WORKING PAPER

TWOSTAGE:
A CODE OF A BASIS DECOMPOSITION METHOD FOR STOCHASTIC PROGRAMMING

B. Strazicky

December 1987
WP-87-126

IIASA
International Institute for Applied Systems Analysis

TWOSTAGE:
A CODE OF A BASIS DECOMPOSITION METHOD FOR STOCHASTIC PROGRAMMING

B. Strazicky

December 1987
WP-87-126

INTERNATIONAL INSTITUTE FOR APPLIED SYSTEMS ANALYSIS
A-2361 Laxenburg, Austria

FOREWORD

The experimental version of the TWOSTAGE code for solving the two stage stochastic linear programs with discretely distributed random right-hand-sides and/or technology matrix gives an alternate possibility to the existing software produced in SDS/ADO.

Alexander B. Kurzhanski
Chairman
System and Decision Sciences Program

CONTENTS

TWOSTAGE: A CODE OF A BASIS DECOMPOSITON METHOD FOR STOCHASTIC PROGRAMMING

B. Strazicky

Computer and Automation Institute
of the Hungarian Academy of Sciences, Budapest

1. INTRODUCTION

The IIASA contracted study "Modeling of Interconnected Power Systems" requires the development of an algorithm for the solution of a model of A. Prékopa [16] fitting to the problem of planning in interconnected power systems.

As the first step in this effort we prepared a first experimental version of the code named TWOSTAGE based on an algorithm developed by B. Strazicky [18] for the solution of the two-stage stochastic programming problem for discrete distributions. The need for such a code is underlined by the fact that the planning model is a combination of chance constrained programming and two-stage programming under uncertainty (which is also called recourse problem).

As summarized in R. Wets' [22] paper there are two main directions in developing algorithms and computer codes for the solution of the two-stage stochastic programming problem with discrete distributions namely the use of the L-shaped algorithm of R. Van Slyke and R. Wets [21] or the use of the above mentioned algorithm of B.Strazicky that relies on the idea of basis decomposition.

The theoretical connections between these algorithms have been discussed – (see [2], [23] and [20]) – and it turned out that the L-shaped algorithm is a version of the Benders decomposition algorithm as specialized for the problem to be solved while the basis decomposition algorithm is a version of the Dantzig-Wolfe decomposition method. This means that the two algorithms are in a sense dual to each other.

Some preliminary computer results concerning the B. Strazicky algorithm are reported in [19]. These results refer to a very early stage of coding where even all the necessary standard linear programming subroutines have been self-coded non-professional codes.

In the present attempt of coding we make a step ahead by using the subroutines of MILP/F linear programming package [11] written in FORTRAN77 for the solution of linear programming problems on personal computers.

In the main part of preparing the code we have been using an IBM PC XT and at the end the resulting code – the experimental version of TWOSTAGE – has been transferred to the VAX machine of the IIASA.

It must be underlined that this version is a first experimental version of TWOSTAGE and we feel that comparing test results with those produced by sophisticated final versions of codes of different methods would not supply much information about the algorithms behind them. The author intends to develop a more sophisticated version of TWOSTAGE well suited to perform computational comparisons with other methods but according to the remark no comparisons are made in this report.

The style of TWOSTAGE is aimed to enable the exchange of the subroutines for some different versions.

2. PROBLEM FORMULATION

As formulated in [4] the two-stage stochastic programming problem to be solved by the algorithm is the following:

$$\text{minimize } c'x + E(\min q'y \mid Tx + My = a,\ y \geq 0 \text{ a.s.})$$

$$\text{subject to } Ax = b \ ,$$

$$x \geq 0$$

where A and M are $m \times n$, $m_0 \times n_0$ matrices, $T(m_0 \times n)$ is a random technology matrix, a is a random vector in R_{m_0}, $b \in R_m$, $q \in R_{n_0}$ $c \in R_n$ are vectors, prime denotes transpose, and E denotes expectation.

The second stage problem is:

$$\text{minimize } q'y$$

$$\text{subject to } My = a - Tx \ ,$$

$$y \geq 0 \ ,$$

where x, T and a are fixed.

