Proceedings of ALGORITMY 2002
Conference on Scientific Computing, pp. 102–112

PARALLEL IMPLEMENTATION OF BIRGE AND QI METHOD FOR THREE-STAGE STOCHASTIC PROGRAMS USING IPM*

G.CH.PFLUG†, L.HALADA‡, AND M.LUCKA§

Abstract.
One approach how to solve a linear optimization problem is based on the interior point method. This method requires a solution of the large system of linear equations. A special matrix factorization techniques that exploit the structure of the constraint matrix has been suggested for its computation. The method of Birge and Qi has been reported as efficient, stable and accurate for two-stage stochastic programs. In this report we present a generalization of this method for three-stage stochastic programs. For this method we have proposed a parallel algorithm based on the Message Passing Interface (MPI). The algorithm was coded in the Fortran 90 programming language, whereby for solving of linear algebra problems the linear algebra package LAPACK has been used.

Key words. Stochastic methods, linear algebra, MPI, LAPACK

AMS subject classifications. 82B31, 60H35, 68W10, 90C51

1. Introduction. Many practical problems with uncertain parameters can be modeled as stochastic programs. In the literature one can find such applications in science, technology and economy. References to the wide range of applications are made in the textbook of Kall and Wallace [1], too. Stochastic programs are usually very difficult and even in the simplest case of linear programming problems with finitely many events (scenarios) they lead to a system of equations with very large numbers of variables and constraints. One approach for solving of such stochastic problems is Interior Point Method (IPM). They are many variants of this method. We will use primal-dual path-following algorithm based on the solution of Kuhn-Karush-Tucker (KKT) equation. Crucial for efficiency of this approach is the solution of linear system of the form

$$(ADA^t)\,\Delta y = b. \tag{1}$$

Solving of this problem requires more then $90 - 95\%$ of total programming time [2]. Birge and Holmes [3] compared different methods for the solution of this system for two-stage stochastic programs. They found that the factorization technique based on the work of Birge and Qi (BQ) [4] is more efficient and stable than other methods. A parallel version of BQ for two-stage stochastic programs was also implemented on an Intel iPSC/860 hypercube and a Connection Machine CM-5 with nearly perfect speedup [5].

The aim of this report is a suggestion how the BQ method can be used for the three-stage stochastic programming. In Section 2 we introduce the problem formulation. An application of the BQ to a three-stage stochastic model together with an

*This work was supported by the Special Research Program SFB F011 "AURORA" of the Austrian Science Fund FWF.

†Department of Statistics and Decision Support Systems, Vienna University, Universitaesstr. 5, A-1010 Wien, (georg.pflug@univie.ac.at).

‡Institute for Informatics, Slovak Academy of Sciences, Dubravska 9, 842 37 Bratislava, Slovakia, (upsyhala@savba.sk).

§Institute for Software Science, University of Vienna, Liechtensteinstr.22, A-1090 Vienna, Austria, (lucka@par.univie.ac.at).

algorithm is given in Section 3. In the last section a parallel implementation of this algorithm is described and the speed-up for different number of parallel processes is illustrated on an example. The results of experiments were obtained on the parallel Beowulf-Cluster Gescher at the VCPC, University of Vienna.

2. The two-stage problem formulation. The deterministic equivalent formulation of the two-stage stochastic linear program have the following form:

$$\begin{aligned} &\text{minimize} \quad \{c^t x + \sum_{i=1}^{k} pr_i d_i y_i\} \\ &\text{subject to} \qquad A_0 x = b_0 \\ &\qquad T_i x + A_i y_i = b_i \\ &x \geq 0,\ y_i \geq 0,\ i = 1, 2, ..., k \end{aligned} \tag{2}$$

where x is the vector of decision variables whose optimal value is not conditioned on the realization of uncertain parameters. The variable y_i denotes the vector of control decision (recourse action). The number of the different possible future outcomes (scenarios) to be assumed is k and probability of its occurrence $pr_i, i = 1, 2, ...k$. Thus, the constraint matrix of the two-stage stochastic program has the following block angular structure

$$A = \begin{pmatrix} A_0 & & & & \\ T_1 & A_1 & & & \\ T_2 & & A_2 & & \\ \vdots & & & \ddots & \\ T_k & & & & A_k \end{pmatrix}. \tag{3}$$

The standard IPM approach for finding the solution of (2) is the primal-dual path-following algorithm based on the solution of perturbed KKT system

$$\begin{pmatrix} & A^t & I \\ A & & \\ Z & & X \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} c \\ b \\ \mu e \end{pmatrix}, \tag{4}$$

where X and Z are diagonal matrices whose diagonal entries come from vectors x and z, respectively and e denotes the vector of all ones. As $\mu \downarrow 0$ the solution of this system converge to the solution of our two-stage problem. Using the Newton method for solving of (4) with the iterative updates $(x, y, z,)$ by the formula

$$\begin{aligned} x &:= x + \alpha_p \Delta x \\ y &:= y + \alpha_d \Delta y \\ z &:= z + \alpha_d \Delta z, \end{aligned} \tag{5}$$

where α_p, $\alpha_d \in (0, 1)$ are chosen to keep $x > 0$ and $z > 0$, we obtain

$$\Delta z = X^{-1} r_c - X^{-1} Z \Delta x \tag{6}$$

$$\Delta x = Z^{-1}(X A^t \Delta y + r_c - X r_d) \tag{7}$$

$$(ADA^t)\ \Delta y = r_p + AZ^{-1}(X r_d - r_c). \tag{8}$$

Here $D = Z^{-1}X$ is a diagonal matrix with all entries strictly positive and r_p, r_d and r_μ are residual vectors of the perturbed KKT system. We note, if matrix A has full row rank then ADA^t is a symmetric positive definite and thus the system (8) has a well defined solution.

