TEST OF ALGORITHMS FOR SEQUENTIAL ADJUSTMENT IN CLOSE RANGE PHOTOGRAMMETRY

Knut Ragnar Holm
RUNIT - The Computing Centre at
the University of Trondheim

ABSTRACT

Photogrammetric dimension control in industrial production requires short turnaround times. An aid for meeting these requirements is on-line triangulation (OLT), in this context meaning block adjustment with blunder detection being carried out along with the photo measurement. Combined with digital cameras, the OLT system will approach the "real-time" situation, and thus even meet the requirements of control tasks with strict time limits.

A search for the most effective algorithm for sequential adjustment is underway. Some algorithms are proposed, and some estimates have been made on the performance, mainly in terms of algebraic operations counts. However, in an actual implementation of OLT, the performance of a specific algorithm also depends on the software structure as well as the data structure used. Although the data structure should meet the algorithm's performance needs, the software structure should reflect the operational procedures of the system, including the specific requirements in close range applications.

Consequently the performance of such algorithms can best be judged with tests in an OLT-like software environment. The paper discusses important aspects and objectives of such a test, and gives results from a comparative test of two specific algorithms.

INTRODUCTION

At the University of Trondheim there has for some years been going on research on photogrammetry for industrial applications (Stenberg, Øfsti,1976; Hådem,1981,1984), and lately particularly for production control on large steel structures, during the FOMAKON project (Holm, Østbye,1982). Procedures and software have been improved, and the advantages of photogrammetric measurements have been recognised. Nevertheless, the method has not yet been adopted for regular use at shipyards, who were the main potential users.

The main reason for this reluctance is the turnaround time. With film developement, photo measurement, computation with blunder removal, and production of final results, turnaround time would normally be some days. Practical applications often require turnaround measured in hours, or even minutes.

Some kind of on-line photogrammetric triangulation (OLT) would bring us a great deal closer to fulfilling such time requirements, especially when performed on digital images supplied by digital cameras. During developement of OLT, the need for quick algorithms for sequential least squares adjustment has become clear.

ON-LINE PHOTOTOGRAMMETRIC TRIANGULATION

The term On Line Triangulation (OLT) has been used in various contexts. With analog stereoplotters it involved the direct editing of data during measuring, making it easy to transfer data to the block adjustment program in a proper format, and with the most obvious errors removed.

When the analytical stereoinstruments arrived OLT was further developed, e.g. with model transfer procedures. As a computer is an integral part of the instrument, it was natural to include more advanced error checking. This error checking has more and more become the important part of OLT, as the potential has become clearer.

To perform more reliable blunder detection, it was desireable to involve more than the traditional two photos of a stereomodel in the same computed unit. Both triplets and quadruplets has been investigated for use in aerotriangulation, and lately sub-blocks of 3x3 photos have been suggested as a working unit for optimal reliability. For close range photogrammetry with more scattered camera positions one should rather look directly at the number and angles of intersecting rays to each object point, to decide an optimal block size.

When choosing sub-block size, the following aspects of the response time also have to be considered:

The OLT system must not cause unnecessary delay compared to a normal measuring procedure. This is an objective demand. By calculation there may be found acceptable waiting times in different stages of the work.

The operator should not get the feeling of waiting for response. This is a subjective demand, but may become the dominant one. If the operator is repeatedly forced to wait a few seconds it will affect his patience and concentration and so the quality of the work.

The conclusion is that whatever the OLT system computes between two consecutive point readings, it should be completed as the operator is ready to press the button for the next point. At certain stages some delay would be acceptable, e.g. for error checking after a number of point readings, or when changing photos.

With the available hardware today it is beyond any expectation to handle an entire aerial block in one piece on-line. The scope for OLT in aerial triangulation is only to perform data collection and blunder detection on-line, in order to provide error-free data and good approximate values for the final off-line adjustment.

There may be some differences when dealing with close range photogrammetry. In addition to the important blunder detection aspect the need for final results without too much delay makes it desireable to treat the entire block in one piece. With today's techniques this could perhaps be possible with the smaller blocks appearing in some applications (e.g. 3-5 photos). The ability to handle larger blocks in an on-line system for close-range photogrammetry will depend on the successful development of adjustment algorithms, as well as the efficiency and price of computer hardware.

SEQUENTIAL ADJUSTMENT ALGORITHMS AND PERFORMANCE

With simultaneous adjustment the time for computing a new solution basicly increases by the third power of the number of parameters. Consequently the response time requirements would soon be violated, even for "block" adjustment of two or three photos. Sequential updating of the inverted or the decomposed normal equation matrix followed by back substitution will increase only by the second power of the number of parameters.

Even with good sequential adjustment algorithms the response time will soon be too long when the number of photos and object points increases. Therefore it will always be interesting to find "the best" algorithm, - in order to be able to operate with the desired block sizes on-line.