We treat this problem in the case when the random vector a and the matrix T have a finite number of possible realizations. Let $f_1, f_2, \cdots f_k$ and $T_1, T_2, \cdots T_k$ denote these realizations and $p_1, p_2, \cdots p_k$ the corresponding probabilities.

The problem is equivalent to the following one:

$$\begin{array}{llllll}
\text{minimize} & (c'x + & p_1 q' y_1 & + p_2 q' y_2 & \ldots + p_k q' y_k) & \\
\text{subject to} & Ax & & & & = b, \\
& T_1 x + & M y_1 & & & = f_1, \\
& T_2 x + & & M y_2 & & = f_2, \\
& \cdot & & & \ddots & \cdot \\
& \cdot & & & & \cdot \\
& T_k x + & & & M y_k & = f_k, \\
& x \geq 0, & y_1 \geq 0, & y_2 \geq 0, & \ldots\ y_k \geq 0 & .
\end{array}$$

The dual of this will be solved by a version of the simplex algorithm. This version uses the fact, that a feasible basis of the special-structured dual problem has favorable special structure too.

The special structure of a feasible basis is as follows:

$$\begin{vmatrix}
B_1 & & & Y_1 & & & \\
& B_2 & & & Y_2 & & \\
& & B_k & & & Y_k & \\
p_1 L_1 & \ldots & p_k L_k & p_1 Z_1 & \ldots & p_k Z_k & Z_0
\end{vmatrix}$$

where $B_1, B_2, \cdots B_k$ are bases of

$$M'w + Is = q,\ s \geq 0\ ,$$

Y_i contains some additional vectors from (M', I) and L_i, Z_i are vectors from $(T_i', 0)$ with the same subscripts as those of B_i and Y_i respectively, for $i = 1, 2, \cdots k$, and Z_0 contains some vectors form (A', I). Here and everywhere in this report I denotes the identity matrix.

Let X_i be defined by the following equality:

$$Y_i = B_i X_i \ \text{ for } i = 1, 2, \ldots k\ .$$

and denote

$$X = \begin{vmatrix}
X_1 & & & \\
& X_2 & & \\
& & & 0 \\
& & X_k &
\end{vmatrix}$$

$$L = |p_1 L_1 \cdots p_k L_k|, \; Z = |p_1 Z_1 \cdots p_k Z_k, \; Z_0| \; .$$

The matrix X is of size $Kn_0 \times n$.

The main idea of the algorithm published in [18] is the following:

With such a feasible basis the corresponding simplex iteration can be carried out without the knowledge of the full basis inverse. By applying a special basis-change strategy the above basis-structure can remain unchanged. By using this strategy in each basis change throughout the simplex iterations the basis structure is always the same and we have to update at most two ones from the necessary inverses of

$$B_i, \; i = 1, 2, \cdots k, \text{ and } (LX - Z) \; .$$

3. RELATION TO OTHER EXISTING ALGORITHMS

Solution methods developed for problems belonging to the family of the so called two-stage programming problems [4] depend very much on te special features of the investigated special case. In this report we are dealing only with those cases where the random elements have discrete distributions and only remark that the algorithms for more general problems are based mostly on approximating the distribution by appropriate discrete ones. ([5], [9], [10]). This means, that the efficiency of the algorithms developed for the discrete problems may strongly influence the efficiency of the algorithms developed for the solution of more general cases.

The two-stage programming problem with discrete random elements may be of three different types:

1. Simple recourse case:

   The recourse matrix M in the second stage problem, representing the additional technology used to have the "random equalities" fulfilled is

   $$|I, \; -I|$$

   where I denotes the identity matrix as before.

2. Only the right-hand-side vector is random.

3. Both the right-hand-side vector and the second stage coefficient matrix T are random.

Methods aiming at the solution of problem of type 1 are based on the fact, that in this case the solution of the second stage problem can be given without solving an LP problem.

Methods for problems of type 2 are based on the fact, that in this case after introducing new variables one gets a staircase structured LP – investigated in the literature of large scale dynamical systems – suitable for developing specialized versions of the simplex method.

