           -1.0
    HOLD1     BALAN1           -1.0    COST              0.3
    SHORT1    DEMD1             1.0    COST              0.1
    SELL1     BALAN1           -1.0    COST             -4.0
    SELL1     AVAIL1            1.0    DEMD1             1.0
    PROD1     CAP1              1.0    COST              2.0
RHS
    RHS       CAP1           5000.0    DEMD1          3000.0
ENDATA
*
* Third-stage operations; one product only
*------------------------------------------
 MK NODE22    NODE12            0.5
ROWS
 N  COST
 E  BALAN2
 E  DEMD2
 L  AVAIL2
COLUMNS
    HOLD1     BALAN2            1.0
    PROD1     BALAN2            1.0    AVAIL2           -1.0
    HOLD2     BALAN2           -1.0    COST              0.3
    SHORT2    DEMD2             1.0    COST              0.1
    SELL2     BALAN2           -1.0    COST             -4.0
    SELL2     AVAIL2            1.0    DEMD2             1.0
RHS
    RHS       DEMD2          4000.0
ENDATA
*
 CP NODE23    NODE12            0.5    NODE22
    RHS       DEMD2          5000.0
*
* Node14 has high demand: introduce the second product
*------------------------------------------------------
 MK NODE14    NODE01            0.2
ROWS
 N  COST
 E  BALAN1
 E  DEMD1
```

```
 L  AVAIL1
 L  CAP1
COLUMNS
    PROD0     BALAN1          1.0    AVAIL1          -1.0
    HOLD1     BALAN1         -1.0    COST             0.3
    SHORT1    DEMD1           1.0    COST             0.1
    SELL1     BALAN1         -1.0    COST            -4.0
    SELL1     AVAIL1          1.0    DEMD1            1.0
    PROD1     CAP1            1.5    COST             2.0
    PRODG1    CAP1            2.0    COST             2.5
RHS
    RHS       CAP1        12000.0    DEMD1         5000.0
ENDATA
*
* Third-stage operations; two products
*------------------------------------
 MK NODE24    NODE14          0.2
ROWS
 N  COST
 E  BALAN2
 E  DEMD2
 L  AVAIL2
 E  BALAN2G
 E  DEMD2G
 L  AVAIL2G
COLUMNS
    HOLD1     BALAN2          1.0
    PROD1     BALAN2          1.0    AVAIL2          -1.0
    PRODG1    BALAN2G         1.0    AVAIL2G         -1.0
    HOLD2     BALAN2         -1.0    COST             0.3
    SHORT2    DEMD2           1.0    COST             0.1
    SELL2     BALAN2         -1.0    COST            -4.0
    SELL2     AVAIL2          1.0    DEMD2            1.0
    HOLD2G    BALAN2G        -1.0    COST             0.3
    SHORT2G   DEMD2G          1.0    COST             0.1
    SELL2G    BALAN2G        -1.0    COST            -4.0
    SELL2G    AVAIL2G         1.0    DEMD2G           1.0
RHS
    RHS       DEMD2G       5000.0    DEMD2         4000.0
ENDATA
*
 CP NODE25    NODE14          0.3    NODE24
    RHS       DEMD2G       4000.0    DEMD2         4000.0
*
 CP NODE26    NODE14          0.4    NODE24
    RHS       DEMD2G       4000.0    DEMD2         5000.0
*
 CP NODE27    NODE14          0.1    NODE24
    RHS       DEMD2G       5000.0    DEMD2         5000.0
ENDATA
```

5. INDEP

The INDEP format is used to build an event tree implicitly from 1D marginal information. We illustrate the format using two versions of a power generation model first employed by Louveaux & Smeers (1988). A decision maker has to decide on the capacities x_j of a number of technologies for the generation of power and has to operate the resulting facility so as to satisfy uncertain demand. Mathematically, this can be formulated as follows:

$$
\begin{aligned}
\min \sum_{j\in J} c_j x_j &+ \sum_{j\in J}\sum_{s\in S} f_j p_s y_{js} \\
\text{s.t.} \sum_{j\in J} c_j x_j &\leqslant b, \\
\sum_{j\in J} x_j &\geqslant M, \\
-x_j + \sum_{s\in S} y_{js} &\leqslant 0, \quad j \in J, \\
\sum_{j\in J} y_{js} &= \omega_{sj}, \quad s \in S.
\end{aligned}
$$

We use data that have been slightly modified from the original source and were taken from Higle & Sen (1996).

Core file:

```
NAME          PGP2
ROWS
 N  FOBJ
 G  MXDEMD
 L  BUDGET
 L  CAPEQ1
 L  CAPEQ2
 L  CAPEQ3
 L  CAPEQ4
 G  DNODE1
 G  DNODE2
 G  DNODE3
COLUMNS
    INVEQ1    FOBJ          10.0        MXDEMD        1.0
    INVEQ1    BUDGET        10.0        CAPEQ1       -1.0
    INVEQ2    FOBJ           7.0        MXDEMD        1.0
    INVEQ2    BUDGET         7.0        CAPEQ2       -1.0
    INVEQ3    FOBJ          16.0        MXDEMD        1.0
    INVEQ3    BUDGET        16.0        CAPEQ3       -1.0
    INVEQ4    FOBJ           6.0        MXDEMD        1.0
    INVEQ4    BUDGET         6.0        CAPEQ4       -1.0
    EQ1ND1    FOBJ          40.0        CAPEQ1        1.0
    EQ1ND1    DNODE1         1.0
    EQ1ND2    FOBJ          24.0        CAPEQ1        1.0
```

