Linear Algebra Kernel Functionality

The BLAST Forum *

February 4, 1997

http://www.netlib.org/utk/papers/blast-forum.html

Abstract

This document lists the perceived functionality of numerical linear algebra library kernels as discussed at the BLAS Technical Forum in Atlanta, May 20 - 21, 1996. Included are kernels needed for dense matrices, sparse direct methods, sparse iterative methods as well as related kernels needed in non-traditional linear algebra computations such as CFD and finite elements.

The focus is on sequential and shared memory computers, but functionality for the parallel case has been borne in mind as well.

This version of the document gives a brief introduction and then presents tables listing the functionality. Further information on the BLAS technical Forum can be found at the World Wide Web address. Comments on this document and active participation in the Forum are encouraged.

*This document so far edited by Jack Dongarra, Sven Hammarling, Antoine Petitet and Barry Smith and Linda Kaufman

1 Introduction

A technical forum has recently been established to consider expanding the Basic Linear Algebra Subprograms (BLAS) in a number of directions in the light of modern software, language, and hardware developments. The first two meetings of the BLAS Technical Forum were held in Nashville on February 19-20, 1996 and in Atlanta on May 20-21, 1996.

Working groups have been established to consider the overall functionality, a lightweight interface, and an object based interface, for the BLAS. Other subgroups have also been established, either to advise the current working groups or as placeholders for future working groups on parallel processing issues, sparse operations, data structures, extensions to the existing BLAS and language binding issues.

This document summarizes, in tabular form, the discussions of the Functionality Working Group. Note that this document does not consider issues such as storage formats or data types.

The list is fairly exhaustive and it is not assumed that everything will necessarily be included in the final report. At the very least we expect to provide a priority guide. The list includes the functionality covered by the existing Level 1, 2 and 3 BLAS [4, 1, 3, 2].

This document is expected to develop over time, and some of the topics to be discussed include:

- Motivation
- Functionality
- Justification
  - Algorithm Use
  - Efficiency Issues. E.g. re-use of data
  - Ease of expression
- Comparison with Existing BLAS
  - What's New
- Important Kernels

– (historical considerations and changes)

In this version of the document, the functionality is categorized more or less along traditional BLAS lines, but the Forum discussed other schemes and it may be appropriate to categorize the functionality in more than one way in the final document.

If what you need is not listed here, please comment and if possible join the Forum, to let us know what you need.

References

[1] D. S. Dodson, R. G. Grimes, and J. G. Lewis. Sparse extensions to the FORTRAN Basic Linear Algebra Subprograms. *ACM Trans. Math. Software*, 17:253–272, 1991. (Algorithm 692).

[2] J. J. Dongarra, J. Du Croz, I. S. Duff, and S. Hammarling. A set of Level 3 Basic Linear Algebra Subprograms. *ACM Trans. Math. Software*, 16:1–28, 1990. (Algorithm 679).

[3] J. J. Dongarra, J. Du Croz, S. Hammarling, and R. J. Hanson. An extended set of FORTRAN Basic Linear Algebra Subprograms. *ACM Trans. Math. Software*, 14:1–32, 1988. (Algorithm 656).

[4] C. L. Lawson, R. J. Hanson, D. Kincaid, and F. T. Krogh. Basic Linear Algebra Subprograms for FORTRAN usage. *ACM Trans. Math. Software*, 5:308–323, 1979. (Algorithm 539).

A Notation

- x_i - an element of a one-dimensional array
- x_I - subset of entries of x indexed by I
- $y[i]$ - an array from an array of arrays
- x_{C} - compressed (sparse) array, only non-zeros are stored, associated with an index set I
- e - vector whose elements are all unity

B Scalar and Vector Operations

This Appendix gives the operations corresponding to the traditional Level 1 BLAS operations. Table 1 lists the scalar and vector reduction operations, Table 2 lists the vector operations and Table 3 lists the sparse vector operations. The operations are intended to apply to both real and complex arguments. For the sake of compactness the complex operators are omitted, so that whenever a transpose operation is given the conjugate transpose should also be assumed for the complex case. The last column of this document has a priority listing. A "0" indicates that it is already part of the Blas. The indication 'k1' indicates top priority by Linda Kaufman.