The most difficult part of the computation (6)-(8) is the solution of the system (8). Birge and Qi [4] suggested to solve such system by the factorization of (ADA^t) matrix that exploits the structure of the constraint matrix A. The basic idea is as follows:

Let (ADA^t) be expressed as

$$ADA^t = \mathcal{R}^{(2)} + U^{(2)}\,[\,\mathcal{D}^{(2)}\,(V^{(2)})^t\,] = \mathcal{R}^{(2)} + U^{(2)}\,[\,(W^{(2)})^t\,], \tag{9}$$

where

$$\mathcal{R}^{(2)} = Diag(I_{m_0},\, A_1D_1A_1^t, \ldots, A_kD_kA_k^t) = Diag(I_{m_0},\, R_1, \ldots, R_k) \tag{10}$$

and

$$U^{(2)}\,\mathcal{D}^{(2)}\,(V^{(2)})^t = \begin{pmatrix} A_0 & I_{m_0} \\ T_1 & \\ T_2 & \\ \vdots & \\ T_k & \end{pmatrix} \begin{pmatrix} D_0 & \\ & I_{m_0} \end{pmatrix} \begin{pmatrix} A_0^t & T_1^t & T_2^t & \ldots & T_k^t \\ -I_{m_0} & & & & \end{pmatrix}.$$

Then for the inverse of (ADA^t) they suggested to use the Sherman-Morrison-Woodbury formula. It holds [6]

$$(ADA^t)^{-1} = (\mathcal{R}^{(2)})^{-1} - (\mathcal{R}^{(2)})^{-1}\,U^{(2)}\,(G^{(2)})^{-1}\,(V^{(2)})^t\,(\mathcal{R}^{(2)})^{-1}, \tag{11}$$

if and only if both $\mathcal{R}^{(2)}$ and $G^{(2)}$ are nonsingular, where

$$(G^{(2)})^{-1} = [I_{n_0+m_0} + (W^{(2)})^t(\mathcal{R}^{(2)})^{-1}U^{(2)}]^{-1}D^{(2)}. \tag{12}$$

In the matrix form

$$G^{(2)} = \begin{pmatrix} D_0^{-1} + A_0^tA_0 + \sum_{i=1}^k T_i^tR_i^{-1}T_i & A_0^t \\ -A_0 & 0 \end{pmatrix} = \begin{pmatrix} \hat{G}^{(2)} & A_0^t \\ -A_0 & 0 \end{pmatrix}. \tag{13}$$

It has been proved [3] that if the matrix A has full row rank then $\hat{G}^{(2)}$ is positive definite and symmetric matrix and $G^{(2)}$ is nonsingular. Hence, the conditions for the validity of (11) are fulfilled. Thus, we can rewrite the solution of the system $(ADA^t)\,dy = b$ by the relation (11) as follows: $dy = p^{(2)} - s^{(2)}$, where

$$\mathcal{R}^{(2)}\,p^{(2)} = b \tag{14}$$

$$G^{(2)}\,q^{(2)} = (V^{(2)})^t\,p^{(2)} \tag{15}$$

$$\mathcal{R}^{(2)}\,s^{(2)} = U^{(2)}\,q^{(2)}. \tag{16}$$

The procedure for sequential computing of the vector dy by (14)-(16) has been formulated in [3] and named Finddy. Formally, parameters of this procedure are

$$\text{Finddy}(\mathcal{R}^{(2)},\, A_0,\, D_0, T_1, \ldots, T_k, b, dy).$$

In the next section we will use this procedure in the formulation of the algorithm for the three-stage stochastic program.

3. BQ Method Applied to Three-Stage Stochastic Model. As in the two-stage case, we can write the deterministic equivalent system of equations for tree-stage problem, too. If we will suppose the two scenarios in second and three scenarios in the third stage then the constraint matrix of the tree-stage problem $A^{(3)}$ in this case will have the following form:

$$A^{(3)} = \begin{pmatrix} A_0 & & & & & & & & \\ T_{10} & A_{10} & & & & & & & \\ & T_{11} & A_{11} & & & & & & \\ & T_{12} & & A_{12} & & & & & \\ & T_{13} & & & A_{13} & & & & \\ T_{20} & & & & & A_{20} & & & \\ & & & & & T_{21} & A_{21} & & \\ & & & & & T_{22} & & A_{22} & \\ & & & & & T_{23} & & & A_{23} \end{pmatrix} \tag{17}$$

where A_0 is $m_0 \times n_0$ and A_{ij} are $m_{ij} \times n_{ij}$ matrices. T_{ij} has the size conformable to the matrices A_0 and A_{ij}. In the block form

