APPLICATIONS OF THE INCOMPLETE CHOLESKY CONJUGATE GRADIENT METHOD IN BLOCK ADJUSTMENT

B.Benciolini*, L.Mussio*, L.Surace**
* Politecnico di Milano , Istituto di Topografia
**Istituto Geografico Militare Italiano (Firenze)
Italy
Commission III

1. The problem

The least squares adjustment of photogrammetric blocks implies the solution of large systems. Many algorithms are applied to this aim: at present the most commonly used algorithm is the Cholesky decomposition, followed by a forward-backward substitution. Another interesting method id the conjugate gradient algorithm, which, although very advantageous from the point of view of the storage-sparing, gives a bad proof as far as the computing time is concerned: this specially when a very accurate solution is required as it is customary in geodetic sciences.
An algorithm, combining an incomplete Cholesky decomposition with conjugate gradients has been proposed in recent years: this method has a velocity performance comparable to the Cholesky method, however with a much smaller storage waste. The method after the Meijerink and Van der Vorst has been called the Incomplete Cholesky-Conjugate Gradients method (ICCG).
The normal matrix C of the system to be solved by ICCG has to satisfy some hypotheses to fulfill theoretical requirements. These hypotheses can be summarized by saying that C must be an M-matrix (see Varga), i.e. C is a non singular matrix, with negative off-diagonal entries ($c_{ij} < 0$, $i \neq j$) and C^{-1} has only non negative entries ($C^{-1} \geq 0$).
The normal matrices of block adjustment do not fit the M-property: nevertheless, following Kershaw the ICCG method has a possibility of successful application, although one is not any more sure of convergence.
The least squares adjustment of photogrammetric blocks usually implies the solution of large linear systems, with very sparse metrices, i.e. with only a small percentage of non-zero entries. Particularly for the solution with an exact method, such as the packed Cholesky algorithm, but also with some iterative methods, is very useful to have a matrix with all non-zero elements as near as possible to the main diagonal. In fact, with such a configuration, the fill in of an exact method is minimized and the number of iterations of some iterative methods is strongly decreased.
The clustering of the elements of a sparse matrix round the main diagonal can be obtained with a reordering of the unknowns. Several reordering methods exist with different phylosophy (see Tewarson); one of the best is the reverse Cuthill-McKee algorithm as modified by Gibbs, Poole and Stockmeyer.

2. The method

Reordering algorithm

A paper by Gibbs, Poole and Stockmeyer which explaines shortly and clearly the aim and the strategies of the reordering algorithm is directly quoted in the following; the prerequisite to understand the sentences is "Sparse matrices" by Tewarson.
- Finding a starting vertex - The level structures of small width

are usually among those of maximal depth. Clearly, increasing the number of levels always decreases the average number of vertices in each level, and tends to reduce the width of the level structure as well. Ideally, then, one would like to generate level structures rooted at endpoints of a diameter. Since there is no known efficient procedure that always finds such vertices, a sub-optimal algorithm is employed to find the endpoints (u and v) of a pseudo-diameter, that is, a pair of vertices that are at nearly maximal distance apart.

- Minimizing level width - In the process of finding a pseudo-diameter, two level structures L_u and L_v rooted at the endpoints u and v, respectively, are constructed. It is possible to combine suitably these two level structures into a new level structure L whose width is usually less than that of either of the original ones.

- Numbering - The numbering procedure is similar to that of the reverse Cuthill-McKee algorithm in that it assigns consecutive positive integers to the vertices of the graph G level by level. A few modifications were necessary, however, since the level structures L obtained by minimizing level width are of a more general type than the rooted ones used in the reverse Cuthill-McKee algorithm. When the resulting numbering is similar to that obtained by the (forward) Cuthill-McKee algorithm, profile can be further reduced by using the reverse numbering.

Preconditioning algorithm

The incomplete Cholesky factorization W is obtained easily by modifying the usual relations of the exact Cholesky factorisation, as follows:

$$w_{ii} = \sqrt{c_{ii} - \sum_{k=1}^{i-1} w_{ki}^2}$$

$$w_{ij} = (c_{ij} - \sum_{k=1}^{i-1} w_{ki}\, w_{kj}) \,/\, w_{ii} \quad , \quad \text{if } c_{ij} \neq 0 \qquad (j > i)$$

$$w_{ij} = 0 \quad , \quad \text{if } c_{ij} = 0$$

If an argument of the square root is negative, it must be replaced with a small positive number. The incompleteness is due to the fact that the indices run only on the non-zero elements of the normal matrix C.