Inner product	$r \leftarrow x^T y,$	0
	$r \leftarrow r \pm x^T y$	k3
	$r \leftarrow \alpha x^T y,$	k3
	$r \leftarrow r + \alpha x^T y$	k3
Vector norms	$r \leftarrow \|\|x\|\|_1,$	0
	$r \leftarrow \|\|x\|\|_\infty$	k1
	$r \leftarrow \|\|x\|\|_2,$	0
	$r \leftarrow \|\|x\|\|_p$	k3
Sum	$r \leftarrow \sum_i x_i$	k1
Max value & location	$i, x_i, ; i = \arg\max_i x_i$	k1
Max absolute value & location	$i, \|x_i\|, i = \arg\max_i \|x_i\|$	k1
Multiple inner products	$r_i \leftarrow x^T y[i]$	k2
Sum of squares	$(a, b) \leftarrow \sum x_i^2,$	k2
	$a \cdot b^2 = \sum x^2$	k3
Generate Givens rotation	$(c, s) \leftarrow \mathrm{rot}(a, b)$	0
Generate muliple Givens rotation	$(c[i], s[i]) \leftarrow \mathrm{rot}(a[i], b[i])$	k2
Generate Modified Givens rotation	$(z, r) \leftarrow \mathrm{mrot}(a, b)$	0
Generate Jacobi rotation	$(c, s) \leftarrow \mathrm{jrot}(a, b)$	k2
Generate Householder transform	$(\alpha, u) \leftarrow \mathrm{house}(x),$	
	$H = I - \alpha u u^T$	k1
Sort into order & return index vector	$(iv, y) \leftarrow \mathrm{sort}(x)$	k1
Sort according to index vector	$y \leftarrow \mathrm{sort}(x, iv)$	k1

Table 1: Reduction Operations

Zero	$x \leftarrow 0$	k1
Constant	$x \leftarrow \alpha e$	k1
Copy	$y \leftarrow x$	0
Conjugate copy	$y \leftarrow \bar{x}$	k2
Swap	$y \leftrightarrow x$	0
Scale	$x \leftarrow \alpha x,$	0
	$y \leftarrow \alpha x$	k2
Group scale	$x_{I_i} \leftarrow a_i x_{I_i}$	k1
Sum	$y \leftarrow \pm x + y,$	k1
	$w \leftarrow \pm x + y$	k2
Shift	$x \leftarrow x + \sigma e,$	k2
	$y \leftarrow x + \sigma e$	k2
AXPY & AXPY like	$y \leftarrow \alpha x + y,$	0
	$w \leftarrow \alpha x + y$	k2
	$y \leftarrow x + \beta y,$	k2
	$w \leftarrow x + \beta y$	k2
	$y \leftarrow \alpha x + \beta y,$	k2
	$w \leftarrow \alpha x + \beta y$	k1
	$y \leftarrow \alpha x + \beta y + \sigma e,$	k2
	$w \leftarrow \alpha x + \beta y + \sigma e$	k2
Absolute value	$x_i \leftarrow \|x_i\|$	k1
	$y_i \leftarrow \|x_i\|$	k2
Elementwise multiply	$x_i \leftarrow x_i y_i,$	k1
	$w_i \leftarrow x_i y_i$	k2
Elementwise divide	$x_i \leftarrow x_i / y_i,$	k1
	$w_i \leftarrow x_i / y_i$	k2
Multiple AXPYs	$y[i] \leftarrow a_i x + y[i]$	k1
Apply plane rotation	$\begin{pmatrix} x & y \end{pmatrix} \leftarrow \begin{pmatrix} x & y \end{pmatrix} R$	0
Apply muliple plane rotation	$\begin{pmatrix} x[i] & y[i] \end{pmatrix} \leftarrow \begin{pmatrix} x[i] & y[i] \end{pmatrix} R[i]$	k2
Apply modified plane rotation	$\begin{pmatrix} \tilde{x} & \tilde{y} \end{pmatrix} D \leftarrow \begin{pmatrix} \tilde{x} & \tilde{y} \end{pmatrix} DR$	0

Table 2: Vector Operations

Scatter	$y_I \leftarrow x_{\mathrm{C}}$	k1
Gather	$x_{\mathrm{C}} \leftarrow y_I$	k1
Scatter add	$y_I \leftarrow \pm x_{\mathrm{C}} + y_I$	k2
Gather add	$x_{\mathrm{C}} \leftarrow x_{\mathrm{C}} \pm y_I$	k2
AXPY	$y_I \leftarrow \alpha x_{\mathrm{C}} + y_I$	k1
(other AXPY and AXPY like operations also?)		
Inner product	$r \leftarrow x_{\mathrm{C}}\, y_I^T$	k1
Apply plane rotation	$\begin{pmatrix} x_{\mathrm{C}} & y_I \end{pmatrix} \leftarrow \begin{pmatrix} x_{\mathrm{C}} & y_I \end{pmatrix} R$	k2

Table 3: Sparse Vector Operations

C Matrix Vector Operations

This Appendix gives the operations corresponding to the traditional Level 2 BLAS operations. Table 4 lists the matrix-vector operations. The operations are intended to apply to both real and complex arguments, and the matrix arguments A, T, L and U are intended to be dense or banded or sparse. In addition, where appropriate, the matrix A can be symmetric (Hermitian) or triangular or general. The matrix T specifically represents a triangular matrix, either upper or lower, L represents a lower triangular matrix and U an upper triangular matrix, all three of which can be unit or non-unit triangular.

