PARVEC Workshop on Very Large Least Squares Problems and Supercomputers

John R. Rice
Purdue University, jrr@cs.purdue.edu

Report Number:
83-464
PARVEC Workshop on
Very Large Least Squares Problems
and Supercomputers

John R. Rice
CSD-TR 464
December 15, 1983

ABSTRACT

This is a summary of an intensive workshop held at Purdue University on November 3-4, 1983. Eleven people of different backgrounds (numerical analysis, supercomputers, photogrammetry, geodetics and molecular structures) met to exchange ideas and to evaluate present and future developments. This report records the results as follows:

1. Summary and Principal Conclusions
2. Workshop Participants
3. Very Large Least Squares Problems
4. Methods and Matrix Structures
5. Discussion: Issues and Responses
6. References
7. Participant's Statements of Background and Interests

Support from the Office of Naval Research and the Army Research Office made this workshop possible and is gratefully acknowledged.
1. SUMMARY AND PRINCIPAL CONCLUSIONS

The goals of the workshop were: (a) to provide an interchange between different groups working on very large least squares problems, (b) to provide an interchange between computer scientists involved with supercomputer systems and scientists using supercomputers and (c) to assess the state of the art in solving very large least squares problems. This workshop is one of a series held by the Purdue Center for Parallel and Vector Computing and supported by the Army Research Office, the National Science Foundation and the Office of Naval Research.

The number of participants was kept small so as to allow for discussions in depth and complete expressions of views. The principal participants were:

James Bethel (photogrammetry) Purdue University
Iquacio Fita (molecular structures) Purdue University
Dennis Gannon (supercomputers) Purdue University
Gene Golub (numerical linear algebra) Stanford University
Wayne Hendrickson (molecular structures) Naval Research Lab
Greg Kramer (applications programmer) Purdue University
Charles Lawson (numerical linear algebra) Jet Propulsion Lab
Robert Plemmons (matrix computation, geodesy) North Carolina State
John Rice (supercomputers) Purdue University
Michael Rossman (molecular structures) Purdue University
Ahmed Sameh (supercomputers) University of Illinois

Twelve sources of very large least squares problems were identified (see Section 3), the ones principally involved in the discussions were:

- Geodetic surveys
- Photogrammetry
- Molecular structures
- Gravity field of the earth
- Partial differential equations

A brief review of the current methods used in these problems is given in Section 4.

The final day was devoted to a discussion of 17 "issue" questions, plus three additional observations (see Section 5 for fuller discussions). These issues covered the problems, the methods, the use of supercomputers and the future. The principal conclusions reached are listed below. Note that this report summarizes lengthy discussions and no participant is likely to agree in detail with all statements made here.

A. Problems. There are several important least squares problems that require supercomputer power. There is substantial similarity in the structure of the problems from different areas; the
matrices possess a block structure (sometimes at two levels) which reflects a "local connection" nature in the underlying physical problem.

B. Methods. Most of the standard methods are being used somewhere. There is a definite need for a comprehensive software package for least squares that includes sparse matrix facilities.

C. Computations. Programming effort is more often a bottleneck than computer time, but neither are likely to be dominant, (the most common dominant effort is to get the data). The preprocessing, postprocessing and general inefficiency in using masses of data on several devices is an important bottleneck. Current pipeline machines and attached processors sacrifice portability and clarity to high efficiency.

D. FUTURE. The most promising area for algorithm improvement is in the handling of sparsity. There are important least squares problems that require much more resources (including computer power) than are currently available. Supercomputer architectures are not going to stabilize, so it is important that high level, somewhat architecture independent languages (even a clean Fortran extension) can be used by scientists. Reasonable profitability is essential for many reasons, including the success of "national resource" supercomputer centers. A very critical need is to make supercomputers easier to use.

2. WORKSHOP PARTICIPANTS

There were eleven principal participants in this workshop.

In addition, there were three graduate students present at times and Edward Mikhail, Professor of Photogrammetry at Purdue University, had to withdraw at the last minute.

James Bethel, Purdue University - (Civil Engineering)

Iquacio Fita, Purdue University - (Biological Sciences)
Postdoc - assisting M. Rossmann in molecular structures studies

Dennis Gannon, Purdue University - (Computer Science)
Assistant Professor - research in areas of supercomputer design, systems and scientific computation. Particularly interested in PDE computations using various computer architectures.

Gene Golub, Stanford University - (Computer Science)
Professor - research in numerical linear algebra, especially iterative methods, least squares and statistical computation.