```
    EQ1ND2    DNODE2         1.0
    EQ1ND3    FOBJ           4.0        CAPEQ1        1.0
    EQ1ND3    DNODE3         1.0
    EQ2ND1    FOBJ          45.0        CAPEQ2        1.0
    EQ2ND1    DNODE1         1.0
    EQ2ND2    FOBJ          27.0        CAPEQ2        1.0
    EQ2ND2    DNODE2         1.0
    EQ2ND3    FOBJ           4.5        CAPEQ2        1.0
    EQ2ND3    DNODE3         1.0
    EQ3ND1    FOBJ          32.0        CAPEQ3        1.0
    EQ3ND1    DNODE1         1.0
    EQ3ND2    FOBJ          19.2        CAPEQ3        1.0
    EQ3ND2    DNODE2         1.0
    EQ3ND3    FOBJ           3.2        CAPEQ3        1.0
    EQ3ND3    DNODE3         1.0
    EQ4ND1    FOBJ          55.0        CAPEQ4        1.0
    EQ4ND1    DNODE1         1.0
    EQ4ND2    FOBJ          33.0        CAPEQ4        1.0
    EQ4ND2    DNODE2         1.0
    EQ4ND3    FOBJ           5.5        CAPEQ4        1.0
    EQ4ND3    DNODE3         1.0
    PEN1      FOBJ        1000.0        CAPEQ1       -1.0
    PEN2      FOBJ        1000.0        CAPEQ2       -1.0
    PEN3      FOBJ        1000.0        CAPEQ3       -1.0
    PEN4      FOBJ        1000.0        CAPEQ4       -1.0
RHS
    RHS       MXDEMD        15.0
    RHS       BUDGET       220.0
    RHS       DNODE1         5.0
    RHS       DNODE2         4.0
    RHS       DNODE3         3.0
ENDATA
```

Time file:

```
TIME          PGP2
PERIODS
    INVEQ1    FOBJ                      TIME1
    EQ1ND1    CAPEQ1                    TIME2
ENDATA
```

The first stoch file sets up independent discrete distributions for the three demands. The demand in node ‘DNODE1’ has nine realizations and the other two have eight realizations each. Since the three random elements are independent of each other, this defines 576 scenarios altogether.

```
STOCH         PGP2
INDEP         DISCRETE
    RHS       DNODE1      0.5                         0.00005
    RHS       DNODE1      1.0                         0.00125
    RHS       DNODE1      2.5                         0.02150
```

```
    RHS       DNODE1      3.5                         0.28570
    RHS       DNODE1      5.0                         0.38300
    RHS       DNODE1      6.5                         0.28570
    RHS       DNODE1      7.5                         0.02150
    RHS       DNODE1      9.0                         0.00125
    RHS       DNODE1      9.5                         0.00005
*
    RHS       DNODE2      0.0                         0.00130
    RHS       DNODE2      1.5                         0.02150
    RHS       DNODE2      2.5                         0.28570
    RHS       DNODE2      4.0                         0.38300
    RHS       DNODE2      5.5                         0.28570
    RHS       DNODE2      6.5                         0.02150
    RHS       DNODE2      8.0                         0.00125
    RHS       DNODE2      8.5                         0.00005
*
    RHS       DNODE3      0.0                         0.00130
    RHS       DNODE3      0.5                         0.02150
    RHS       DNODE3      1.5                         0.28570
    RHS       DNODE3      3.0                         0.38300
    RHS       DNODE3      4.5                         0.28570
    RHS       DNODE3      5.5                         0.02150
    RHS       DNODE3      7.0                         0.00125
    RHS       DNODE3      7.5                         0.00005
ENDATA
```

A second stoch file is provided to show continuous distributions, as originally envisioned in Louveaux & Smeers (1988). Here, the demand is normally distributed with equal variance. The expected demands in the three locations are 5, 4 and 3, respectively.

```
STOCH         PGP2
INDEP         NORMAL
    RHS       DNODE1      5.0                         1.5625
*
    RHS       DNODE2      4.0                         1.5625
*
    RHS       DNODE3      3.0                         1.5625
ENDATA
```

6. Blocks

This third version of the power generation problem of Section 5 indicates the use of random vectors that are assumed to be independent from period to period although they may exhibit intra-period dependence. This stoch file sets up a discrete random vector with six realizations, i.e. six scenarios.

```
STOCH         PGP2
BLOCKS        DISCRETE
 BL BLOCK_1   PERIOD_2    0.005
    RHS       DNODE1      1.0
    RHS       DNODE2      1.5
```

```
    RHS        DNODE3       0.5
 BL BLOCK_1    PERIOD_2     0.045
    RHS        DNODE1       2.5
    RHS        DNODE2       2.5
    RHS        DNODE3       1.5
 BL BLOCK_1    PERIOD_2     0.45
    RHS        DNODE1       4.0
    RHS        DNODE2       3.0
    RHS        DNODE3       2.0
 BL BLOCK_1    PERIOD_2     0.45
    RHS        DNODE1       6.0
    RHS        DNODE2       5.0
    RHS        DNODE3       4.0
 BL BLOCK_1    PERIOD_2     0.045
    RHS        DNODE1       8.0
    RHS        DNODE2       7.5
    RHS        DNODE3       6.5
 BL BLOCK_1    PERIOD_2     0.005
    RHS        DNODE1       9.5
    RHS        DNODE2       8.5
    RHS        DNODE3       7.5
ENDATA
```

7. Network

The problem in this section is taken from Mulvey & Vladimirou (1989). It represents a simplified investment problem, which is given as a generalized network, with stochastic gains and losses on the arcs, representing random investment returns. The problem has three time periods, but it is set up as a two-stage problem.

The mathematical formulation of this problem is as follows:

$$
\begin{aligned}
\min\quad & \sum_{s\in S} p_s w_s \\
\text{s.t.}\quad & x_{f0} + y_{f0} = B_f, \qquad f \in F, \\
& x_{b0} + y_{b0} = B_b, \\
& -\sum_{f\in F}(1-\xi_{f0})x_{f0} + \sum_{f\in F} u_{f0} + x_{b0} - u_{b0} + \sum_{t=1}^{T} v_{0t} = C_0, \\
& y_{f0} + (1-\eta_{f0})u_{f0} - z_{f0} = 0, \qquad f \in F, \\
& y_{b0} + u_{b0} - (1-R_{b0})z_{b0} = 0, \\
& (1+R_{fs0})z_{f0} - x_{fs1} - y_{fs1} = 0, \qquad f \in F,\ \ s \in S_1, \\
& z_{b0} - x_{bs1} - y_{bs1} = 0, \qquad s \in S_1, \\
& -\sum_{f\in F}(1-\xi_{ft})x_{fst} + \sum_{f\in F} u_{fst} + x_{bst} - u_{bst}, \\
& -\sum_{p<t}(1+R_{pst})v_{pst} + \sum_{p>t} v_{tsp} = C_{st}, \qquad t = 1,\dots,T,\ \ s \in S_t,
\end{aligned}
$$

$$y_{fst} + (1 - \eta_{ft})u_{fst} - z_{fst} = 0, \quad f \in F, \ t = 1, \ldots, T, \ s \in S_t,$$
$$y_{bst} + u_{bst} - (1 - R_{bst})z_{bst} = 0, \quad t = 1, \ldots, T, \ s \in S_t,$$
$$(1 + R_{fst})z_{f,a(s),t-1} - x_{fst} - y_{fst} = 0, \quad f \in F, \ t = 1, \ldots, T-1, \ s \in S_t,$$
$$z_{b,a(s),t-1} - x_{bst} - y_{bst} = 0, \quad t = 1, \ldots, T-1, \ s \in S_t,$$
$$\sum_{f \in F}(1 + R_{fsT})z_{fsT} + \sum_{p \leqslant T}(1 + R_{p,s,T+1})v_{p,s,T+1} - z_{bsT} - w_s = 0, \quad s \in S,$$

where

F is the set of risky assets,
T is the number of time stages,
S is the set of scenarios and S_t is the set of scenarios in stage t, where $S_T = S$,
$a(s)$ is the ancestor scenario of scenario s,
B_f are the initial holdings in asset f in F,
B_b is the initial liability,
ξ_{ft} is the transaction cost for selling one unit of asset f in period t,
η_{ft} is the transaction cost for buying one unit of asset f in period t,
R_{fst} is the rate of return for risky asset f in period t under scenario s,
R_{bst} is the interest rate for borrowing in period t under scenario s,
R_{pst} is the rate of return for the riskless asset purchased at time p and maturing at time t,
C_{st} is the cash inflow (if >0) or outflow (if <0) in period t under scenario s,
x_{fst} is the amount of risky asset f sold in period t under scenario s,
y_{fst} is the amount of risky asset f carried forward (held) in period t under scenario s,
u_{fst} is the amount of risky asset f purchased in period t under scenario s,
x_{bst} is the amount of liability paid back at time t under scenario s,
y_{bst} is the amount of liability carried forward at time t under scenario s,
u_{bst} is the amount of new borrowing in period t under scenario s,
v_{pst} is the amount of riskless asset purchased at time p and maturing at time t under scenario s,
z_{fst} are the holdings in asset f after the portfolio revision of stage t under scenario s,
z_{bst} is the amount of debt after the portfolio revision of stage t under scenario s,
w_s is the net wealth at the end of the horizon in scenario s,
p_s is the probability that scenario s occurs.

The network can also be represented pictorially as in Fig. 4.

The core file for this problem uses ‘NODES’ and ‘ARCS’ headers instead of ‘ROWS’ and ‘COLUMNS’. The structure of the data records in the ARCS section is slightly different from the MPS form.