Methods for the solution of problems of type 3 are different mainly in the way of solving the primal problem or the dual one of the special structured large scale LP problem corresponding to this case.

3.1. The L-shaped method [21]

Aiming to solve the primal problem leads to a method of R. Wets which can be considered as a version of the Benders decomposition method applied to the large scale LP problem. It involves two main steps and the coordination of these.

Step 1 Solve the LP problem:

$$\begin{array}{lll} & Ax & = b, \\ & D'_k x & \geq d_k,\ k = 1,\dots r, \\ & E'_k x + u & \geq e_k,\ k = 1,\dots s, \\ & x & \geq 0, \\ \min & c'x + u & = z, \end{array}$$

where r and s the numbers of constraints are zero in the first iteration and changing during the iterations. The new coefficient vectors D_{r+1} and E_{s+1} and the right-hand-side values d_{r+1} and e_{s+1} are generated as a result of Step 2.

Step 2/a For each realization of the discrete random matrix and right-hand-side vector make a first phase of the simplex method as applied to the second stage problem with fixed values of the x variable. (The fixed value is the solution of Step 1.)

If there is a realization of the random matrix and the right-hand-side vector for which the optimal value in the first phase is positive, then

let D_{r+1} and d_{r+1} be the linear combination of the rows of that realization of the random matrix and components of that right-hand-side vector respectively with the corresponding optimal simplex multipliers as coefficients,

and go to Step 1.

Step 2/b For each realization of the discrete random matrix and right-hand-side vector solve the second stage problem with fixed values of the x variable. (The fixed value is the solution of Step 1.)

Using the optimal simplex multipliers as coefficients make linear combinations of the rows of the corresponding realizations of the random coefficient matrix and components of the right-hand-side vectors and sum up these linear combinations weighted by the corresponding probabilities. Denote by E_{s+1} the resulting vector and e_{s+1} the resulting constant.

If an optimality condition (see [23]) holds then stop, otherwise go to Step 1.

3.2. The basis decomposition method [18]

Aiming to solve the dual problem results in an algorithm that might also be interpreted as an algorithm involving two main steps and the corresponding coordination of these.

These two steps are as follows:

Step 1 Make one simplex iteration (basis change) ahead to solve the LP problem:

$$\begin{aligned} & A'u + Is + Rv = c, \\ & \quad s \geq 0, \\ \text{max: } & b'u + r'v, \end{aligned}$$

where the matrix R is different in each iteration having at most n columns which are combinations of the columns of the transposed of the realizations T_i, $i = 1, \cdots k$ of the random coefficient matrix T. The second part of the objective function also changes in each iteration.

Step 2 Using the solution of Step 1 make one simplex iteration (basis change) ahead to solve the following problem for each realization of the random coefficient matrix and right hand side vector:

$$M'w + Iz = q,$$
$$z \geq 0,$$
max: $p'w$,

where the objective function coefficients depend on the realizations of the random right hand side vector and of the random coefficient matrix, and are corrected during the iterational process. As starting basis use the optimal basis of the last solution of the problem with the same realization of the random coefficient matrix and right hand side (if it has been solved already before).

If for each i the starting basis is an optimal basis then:

stop .

Otherwise:

Correct the R part of the coefficient matrix and the corresponding objective coefficients and

go to Step 1.

It is obvious that these two algorithms are dual to each other in the sense, that problems solved iteratively are dual to each other.

4. SOME DETAILS ABOUT THE ALGORITHM

4.1. Notation

Notation used in this part of the report are borrowed from the code itself and are of interest only for those who want to modify the code. To help modifications we shortly summarize the most important notations.