Conjugate gradient

Set, as starting point, where C and d are respectively the normal matrix and the known vector of the normal system:

$$r_1 = -(Cx_o + d)$$

$$p_1 = (W'W)^{-1} r_1$$

the algorithm runs with the following recurrent scheme:

$$\alpha_i = (r_i'(W'W)^{-1} r_i) \,/\, (p_i'\, C\, p_i)$$

$$x_i = x_{i-1} + \alpha_i\, p_i \qquad i = 1, 2, \ldots, \tilde{n}$$

$$r_{i+1} = r_i - \alpha_i C p_i \qquad i = 1, 2, \ldots, \tilde{n}$$

$$\beta_i = (r'_{i+1}(W'W)^{-1} r_{i+1}) / (r'_i(W'W)r_i)$$

$$p_{i+1} = (W'W)^{-1} r_{i+1} + \beta_i p_i$$

When the relative error between x_i and x_{i+1} is less than ε (small suitably chosen number), the algorithm stops and the solution is found:

$$x = x_{\tilde{n}}$$

At present no convenient method has been found to get in general an information on the elements of the inverse matrix C^{-1}. Nevertheless it is possible to say that the solution with the conjugate gradient of a system with a column of the identity matrix, as known vector, furnishes the correspondent column of the inverse matrix. Therefore, since the variances in a photogrammetric block are in general only slowly varying from place to place as it is seen by computing the variances with the exact Cholesky method, it is possible to give some information on these variances by solving only very few systems.

3. The examples

The running programs are written in the FORTRAN '77 language, using the multibanking technique for the collector; the variables are all in double precision. Some simulated examples are performed on the UNIVAC 1100/80 computer by CILEA; their results are listed in the enclosed table.

From the analysis of the table it is possible to make some remarks. The core storage requirement for the solution with the iterative ICCG method is always less than the one with the exact Cholesky method; particularly for the large systems the difference of storage requirement is significant.

The CPU execution time is not much higher for the relatively small systems, comparable for the large systems and much lower for the very large systems. Note that for the very large systems it is impossible to use the exact Cholesky method without mass storage, as the program of the Stuttgart University (1973), version PAT-M43.

The increase of the number of iterations (i) corresponding to the increase of the number of unknowns (n) is rather low; the relation between the two increases, following Kershaw, is $i = a \log n + b$. Note that the stopping test for the conjugate gradient must be fixed suitably in order to reach a sufficiently accurate solution, but avoiding an unessentially large number of iterations.

The application of the reordering algorithm and of the preconditioning algorithm, before the conjugate gradient method, is essential to obtain a good convergence; the same problem without reordering and preconditioning would require an enormous number of iterations. Since the fill in coefficient of the design matrix is always less than the one of the normal matrix, and the operations for the normalisation and the preconditioning are just the longest, an iterative method working directly on the design matrix could be useful. A new algorithm has been recently proposed by Paige and Saunders that seems to go in this direction.

Blocks	1	2	3	4	5	6	7	8	9	10
.strips	3	6	9	12	15	18	21	24	28	30
.models for strips	4	8	12	16	20	24	28	32	36	40
.points at the ground	30	80	154	252	374	520	690	884	1102	1344
.points in the models	108	432	972	1728	2700	3888	5292	6912	8748	10800
.horizontal control p.	4	8	10	14	18	18	22	24	28	32
.vertical control p.	4	9	12	20	30	30	42	48	63	80
.observations	324	1296	2916	5184	8100	11664	15876	20736	26244	32400
.unknowns	174	576	1218	2100	3222	4584	6186	8028	10110	12432
.constraints	12	25	32	48	66	66	86	96	119	144
.redundancy	162	745	1730	3132	4944	7146	9776	12804	16253	20112
design matrix:										
.non-zero elements	3240	12960	29160	51840	81000	116640	158760	207360	262440	324000
.fill in coefficient	1:17	1:57	1:121	1:210	1:322	1:458	1:618	1:802	1:1011	1:1243
normal matrix:										
.non-zero elements	2784	10896	24360	43176	67344	96864	131736	171960	218544	268464
.fill in coefficient	1:11	1:30	1:61	1:102	1:154	1:217	1:290	1:375	1:468	1:576
.minimum profile	8004	49536	153468	348600	663732	1127664	1769196	2617128	3700260	5047392
.fill in coefficient	1:3	1:6	1:9	1:12	1:15	1:18	1:21	1:24	1:27	1:30
.reordering algor. (CPU)	0".4	1".8	5".0	11".1						
.exact Cholesky	1".9	17".1								
.precondit. algorithm	4".4	20".0	53".0	1'29".0						
.conjugate gradient	2".9	12".1	30".0	1'6".0						
.number of iterations	14	16	18	20	(21)*	(21)	(22)	(23)	(23)	(24)
.PAT-M43** (CPU+I/O)	1'.31".6	1'.51".7	3'.14".6	5'.22".9						

* () indicates interpolated values
** program of the Stuttgart University (1973)

No investigation has yet been possible, but the importance of the argument suggests to regard with care this method.