D represents at least a diagonal matrix, and maybe also a 2×2 blocked diagonal or a tridiagonal matrix. Q represents an orthogonal (unitary) matrix and it is an open question as to how Q should be stored.

For the sake of compactness the complex operators are omitted, so that whenever a transpose operation is given both the conjugate and conjugate transpose should also be assumed for the complex case.

Matrix vector product	$x \leftarrow Ax,\ x \leftarrow A^T x$	k2
	$y \leftarrow \pm Ax,\ y \leftarrow \pm A^T x$	k1
	$y \leftarrow \pm Ax + y,\ y \leftarrow \pm A^T x + y$	k2
	$y \leftarrow \alpha Ax + \beta y,\ y \leftarrow \alpha A^T x + \beta y$	0
Diagonal extract	$d \leftarrow \mathrm{diag}(A)$	k2
Triangular solve	$x \leftarrow T^{-1}x,\ x \leftarrow T^{-T}x$	0
	$x \leftarrow \alpha T^{-1}x,\ x \leftarrow \alpha T^{-T}x$	k2
	$y \leftarrow T^{-1}x,\ y \leftarrow T^{-T}x$	k2
	$y \leftarrow \alpha T^{-1}x,\ y \leftarrow \alpha T^{-T}x$	k2
Double triangular solve,	$x \leftarrow U^{-1}L^{-1}x,\ y \leftarrow U^{-1}L^{-1}x$	k3
– including permutations	$x \leftarrow U^{-1}D^{-1}L^{-1}x,\ y \leftarrow U^{-1}D^{-1}L^{-1}x$	k3
(Also do these with inverse	$x \leftarrow L^{-1}U^{-1}x,\ y \leftarrow L^{-1}U^{-1}x$	k3
transpose, and with TT^T	$x \leftarrow L^{-1}D^{-1}U^{-1}x,\ y \leftarrow L^{-1}D^{-1}U^{-1}x$	k3
and TDT^T when SPD)		
Rank one and symmetric ($A = A^T$)	$A \leftarrow \beta xy^T + \alpha A$	k2
Rank one and symmetric ($A = A^T$)	$A \leftarrow \beta xy^T + A$	0
rank one & two updates	$A \leftarrow \beta xx^T + A$	k1
rank one & two updates	$A \leftarrow \beta xx^T + \alpha A,\ A \leftarrow \beta xDx^T + \alpha A$	k2
($\alpha = 1$ and $\beta = \pm 1$	$A \leftarrow (\beta x)y^T + y(\beta x)^T + \alpha A$	k3
as special cases?)	$A \leftarrow (\beta xD)y^T + y(\beta xD)^T + \alpha A$	k3
Application of Q from	$x \leftarrow Qx,\ y \leftarrow Qx$	k1
QR factorization	$x \leftarrow Q^T x\ y \leftarrow Q^T x$	k1
Also apply these operations		k3
to sparse vectors?		k3

Table 4: Matrix Vector Operations

D Matrix Operations

This Appendix gives the operations corresponding to the traditional Level 3 BLAS operations, as well as a number of additional single matrix operations. Table 5 lists the matrix and matrix-matrix operations. The operations are intended to apply to both real and complex arguments. Where appropriate one or more of the matrices can also be symmetric (Hermitian) or triangular or general. The matrix T specifically represents a triangular matrix, either upper or lower, L represents a lower triangular matrix and U an upper triangular matrix, all three of which can be unit or non-unit triangular.

D represents at least a diagonal matrix, and maybe also a 2×2 blocked diagonal or a tridiagonal matrix. Q represents an orthogonal (unitary) matrix and it is an open question as to how Q should be stored.

For the sake of compactness the complex operators are omitted, so that whenever a transpose operation is given both the conjugate and conjugate transpose should also be assumed for the complex case.