Wayne Hendrickson, Naval Research Lab - (Biophysics)
Research Biophysicist - research in molecular bio-
physics, biological macromolecules and crystallographic analysis. Extensive experience in developing methods and software for determining molecular structures on the TI-ASC. This work involves determining models via large nonlinear least squares computations.

Greg Kramer, Purdue University - (Biological Sciences)
Application programmer - assisting M. Rossman in transferring the molecular structures programs to the Cyber 205.

Charles Lawson, Jet Propulsion Laboratory - (Numerical Analysis)
Section Chief - research in approximation theory, least squares computations, computational geometry and celestial mechanics. Extensive experience in large scale scientific computing involving least squares, especially gravitational fields and orbit determinations.

Robert Plemmons, North Carolina State - (Computer Science)
Professor - research in matrix computations, especially for least squares adjustments in geodesy, structural analysis and Markov processes.

John Rice, Purdue University - (Computer Science)
Professor - research in approximation theory, mathematical software and large scale scientific computation. Current research is in solving partial differential equations on vector and parallel computers. Past research has involved various aspects of least squares.

Michael Rossmann, Purdue University - (Biological Sciences)
Distinguished Professor - research on crystallography and the determination of the structure of proteins and viruses. Extensive experience in the use of least squares computation as part of the analysis of large biological molecules.

Ahmed Sameh, University of Illinois - (Computer Science)
Professor - research in numerical linear algebra and computation on multiprocessor computers. Extensive experience in the analysis of sparse matrix computations where interprocessor communication and synchronization are critical. Current research is in sparse matrix computations on multiprocessors.

3. VERY LARGE LEAST SQUARES PROBLEMS

Most of the first day of the workshop was spent in general discussions of various least squares problems and the methods to solve them. Most of the problems discussed are briefly described in this section. References are given for more information about most of the problems; there were a few of them about which very little was known first hand.

A. The Geodetic Survey Problem. The existing geographical survey
points are not completely consistent because of errors in the measurements. A classical procedure has been to adjust the measurements to obtain a best least squares fit to the non-linear relationships that must hold. The National Geodetic Survey (NGS) currently has a program under way to adjust the measurements for the entire North American continent. This computation will involve about 540 thousand variables and 6.5 million relationships. The adjustment of the geodetic measurements for the entire earth is planned for the future. See [Golub and Plemmons, 1980, Kolata, 1978, Plemmons, 1979] for details of this problem. Its main features are:

(i) A natural multilevel block structure. Data and computations are usually organized by, say, counties, then by states, then by countries.

(ii) Highly variable accuracy in data. Some survey data is over 50 years old and much less accurate than recent data.

(iii) Very good approximate solutions available for iteration on the nonlinearities.

(iv) Data is expensive to obtain; the least squares computation costs are a moderate part of the whole process.

B. The Photogrammetry Problem. When one takes a series of aerial photographs, one neither knows the locations on the photographs nor the locations of the cameras. One identifies some points on the photographs whose ground locations are known precisely (these are often marked on the ground so they show up clearly in the photos). Overlapping photos are taken showing these points several times; this information is combined with knowledge of the camera properties to create a model of the camera locations. The parameters are then determined as a least squares solution of this nonlinear model. The camera parameters and the ground point parameters are obtained in a simultaneous solution. In large systems a block elimination scheme is often employed so that a reduced system of only camera parameters is solved first, followed by a "back solution" for the others. The total number of unknown parameters (6 per camera, 3 per ground point) can number in the hundreds for a modest system, and in the thousands for a large system.

Similar computations occur in the precise measurement of the position and shape of large structures such as radio telescopes. The main features of this problem are:

(i) There is a natural block structure within the least squares normal equations' coefficient matrix. Each photograph and each ground point contribute such a block. Non-zero off-diagonal terms are limited to a band which arises from the "local connection" nature (similar to the geodetic problem) of the photographs and ground point.

(ii) Raw data is collected in two places: (a) obtaining the photographs, and (b) measuring the point locations.
on the photographs. Obtaining the data is an expensive process, however individual point measurements may be repeated or added relatively inexpensively. The least squares computation is the other major step in producing the final results.

C. The Molecular Structures Problem. The linear least squares problem is in the inner loop of a complex process depicted by the following (simplified) steps.

1. **Collect Data**
   - Grow single crystals of a pure macromolecular substance of sufficient size
   - Obtain x-ray diffraction patterns
   - Preprocess pictures and use symmetry to enhance data quality

2. **Determine approximate molecular structure**
   - Uses various chemical and physical procedures plus considerable analysis of data

3. **Create nonlinear least squares problem**
   - Include basic covalent process of atomic interactions
   - Include 10 to 15 types of "restraints" which incorporate various chemical and molecular facts.