A number of algorithms for sequential adjustment have been suggested. Some of them are in use or have been tested. Others have merely been mathematicly described.

Mikhail, Helmering, 1973 gives algorithms for updating of the inverse. This approach has been adopted in several OLT systems, and is known as "Kalman-form".

Grün, 1982 describes an algorithm called Triangular Factor Update (TFU), which updates the decomposed normals, based on Gauss decomposition. Taking care of the sparse matrix structure is an integral part of the algorithm.

The procedures of TFU are described in more detail in Wyatt, 1982, which is related to an actual implementation of the algorithm. Furthermore, this reference describes a test where a Kalman-form algorithm is compared to the TFU algorithm. A block adjustment is carried out with 9 photos, with the addition of various numbers of new observations at different stages. The program also includes blunder detection according to a method shown in Grün,1982, followed by deletion of observations pointed out. The conclusion is that the TFU algorithm is superior to the Kalman-form algorithm both concerning time consumption and storage requirement.

The use of Givens transformations is suggested in Blais, 1983. He also shows how the inverse may be updated along with the Cholesky factor of the normal equations, and indicates that matrix sparsity may be exploited. However, he does not go into details.

Inkilä, 1984 describes and compares methods for updating Cholesky factorization. The comparison is made theoretically by operation counts, and shows basicly no difference between Agee-Turner updating and orthogonal transformations, e.g. Givens transformations.

An extensive review of the development of OLT is given in Grün,1984, including different methods for sequential adjustment.

Encouraged by these and other publications this author decided to implement and test sequential adjustment based on Givens transformations (GT). One of the appealing features of GT was its true sequential nature, with row by row updating, while for instance TFU seemed to work best with groups of observations.

GIVENS TRANSFORMATIONS IN SEQUENTIAL ADJUSTMENT

The normal adjustment problem may be written

$$v = A\,x - l \qquad (1)$$

where A is the (m,n) design matrix, l is the observation vector, and v is the vector of residuals. In the sequential case we want to add a new observation equation to the original problem

$$v_a = a\,x - b \qquad (2)$$

so that the complete problem is

$$\begin{bmatrix} v \\ v_a \end{bmatrix} = \begin{bmatrix} A \\ a \end{bmatrix} x - \begin{bmatrix} l \\ b \end{bmatrix} \qquad (3)$$

The solutions of (1) and (3) respectively are given by:

$$A^T A\,x = A^T\,l \qquad (4)$$

$$(A^T A + a^T a)\,x = A^T\,l + a^T\,b \qquad (5)$$

It is shown (in, for instance, Inkilä,1984) that, having reduced (4) to

$$R\,x = t \tag{6}$$

by Cholesky factorization, the updated Cholesky reduction

$$\bar{R}\,x = \bar{t} \tag{7}$$

of (5) (after adding (2)) may be obtained by an orthogonal transformation, expressed as

$$Q \begin{bmatrix} R & t \\ a & b \end{bmatrix} = \begin{bmatrix} \bar{R} & \bar{t} \\ 0 & e \end{bmatrix} \tag{8}$$

By further augmenting the matrix (8), the value $s = \sqrt{\bar{v}^T \bar{v}}$ will also be maintained (Gentleman,1973):

$$Q \begin{bmatrix} R & t \\ 0 & s \\ a & b \end{bmatrix} = \begin{bmatrix} \bar{R} & \bar{t} \\ 0 & \bar{s} \\ 0 & 0 \end{bmatrix} \tag{9}$$

Concerning weights, the row (a,b) must be multiplied by the square root of its weight. For correlated observations, the correlated observation equations should be multiplied by the Cholesky factor of the weight matrix.

There are different methods for constructing the orthogonal matrices. In practical use, the orthogonal matrices themselves need not be constructed as such. Each Q is normally a product of several orthogonal matrices with rather simple structure, and the effect of the individual multiplications may be computed directly into the elements of the matrix operated upon.

One of the methods is to apply Givens transformations to pairs of rows in (8). Q is then constructed as a product of n matrices Q_i (i=1,...,n), each representing one Givens transformation. Each Q_i will affect two rows of the matrix it is applied to, being of the form

$$Q_i = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & c & 0 & s \\ 0 & 0 & I & 0 \\ 0 & -s & 0 & c \end{bmatrix} \tag{10}$$

Calling the two rows

$$\begin{aligned} r &= (r_1, r_2, \ldots, r_j, \ldots) \\ a &= (a_1, a_2, \ldots, a_j, \ldots) \end{aligned} \tag{11}$$

the elements are transformed by

$$\begin{aligned} \bar{r}_j &= c\, r_j + s\, a_j \\ \bar{a}_j &= c\, a_j - s\, r_j \end{aligned} \tag{12}$$

Thus, the desired operation (9) may be executed by sequentially applying Givens transformations to the last row (a,b) and the i'th row of R. For each pair of rows, c and s are constructed so that $\bar{a}_i$ becomes zero. This is achieved when

$$\begin{aligned} c &= r_i \,/\, d \\ s &= a_i \,/\, d \end{aligned} \tag{13}$$

where

$$d = \sqrt{r_i^2 + a_i^2} \qquad (14)$$

To remove observations, Blais,1983 recommends to apply negative weights and use the same formulae. After elaborating (12),(13), and (14) with the complex numbers caused by the square root of the negative weight, it turns out that the problem may be solved with only real numbers in slightly modified formulae.