$$A^{(3)} = \begin{pmatrix} A_0 & & \\ T_1^{(3)} & A_1^{(2)} & \\ T_2^{(3)} & & A_2^{(2)} \end{pmatrix}, \tag{18}$$

where $A_i^{(2)}, i = 1, 2$ represent the two-stage problem in the frame of the whole three-stage problem. Let $D^{(3)}$ be diagonal and positive definite matrix

$$D^{(3)} = Diag(D_0;\ D_{10}, \dots, D_{13};\ D_{20}, \dots, D_{23}) = Diag(D_0, D_1^{(2)}, D_2^{(2)}), \tag{19}$$

where D_0 and D_{ij} are diagonal $n_0 \times n_0$ and $n_{ij} \times n_{ij}$ matrices with positive entries, $i = 1, 2;\ j = 0, 1, 2, 3.$. Again, we assume that A_0 and A_{ij} have full row rank and $m_0 \le n_0,\ \ m_{ij} \le n_{ij},\ \ i = 1, 2;\ j = 0, 1, 2, 3$. Now, our main problem is to find the solution of the system

$$[A^{(3)} D^{(3)} (A^{(3)})^t] dy^{(3)} = b^{(3)}$$

The matrix $R^{(3)} = A^{(3)} D^{(3)} (A^{(3)})^t$ in the denotation of the largesized blocks matrix $A^{(3)}$ can be decomposed as follows:

$$R^{(3)} = \mathcal{R}^{(3)} + U^{(3)} [\, \mathcal{D}^{(3)}\ (V^{(3)})^t\,] = \mathcal{R}^{(3)} + U^{(3)} [\, (W^{(3)})^t\,],$$

where

$$\mathcal{R}^{(3)} = Diag(I_{m_0},\ A_1^{(2)} D_1^{(2)} (A_1^{(2)})^t,\ A_2^{(2)} D_2^{(2)} (A_2^{(2)})^t) = Diag(I_{m_0},\ R_1^{(2)},\ R_2^{(2)})$$

and

$$U^{(3)} \mathcal{D}^{(3)}\ (V^{(3)})^t = \begin{pmatrix} A_0 & I_{m_0} \\ T_1^{(3)} & \\ T_2^{(3)} & \end{pmatrix} \begin{pmatrix} D_0 & \\ & I_{m_0} \end{pmatrix} \begin{pmatrix} A_0^t & (T_1^{(3)})^t & (T_2^{(3)})^t \\ -I_{m_0} & & \end{pmatrix}.$$

By the Sherman-Morrison-Woodbury formula we obtain again

$$(\, R^{(3)}\,)^{-1} = (\mathcal{R}^{(3)})^{-1} - (\mathcal{R}^{(3)})^{-1} U^{(3)}\ (G^{(3)})^{-1}\ (V^{(3)})^t\ (\mathcal{R}^{(3)})^{-1}, \tag{20}$$

if and only if $\mathcal{R}^{(3)}$ and $G^{(3)}$ are nonsingular. Here

$$(G^{(3)})^{-1} = [\, I_{n_0+m_0} + \mathcal{D}^{(3)}\ (V^{(3)})^t\ (\mathcal{R}^{(3)})^{-1}\ U^{(3)}]^{-1}\ \mathcal{D}^{(3)}$$

$$G^{(3)} = (\mathcal{D}^{(3)})^{-1} + (V^{(3)})^t\ (\mathcal{R}^{(3)})^{-1}\ U^{(3)}. \tag{21}$$

In the matrix form:

$$G^{(3)} = \begin{pmatrix} D_0^{-1} + A_0^t A_0 + \sum_{i=1}^{2} (T_i^{(3)})^t (R_i^{(2)})^{-1} T_i^{(3)} & A_0^t \\ -A_0 & 0 \end{pmatrix} = \begin{pmatrix} \hat{G}^{(3)} & A_0^t \\ -A_0 & 0 \end{pmatrix}. \tag{22}$$

The validity of (20) follows from the same reason as for the two-stage model. Thus, the solution of $R^{(3)}\, dy^{(3)} = b^{(3)}$ can be expressed by the inversion as $dy^{(3)} = p^{(3)} - s^{(3)}$ while

$$\mathcal{R}^{(3)} p^{(3)} = b^{(3)}, \tag{23}$$

$$G^{(3)} q^{(3)} = (V^{(3)})^t p^{(3)}, \tag{24}$$

$$\mathcal{R}^{(3)} s^{(3)} = U^{(3)} q^{(3)}. \tag{25}$$

The equations (23)-(25) represent the decomposition of the original problem into three sub-problems. An advantage of such decomposition is that $\mathcal{R}^{(3)}$ is the block-diagonal matrix amenable to further decomposition.