```
NAME          Invest
NODES
 N  OBJECT
 E  ND_11
 E  ND_12
 E  ND_13
 E  ND_14
```

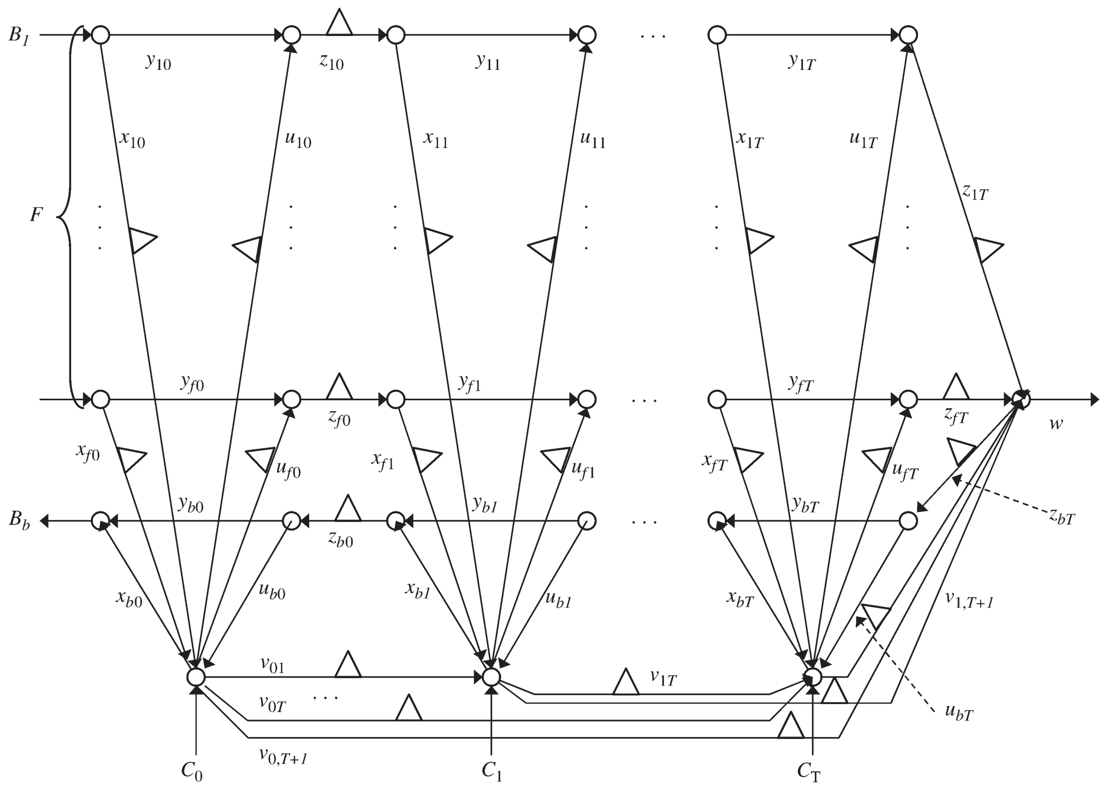

FIG. 4. Generalized network representing the investment problem of Mulvey and Vladimirou.