IPHASE phase of the algorithm,

IEGYMO number of the subproblems that have been solved and are optimal,

IMUT type of the entering vector,

IBLEP identifier of the part of the coefficient matrix where an entering vector has been found,

IBEV identifier of the entering vector,

$IKLEP$ identifier of the part of the coefficient matrix where the leaving vector belongs to,

$IKIM$ identifier of the leaving vector,

$IPART$ identifier of the part of the basis where the leaving vector belongs to,

$BLXZ$ inverse matrix of $LX - Z$,

$IBLX$ subscripts, characterizing the second part of the basis,

BAZ_i inverse matrix of B_i,

$IBAZ_i$ subscripts of vectors belonging to B_i,

$X0_i$ solution of the i-th subproblem,

$X1_i$ solution belonging to the i-th subbasis in the first part of the basis,

$X2$ solution, belonging to the second part of the basis,

X_i matrix denoted by the same letter in the Problem formulation,

X matrix denoted by the same letter in the Problem formulation, but stored in a compressed form of the size $n_0 \times n$,

$D0_i$ parts of the updated entering vector,

$D1_i$ modificated updated entering vector,

$D2$ last part of the updated entering vector.

4.2. The main steps

Step 0 Initialize

Step 1 Let $u = c$.

For $i = 1, 2, \cdots k$

Solve the subproblems:

$$\max p_i fi_i w,$$

$$M'w + Is = q,$$

$$s \geq 0 \quad .$$

Let $u = u - p_i T_i' w$.

Store BAZ_i, $IBAZ_i$, $X0_i$

If a subproblem is infeasible: TERMINATE

Step 2 If $u \geq 0$ then:

Let $IPHASE = 2$, $Z_0 = I$.

Otherwise let $IPHASE = 1$, $Z_0 = I_+$.

Step 3 Let $BLXZ = - Z_0$, $IBLX$ the corresponding subscripts, $X = (X_1, X_2, \cdots X_k, 0)$

Step 4 Let $X1_j = X0_j$, $X2 = - BLXZu$.

Step 5 Let $IEGYMO = 0$.

If $IPHASE = 1$ then:

Let $CB1 = 0$, $CB2_i = 0$ or -1, $CB3 = - CB2$.

Otherwise:

Compute $CB1$, $CB2$, $CB3$ using $IBLX$, $IBAZ_i$ and X.

Step 6 "OPTVI0"

Let $IMUT = 0$.

Let $IBLEP = 0$, $PI2 = CB3 * BLXZ$.

If there is a j, such that $PI2_j < 0$ then:

let $IMUT = 1$, $IBEV = j$.

If there is a j, such that $(PI2 * A')_j < b_j$ then:

let $IMUT = 2$, $IBEV = j$.

If there is a j, such that $(PI2 * A')_j > b_j$ then:

let $IMUT = 3$, $IBEV = j$.

If $IMUT > 0$ then: let $IEGYMO = 0$ and GO TO Step 8.

If $IMUT = 0$ then: let $IEGYMO = IEGYMO + 1$.

If $IEGYMO = k + 1$ then:

If $IPHASE = 2$ then OPTIMAL SOLUTION, TERMINATE.

If $IPHASE = 1$ then:

let $IPHASE = 2$ and GO TO Step 5.

Otherwise:

Let $J = 1$, GO TO Step 7.

Step 7 "OPTVIJ"

Let $IMUT = 0$, $IBLEP = J$.

$PI1_J = (CB1_J - PI2^*L_J)^*BAZ_J$.

If there is a j, such that $PI1_J(j) < 0$ then:

let $IMUT = 1$, $IBEV = j$.

If there is a j, such that

$$(PI1_J{}^*M' + p_J{}^*PI2^*T_J')_J < (p_Jf_J)_j$$

then: let $IMUT = 2$,$IBEV = j$.

If there is a j, such that

$$(PI1_J{}^*M' + p_J{}^*PI2^*T_J')_j > (p_Jf_J)_j$$

then: let $IMUT = 3$, $IBEV = j$.

If $IMUT > 0$ then: let $IEGYMO = 0$ and GO TO Step 9.

If $IMUT = 0$ then: let $IEGYMO = IEGYMO+1$.

If $IEGYMO = k + 1$ then:

If $IPHASE = 2$ then OPTIMAL SOLUTION, TERMINATE.

If $IPHASE = 1$ then:

let $IPHASE = 2$ and GO TO Step 5.

Otherwise:

If $J = k$ then GO TO step 6.

Otherwise $J = J + 1$ and GO TO Step 7.

Step 8 If $IMUT = 1$ then: $D2 = - BLXZ^*e_{IBEV}$, $D1_J = - X_J{}^*D2$, $J = 1, 2, \cdots k$.