4. **Iterate**
   - Linearize problem by Newton's method
   - Discard "most" terms in the Jacobian matrix $J$
   - Solve $J^t \Delta x = b$ for the Newton step as a linear least squares problem

5. **Reconstruct Computed Molecule**
   - The numbers are postprocessed to produce a visual model

6. **Evaluate Computed Molecule**
   - The computed models and observed electron densities are compared visually. If they agree to within the uncertainty of the nonlinear least squares, the model is accepted
   - Otherwise:
     - identify water and other solvent structures
     - reorient certain submolecules
     - add or delete atoms or submolecules
     - add restraints on the structure
     - go back to step 3

There are three positional variables for each atom in the molecule; a simple molecule has a few hundred atoms, a complex one (e.g.) a small virus has a few million. Current work involves molecules with several tens of thousands of atoms. This least squares problem is thus embedded in a large, complex 10-30 minutes on a Class VI computer and the nonlinear iteration requires solving many of these. Even so, this is not necessarily the dominant part of the computation. There may be a stack several feet high of X-ray films with thousands of information spots on
each; runs to preprocess this data can require over many hours or a day even on a Class VI computer. See [Hendrickson and Konnert, 1980] for more details on the overall problem and [Blumde11 and Johnson, 1976] for more details on the mathematical model and least squares problems.

This problem may be interpreted as 2 1/2 dimensional, the molecule is like a long sausage that winds around itself. Most of the terms in the model refer to local relationships (positions or angles) along the molecule. With an appropriate numbering of the atoms, these relationships produces a "local connection" nature in the least squares problem. All of the important local connection terms in $J$ are near the main diagonal and the others are negligible. However, where the sausage folds over itself, there are non-local (in the numbering system) effects. The folding is not random, one of the major unsolved problems in molecular structures is how and why these giant molecules fold. These non-local terms are in the "restraints" and are a crucial part of determining the structure; they produce "randomly" scattered small blocks away from the main diagonal of $J$.

The main features of this problem are:

(i) There are several large scale computational steps involved

(ii) The least squares problem has a local-connection structure modified by a relatively small number of other terms.

(iii) The blocks in the matrix $J$ are small (3 by 3 to 20 by 20 or so).

(iv) There is only a very rough initial approximation for the nonlinear problem, it probably has terms missing (at least in the beginning) so the least squares residuals are not "small".

(v) The outer loop involves someone visually comparing electron density maps with the current model, usually using a computer graphics system. This is the most time-consuming aspect of the work.

D. Gravity Field of the Earth. There is a standard model of gravity using spherical harmonics which is derived from viewing the earth as a homogeneous ellipsoidal planet. As more accuracy is desired, one adds more terms to compensate for the non-homogeneous mass distribution and the actual shape of the earth. NASA has a mission GRM (Geopotential Research Mission) to collect a massive amount of near earth data to be used to determine many thousands of terms in the expansion in spherical harmonics. This will be a standard least squares fitting problem, it is not a sparse matrix problem because the spherical harmonics do not have any "local support" behavior.

An alternative (not part of NASA's plan) is to use a piecewise polynomial representation of the gravity field. The idea behind this is that the detailed effects of irregular shapes and masses are not well modeled by spherical harmonics and one is going to obtain the usual slow convergence properties of polynomial and trigonometric approximations will lead to a least squares problem with quite regular sparsity structure. For more details on this problem see [NASA, 1982], [Moore et al, 1982]
The main features of this problem are:

(i) A massive amount of data, very expensive to collect.
(ii) Very regular and uniform structure in the data, the problem and the underlying models.
(iii) The classical model leads to a full matrix least squares problem.
(iv) The possible piecewise polynomial model leads to a sparse least squares problem with a very regular structure. The blocks in the matrix are relatively small, about 10 by 10.

E. The Least Squares Method for Partial Differential Equations (PDEs). There is a classical least squares (finite element) method for solving PDEs that is rarely used in practice. It is closely related to their widely used methods (e.g. collocation and Galerkin) and the probable reason for its "neglect" is that people feel that it offers no apparent advantage over the more standard methods. See [Rice, 1983] for an elementary discussion of this method.

This method would be used primarily with piecewise polynomial basis functions which would lead to least squares problem with a regular block sparsity structure, similar to that which appears in the more standard PDE methods. The number of unknowns can easily reach 1 million for three dimensional PDEs, there would be a small number (1 to 10) of equations per unknown.