3.1. Solving the Equation $\mathcal{R}^{(3)} p^{(3)} = b^{(3)}$. It is easy to see from the equation

$$\mathcal{R}^{(3)} p^{(3)} = \begin{pmatrix} I_{m_0} & & \\ & R_1^{(2)} & \\ & & R_2^{(2)} \end{pmatrix} \begin{pmatrix} p_0^{(3)} \\ p_1^{(3)} \\ p_2^{(3)} \end{pmatrix} = \begin{pmatrix} b_0^{(3)} \\ b_1^{(3)} \\ b_2^{(3)} \end{pmatrix} \tag{26}$$

that this system represents the following independent systems

$$p_0^{(3)} = b_0^{(3)} \tag{27}$$

$$R_i^{(2)} p_i^{(3)} = b_i^{(3)}, \quad i = 1, 2, \tag{28}$$

where $R_i^{(2)} = A_i^{(2)} D_i^{(2)} (A_i^{(2)})^t$, $i = 1, 2$ represent the matrices of the two-stage model problem, which has been described in Sect.2. Therefore, (28) can be solved by the procedure Finddy. Its input parameters are readable from the entries of matrix $A_i^{(2)}, D_i^{(2)}$. The right-hand side and the solution vector are $b_i^{(3)}$ and $p_i^{(3)}$, $i = 1, 2$, respectively. It is clear that in our case the parameters are

$$\text{Finddy}(\mathcal{R}_i^{(2)},\ A_{i0},\ D_{i0}, T_{i1}, \ldots, T_{i3}, b_i^{(3)}, p_i^{(3)}), \quad i = 1, 2$$

where $\mathcal{R}_i^{(2)}$ is the diagonal matrix in the decomposition $R_i^{(2)}$, i.e.

$$R_i^{(2)} = \mathcal{R}_i^{(2)} + U_i^{(2)} (W_i^{(2)})^t \quad i = 1, 2 \tag{29}$$

and

$$\mathcal{R}_i^{(2)} = Diag(I_{m_{i0}},\ R_{i1}, R_{i2}, R_{i3}), \quad R_{ij} = A_{ij} D_{ij} A_{ij}^t, \quad i = 1, 2 \quad j = 1, 2, 3$$

$$U_i^{(2)} = \begin{pmatrix} A_{i0} & I_{m_{i0}} \\ T_{i1} & \\ T_{i2} & \\ T_{i3} & \end{pmatrix}, \quad (W_i^{(2)})^t = \begin{pmatrix} D_{i0} & \\ & I_{m_{i0}} \end{pmatrix} \begin{pmatrix} A_{i0}^t & T_{i1}^t & T_{i2}^t & T_{i3}^t \\ -I_{m_{i0}} & & & \end{pmatrix}.$$

If this procedure is applied for the given values, having in the mind the relations (27)-(28) we are able to compose the vector $p^{(3)}$.

3.2. Solving the Equation $G^{(3)}q^{(3)} = (V^{(3)})^t p^{(3)}$. Solving of this equation requires to have the entries of the right-hand side vector and the sub-block matrix $\hat{G}^{(3)}$ available .With this aim we denote the elements of the vector $(V^{(3)})^t p^{(3)}$ as $(\hat{v}_1^{(3)}, \hat{v}_2^{(3)})^t$. Then we have

$$\left(\begin{array}{c} \hat{v}_1^{(3)} \\ \hat{v}_2^{(3)} \end{array} \right) = \left(\begin{array}{c} A_0^t p_0^{(3)} + \sum_{i=1}^2 T_{i0}^t p_{i0}^{(3)} \\ -p_0^{(3)} \end{array} \right), \tag{30}$$

where $p_{i0}^{(3)}$, $i = 1,2$ is the vector of the first m_{i0}-elements of $p_i^{(3)}$. We know from (22) that

$$\hat{G}^{(3)} = D_0^{-1} + A_0^t A_0 + \sum_{i=1}^{2} (T_i^{(3)})^t (R_i^{(2)})^{-1} T_i^{(3)}. \tag{31}$$

For the relatively complicated expression $(T_i^{(3)})^t (R_i^{(2)})^{-1} T_i^{(3)}$ we can prove that

$$(T_i^{(3)})^t (R_i^{(2)})^{-1} T_i^{(3)} = T_{i0}^t (\hat{T}_{i0} - T_{i0}), \quad i = 1,2 \tag{32}$$

where $\hat{T}_{i0}$ is the solution of the equation

$$[\, A_{i0}\, (\hat{G}_i^{(2)})^{(-1)}\, A_{i0}^t \,]\, \hat{T}_{i0} = T_{i0}, \quad i = 1,2. \tag{33}$$

Really, according to (11) we have

$$\begin{aligned} &(T_i^{(3)})^t (R_i^{(2)})^{-1} T_i^{(3)} = \\ &(T_i^{(3)})^t \,[\, (\mathcal{R}_i^{(2)})^{-1} - (\mathcal{R}_i^{(2)})^{-1} U_i^{(2)} (G_i^{(2)})^{-1} (V_i^{(2)})^t (\mathcal{R}_i^{(2)})^{-1} \,]\, T_i^{(3)} = \\ &(T_i^{(3)})^t (\mathcal{R}_i^{(2)})^{-1} T_i^{(3)} - (T_i^{(3)})^t (\mathcal{R}_i^{(2)})^{-1} U_i^{(2)} (G_i^{(2)})^{-1} (V_i^{(2)})^t (\mathcal{R}_i^{(2)})^{-1} T_i^{(3)}. \end{aligned}$$

It holds

$$(T_i^{(3)})^t (\mathcal{R}_i^{(2)})^{-1} T_i^{(3)} = T_{i0}^t T_{i0}, \quad (T_i^{(3)})^t (\mathcal{R}_i^{(2)})^{-1} U_i^{(2)} = (T_{i0}^t A_{i0}, T_{i0}^t), \tag{34}$$

$$(V_i^{(2)})^t (\mathcal{R}_i^{(2)})^{-1} T_i^{(3)} = \left(\begin{array}{c} A_{i0}^t T_{i0} \\ -T_{i0}) \end{array} \right). \tag{35}$$