```
 E  ND_15
 E  ND_21
 E  ND_22
 E  ND_23
 E  ND_24
 E  ND_25
 E  ND_31
 E  ND_32
 E  ND_33
 E  ND_34
 E  ND_35
 E  ND_41
ARCS
       ND_11 ND_15    0.0     0.3        0.0       0.99700
       ND_11 ND_21    0.0     0.6        0.0       1.02871
       ND_12 ND_22    0.0     0.4        0.0       1.04982
       ND_13 ND_15    0.0     0.2        0.0       1.00000
       ND_13 ND_23    0.0     0.8        0.0       1.01973
       ND_14 ND_24    0.0     0.3        0.0       1.02534
```

```
      ND_15 ND_11    0.0     0.6        0.0      0.99700
      ND_15 ND_12    0.0     0.4        0.0      0.99000
      ND_15 ND_13    0.0     0.8        0.0      1.00000
      ND_15 ND_14    0.0     0.3        0.0      0.99500
      ND_15 ND_25    0.0     0.0        0.0      1.01650
      ND_21 ND_25    0.0     1.E+30     0.0      0.99700
      ND_21 ND_31    0.0     1.E+30     0.0      1.02871
      ND_22 ND_25    0.0     1.E+30     0.0      0.99000
      ND_22 ND_32    0.0     1.E+30     0.0      1.04982
      ND_23 ND_25    0.0     1.E+30     0.0      1.00000
      ND_23 ND_33    0.0     1.E+30     0.0      1.01973
      ND_24 ND_25    0.0     1.E+30     0.0      0.99500
      ND_24 ND_34    0.0     1.E+30     0.0      1.02534
      ND_25 ND_21    0.0     1.E+30     0.0      0.99700
      ND_25 ND_22    0.0     1.E+30     0.0      0.99000
      ND_25 ND_23    0.0     1.E+30     0.0      1.00000
      ND_25 ND_24    0.0     1.E+30     0.0      0.99500
      ND_25 ND_35    0.0     1.E+30     0.0      1.01650
      ND_31 ND_35    0.0     1.E+30     0.0      0.99700
      ND_31 ND_41    0.0     1.E+30     0.0      1.02871
      ND_32 ND_35    0.0     1.E+30     0.0      0.99000
      ND_32 ND_41    0.0     1.E+30     0.0      1.04982
      ND_33 ND_35    0.0     1.E+30     0.0      1.00000
      ND_33 ND_41    0.0     1.E+30     0.0      1.01973
      ND_34 ND_35    0.0     1.E+30     0.0      0.99500
      ND_34 ND_41    0.0     1.E+30     0.0      1.02534
      ND_35 ND_31    0.0     1.E+30     0.0      0.99700
      ND_35 ND_32    0.0     1.E+30     0.0      0.99000
      ND_35 ND_33    0.0     1.E+30     0.0      1.00000
      ND_35 ND_34    0.0     1.E+30     0.0      0.99500
      ND_35 ND_41    0.0     1.E+30     0.0      1.01650
      ND_41 OBJECT  -1.0
SUPPLY
    RHS       ND_11    0.3000000000
    RHS       ND_13    0.2000000000
ENDATA
```

The columns in this network problem are named implicitly, so the time file must include a special marker in order to be parsed properly:

```
TIME          Invest
PERIODS
    'NETWRK'  OBJECT                    STAGE_1
    'NETWRK'  ND_11                     STAGE_2
ENDATA
```

The stoch file defines a 12D random vector with three possible realizations:

```
STOCH         Invest
BLOCKS        DISCRETE
```

```
 BL  BLOCK_1  STAGE_2  0.3
 M   ND_11    ND_21      0.92596
 M   ND_12    ND_22      0.8557
 M   ND_13    ND_23      1.00658
 M   ND_14    ND_24      0.90479
 M   ND_21    ND_31      0.94223
 M   ND_22    ND_32      0.89201
 M   ND_23    ND_33      1.00658
 M   ND_24    ND_34      0.9319
 M   ND_31    ND_41      1.04075
 M   ND_32    ND_41      1.09643
 M   ND_33    ND_41      1.00658
 M   ND_34    ND_41      1.04699
 BL  BLOCK_1  STAGE_2  0.4
 M   ND_11    ND_21      1.09407
 M   ND_12    ND_22      1.15979
 M   ND_13    ND_23      1.00658
 M   ND_14    ND_24      1.10405
 M   ND_21    ND_31      1.04984
 M   ND_22    ND_32      1.13555
 M   ND_23    ND_33      1.00658
 M   ND_24    ND_34      1.06147
 M   ND_31    ND_41      0.94542
 M   ND_32    ND_41      1.11137
 M   ND_33    ND_41      1.00658
 M   ND_34    ND_41      0.95535
 BL  BLOCK_1  STAGE_2  0.3
 M   ND_11    ND_21      1.02486
 M   ND_12    ND_22      0.98336
 M   ND_13    ND_23      1.00658
 M   ND_14    ND_24      1.04039
 M   ND_21    ND_31      1.05786
 M   ND_22    ND_32      1.05695
 M   ND_23    ND_33      1.00658
 M   ND_24    ND_34      1.05855
 M   ND_31    ND_41      1.13595
 M   ND_32    ND_41      1.11459
 M   ND_33    ND_41      1.00658
 M   ND_34    ND_41      1.13907
ENDATA
```

8. Mixing LP and network formats

The problem in this section was inspired by the work done by Wallace (1986). It concerns a hypothetical fish-processing company with two processing plants and a fleet that can fish in five different locations. The aim is to expand the capacity of both the fleet and the production facilities, subject to a joint budget constraint, to send the fleet to the locations, to land the ensuing catch and finally to process the catch into three products. The objective is to minimize net cost, which is subject to uncertainty on both the supply side (availability of fish) and the demand side (price to customers).

We will give the problem in three separate stages. The first stage concerns the capacity expansion and original allocation of the fishing fleet. The mathematical formulation of this problem is

$$
\begin{aligned}
\min \quad & \sum_{f=1}^{F}\sum_{m=0}^{M} e_{fm}x_{fm} + \sum_{f=1}^{F}\sum_{g=1}^{G} c_{fg}u_{fg} + \mathrm{E}_{\xi_1} R_1(x,u,\xi_1) \\
\text{s.t.} \quad & \sum_{f=1}^{F}\sum_{m=0}^{m} e_{fm}x_{fm} \leqslant B, \\
& -x_{f0} + \sum_{g=1}^{G} u_{fg} \leqslant i_{f0}, \quad f=1,\ldots,F, \\
& x_{fm}, u_{fg} \geqslant 0, \quad f=1,\ldots,F, \ \ m=0,\ldots,M, \ \ g=1,\ldots,G,
\end{aligned}
$$

where

F is the number of processing plants,
M is the number of resources considered (the resource numbered 0 represents the fishing fleet),
G is the number of fishing grounds,
B is the available budget,
i_{fm} is the existing capacity of resource m in plant f,
e_{fm} is the cost of adding one unit of capacity of resource m in plant f,
x_{fm} is the capacity of resource m added in plant f,
c_{fg} is the cost of sending one unit of fishing capacity from plant f to fishing ground g,
u_{fg} is the amount of fishing capacity sent from plant f to fishing ground g.

The quantity $R_1(x,u,\xi_1)$ is the recourse cost. Once the fishing fleet is in location, the amount of fish is revealed and the fishing fleet can be relocated from one ground to another. Mathematically, this amounts to solving the following problem:

$$
\begin{aligned}
R_1(x,u,\xi_1) = \min & \sum_{n=1}^{N_1}\sum_{g=1}^{G}\sum_{j=1}^{G} \pi_n t_{gj} y_{gjn} + \sum_{n=1}^{N_1}\sum_{g=1}^{G}\sum_{f=1}^{F} \pi_n r_{gf} v_{gfn} \\
& + \mathrm{E}_{\xi_2} R_2(x,u,\xi_1,y,v,w,\xi_2) \\
\text{s.t.} & \sum_{j=1}^{G} y_{gjn} + \sum_{f=1}^{F} v_{gfn} = \sum_{f=1}^{F} u_{fg}, \quad g=1,\ldots,G, \ \ n=1,\ldots,N_1, \\
& v_{gfn} - w_{gfn} \geqslant 0, \quad g=1,\ldots,G, \quad f=1,\ldots,F, \ n=1,\ldots,N_1, \\
& \sum_{f=1}^{F} w_{gfn} \leqslant s_{gn}, \quad g=1,\ldots,G, \ \ n=1,\ldots,N_1, \\
& y_{gjn} \text{ unrestricted}, \quad v_{gfn}, w_{gfn} \geqslant 0,
\end{aligned}
$$

where

t_{gj} is the cost of relocating one unit of fishing capacity from fishing ground g to fishing ground j,
y_{gjn} is the amount of fishing capacity relocated from fishing ground g to fishing ground j in node n,
r_{gf} is the cost of returning one unit of fishing capacity from fishing ground g to plant f,
v_{gfn} is the amount of fishing capacity returned from fishing ground g to plant f in node n,
w_{gfn} is the amount of fish harvested from fishing ground g and returned to plant f in node n,
s_{gn} is the amount of fish available at fishing ground g in node n,
π_n is the probability that node n occurs.

The amount of resource transported from fishing ground g to ground j is unrestricted, in order to allow for a more compact representation. If $v_{gjn} < 0$, then shipping movement occurs from j to i.

The third stage of the problem concerns the processing of the fish. This can be expressed mathematically as

$$R_2(x, u, \xi_1, y, v, w, \xi_2) = \min \left[-\sum_{n=N_1+1}^{N} \sum_{f=1}^{F} \sum_{q=1}^{Q} \pi_n h_{fqn} z_{fqn} \right]$$

$$\text{s.t.} \sum_{q=1}^{Q} z_{fqn} \leqslant \sum_{g=1}^{G} w_{gfa(n)}, \quad f = 1, \ldots, F, \quad n = N_1 + 1, \ldots, N,$$

$$\sum_{q=1}^{Q} a_{fqm} z_{fqn} \leqslant i_{fm} + x_{fm}, \quad f = 1, \ldots, F, \quad m = 1, \ldots, M,$$
$$n = N_1 + 1, \ldots, N$$

$$z_{fqn} \geqslant 0, \quad f = 1, \ldots, F, \quad q = 1, \ldots, Q, \quad n = N_1 + 1, \ldots, N,$$

where

Q is the number of different fish products that can be produced,
h_{fqn} is the net profit obtained from one unit of product q produced at plant f in node n,
a_{fqm} is the amount of resource m needed to produce one unit of product q at plant f,
z_{fqn} is the amount of product q produced at plant f in node n,
p_n is the (path) probability of reaching node n,
$a(n)$ is the predecessor node of node n in the event tree,
$\mathcal{N} = \{0, \ldots, N\}$ is the set of nodes; 0 is the root node, $\{1, \ldots, N_1\}$ are the second-stage nodes and the remainder are third-stage nodes.

The SMPS files below combine all three problems into a single formulation. We will use an LP formulation for the capacity expansion problem in stage 1 and the processing problem in stage 3, and a network formulation for the fleet allocation and relocation problem in stage 2.

```
NAME          MIXED_LP
* Start with ordinary LP section
ROWS
 N  TCOST
 L  BUDGET
 L  FLEETA
 L  FLEETB
```

```
* network section
NODES
 E  LOC1
 E  LOC2
 E  LOC3
 E  LOC4
 E  LOC5
* back to LP
ROWS
 L  AVAIL1
 L  AVAIL2
     ...
COLUMNS
    EXPA1     TCOST             5.      BUDGET           5.0
    EXPA1     PCAPA1           -1.
    EXPA2     TCOST             5.      BUDGET           5.0
    EXPA2     PCAPA2           -1.
     ...
* network section with implicitly named decisions
* The keyword UNDIR indicates an undirected network with
* no capacity restrictions on the arcs; flow can be reversed
ARCS          UNDIR
      LOC1  LOC2       0.10
      LOC1  LOC3       0.20
      ...
* back to LP
COLUMNS
    HARV1A    CAPA1             1.      AVAIL1           1.0
    HARV1A    CATCHA           -1.
     ...
RHS
    RHS       BUDGET        10000.
    RHS       FLEETA          500.      FLEETB         300.
    RHS       AVAIL1          700.      AVAIL2         500.
    RHS       AVAIL3         1000.      AVAIL4        1200.
    RHS       AVAIL5          800.
    RHS       PCAPA1         1000.      PCAPA2        1000.
    RHS       PCAPA3            0.
    RHS       PCAPB1         1500.      PCAPB2        1000.
    RHS       PCAPB3         1000.
ENDATA
```

The problem has three distinct stages as defined in the time file:

```
TIME          MIXED_LP
PERIODS
    EXPA1     TCOST                     STAGE1
    'NETWRK'  LOC1                      STAGE2
    PRODA1    CATCHA                    STAGE3
ENDATA
```

The stoch file sets up four scenarios. There are two possible levels of fish stocks, and for each level there are two possible sets of prices for the fish products.