The principal features of this problem are:

(i) There is very little data, the equations are generated mathematically.
(ii) There is a very regular block sparsity structure to the problem. The blocks are small to moderate in size (4 by 4 to 50 by 50).
(iii) The number of equations can be very large, there are applications where one solves a large sequence of very similar problems.

F. Tomography. This is a specialized application where one reconstructs an object by taking X-ray cross sections. It is similar to data filling in that one has a fixed number of data from a continuum; it differs from data filling in that one observes various linear functionals (e.g. integrals) from the continuum rather than actual values. See [Herman, 1976, 1978 and 1980] for more information.

The principal characteristics of these problems are:

(1) The systems of equalities and inequalities are huge, order about a million.
(2) The sparsity is somewhat haphazard, less than 1 percent of the matrix elements are non-zero.
(3) The principal computational tool is the row-action method, see [Censor, 1981].

G. Force Method in Structural Analysis. There are two principal methods of matrix structural analysis, the displacement (or stiffness) method and the force (or flexibility) method. The force method has certain advantages for multiple redesign
problems or nonlinear elastic analysis because it allows the solution of modified problems, by least squares computations, without restarting the total computation from the beginning. This can result in significant savings for large scale problems. See [Kaneko, Lawo and Thierauf, 1982] and [Kaneko and Plemmons, 1984] for details.

The main features of this problem are:

(i) The force method consists of two stages. Stage 1 involves the computation of a basis matrix $B$ for the null space of the equilibrium matrix $E$ for the structure and state 2 involves the solution of a certain least squares problem with $B$ serving as the observation matrix. $B$ can be dense even though $E$ is sparse, depending upon the method for computing $B$.

(ii) There is very little data. The elements of $E$ are generated mathematically and $B$ is computed from $E$.

(iii) Engineering substructuring methods can lead to a block angular form for the least squares matrix $B$, similar in form to those of the observation matrices in the Geodetic and Photogrammetry problems.

H. Very Long Base Line Problem. The object is to measure astronomical distances by using interferometer methods with base lines that are thousands of miles (using geographically separated radio telescopes) or millions of miles (using observations taken at different points on the earth's orbit around the sun). There are enormous quantities of data that are relatively inaccurate.

I. Surface Fitting. One has a physical surface where many positions are known. The surface is modeled by piecewise polynomials of modest degree (1 to 3) joined with some smoothness (continuity, perhaps less, perhaps one or two continuous derivatives). The model has parameters which are determined by a least squares fit to the observed data.

The size of these problems commonly varies from rather small, say a few dozen parameters, up to fairly large, perhaps a thousand parameters. One can, of course, visualize almost arbitrarily large problems, especially if one goes to three dimension problems. The matrices involved have the block structure expected from a "local basis" model of the surface.

The principal characteristics of these problems are:

(i) Usually modest to moderate in size, that is 50 to 1000 unknowns and 2 to 5 observations per unknown.

(ii) A fairly regular block structure in the matrices with modest sized blocks, say 4 by 4 to 16 by 16.

J. Cluster Analysis and Pattern Matching. Some pattern recognition algorithms are essentially least squares problems (usually nonlinear). One usually has a modest sample of values and a very flexible model with a relatively small number of parameters; a few hundred values and 5 to 50 parameters are common. As we become more adept at these problems, we can expect the size of the problem to grow very substantially.
The principal characteristics of these problems are:

(i) Modest to moderate in size, but potentially quite large
(ii) Considerable variation in structure as widely different models may be used. Many models probably give full matrices.

4. METHODS AND MATRIX STRUCTURES

The linear least squares problem is formulated mathematically with an $n \times m$ matrix $A=(a_{ij})$, unknowns $x_i$, $i=1$ to $m$ and data $b_j$, $j=1$ to $n$. One wishes to solve $Ax=b$, but $n \neq m$ so this system is generally inconsistent. Thus one determines the least squares solution $x$ so that

$$\sum_{j=1}^{m} \left( y_j \mathbf{x}_j - b_j \right)^2 = \|Ax-b\|^2 = \text{minimum}$$

In the discussion that follows, we assume that $n \leq m$ and $m$ is large.

A. The Normal Equations. A simple analysis shows that the least squares solution $x$ satisfies the linear system

$$A^T A x = A^T b$$

which is an $m \times m$ system with a symmetric and (normally) positive definite coefficient matrix $A^T A$. The total work for this solution method is, including forming $A^T A$, $m^2/2 + m^3/3$ multiplications.