A COMPARATIVE TEST OF GT AND TFU

Aspects of Testing

Different adjustment methods have been evaluated with respect to the computing time, either by counting the number of algebraic operations necessary or by running test programs. The theoretical operation count may give a good indication of performance. However, there are other factors which will influence the final performance in a specific application.

Naturally, the data structure should fit the algorithm, to make it as effective as possible. The sparsity pattern of the normals should be exploited if possible, both with respect to storage and execution time. However, it should be considered whether the sparsity pattern is substantially different in different applications.

On the other hand, the procedural structure of the software should reflect the operational procedures of the OLT system. This includes, for instance, the number of observations per update and how often blunder detection is wanted. Whether object control is to be included or not, and in which form it is included, will also have considerable impact on the program structure.

The conclusion is that a specific algorithm could fit nicely into one application, but not perform so well in another. Consequently, performance tests of algorithms should preferably be run in a software environment similar to the actual application software.

The Actual Test

To test the performance of the GT algorithm it was decided to compare it with the TFU algorithm, since the latter had already been found useful in an earlier test.

The application in mind was close range photogrammetry, including multistation photography and monocomparator measurements. Hence the test should have been performed in a system for these applications.

However, the special close range requirements were put aside for a while. This was in order to determine, as early as possible, whether the GT algorithm would perform as well as expected. At that stage a copy of the program written and used by Wyatt for testing TFU (Wyatt,1982), was obtained. This program simulates aerial block triangulation.

A new program was written with the same operational specifications, but based on the GT algorithm for sequential adjustment. Most of the data structures were identical, changes being introduced only where necessary for the correct operation of the GT algorithm.

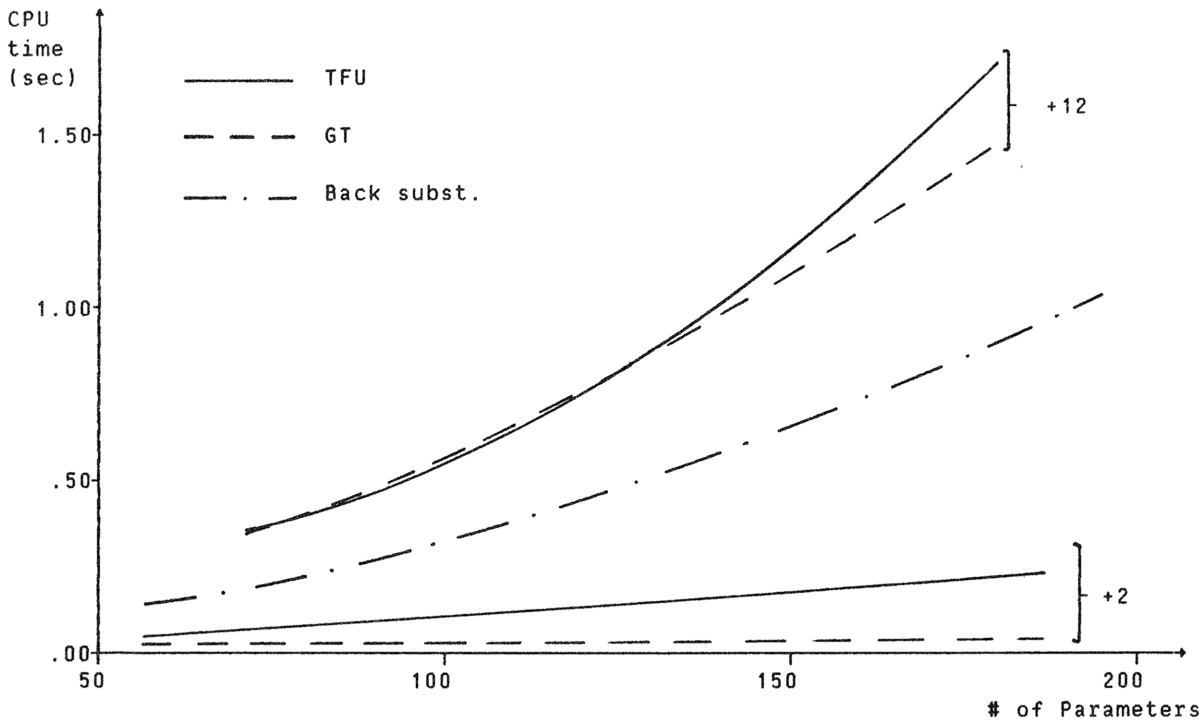
The test now consists of a simulated aerotriangulation of a block of 3x3 photos in an analytical stereoinstrument. Various numbers of new point observations are added at different stages of the triangulation. Single observations are removed in connection with blunder detection.