```
STOCH          MIXED_LP
SCENARIOS
 SC SCEN_1     'ROOT'           0.3     STAGE1
 SC SCEN_2     SCEN_1           0.4     STAGE2
    RHS        AVAIL1         900.
    RHS        AVAIL2         700.
    RHS        AVAIL3        1300.
    RHS        AVAIL4        1700.
    RHS        AVAIL5        1000.
    PRODA1     TCOST           -2.
    PRODA2     TCOST           -2.5
    PRODA3     TCOST           -0.1
    PRODB1     TCOST           -3.
    PRODB2     TCOST           -3.5
    PRODB3     TCOST           -0.8
    PRODC1     TCOST           -2.5
    PRODC2     TCOST           -3.5
    PRODC3     TCOST           -0.3
 SC SCEN_3     SCEN_1           0.2     STAGE3
    PRODA1     TCOST           -4.
    PRODA2     TCOST           -5.5
    PRODA3     TCOST           -1.5
    PRODB1     TCOST           -5.
    PRODB2     TCOST           -6.
    PRODB3     TCOST           -1.8
    PRODC1     TCOST           -4.5
    PRODC2     TCOST           -6.5
    PRODC3     TCOST           -2.2
 SC SCEN_4     SCEN_2           0.1     STAGE3
    PRODA1     TCOST           -3.
    PRODA2     TCOST           -4.5
    PRODA3     TCOST           -2.
    PRODB1     TCOST           -4.
    PRODB2     TCOST           -4.5
    PRODB3     TCOST           -1.1
    PRODC1     TCOST           -4.0
    PRODC2     TCOST           -5.5
    PRODC3     TCOST           -1.2
ENDATA
```

9. Simple recourse

Simple recourse problems feature a very special form of the recourse matrix. Deviations from a target value are penalized by a linear penalty. We illustrate the use of this feature with one of the first stochastic linear programs ever formulated, an airline fleet allocation problem due to Dantzig (1963) and Ferguson & Dantzig (1956). In this problem, a fleet of airplanes must be assigned to different routes so as to minimize the operating costs. The demands along the routes are stochastic, and penalties are incurred for lost sales due to insufficient capacity:

$$\begin{aligned}
\min \sum_{i\in I}\sum_{r\in R(i)} c_{ir}x_{ir} + \sum_{s\in S} p_s \left[\sum_{r\in R} q_{rs}y_{rs}\right] \\
\text{s.t.} \sum_{r\in R(i)} x_{ir} \leqslant b_i, \quad i \in I, \\
\sum_{i\in I, r\in R(i)} t_{ir}x_{ir} + y_{rs} - z_{rs} = h_{rs}, \quad r \in R,\ s \in S, \\
x_{ir} \geqslant 0,\ i \in I,\ r \in R(i), \quad y_{rs}, w_{rs} \geqslant 0,\ r \in R,\ s \in S,
\end{aligned}$$

where

I is the set of aircraft to be used,
R is the set of routes to be serviced,
$R(i)$ is the set of routes within R that can be serviced by aircraft of type i,
b_i is the number of aircraft available of type i,
c_{ir} is the cost of operating an aircraft of type i along route r,
t_{ir} is the passenger capacity of aircraft i on route r,
h_{rs} is the passenger demand on route r under scenario s,
q_{rs} is the revenue lost per passenger turned away on route r under scenario s,
x_{ir} is the number of aircraft of type i assigned to route r,
y_{rs} is the number of passengers turned away on route r under scenario s,
z_{rs} is the number of empty seats on route r under scenario s.

```
NAME          AIRLINE
ROWS
 N   COST
 L   AVAIL_A
 L   AVAIL_B
 L   AVAIL_C
 L   AVAIL_D
 G   DEMAND1
 G   DEMAND2
 G   DEMAND3
 G   DEMAND4
 G   DEMAND5
COLUMNS
    ALLOC1A   AVAIL_A        1.0        DEMAND1       16.0
    ALLOC1A   COST          18.0
    ALLOC1D   AVAIL_D        1.0        DEMAND1        9.0
    ALLOC1D   COST          17.0
*
    ALLOC2A   AVAIL_A        1.0        DEMAND2       15.0
     ...
RHS
    RHS1      AVAIL_A       10.0        AVAIL_B       19.0
```

```
    RHS1      AVAIL_C       25.0        AVAIL_D       15.0
    RHS1      DEMAND1      250.0        DEMAND2      120.0
    RHS1      DEMAND3      180.0        DEMAND4       90.0
    RHS1      DEMAND5      600.0
ENDATA
```

Since the second-stage recourse variables have not been set up in the core file, the time file must take on a special form. The marker 'PENLTY' alerts the system to what is happening.

```
TIME          AIRLINE
PERIODS
    ALLOC1A   COST                      STAGE1
    'PENLTY'  DEMAND1                   STAGE2
ENDATA
```

The stoch file has two sections. The first section gives the penalty parameters for violating the demand constraints, while the second section sets up discrete distributions (independently of each other) for the demands along the different routes.

```
STOCH         AIRLINE
SIMPLE
    RHS1      DEMAND1       13.0
    RHS1      DEMAND2       13.0
    RHS1      DEMAND3        7.0
    RHS1      DEMAND4        7.0
    RHS1      DEMAND5        1.0
INDEP         DISCRETE
    RHS1      DEMAND1      200.0                         0.20
    RHS1      DEMAND1      220.0                         0.05
    RHS1      DEMAND1      250.0                         0.35
    RHS1      DEMAND1      270.0                         0.2
    RHS1      DEMAND1      300.0                         0.2
    ...
ENDATA
```

SMPS supports other forms of penalties as well, namely, purely quadratic penalties and a form of piecewise linear-quadratic penalties popularized by King (1988a). The format is very similar to the linear penalties; hence, an example is omitted for reasons of space.

10. Chance-constrained problem

This problem was taken from King (1988b) who attributes it to Prékopa & Szántai (1978). It represents a water management problem, whereby a number of reservoirs must be designed in order to control flooding due to random stream inflows.

As shown in Fig. 5, five dams are to be built to deal with random inflows in five locations. Flow in this system is from north to south. The objective is to protect against floods in location 10 with probability 0.9. This somewhat complicated mathematical condition can be reformulated into an equivalent system of inequalities, which turns the problem into a chance constraint problem with a single joint

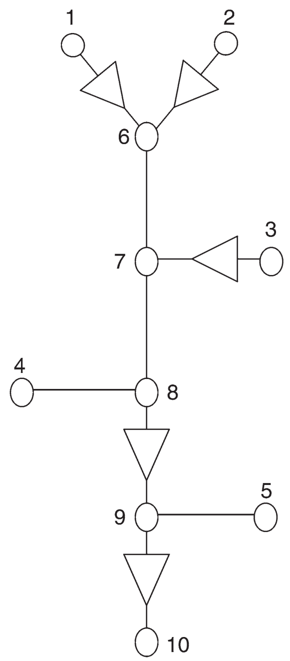

FIG. 5. A river system with dams for flood control.

chance constraint. Its mathematical formulation is given below:

$$\min \sum_{j \in J} c_j x_j$$

$$\text{s.t. } 0 \leqslant x_j \leqslant u_j, \quad j \in J,$$

$$\Pr\left[\sum_{j \in J} t_{kj} x_j \geqslant \sum_{i \in I} s_{ki} \xi_i, k \in K\right] \geqslant p,$$

where

I is a set of inflows into the system,
J is a set of reservoirs,
x_j is the capacity of reservoir j,
u_j is an upper bound on the capacity of reservoir j,
c_j is the cost per unit capacity of reservoir j,
ξ_i is the random inflow from source i,
K is the number of simple constraints used to represent the no-flood condition,
$T = [t_{kj}]$ and $S = [s_{ki}]$ are incidence matrices for the no-flood condition,
p is the desired level of confidence that the river system will not be flooded.

The core file for this problem has the following form.