The main advantage of this approach is simplicity, the disadvantages are (i) the computation might be less stable numerically and (ii) any sparsity structure in $A$ is usually destroyed.

B. The Residual Equations. Let $r_j = a_{ij}x_i - b_j$ be the residual of the $j$-th equation. Then a simple analysis shows that $x$ and $r$ solve the system.

$$\begin{bmatrix} I & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} r \\ x \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}$$

This system is larger than the normal equations, $(n+m)$ by $(n+m)$, but retains the sparsity of $A$. This system is indefinite.

C. Orthogonalization. One may apply an orthogonal matrix $Q$ to $Ax=b$ to obtain

$$QAx=Qb$$

and determine $Q$ so that $QA=R$ is "upper triangular". That is

$$R = \begin{bmatrix} T \\ 0 \end{bmatrix}$$

where $T$ is square and upper triangular. One then solves $Tx=b'$ where $b'$ is the first $n$ elements of $Qb$. The elementary reflections or elementary rotation matrices are usually recommended to
The total work for this solution method is $m n^2/6$. The main advantages of this method are numerical stability and the potential of using any sparsity that $A$ might have. The disadvantage is that it is twice as much work as the normal equations (assuming that $m$ dominates $n$, as it usually does).

D. Iteration. Splitting and Conjugate Gradient Method. Since the normal equations are symmetric and positive definite, most standard iteration methods are applicable. The convergence of such methods can often be accelerated by splitting the linear system $Ax = b$ written in the form $Mx = d$, where $C = M - N$ and $N$ is a good approximation to $C$. A particularly effective iteration is the conjugate gradient method where

$$x^{(k+1)} = x^{(k)} + \frac{d - Cx^{(k)}}{\nabla^2 x^{(k)} - Cx^{(k)}}$$

is solved for $x^{(k+1)}$. The parameters $\omega_k$ and $\beta_k$ are determined by separate computations. See [Concus, Golub, and O'Leary, 1976] for further details.

5. DISCUSSION: ISSUES AND RESPONSES

A set of issues was prepared before the workshop and most of the second day was spent in discussing them. The more significant discussions are already summarized in Section 4. The issues as originally presented are listed along with a summary of the responses.

PROBLEMS

A. What are the very large least squares problems that arise in scientific areas? Section II of this report contains the response to this question.
B. Do the problems from different areas have similarities?

There is a surprising amount of similarity. The matrix \( A \) can almost always be put in the following form (sometimes called the dual block angular form):

\[
\begin{pmatrix}
A_1 & B_1 \\
A_2 & B_2 \\
A_3 & B_3 \\
\vdots & \vdots \\
A_k & B_k \\
D
\end{pmatrix}
\]

This reflects a "local connection" structure in the underlying physical problem (the spherical harmonics expansion of gravity is one exception). There is a wide variation in the number and size of the blocks. Some problems have large blocks with \( k \) modest in size (10-100) while others have much smaller blocks but many more of them. A number of the problems have two levels of sparsity structure. That is, the blocks \( A_i \) and/or \( B_i \) are themselves large sparse matrices, usually with this same general pattern of sparsity. There might be some difference in the sparsity patterns between the two levels. The molecular structures problem has this structure with a relatively small number of other blocks scattered through the matrix.

C. What is the scientific significance of these problems?

Some of these problems are integral parts of large national scientific programs (e.g. geodetic survey, very long base line and gravity model). Others are ubiquitous in some important areas (e.g. PDE computations, digital terrain modeling, structural analysis, photogrammetry). Still others are integral parts of the developing frontiers of significant scientific research programs (e.g. molecular structures, tomography, pattern analysis).

D. Are there very large least squares problems of potential interest that have not yet been seriously attempted?

Three problems were mentioned: PDE computations, geological structure (the analog of the gravity problem, but below the surface of the earth) and cell biology (the natural long range extension of the molecular structures problem).

E. What methods are currently being used for these problems?

Section IV of this report contains the response to this question.

F. What methods are thought to be the most suitable for these problems?

There is no clear winner yet. The exploitation of sparsity is not yet thoroughly explored; different patterns of sparsity give the advantage to different methods. The
normal equations and conjugate gradient (applied to the residual equations) are the most widely used. A drawback of the conjugate gradient method is the difficulty in simultaneously obtaining variance and covariance information.

The impact of vector computers will be substantial but, again, no clear pattern has yet emerged. These calculations deal primarily with very long, very sparse vectors. Substructuring is naturally applicable to these problems for the multiprocessor computers. Again, the algorithmic questions are mostly open.