Therefore

$$(T_i^{(3)})^t (R_i^{(2)})^{-1} T_i^{(3)} = T_{i0}^t\, T_{i0} - (T_{i0}^t A_{i0},\ T_{i0}^t) \left(\begin{array}{cc} \hat{G}_i^{(2)} & A_{i0}^t \\ -A_{i0} & 0 \end{array} \right)^{-1} \left(\begin{array}{c} A_{i0}^t T_{i0} \\ -T_{i0} \end{array} \right). \tag{36}$$

Now let

$$\left(\begin{array}{cc} K & L \\ M & N \end{array} \right) = \left(\begin{array}{cc} \hat{G}_i^{(2)} & A_{i0}^t \\ -A_{i0} & 0 \end{array} \right)^{-1}. \tag{37}$$

According to [7]

$$N = [\, A_{i0} (\hat{G}_i^{(2)})^{-1} A_{i0}^t \,]^{-1} \tag{38}$$

$$L = -(\hat{G}_i^{(2)})^{-1} A_{i0}^t \; [\, A_{i0} (\hat{G}_i^{(2)})^{-1} A_{i0}^t \,]^{-1} \tag{39}$$

$$M = [\, A_{i0} (\hat{G}_i^{(2)})^{-1} A_{i0}^t \,]^{-1}\; A_{i0} (\hat{G}_i^{(2)})^{-1} \tag{40}$$

$$K = (\hat{G}_i^{(2)})^{-1} - (\hat{G}_i^{(2)})^{-1} A_{i0}^t [\, A_{i0} (\hat{G}_i^{(2)})^{-1} A_{i0}^t \,]^{-1} A_{i0} (\hat{G}_i^{(2)})^{-1}. \tag{41}$$

Now, if we use (38)-(41) in (36) we obtain

$$(T_i^{(3)})^t (R_i^{(2)})^{(-1)} T_i^{(3)} = T_{i0}^t [\, A_{i0} (\hat{G}_i^{(2)})^{-1} A_{i0}^t \,]^{-1} T_{i0} - T_{i0}^t \, T_{i0}. \tag{42}$$

Thus,

$$\hat{G}^{(3)} = D_0^{-1} + A_0^t A_0 + \sum_{i=1}^{2} T_{i0}^t (\hat{T}_{i0} - T_{i0}). \tag{43}$$

We remember that the Cholesky decomposition of matrix $A_{i0}(\hat{G}_i^{(2)})^{-1} A_{i0}^t$ has been performed during the procedure Finddy applied on matrix $R_i^{(2)}$, $i = 1, 2$. Thus, this decomposition is available already, only triangular solver is used for the computation of $\hat{T}_{i0}$, $i = 1, 2$ in this step. Having the values of $\hat{G}^{(3)}$ and $(\hat{v}_1^{(3)}, \hat{v}_2^{(3)})^t$ we can solve the system

$$\begin{pmatrix} \hat{G}^{(3)} & A_0^t \\ -A_0 & 0 \end{pmatrix} \begin{pmatrix} q_1^{(3)} \\ q_2^{(3)} \end{pmatrix} = \begin{pmatrix} \hat{v}_1^{(3)} \\ \hat{v}_2^{(3)} \end{pmatrix}. \tag{44}$$

The standard elimination process applied to this system yields

$$[(A_0 \, (\hat{G}^{(3)})^{-1} \, A_0^t] \, q_2^{(3)} = A_0 \, (\hat{G}^{(3)})^{-1} \, \hat{v}_1^{(3)} + \hat{v}_2^{(3)} \tag{45}$$

$$\hat{G}^{(3)} \, q_1^{(3)} = \hat{v}_1^{(3)} - A_0^t \, q_2^{(3)}. \tag{46}$$

Thus, to solve (44) the following procedure is required:

PROCEDURE Updy($\hat{G}^{(3)}$, A_0, $\hat{v}_1^{(3)}, \hat{v}_2^{(3)}$)

(a) Form the Cholesky decomposition of $\hat{G}^{(3)}$

(b) Solve the systems $\hat{G}^{(3)} B_0 = A_0^t$

(c) Form the Cholesky decomposition of $A_0 B_0$

(d) Solve the systems (45) and (46).