If $IMUT = 2$ then: $D2 = - BLXZ^*A'_{IBEV}$, $D1_J = - X_J{}^*D2$, $J = 1, 2, \cdots k$.

If $IMUT = 3$ then: $D2 = BLXZ^*A'_{IBEV}$, $D1_J = - X_J{}^*D2$, $J = 1, 2, \cdots k$.

Let $IKIM = 0$, $IKLEP = 0$, $IPART = 0$ and GO TO Step 10.

Step 9 If $IMUT = 1$ then:

$$D0 = BAZ_{IBLEP} e_{IBEV},$$
$$D2 = p_{IBLEP} {}^* BLXZ {}^* T'_{IBLEP} {}^* D0,$$
$$D1_J = - X_J {}^* D2,$$
$$D1_{IBLEP} = D0 - X_{IBLEP} {}^* D2.$$

If $IMUT = 2$ then:

$$D0 = BAZ_{IBLEP} M'_{IBLEP},$$
$$D2 = p_{IBLEP} {}^* BLXZ {}^* (T'_{IBLEP} {}^* D0 - T'_{IBLEP, IBEV})$$
$$D1_J = - X_J {}^* D2,$$
$$D1_{IBLEP} = D0 - X_{IBLEP} {}^* D2.$$

If $IMUT = 3$ then:

$$D0 = - BAZ_{IBLEP} M',$$
$$D2 = p_{IBLEP} {}^* BLXZ {}^* (T'_{IBLEP} {}^* D0 + T'_{IBLEP, IBEV})$$
$$D1_J = - X_J {}^* D2,$$
$$D1_{IBLEP} = D0 - X_{IBLEP} {}^* D2.$$

Let $IKIM = 0$, $IKLEP = 0$, $IPART = 0$ and GO TO Step 10.

Step 10 Find:

$$\min \left[\min_{(D1_J)_i > 0 \; J = 1, 2, \ldots, k} \frac{(X1_J)_i}{(D1_J)_i}, \; \min_{D2_i > 0} \frac{X2_i}{D2_i}, \; + \inf \right]$$

Define $IPART$, $IKIM$, $IKLEP$ correspondingly.

If $IPART = 0$ UNBOUNDED PROBLEM, TERMINATE.

If $IPART = 1$ GO TO Step 11.

If $IPART = 2$ GO TO Step 14.

Step 11 If there is a non-zero element in the $IKIM$-th row of X_{IKLEP} then: GO TO Step 12.

Otherwise: GO TO Step 13.

Step 12 Update:

BAZ_{IKLEP}, $IBAZ_{IKLEP}$, X_{IKLEP}, $BLXZ$

$CB1$, $CB2$, $CB3$, $X0$, $X1$, $X2$.

GO TO Step 7.

Step 13 Update:

BAZ_{IKLEP}, $IBAZ_{IKLEP}$, X_{IKLEP}, $BLXZ$

$IBLXZ$,

$CB1$, $CB2$, $CB3$, $X0$, $X1$, $X2$.

GO TO Step 14.

Step 14 Update:

$BLXZ$, $IBLXZ$, X_{IKLEP}, X_{IBLEP},

$CB1$, $CB2$, $CB3$, $X0$, $X1$, $X2$.

If $J=0$ then GO TO Step 6.

Otherwise GO TO Step 7.

5. DESCRIPTION OF TWOSTAGE

The code is a first experimental version using the subroutines of $MILP/F$ linear programming package [11] written in FORTRAN77 for the solution of linear programming problems on professional personal computers.

5.1. Input format

The system starts by a dialogue asking information about the type of the problem, namely if

1 only the right hand side vector is random,

2 only the technology matrix is random,

3 both, the right hand side vector and the technology matrix are random,

4 the problem is an LP.

The name of the input data file has to be typed in. The structure of it depends on the information given before. For an LP problem the input format is as defined in [11].