G. Is it practical to use the same methods - or same algorithms - or same software - in different applications areas?

There are definite similarities in the problems from different application areas; this implies that similar methods are applicable. There is not enough generally used software to give as real experience in applying the same software in different applications areas. However, limited experience plus informed conjecture suggests that some software can be used widely. Well designed software could be modified or parameterized to give good efficiency in a variety of applications.

H. How much exchange of know-how is there between scientists in different application areas? between numerical analysts or computer scientists and scientists?

There is some exchange of know-how, but it is not systematic nor uniform. The amount of isolation among groups interested in essentially the same problems seems to be typical of science in general.

**COMPUTATIONS**

I. Is the vectorization of the linear algebra the major step in adapting methods to current supercomputers?

There is definitely much more to be done than to vectorize the linear algebra (although this must be done also). The principal task is to reorganize the algorithms so as to exploit the natural sparsity in the problems and yet also exploit the vector processing power of the supercomputers. Experiences were reported where it was as difficult to overcome "non-numerical" bottlenecks (like I/O or page thrashing) as to make the arithmetic run fast. The opinion was expressed that obtaining efficient, well organized software is a bigger hurdle than devising vector algorithms or reorganizing algorithms to be vectorizable.

Current supercomputers were strongly criticized for inadequate Fortran support. To obtain good performance on Cyber 205 or Cray 1 requires a lot of detailed idiosyncratic changes in the codes which renders them totally useless for any other computer. The view was expressed that many people do not want to invest years in codes that cannot be used by their colleagues and which become useless once a new machine is acquired.

J. Is least squares computation the major part of the total computations?

The least squares computation is almost always in the "inner loop" of the computations and thus a significant
computational expense. However, it is rarely the dominant part of the computation. Input/output, data processing, preprocessing and postprocessing are also significant computations and some applications also involve significant numerical computations of other types (e.g. nonlinear systems of equations).

K. What is the nature of the difficulty in getting the data for very large least squares computations?
   There are a couple of areas (PDEs and quantum mechanics) where obtaining the data is a minor part of the problem. For most applications, this is a major part of the problem and for some (e.g. geodetic survey, molecular structures determination, and gravity field analysis) the cost of obtaining the data completely dominates the computational (and programming) costs.

L. How does programming supercomputers for very large least squares computations compare with programming ordinary machines?
   A high level of general dissatisfaction was expressed for programming the current Class VI machines. They were described as "a pain"; the resulting software is totally non-transportable and generally obscure. The attached array processors are no better. This is not inherently the nature of supercomputers; one participant had considerable experience with the TI-ASC machine and felt it was much more "usable" than his current experience with the Cyber 205.

M. Is computer time a major bottleneck in getting results for these problems?
   Yes, but it is not dominant in most cases. The preprocessing and postprocessing of results tends to require a lot of human attention and involve delays of various kinds (e.g. getting files from one machine to another, getting output plotted, making tapes, etc.). These activities slow down the whole process much more than the few hours that one is waiting for the "scientific computations" to be done. One sometimes has to wait many hours (or even days) to obtain adequate amounts of computer time.

N. Is programming effort a major bottleneck in getting results for these problems?
   Yes. It is sometimes more of a bottleneck than computer time, but still usually not the dominant factor. There is often considerable difficulty in finding people who have the desired knowledge of supercomputers, programming and the application area.

THE FUTURE

O. What are the prospects for being able to solve the very large least squares problems at the frontiers of science? Do we need much faster computers - or much faster algorithms - or both?
   The prospects are good. Both faster computers and faster algorithms are needed; neither one obviously dominates the others. It is just as important to have better user interfaces, better languages and supporting tools as it is to have faster computers and algorithms.
P. Is it more important to make the computer faster or easier to use?  
The question is misleading; the critical task is to make the very fast computers easy to use.

Q. What would be the scientific impact of much greater computational powers in these areas?  
It would do a lot of good (no specific list of impact areas was generated). Perhaps the greatest impact would come from the ability to do conceptually straightforward things better. A great deal of effort is now required to solve a lot of problems that have little technical difficulty or novelty; this is taking away from the time available for problems that require a lot of thought.

R. What are the prospects of discovering significantly better supercomputer algorithms for least squares?  
They seem good for two reasons. First, one can see that it is possible to devise better ways of handling sparsity, data and memory space. Second, history tells us that it is unwise to believe that better methods will not appear.

WRAP-UP OBSERVATIONS

S. There is a strong need for a flexible package (or several packages) of sparse least squares routines.

T. Supercomputer hardware is not going to stabilize. People cannot rewrite and tailor massive codes for each new architecture (never mind variances on a theme) that appears. Thus scientists must keep programs expressed at high levels and processors for these languages must be developed for each new architecture.