The programs were run on a VAX 11/750 at the Departement of Civil Engineering at the Norwegian Institute of Technology, Trondheim. CPU time was measured for different phases of the computations, such as normals formation and factorization, update of normals, and parameter solution.

Results

Since this test concerns update algorithms, only the CPU-times for updating the factorized normals are reported here.

The diagram summarises the main trends of the results. It shows how the update time increases with the number of unknown parameters in the normal equations. Times for TFU and GT are shown for two different amounts of additional observations (+2 and +12). Just to get an idea of the magnitude in relation to other parts of the computation, time for back substitution is shown as well. Note that one "observation" here means one observed photo point, implying two observation equations (for x and y).



CPU time versus number of parameters

Note that the CPU values for TFU are not the directly measured ones. An improvement of the TFU program suggested in Wyatt,1982 was not implemented in the program used in the test, while an equivalent improvement was incorporated in the GT program. Thus the measured CPU times were not quite compatible. However, it was possible to isolate the time for a program sequence which would be involved by the improvement in the TFU program, and a part of this time was subtracted, based on judgement of the expected effect of the improvement.

DISCUSSION AND CONCLUSIONS

When drawing conclusions from the present data, one should keep in mind the limited accuracy of the values for TFU in the diagram. It should also be noted that the absolute values of the times are not representative for an operating OLT system, since the computer of an OLT system is not expected to be of the same size or speed as the one used in the test.

The diagram indicates that, with the data set used, the GT algorithm works faster than the TFU algorithm when adding few new observations. It seems to be confirmed that TFU operates more effectively for a larger group of additional equations, while GT has its strength for few new observations. When measuring on single photos, which is often the case in close range photogrammetry, this feature of GT will be advantageous.

GT will not compete with TFU if the updates are to be performed with larger groups of observations. But the time for the smaller updates with GT are expected to allow the update to take place between point readings. Thus only the back substitution is required whenever the parameters are to be computed.

Further tests are required to find out how the performance of the algorithms vary with other parameters than shown here, e.g number of photos involved in an update and their position in the normals. This might confirm or reduce the indicated advantage of GT.

The GT algorithm may be further improved. The Givens transformations without square roots presented in Gentleman,1973 would speed up the process. There are also potentials for efficiency improvements in the current program. As indicated earlier, that is the case for TFU as well.

As a final conclusion the test has shown that an algorithm based on Givens transformations is at least competitive, and may be especially suited for near-real-time systems for close range photogrammetry.

ACKNOWLEDGEMENTS

The author wish to express his gratitude to those having contributed to the present work. Special thanks to professor Armin Grün at ETH, Zürich, for making the TFU program available for the test, and to Norges Tekniske Høgskoles Fond for financial support.

REFERENCES

Blais J. A. R., 1983: Linear least-squares computations using Givens transformations. The Canadian Surveyor 37(4):225-233.

Gentleman W. M., 1973: Least squares computations by Givens transformations without square roots. Journal of the Institute of Mathematical Applications 12:329-336.

Grün A., 1982: An optimum algorithm for on-line triangulation. Proceedings of ISPRS Comm. III Symposium, Helsinki.

Grün A., 1984: Algorithmic aspects in on-line triangulation. Proceedings of ISPRS Congress, Rio de Janeiro, Comm. III.

Holm K. R., Østbye B., 1982: The FOMAKON project, photogrammetry on marine structures. Proceedings of ISPRS Comm. V Symposium, York.

Hådem I., 1981: Bundle adjustment in industrial photogrammetry. Photogrammetria 37:45-60.

Hådem I., 1984: Generalized relative orientation in close-range photogrammetry - a survey of methods. Proceedings of ISPRS Congress, Rio de Janeiro, Comm. V.

Inkilä K., 1984: Recursive least squares estimation by updating the Cholesky factorization. Proceedings of ISPRS Congress, Rio de Janeiro, Comm. III.

Mikhail E. M., Helmering R. J., 1973: Recursive methods in photogrammetric data reduction. Photogrammetric Engineering 39(9):983-989.

Stenberg A., Øfsti O., 1976: Photogrammetric determination of distribution and thickness of cavitation on ship propellers. Proceedings of ISP Congress, Helsinki, Comm. V.

Wyatt A. H., 1982: On-line photogrammetric triangulation - an algorithmic approach. Master Thesis, Department of Geodetic Science and Surveying, The Ohio State University.