Otherwise it contains the following information:

1 Problem name

2 Sizes:

number of the first stage constraints

number of the second stage constraints

number of the first stage variables

number of the second stage variables

number of the realizations of the random part

FORMAT:5I5

3 First stage objective:

name

number of nonzero coefficients

index and value of nonzero coefficients

FORMAT: I5 and 5(I5,F12.4)

4 Second stage objective:

name

number of nonzero coefficients

index and value of nonzero coefficients

FORMAT: I5 and 5(I5,F12.4)

5 First stage rhs:

name

number of nonzero coefficients

index and value of nonzero coefficients

FORMAT: I5 and 5(I5,F12.4)

6 Transposed of the first stage coefficient matrix:

name

for each column:

number of nonzero coefficients

index and value of nonzero coefficients

FORMAT: I5 and 5(I5,F12.4)

7 Transposed of the second stage recourse matrix:

name

for each column:

number of nonzero coefficients

index and value of nonzero coefficients

FORMAT: I5 and 5(I5,F12.4)

and thereafter:

in Case 1.

8 Transposed of the second stage coefficient matrix:

(not random in this case)

name

for each column:

number of nonzero coefficients

index and value of nonzero coefficients

FORMAT: I5 and 5(I5,F12.4)

9 Random part:

for each possible realization

name

probability FORMAT F12.4

name of rhs

number of nonzero coefficients

index and value of nonzero coefficients

FORMAT: I5 and 5(I5,F12.4)

in Case 2.

8 Second stage rhs:

(not random in this case)

name

number of nonzero coefficients

index and value of nonzero coefficients

FORMAT: I5 and 5(I5,F12.4)

9 Random part:

for each possible realization

name

probability FORMAT F12.4

9/a. Transposed of the second stage coefficient matrix:

name

for each column:

number of nonzero coefficients

index and value of nonzero coefficients

FORMAT: I5 and 5(I5,F12.4)

in Case 3.

8 Random part:

for each possible realization

name

probability FORMAT F12.4

9/a. Second stage rhs:

name

number of nonzero coefficients

index and value of nonzero coefficients

FORMAT: I5 and 5(I5,F12.4)

9/b. Transposed of the second stage coefficient matrix:

name

for each column:

number of nonzero coefficients

index and value of nonzero coefficients

FORMAT: I5 and 5(I5,F12.4)

5.2. Routines

MAIN PROGRAM: TWOSTAGE

SUBROUTINES: COPYRG
PROBLIDENT
FILIN(irrh,irst,irg,ipri)

ADATKI
STE1
STE2
OPTVIJ(jjj,kez,ians)
OPTVI(kez,ians)
KILEP(ian2)
CSERE
VEGE
NEMKVEG
RDUMUJ(jbas)
BLXZ(j,jj,sxz)
LPINIT(jas,jbas,jats)
LINPRO(jas,jbas,jats)
TM(io)
TT(idb,io)
TA(io)
F(idb)

SUBROUTINES belonging to MILP/F are not mentioned above but these are also included into the code.

5.3. Result file

The name of the result file is: ERED.

The dialogue system asks if one requires solution of subproblems.

5.4. Testing of the code

For to test the code for its correctness we solved some test problems by using TWOSTAGE and parallelly also by using the MILP/F linear programming package and compared the solutions.

Additional testing efforts are necessary to see and to increase the efficiency of TWOSTAGE. As a consequence of these test runs minor changes in the data storage scheme may be expected.

It would be interesting to test for the size of that two-stage programming problem for which the efficiency of using the rather complicated basis decomposition algorithm is superior as compared with the possibility of solving it as a large scale LP in a straightforward mannner.

The comparison of the code NDSP [7] resp. QDECOM [17] and an advanced version of TWOSTAGE could lead also to interesting conclusions about the sizes and/or structure of problems for which NDSP, QDECOM or TWOSTAGE is more suitable.

REFERENCES

[1] Birge, J.R.: Decomposition and Partitioning Methods for Multi-Stage Stochastic Linear Programs, Operations Research Vol. 33, No.5, September-October 1985, 989–1007.

[2] Birge, J.R.: A Dantzig-Wolfe Decomposition Variant Equivalent to Basis Factorization, Mathematical Programming Study 24 (1985), 43–64.