U. The concept of a set of "national resource" supercomputer access sites is not viable without reasonable transportability of working programs among the super computers.

6. REFERENCES


Buehner, M., Ford, G.C., Moras, D., Olsen, K.W., and Rossman, M.C.,


Lawson, C.L., Solving large dense least squares problems using two levels of storage, JPL internal report CM 366-466, (1979).

Lawson, C.L. and Hanson, R.J., Solving least squares problems, Prentice-Hall, (1974).


7. PARTICIPANTS OF BACKGROUNDS AND INTEREST

Some workshop participants prepared a statement of their background and interests relevant to this workshop. These presented are here in considerably edited form for:

Wayne Henderickson    Dennis Gannon
Charles Lawson         Gene Golub
Edward Mikhail         Robert Plemmons
John Rice              Michael Rossmann
Ahmed Sameh

DENNIS GANNON

1977 Ph.D., University of California Berkeley (Mathematics)
1980 Ph.D., University of Illinois (Computer Science)
1980- Computer Science faculty, Purdue University

Considerable work has been done on the design and analysis of PDE methods for multiprocessors machines. The emphasis has been on multigrid methods. Presently directing the construction of a 64 processor machine (the Pringle, a prototype of the Blue Chip machine) and the design and construction of the I/O system for the machine. Also involved in the design, analysis and creation of language processors and math software for the Pringle.

GENE H. GOLUB
Experience with Least Squares Problems

Research over many years has heavily involved the theory and practice of least squares computations. Numerous publications involving statistical computation as well as numerical linear algebra.

WAYNE A. HENDRICKSON

1968 Ph.D., Johns Hopkins University (Biophysics)
1968-69 Post-doc, Johns Hopkins University
1969-71 Post-doc, Naval Research Laboratory
1971- Research Biophysicist, Naval Research Laboratory

Research Specialties:
- Molecular Biophysics
- Structure and Function of Biological macromolecules
- Methods for crystallographic analysis of structures.

Experience with Least Squares Problems

Least-squares problems are commonly encountered in crystallography. Perhaps the most demanding of these occurs in the refinement of the parameters of an atomic model against the observed diffraction data from crystals. This is a non-linear problem fraught with false minima and, in the case of biological macromolecules, the many thousands of parameters make these refinements large least squares problems.

We have developed a procedure for the refinement of protein structures subject to certain stereochemical conditions. Only a sparse matrix of second derivations is stored and minimization of the linearized least-squares problem in each cycle is done by a conjugate gradients procedure. The escape from false minima requires manual intervention through the use of interactive computer graphics.

Experience with Supercomputers

At NRL, we presently use the Texas Instruments Advanced Scientific Computer (TI-ASC) for much of our refinement work. This is a fast vectorizing machine. Our refinement program, PROLSQ, has been optimized for vector processing through standard FORTRAN and the resulting code is quite transportable to other vectorizing machines (e.g. Cray-1) as well as to non-vectorizing machines (e.g. VAX).

CHARLES L. LAWSON

1961 Ph.D., UCLA (Mathematics)
1960- Research scientist and supervisor, Jet Propulsion Laboratory

Research Specialties
- Function Approximation, Curve and Surface Fitting
- Least Squares Algorithms
- Mathematical Software

Experience with Least Squares Problems
I am presently working with JPL scientists on approaches for improving the accuracy of the NASA model of the earth's gravity field through use of data from a proposed earth satellite project, GRM. This appears to require a least squares solution for about 30,000 to 40,000 coefficients. The VLBI (Very Long Baseline Interferometer) research at JPL involves the solution of least square problems with 1000 unknowns.

In the 1969-1972 period, R.J. Hanson and I developed FORTRAN programs for univariate least squares spline curve fitting with equality and inequality constraints, and for bivariate cubic spline least squares surface fitting. The latter problem can lead to least squares problems with 500 to 2000 unknowns.

During 1977-1979, J.E. Ekelund and I developed a set of Fortran programs using two levels of storage to solve large dense least squares problems using Givens rotations. This package includes optional computation of the covariance matrix and a singular value analysis.

Numerous reports and papers involve least squares and linear algebra algorithms as well as the book with R.J. Hanson: Solving Least Squares Problems, Prentice-Hall, 1974.