4. The application to the on-line triangulation

The on-line triangulation is well becoming one of the most important field of application of analytical plotters and, generally, of the computer-assisted photogrammetric instruments.
As it is well known, an aerial triangulation is performed on-line when the instrument used for the data acquisition is interfaced with a computer capable to process those data in real time or near-real time in comparison with the acquisition time.
The application of the above described ICCG method to the on-line triangulation requires to emphasize that an on-line triangulation can be divided in three phases:

- data acquisition,
- error detection,
- final adjustment.

Two different approaches to on-line triangulation are possible with reference to the two last points.
First, the software implemented on a computer-assisted photogrammetric instrument must primarily be able to check the quality of the acquired data and to detect the middle gross errors and the blunders. The on-line computations have only a temporary character and do not necessary achieve a rigorous solution, since an off-line final and usually more sophisticated adjustment is performed afterwards.
Second, an on-line adjustment is executed to give final results available immediately after the last measurement together with gross error detection and elimination.
The former way is more likely to be used in practice and is also one of the best kinds of applications of the ICCG method.
Indeed the phase of gross error detection and elimination is surely more expensive than that of final adjustment. Consequently, the improvement of the efficiency of an on-line triangulation procedure, to reduce the CPU execution time in this phase, is just possible with a core storage sparing iterative solution, like ICCG, applicable also to a very small computer interfaced with photogrammetric instruments. Moreover, since the numeration of the points is established by assigning suitably chosen numbers to the observed points, the blunders or the gross errors that more frequently occur are those of incorrect attribution and/or identification of the point number. Now, the advantage of real time gross error detection and elimination as well as that of the evaluation of the intrinsic accuracy of the various sub-blocks, useful to decide eventual reobservations, is achieved with an on-line procedure. Note that in this way by the ICCG method it is not needed to repeat the application of the preconditioning algorithm.

Aknowledgments
The authors wish to thank Mr. Forlani, for the precious aid given in the programs implementation and in performing the numerical examples.

References

Benciolini B., Betti B., Mussio L.: "A remark on the application of the GPS procedure to very irregular graph"; Proceedings of the Meeting of FIG-Study Group 5B Survey control networks; Schriftenreihe HSBw Heft 7, Munich,'82.

Benciolini B., Mussio L.: "Test on reordering algorithm for geodetic network and photogrammetric block adjustment"; Proceedings of the Int. Symp. Management of geodetic data, Geodätisches Institut, Meddelelse n. 55, Copenhagen, 1981.

Benciolini B., Mussio L., Sansò F.: "The incomplete Cholesky conjugate gradient method in network adjustment"; Proceedings of the 6th Int. Symp. on Geodetic Network and Computations, DGK Reihe B Heft 258, 1981.

Gibbs N.E., Poole W.G., Stockmeyer P.K.: "An algorithm for reducing the bandwidth and profile of a sparse matrix"; SIAM Journal of numerical analysis vol. 13, n.2, April 1976.

Kershaw D.S.: "The incomplete Cholesky conjugate gradient method for the iterative solution of system of linear equations"; Journal of computational physics, n.26, Jan.'78.

Meijerink J.A., Van der Vorst H.A.: "An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix"; Mathematics of computation, vol. 31, n. 137, January 1977.

Paige C.C., Saunders M.A.: "LSQR: An algorithm for sparse linear equations and sparse least squares"; ACM Transactions on Mathematical Software, vol. 8, n. 1, March 1982.

Paige C.C., Saunders M.A.: "LSQR: Sparse linear equations and least squares problems"; ACM Transactions on mathematical software, vol. 8, n. 2, June 1982.

Tewarson R.P.: "Sparse matrices"; Academic Press, New York, 1973.

Varga M.S.: "Matrix iterative analysis"; Prentice Hall, Englewood Cliffs (New Jersey), 1962.