[3] Dantzig, G.B.: Linear Programming under Uncertainty, Management Science 1 (1955), 197–206.

[4] Dantzig, G.B. and A. Madansky: On the Solution of Two-stage Linear Programs under Uncertainty, In: Proc. of the Fourth Berkeley Symposium on Math. Stat. and Probability, Vol. I. University of California Press, Berkeley, 165–176.

[5] Dupačová, J.: Minimax Approach to Stochastic Linear Programming and the Moment Problem. Selected Results. ZAMM 58, T466–T467 (1978).

[6] Edwards, J., J. Birge, A. King and L. Nazareth: A Standard Input Format for Computer Codes which Solve Stochastic Programs with Recourse and a Library of Utilities to simplify its use, IIASA Working Paper WP-85-03, (1985).

[7] Edwards, J.: NDSP User's Manual in: Documentation for the ADO/SDS Collection of Stochastic Programming Codes, Ed: Edwards, J. IIASA Working Paper, WP-85-02, (1985).

[8] Ho, J.K.: A Comparative Study of Two Methods for Staircase Linear Programs, ACM Transactions on Mathematical Software, Vol. 5, No.4, (1979), 17–30.

[9] Kall, P.: Approximations to Stochastic Programming with Complete Fixed Recourse, Num. Math. 22 (1974), 333–339.

[10] Kall, P.: Computational Methods for Solving Two-Stage Stochastic Linear Programming Problems, Z. Angew. Math. Phys. 30 (1979), 261–271.

[11] Maros, I.: MILP/F linear programming system V1.00 (1985) Budapest, Hungary.

[12] Nazareth, J.L. and R. Wets: Algorithms for Stochastic Programs: The Case of Nonstochastic Tenders, IIASA Working Paper, WP-83-5, (1983).

[13] Nazareth, J.L.: Hierarchical Implementation of Optimization Methods, in: Numerical Optimization 1984, Eds: P.T. Boggs, R.H. Byrd and R.B. Schnabel, SIAM Philadelphia/1985.

[14] Nazareth, J.L.: Design and Implementation of a Stochastic Programming Optimizer with Recourse and Tenders, IIASA Working Paper, WP-85-063, (1985).

[15] Prékopa, A.: Recent Results in Optimization of Electro-energetic Systems, in: Proceedings of the Conference Applied Optimization Techniques in Energy Problems, Ed. Hj. Wacker, B.G. Teubner Stuttgart, 1985, 354–383.

[16] Prékopa, A.: Network Planning Using Two-Stage Programming under Uncertainty, in: Recent Results in Stochastic Programming, Proceedings, Oberwolfach 1979, Eds: P. Kall, A. Prékopa, Springer Verlag, (1980), 215–237.

[17] Ruszczynski, A.: QDECOM: The Regularized Decompositon Method, User's Manual, Manuscript, Institut für OR der Universität Zürich, October 1985.

[18] Strazicky, B.: On an Algorithm for Solution of the Two-Stage Stochastic Programming Problem, Methods of Operations Research XIX (1974), 142–156.

[19] Strazicky, B.: Some Results Concerning an Algorithm for the Discrete Recourse Problem, in: Stochastic Programming, Ed. M. Dempster, Academic Press, London, 1980, 263–274.

[20] Strazicky, B.: On the Solution of the Two-Stage Stochastic Programming Problem, Lecture held at the IIASA Workshop on "Numerical Methods for Stochastic Optimization", 28 November - 2 December 1983, Laxenburg, Austria.

[21] Van Slyke, R.M. and R. Wets: L-Shaped Linear Programs with Applications to Optimal Control and Stochastic Programming, SIAM J. Appl. Math. Vol. 17, No. 4, (1969), 638–663.

[22] Wets, R.: Stochastic Programming: Solution Techniques and Approximation Schemes, in: Mathematical Programming: State of the Art Bonn 1982, Eds: A. Bachem, M. Grotschel, B. Korte, Springer Verlag, Berlin, (1983), 506–603.

[23] Wets, R.: Large Scale Linear Programming Techniques in Stochastic Programming, IIASA Working Paper, WP-84-90, (1984).