EDWARD M. MIKHAIL

1963 Ph.D., Cornell (Photogrammetry ad Geodesy)
1963-65 Photogrammetrist in industry
1965- Civil Engineering faculty, Purdue University

Research Specialties

Photogrammetry
Data Adjustment
Metric Remote Sensing and Image Processing

Experience with Least Squares Problems

I have not worked directly on developing methods and algorithms for least squares solutions, but I work in general areas where least squares are used heavily. These include:

1. Photogrammetry, where one has the simultaneous adjustment of a large block of photographs.
2. Image processing, our main computational problem is the two-dimensional correlation of conjugate images.
3. Remote sensing, the problem is not of least squares itself, but of handling the information in large images in preparation for a least squares computation.
4. Digital terrain modeling, least squares are used to approximate elevation data using bilinear or bicubic spline functions.

Numerous publications involve least squares including the two books Observations and Least Squares and Analysis and Adjustment of Survey Measurements.

MICHAEL G. ROSSMAN

1956 Ph.D., University of Glasgow (Crystallography)
1956-58 Post-doc, University of Minnesota
1958-64 Research scientist, MRC Laboratory of Molecular Biology
1964- Biological sciences faculty, Purdue University

Research Interests

- Molecular structure determination of proteins and viruses
- Analysis of protein structures
- Crystallography

Experience with Least Squares Problems

I was one of the first to apply general least-squares methods to the refinement of molecular structures based on x-ray diffraction data. This was on the UNIVAC 1103 in the mid 1950's. This approach has been used throughout my research on the structure of large biological molecules. I currently carry out nonlinear least squares computations involving tens and hundreds of thousands of variables as part of the project to determine the structure of certain viruses.

ROBERT J. PLEMMONS

1965  Ph.D., Auburn University (Mathematics)
1966-67 Analyst, National Security Agency
1968-81 Mathematics and Computer Science faculty, University of Tennessee
1981- Mathematics and Computer Science Faculty, N C State University

Research Specialties

- Numerical Analysis, including
  least squares adjustments in geodesy
  matrix structural analysis
  queueing network analysis

Experience with Least Squares Problems

About a dozen papers have been written on the general topic of least squares computations, both of iterative and direct methods. In the area of large-scale least squares adjustment methods for geodetic data, four papers have been written. The paper "Large-scale least squares adjustment by dissection and orthogonal decomposition" deals especially with algorithms adaptable to parallel and vector processing. Current work here involves a study of direct-iterative block SOR type methods.

Experience with Supercomputers

Experience here involves the use of the CYBER 205 installation at Colorado State University, accessed through our Triangle Universities Computing Center (TUCC). Additional experience has been gained on a configuration of FPS-164 attached array processors at TUCC. Recent efforts have been made in working with representations from Duke University and the University of North Carolina to obtain a supercomputer in the Research Triangle area.

JOHN R. RICE

1959  Ph. D., California Institute of Technology (Mathematics)
1959-60 Post-doc, National Bureau of Standards
1960-64  Research Scientist, General Motors Research Laboratories
1964-    Math and Computer Science faculty, Purdue University

Research Interests

Approximation Theory, Curve and Surface Fitting
Mathematical Software
Numerical Solution of Partial Differential Equations
Parallel Computation

Experience with Least Squares Problems

Publications include a number of papers involving least squares, primarily for curve fitting applications. Three publications include several algorithms, but none of them involved exceptionally large data sets.

Experience with Supercomputers

Several algorithms for quadrature and approximation have been analyzed on idealized MIMD machines. Work closely related to real computer includes the development and analysis of quadrature algorithms for the IUIAC IV and studies of the implementation of elliptic PDE algorithms on the CYBER 205. At an intermediate level is work on the automatic distribution of PDE computations onto parallel machines. Several reports and studies involve the design of languages for scientific computation on supercomputers.

AHMED H. SAMEH

1968    Ph.D., University of Illinois (Structural Engineering)
1968-74  IUIAC IV Project, University of Illinois
1974-    Computer Science faculty, University of Illinois

Research Interests

Numerical Linear Algebra
Algorithms for multi-processor computers

Experience with Least Squares Problems

I have developed a number of algorithms for large sparse linear algebra computations designed for multiprocessor systems such as the Illiac IV and the Cedar machine. Some of these algorithms are directly applicable as parts of solving least square problems. I have also developed and analyzed least squares algorithms for problems with a dense or regular block structure nature. Again, these algorithms are intended for multiprocessor machines. These results appear in numerous papers and reports.

Experience with Supercomputers

I have used the Illiac IV for which our group at Urbana developed numerous algorithms. I have also been involved in developing numerical algorithms for the Burroughs Scientific Processor and the Cray-XMP.