3.3. Solving Equation $\mathcal{R}^{(3)} s^{(3)} = U^{(3)} q^{(3)}$. The system $\mathcal{R}^{(3)} \, s^{(3)} = U^{(3)} q^{(3)}$ could be solved in a similar way as in Sect. 3.1. The right-hand side equals

$$U^{(3)} q^{(3)} = \begin{pmatrix} A_0 & I_{m_0} \\ T_1^{(3)} & 0 \\ T_2^{(3)} & 0 \end{pmatrix} \begin{pmatrix} q_1^{(3)} \\ q_2^{(3)} \end{pmatrix} = \begin{pmatrix} A_0 \, q_1^{(3)} + q_2^{(3)} \\ T_1^{(3)} \, q_1^{(3)} \\ T_2^{(3)} \, q_1^{(3)} \end{pmatrix}. \tag{47}$$

Thus, the system has the form

$$\mathcal{R}^{(3)} s^{(3)} = \begin{pmatrix} I_{m_0} & & \\ & R_1^{(2)} & \\ & & R_2^{(2)} \end{pmatrix} \begin{pmatrix} s_0^{(3)} \\ s_1^{(3)} \\ s_2^{(3)} \end{pmatrix} = \begin{pmatrix} A_0 \, q_1^{(3)} + q_2^{(3)} \\ T_1^{(3)} \, q_1^{(3)} \\ T_2^{(3)} \, q_1^{(3)} \end{pmatrix}, \tag{48}$$

from which we obtain independent equations

$$s_0^{(3)} = A_0 \, q_1^{(3)} + q_2^{(3)} \tag{49}$$

$$R_i^{(2)} s_i^{(3)} = T_i^{(3)} \, q_1^{(3)}, \; i = 1, 2. \tag{50}$$

The last two equations are again solvable by the procedure Finddy as in Sect. 3.1 with the right-hand side $T_i^{(3)} q_1^{(3)}$, $i = 1, 2$. But, owing to the structure of this vector, where

only the first m_{i0}- entries are nonzero, we suggest for its computation a modification the already mentioned procedure Finddy.
The solution $s_i^{(3)}$ of (50) can be expressed as $s_i^{(3)} = \hat{p}_i^{(3)} - \hat{s}_i^{(3)}$, where $\hat{p}_i^{(3)}$ and $\hat{s}_i^{(3)}$, $i = 1, 2$ fulfill

$$\mathcal{R}_i^{(2)}\,\hat{p}_i^{(3)} = T_i^{(3)}\,q_1^{(3)} \tag{51}$$

$$G_i^{(2)}\,\hat{q}_i^{(3)} = (V_i^{(2)})^t\,\hat{p}_i^{(3)} \tag{52}$$

$$\mathcal{R}_i^{(2)}\,\hat{s}_i^{(3)} = U_i^{(2)}\,\hat{q}_i^{(3)}. \tag{53}$$

In the matrix form

$$\mathcal{R}_i^{(2)}\,\hat{p}_i^{(3)} = \begin{pmatrix} I_{m_{i0}} & & & \\ & R_{i1} & & \\ & & R_{i2} & \\ & & & R_{i3} \end{pmatrix} \begin{pmatrix} \hat{p}_{i0}^{(3)} \\ \hat{p}_{i1}^{(3)} \\ \hat{p}_{i2}^{(3)} \\ \hat{p}_{i3}^{(3)} \end{pmatrix} = \begin{pmatrix} T_{i0}\,q_1^{(3)} \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{54}$$

from which we have immediately

$$\hat{p}_{i0}^{(3)} = T_{i0} q_1^{(3)}, \quad i = 1, 2 \tag{55}$$

$$\hat{p}_{ij}^{(3)} = 0, \quad i = 1, 2 \quad j = 1, 2, 3. \tag{56}$$

To find the solution of (52) means to solve the following matrix equation

$$\begin{pmatrix} \hat{G}_i^{(2)} & A_{i0}^t \\ -A_{i0} & 0 \end{pmatrix} \begin{pmatrix} \hat{q}_{i1}^{(3)} \\ \hat{q}_{i2}^{(3)} \end{pmatrix} = \begin{pmatrix} A_{i0}^t\,\hat{p}_{i0}^{(3)} \\ -\hat{p}_{i0}^{(3)} \end{pmatrix}. \tag{57}$$

The last equation (53) represents the system

$$\mathcal{R}_i^{(2)} \hat{s}_i^{(3)} = \begin{pmatrix} I_{m_{i0}} & & & \\ & R_{i1} & & \\ & & R_{i2} & \\ & & & R_{i3} \end{pmatrix} \begin{pmatrix} \hat{s}_{i0}^{(3)} \\ \hat{s}_{i1}^{(3)} \\ \hat{s}_{i2}^{(3)} \\ \hat{s}_{i3}^{(3)} \end{pmatrix} = \begin{pmatrix} A_{i0}\hat{q}_{i1}^{(3)} + \hat{q}_{i2}^{(3)} \\ T_{i1}\hat{q}_{i1}^{(3)} \\ T_{i2}\hat{q}_{i1}^{(3)} \\ T_{i3}\hat{q}_{i1}^{(3)} \end{pmatrix}, \tag{58}$$

from which we have

$$\hat{s}_{i0}^{(3)} = A_{i0}\,\hat{q}_{i1}^{(3)} + \hat{q}_{i2}^{(3)}, \quad i = 1, 2 \tag{59}$$

$$R_{ij}\,\hat{s}_{ij}^{(3)} = T_{ij}\,\hat{q}_{i1}^{(3)}, \quad i = 1, 2 \quad j = 1, 2, 3. \tag{60}$$

With the vector $\hat{s}_{ij}^{(3)}$ available, we have the result

$$s_i^{(3)} = \hat{p}_i^{(3)} - \hat{s}_i^{(3)} = \begin{pmatrix} \hat{p}_{i0}^{(3)} - \hat{s}_{i0}^{(3)} \\ -\hat{s}_{i1}^{(3)} \\ -\hat{s}_{i2}^{(3)} \\ -\hat{s}_{i3}^{(3)} \end{pmatrix}, \quad i = 1, 2. \tag{61}$$