Matrix copy	$B \leftarrow A$	k1
Matrix transpose	$A \leftarrow A^T,\ B \leftarrow A^T$	k2
Matrix set	$A \leftarrow \alpha I,$	k2
	$A \leftarrow \alpha I + \beta(e^T e - I)$	k3
Matrix norms	$r \leftarrow \lvert\lvert A\rvert\rvert_1, r \leftarrow \lvert\lvert A\rvert\rvert_F, r \leftarrow \lvert\lvert A\rvert\rvert_\infty$	k1
Maximum absolute value	$r \leftarrow \max \lvert A_{i,j}\rvert$	k2
Matrix scaling	$A \leftarrow \alpha A,\ B \leftarrow \alpha A$	k2
Diagonal scaling	$A \leftarrow DA,\ A \leftarrow AD,\ A \leftarrow D_1 A D_2$	k2
Matrix add	$A \leftarrow \pm A + B,\ C \leftarrow \pm A + B$	k2
Matrix add and scale	$A \leftarrow \alpha A + \beta B,\ C \leftarrow \alpha A + \beta B$	k2
Get and put submatrix	$B \leftarrow A_{I,J}$	k2
(dense and sparse)	$B_{I,J} \leftarrow A,\ B_{I,J} \leftarrow A + B_{I,J}$	k2
Matrix matrix product	$A \leftarrow AB,\ A \leftarrow A^T B,\ A \leftarrow AB^T$	k3
	$C \leftarrow AB,\ C \leftarrow A^T B,\ C \leftarrow AB^T$	k2
	$C \leftarrow \pm AB + C,\ C \leftarrow \pm A^T B + C,\ C \leftarrow \pm AB^T + C$	k3
	$D \leftarrow \pm AB + C,\ D \leftarrow \pm A^T B + C,\ D \leftarrow \pm AB^T + C$	k3
	$C \leftarrow \alpha AB + \beta C,$	0
	$D \leftarrow \alpha AB + \beta C$	k3
	$C \leftarrow \alpha A^T B + \beta C$	0
	$D \leftarrow \alpha A^T B + \beta C$	k3
	$C \leftarrow \alpha AB^T + \beta C$	0
	$D \leftarrow \alpha AB^T + \beta C$	k3
Triangular	$B \leftarrow \alpha T^{-1} B$	0
Triangular & double	$B \leftarrow T^{-1}B,\ C \leftarrow T^{-1}B,\ C \leftarrow \alpha T^{-1}B$	k3
(with permutations)	$B \leftarrow BT^{-1},\ B \leftarrow \alpha BT^{-1},\ C \leftarrow BT^{-1},\ C \leftarrow \alpha BT^{-1}$	k3
triangular solve	$B \leftarrow U^{-1}L^{-1}B,\ C \leftarrow U^{-1}L^{-1}B$	k3
(Also with inverse	$B \leftarrow U^{-1}DL^{-1}B,\ C \leftarrow U^{-1}DL^{-1}B$	k3
transpose, and for TT^T	$B \leftarrow L^{-1}U^{-1}B,\ C \leftarrow L^{-1}U^{-1}B$	k3
& TDT^T when SPD)	$B \leftarrow L^{-1}DU^{-1}B,\ C \leftarrow L^{-1}DU^{-1}B$	k3
Symmetric rank k & $k2$	$C \leftarrow \alpha AA^T + \beta C,\ C \leftarrow \alpha A^T A + \beta C$	0
updates ($C = C^T$)	$C \leftarrow \alpha ADA^T + \beta C,\ C \leftarrow \alpha A^T DA + \beta C$	k1
($\alpha = 1$ and $\beta = \pm 1$	$C \leftarrow (\alpha A)B^T + B(\alpha A)^T + \beta C$	0
as special cases?)	$C \leftarrow (\alpha A)^T B + B^T(\alpha A) + \beta C$	k3
	$C \leftarrow (\alpha AD)B^T + B(\alpha AD)^T + \beta C$	k2
	$C \leftarrow (\alpha DA)^T B + B^T(\alpha DA) + \beta C$	k3
Application of Q from	$X \leftarrow QX,\ X \leftarrow XQ,\ X \leftarrow Q^T X,\ X \leftarrow XQ^T$	k2
QR factorization	$Y \leftarrow QX,\ Y \leftarrow XQ,\ Y \leftarrow Q^T X,\ Y \leftarrow XQ^T$	k2
Elementwise multiply	$A_{ij} \leftarrow A_{ij}B_{ij},\ C_{ij} \leftarrow A_{ij}B_{ij}$	k3
Elementwise divide	$A_{ij} \leftarrow A_{ij}/B_{ij},\ C_{ij} \leftarrow A_{ij}/B_{ij}$	k3

Table 5: Matrix Matrix Operations