```
NAME          WATERMGT
ROWS
 N  COST
 G  FLOW1
 G  FLOW2
 G  FLOW3
 G  FLOW4
 G  FLOW5
 G  FLOW6
 G  FLOW7
 G  FLOW8
 G  FLOW9
COLUMNS
    CAP1      COST         0.4
    CAP1      FLOW3        1.0          FLOW6        1.0
    CAP1      FLOW7        1.0          FLOW9        1.0
*
    CAP2      COST         0.5
    CAP2      FLOW4        1.0          FLOW6        1.0
    CAP2      FLOW8        1.0          FLOW9        1.0
*
    CAP3      COST         0.6
    CAP3      FLOW5        1.0          FLOW7        1.0
    CAP3      FLOW8        1.0          FLOW9        1.0
*
    CAP4      COST         1.2
    CAP4      FLOW2        1.0          FLOW3        1.0
    CAP4      FLOW4        1.0          FLOW5        1.0
    CAP4      FLOW6        1.0          FLOW7        1.0
    CAP4      FLOW8        1.0          FLOW9        1.0
*
    CAP5      COST         1.8
    CAP5      FLOW1        1.0          FLOW2        1.0
    CAP5      FLOW3        1.0          FLOW4        1.0
    CAP5      FLOW5        1.0          FLOW6        1.0
    CAP5      FLOW7        1.0          FLOW8        1.0
    CAP5      FLOW9        1.0
RHS
    RHS       FLOW1        1.0          FLOW2        1.0
    RHS       FLOW3        1.0          FLOW4        1.0
    RHS       FLOW5        1.0          FLOW6        1.0
    RHS       FLOW7        1.0          FLOW8        1.0
    RHS       FLOW9        1.0
BOUNDS
 UP BOUND     CAP1         1.0
 UP BOUND     CAP2         1.0
 UP BOUND     CAP3         1.0
 UP BOUND     CAP4         2.0
 UP BOUND     CAP5         3.0
ENDATA
```

This problem has a single period only, but the time file needs to be present in order to process the distribution information in the stoch file.

```
TIME            WATERMGT
PERIODS
    COST        CAP1                         PERIOD1
ENDATA
```

The stoch file has three sections. The DISTRIB section sets up the multivariate normal random variable ξ and then links it to the random right-hand sides of the problem using the linear transformation $r = D\xi$. The matrix D is defined in the BLOCKS section in column order. Finally, a multidimensional (joint) probabilistic constraint is set up in the CHANCE section.

```
STOCH           WATERMGT
DISTRIB         MVNORMAL
 BL  BL
*                           mean                    variance
     XI_1                    0.8                      0.04
     XI_2                    1.5                      0.09
     XI_3                    1.2                      0.36
     XI_4                    0.5                      0.16
     XI_5                    0.7                      0.09
*
*    Correlation matrix
 CR
     XI_1       XI_2         0.0
     XI_1       XI_3         0.6
     XI_1       XI_4         0.4
     XI_1       XI_5         0.0
     XI_2       XI_3         0.5
     XI_2       XI_4         0.3
     XI_2       XI_5         0.3
     XI_3       XI_4         0.7
     XI_3       XI_5         0.6
     XI_4       XI_5         0.4
BLOCKS          LINTR
    RHS         FLOW1        0.0
    RHS         FLOW2        0.0
    RHS         FLOW3        0.0
    RHS         FLOW4        0.0
    RHS         FLOW5        0.0
    RHS         FLOW6        0.0
    RHS         FLOW7        0.0
    RHS         FLOW8        0.0
    RHS         FLOW9        0.0
 RV XI_1
    RHS         FLOW3        1.0
    RHS         FLOW6        1.0
    RHS         FLOW7        1.0
    RHS         FLOW9        1.0
```