Finally, the computing of $s_i^{(3)}$, $i = 1, 2$ consists of the following steps:

PROCEDURE Finddysparse($\mathcal{R}_i^{(2)}, A_{i0}, T_{i1}, T_{i2}, T_{i3}, T_i^{(3)} q_1^{(3)}, s_i^{(3)}$)

1. Set $\hat{p}_{i0}^{(3)} = T_{i0} q_1^{(3)}, \quad i = 1, 2$

2. Solve the system (57)
3. (a) Set $\hat{s}_{i0}^{(3)} = A_{i0}\hat{q}_{i1}^{(3)} + \hat{q}_{i2}^{(3)}, \quad i = 1,2$
   (b) Solve $R_{ij}\hat{s}_{ij}^{(3)} = T_{ij}\hat{q}_{i1}^{(3)}, \quad i = 1,2;\ j = 1,2,3$
4. Set $s_i^{(3)} = \hat{p}_i^{(3)} - \hat{s}_i^{(3)} \quad i = 1,2.$

Note that the Cholesky decomposition of the system matrices is available in step 2 and 3(b). These decomposition has been computed by the procedure Finddy.
In the end, the result of a three-stage stochastic model problem equals

$$dy^{(3)} = p^{(3)} - s^{(3)}.$$

This difference is obtained by the following computational process:
1. Call Finddy $(\mathcal{R}_i^{(2)},\ A_{i0},\ D_{i0}, T_{i1}, \ldots, T_{i3}, b_i^{(3)}, p_i^{(3)}), \quad i = 1,2,$
2. Call Updy$(\hat{G}^{(3)},\ A_0,\ \hat{v}_1^{(3)},\ \hat{v}_2^{(3)})$
3. Call Finddysparse$(\mathcal{R}_i^{(2)}, A_{i0}, T_{i1}, T_{i2}, T_{i3}, T_i^{(3)} q_1^{(3)}, s_i^{(3)})$ i=1,2
4. Form $dy^{(3)}$ as difference $p^{(3)} - s^{(3)}$.

The procedures in steps 1 and 3 are independent and can be computed at the same time. Step 2 represents a binding of existing two-stage models and enables to calculate $s_0^{(3)}$ and the parameter $q_i^{(3)}$, i=1,2 for Finddysparse(). Roughly, the process may be symbolically written as:

$$(62)\quad dy^{(3)} = \begin{pmatrix} b_0 \\ Finddy(\mathcal{R}_1^{(2)}, ..., p_1^{(3)}) \\ Finddy(\mathcal{R}_2^{(2)}, ..., p_2^{(3)}) \end{pmatrix} - \begin{pmatrix} A_0 q_1^{(3)} + q_2^{(3)} \\ Finddysparse(\mathcal{R}_1^{(2)}, ..., s_1^{(3)}) \\ Finddysparse(\mathcal{R}_2^{(2)}, ..., s_2^{(3)}) \end{pmatrix}.$$

4. Parallel implementation. The procedure

$$\text{Finddy}(\mathcal{R}^{(2)},\ A_0,\ D_0, T_1, \ldots, T_k, b, dy)$$

formulated for sequential calculation of the two-stage problem comprise three basic parts expressed in the equations (14)-(16). These equations constitute a method for computing of the vector dy. The vector dy consists in principle of k vectors that correspond to the block matrices $A_i, i = 1,2,...k$ (see 3) and the length of each is equal to the number of rows of the corresponding matrix A_i. The parallel computation can benefit from the fact that all procedures comprised in equations (14) and (16), can be performed independently for every block row of the matrix $\mathcal{R}^{(2)}$. The only communication is inevitable in the equation (15) by establishing of the matrix $\mathcal{G}^{(2)}$ (13), where the elements of the matrix $\hat{G}^{(2)}$ demand values that are dependent on the input matrices $A_i, i = 1,2,...k$.

Similar practise can be used in the computation of the the three-stage algorithm. The computational process according to the previous section can be expressed as

Step 1 Call Finddy $(\mathcal{R}_i^{(2)},\ A_{i0},\ D_{i0}, T_{i1}, \ldots, T_{i3}, b_i^{(3)}, p_i^{(3)}), \quad i = 1,2,...kk$
Step 2 Call Updy$(\hat{G}^{(3)},\ A_0,\ \hat{v}_1^{(3)},\ \hat{v}_2^{(3)}), ...\hat{v}_{kk}^{(3)})$
Step 3 Call Finddysparse$(\mathcal{R}_i^{(2)}, A_{i0}, T_{i1}, T_{i2}, T_{i3}, T_i^{(3)} q_1^{(3)}, s_i^{(3)})$ i=1,2,...kk
Step 4 Form $dy^{(3)}$ as difference $p^{(3)} - s^{(3)}$.

The calculations in the Step 1 and Step 3 are independent for all $i, i = 1,...kk$ and as a consequence all $Finddy()$ and $Finddysparse()$ procedures can be computed in parallel. As it was the case in the two-stage problem the only communication is present in the Step 2. As it is apparent from the Steps 1-4 the parallel algorithm of the three-stage stochastic model based on the two-stage procedure has two levels of parallelism. The outer level of parallelism reside in the parallel execution of the $Finddy()$ and $Finddysparse()$ procedures for all $i, i = 1, ...kk$. The inner level of parallelism can be found in the parallel execution of the procedure $Finddy()$ (and $Finddysparse()$, too) as it is described before. The independence of the most calculations comprised in both algorithms constitute a very natural assumption for a MPI-based programming framework. In all our experiments we have used the distributed memory programming model. The calculations was executed on block matrices in parallel and the communication between the processes was done by the Message Passing Interface (MPI). From

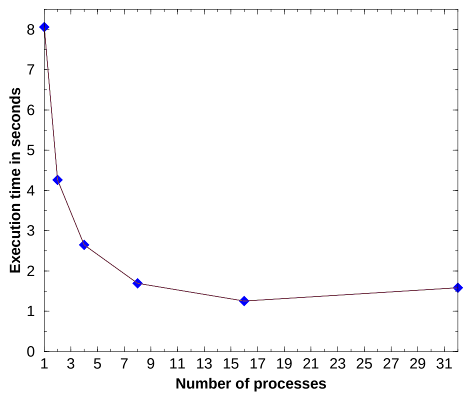

FIG. 1. *Timing results for the two-stage problem with k=99 block matrices A_i*

the numerical point of view the mathematical algorithms used in both described methods were based on the solving of systems of linear equations. A solving of a system of linear equations was realized by factorization of the matrix of the system by the Cholesky decomposition and solving of the linear system, whereby the triangular matrix arised from the Cholesky factorization was used. In our experiments we have used the the software linear algebra package LAPACK (Linear Algebra Package) [8] for all linear algebra calculations. The high efficiency of this software package is based on the use of a standard set of Basic Linear Algebra Subprograms (BLAS), which are optimized for each computing environment and are transportable and efficient across a wide range of computers.

k	NP=2	NP=4	NP=8	NP=16	NP=32
16	8.81	4.27	2.77	3.18	4.38
30	13.92	8.48	6.11	5.53	5.84
40	20.66	9.81	7.43	6.67	5.76

TABLE 1
Timing results for the three-stage problem composed of 16 two-stage problems. The k denote the number of block matrices in the two-stage problem. NP is the number of processors utilized. The size of a block matrix $A_i$is 20×40

All experiments presented in this paper were executed on the Beowulf-Cluster Gescher at the VCPC, University of Vienna. This consists of one front-end with two 400MHz Pentium II processors, a subcluster with eight compute nodes, each with two 400MHz Pentium II processors and a subcluster with sixteen compute nodes, each having four 700MHz Pentium III Xeon processors [9]. The Figure 1 show the dependence of the execution time on the number of processors, whereby every process was running on a different processor. In this example the size of all block matrices $A_i, i = 0, 1, ...k$ was 50×70 and the number of blocks $k = 99$. This experiment proved the high degree of parallelism and almost linear speedup for number of processors NP less than sixteen. For experiments with the three-stage algorithm three different values for $k, k = 16, 30, 40$ (the size of the two-stage problem) were compared.The size of matrices $A_i, i = 1, 2, ...k$ was 20×40 and was the same in all three cases. The value kk (the number of two-stage problems) in experiments illustrated in the Table 1 was equal to 16 (outer loop). As it was the case by the two-stage problem the best speed-up was achieved for smaller number of processes (processors). Decreasing speed-up with increasing number of processors was caused by increasing overhead and handling the I/O operations. Nevertheless an optimization of the code is possible and experimenting with different parallelization strategies could help to improve the performance.

5. Conclusion. The aim of our paper has been to use the BQ factorization technique for three-stage stochastic program in a framework of an interior point method. As we can see, this technique leads to the solution of independent subproblems. Moreover, these subproblems are again scalable into smaller linear system of equations. The whole process contains a serial coordination step, but the range of a sequential computation is not critical for large-scale stochastic program. Experimenting with the parallel code, testing the real problems and an extension to the multistage model will be topics of our future work.

REFERENCES

[1] P.Kall, S.W.Wallace: Stochastic Programming, John Wiley and Sons: New York 1994.
[2] H. Vladimirou, S.A. Zenios: Scalable parallel computations for large-scale stochastic programming, Annals of Oper. Res. 90,1999, pp.87-129.
[3] J.R. Birge, D.F. Holmes: Efficient solution of two-stage stochastic linear programs using interior point methods, Comput. Optim. Appl., 1, 1992, pp.245-276.
[4] J.R. Birge, L. Qi: Computing block-angular Karmarkar projections with applications to stochastic programming, Management Sci., 34, 1988, pp.1472-1479.
[5] E.R. Jessup, D. Yang, S.A. Zenios: Parallel factorization of structured matrices arising in stochastic programming, SIAM J.Optimization, Vol. 4., No.4,1994, pp. 833-846.
[6] G.H. Golub, Ch.F. Van Loan: Matrix Computations, The Johns Hopkins University Press, 1996.
[7] A. Ralston : A first course in numerical analysis, McGraw-Hill, inc., New York, 1965.
[8] Blackford L.S. and Dongarra J.: Lapack Working Note 81, Revised version 3.0, June 30, 1999
[9] http://gescher.vcpc.univie.ac.at/