```
 RV XI_2
    RHS        FLOW4         1.0
    RHS        FLOW6         1.0
    RHS        FLOW8         1.0
    RHS        FLOW9         1.0
 RV XI_3
    RHS        FLOW5         1.0
    RHS        FLOW7         1.0
    RHS        FLOW8         1.0
    RHS        FLOW9         1.0
 RV XI_4
    RHS        FLOW2         1.0
    RHS        FLOW3         1.0
    RHS        FLOW4         1.0
    RHS        FLOW5         1.0
    RHS        FLOW6         1.0
    RHS        FLOW7         1.0
    RHS        FLOW8         1.0
    RHS        FLOW9         1.0
 RV XI_5
    RHS        FLOW1         1.0
    RHS        FLOW2         1.0
    RHS        FLOW3         1.0
    RHS        FLOW4         1.0
    RHS        FLOW5         1.0
    RHS        FLOW6         1.0
    RHS        FLOW7         1.0
    RHS        FLOW8         1.0
    RHS        FLOW9         1.0
CHANCE
 JG CHANCE1    CC1           0.9
    FLOW1
    FLOW2
    FLOW3
    FLOW4
    FLOW5
    FLOW6
    FLOW7
    FLOW8
    FLOW9
ENDATA
```

Integrated chance constraints introduced by Haneveld (1986) and used in the finance community as a measure of risk (see, e.g. Uryasev & Rockafellar, 2001) can be handled in SMPS in a similar manner. Once again we omit the example for space reasons.

11. Concluding remarks

While the examples in this paper have by necessity been abbreviated and kept artificially small, the variety of models and approaches nonetheless served to illustrate the flexibility of the SMPS format.

By making available the algebraic formulation, MPL model files (Kristjánsson, 2002) and the full collection of SMPS files for easy download over the internet, we hope to stimulate some interest in the SMPS format and thereby encourage its greater use.

Acknowledgements

H.I.G. was supported in part by a grant from the National Science and Engineering Research Council of Canada. We thank two anonymous referees for their thorough review of an earlier draft. Their comments have served to greatly improve the presentation of this paper.

References

ARGONNE NATIONAL LABORATORY (1996) NEOS Guide to Optimization. Available at http://www-fp.mcs.anl.gov/otc/Guide/OptWeb/continuous/constrained/linearprogs/mps.html. Accessed 18 February 2007.

BIRGE, J. R., DEMPSTER, M. A. H., GASSMANN, H. I., GUNN, E. A., KING, A. J. & WALLACE, S. W. (1987) A standard input format for multiperiod stochastic linear programs. *COAL Newsl.*, **17**, 1–19.

BIRGE, J. R. & LOUVEAUX, F. (1997) *Introduction to Stochastic Programming*. Springer Series in Operations Research. New York: Springer.

DANTZIG, G. B. (1963) *Linear Programming*. Princeton, NJ: Princeton University Press.

DUPAČOVÁ, J., CONSIGLI, G. & WALLACE, S. W. (2000) Scenarios for multistage stochastic programs. *Ann. Oper. Res.*, **100**, 25–53.

DUPAČOVÁ, J., GRÖWE-KUSKA, N. & RÖMISCH, W. (2003) Scenario reduction in stochastic programming—An approach using probability metrics. *Math. Program.*, **85**, 493–511.

EDWARDS, J. (1988) A proposed standard input format for computer codes which solve stochastic programs with recourse. *Numerical Techniques for Stochastic Optimization* (Yu. Ermoliev & R. J.-B. Wets eds). Berlin, Germany: Springer.

FELT, A., SARICH, J. & ARIYAWANSA, K. A. (2001) *SLPlib: Input Routines and Data Structures for Stochastic Linear Programming*. Available at http://www.uwsp.edu/math/afelt/slpinput.html. Accessed 16 November 2004.

FERGUSON, A. R. & DANTZIG, G. B. (1956) The allocation of aircraft to routes—an example of linear programming under uncertain demand. *Manage. Sci.*, **3**, 45–73.

FLETEN, S. E., WALLACE, S. W. & ZIEMBA, W. T. (2002) Hedging electricity portfolios using stochastic programming. *Decision Making under Uncertainty: Energy and Power*, vol. 128 (C. Greengard & A. Ruszczyński eds). IMA Volumes on Mathematics and Its Applications pp. 71–94.

GASSMANN, H. I. (2005) *The SMPS Format for Stochastic Linear Programs*. Available at http://myweb.dal.ca/gassmann/smps2.htm. Accessed 16 November 2004.

GASSMANN, H. I. & IRELAND, A. M. (1996) On the formulation of stochastic linear programs using algebraic modelling languages. *Ann. Oper. Res.*, **64**, 83–112.

GASSMANN, H. I. & SCHWEITZER, E. (2001) A comprehensive input format for stochastic linear programs. *Ann. Oper. Res.*, **104**, 89–125.

HANEVELD, W. K. K. (1986) *Duality in Stochastic Linear and Dynamic Programming*. Lecture Notes in Economics and Mathematical Systems, vol. 274. Berlin, Germany: Springer.

HIGLE, J. L. & SEN, S. (1996) *Stochastic Decomposition: A Statistcal Method for Large Scale Stochastic Linear Programming*. Dordrecht, The Netherlands: Kluwer Academic Publishers.

HOLMES, D. *A (PO)rtable (S)tochastic Programming (T)est (S)et (POSTS)*. Available at http://users.iems.nwu.edu/~jrbirge/html/dholmes/post.html. Accessed 16 November 2004.

KING, A. J. (1988a) An implementation of the Lagrangian finite-generation method. *Numerical Techniques for Stochastic Optimization* (Yu. Ermoliev & R. J.-B. Wets eds). Berlin, Germany: Springer.

KING, A. J. (1988b) Stochastic programming problems: examples from the literature. *Numerical Techniques for Stochastic Optimization* (Yu. Ermoliev & R. J.-B. Wets eds). Berlin, Germany: Springer.

KLINGMAN, D., NAPIER, A. & STUTZ, J. (1974) NETGEN: a program for generating large scale capacitated assignment, transportation, and minimum cost flow network problems. *Manage. Sci.*, **20**, 814–821.

KRISTJÁNSSON, B. (2002) *MPL for Windows Manual*. Available at http:www.maximalsoftware.com/mplman/mplwtoc.html. Accessed 19 December 2005.

LANE, M. & HUTCHINSON, P. (1980) A model for managing a certificate of deposit portfolio under uncertainty. *Stochastic Programming* (M. A. H. Dempster ed.). London: Academic Press, pp. 473–495.

LOUVEAUX, F. V. & SMEERS, Y. (1988) Optimal investments for electricity generation: A stochastic model and a test problem. *Numerical Techniques for Stochastic Optimization* (Yu. Ermoliev & R. J.-B. Wets eds). Berlin, Germany: Springer.

MULVEY, J. M. & VLADIMIROU, H. (1989) Stochastic network optimization models for investment planning. *Ann. Oper. Res.*, **20**, 187–217.

PFLUG, G. C. (2001) Scenario tree generation for multiperiod financial optimization by optimal discretization. *Math. Program.*, **89**, 251–271.

PRÉKOPA, A. & SZÁNTAI, T. (1978) Flood control reservoir system design using stochastic programming. *Math. Program. Study*, **9**, 138–151.

ROCKAFELLAR, R. T. & WETS, R. J.-B. (1991) Scenario and policy aggregation in optimization under uncertainty. *Math. Oper. Res.*, **16**, 119–147.

URYASEV, S. P. & ROCKAFELLAR, R. T. (2001) Conditional value-at-risk: optimization approach. *Stochastic Optimization: Algorithms and Applications (Gainesville, FL, 2000)*, vol. 54. (S. P. Uryasev & P. M. Pardalos eds). Applied Optimization. Dordrecht, The Netherlands: Kluwer Academic Publishers, pp. 411–435.

WALLACE, S. W. (1986) Solving stochastic programs with network recourse. *Networks*, **16**, 295–317.