

FUJITSU Software Technical Computing Suite V4.0L20



Development Studio BLAS LAPACK ScaLAPACK 使用手引書

J2UL-2575-01Z0(00)
2020年2月

まえがき

本製品は、Prof. Dongarra (テネシー大学)を中心とした研究グループで開発され Netlib (<http://www.netlib.org/>)で公開されている BLAS (Basic Linear Algebra Subprograms), LAPACK (Linear Algebra PACKage)及び ScaLAPACK (Scalable LAPACK)を、当社サーバ向けに高度に性能チューニングを行い提供するものです。BLAS 及び LAPACK は、逐次版に加えて、並列処理によって処理時間を短縮することができるスレッド並列版も提供します。また、ScaLAPACK は弊社の MPI、並列実行環境と組み合わせて使用できます。

本書の構成は以下のとおりです。

1. 概説

本製品で提供される機能の概要と、Netlib で公開されている BLAS, LAPACK, ScaLAPACK との関係について記述しています。

2. 呼出し方法

本マニュアルでは各ルーチンの呼出し方法についての記述は省略しています。ここでは、各ルーチンの呼出し方法について記述した書籍や、WWW で公開されているオンラインドキュメントのアドレスを紹介しています。

3. BLAS, LAPACK の使用例

BLAS, LAPACK を使った簡単なサンプルプログラムを使って、OpenMP Fortran プログラムから利用する方法を説明します。

付録 A ルーチン一覧

本製品で提供されるルーチンの一覧表です。

付録 B ライセンス

本製品で使用しているフリーソフトウェアのライセンスを記述しています。

BLAS, LAPACK 及び ScaLAPACK の一般的な使用方法については“2. 呼出し方法”で示すドキュメントを参照してください。本書では本製品固有の情報を記述しています。

本書の内容を理解するための参考資料としては、以下に示すマニュアルがあります。

“Fortran 使用手引書”

“MPI 使用手引書”

OpenMP Fortran の言語仕様については、以下で公開されている仕様書を参照してください。

<http://www.openmp.org/>

謝辞：

BLAS, LAPACK 及び ScaLAPACK は Prof. Dongarra (テネシー大学) を中心とした研究グループで開発され, Netlib で公開されています.

輸出管理規制について

本ドキュメントを輸出または第三者へ提供する場合は、お客様が居住する国および米国輸出管理関連法規等の規制をご確認のうえ、必要な手続きをおとりください。

出版年月および版数

版数	マニュアルコード
2020 年 2 月 初版	J2UL-2575-01Z0(00)

著作権表示

Copyright FUJITSU LIMITED 2020

お願い

- 本書を無断で他に転載しないようお願いします。
- 本書は予告なしに変更されることがあります。

目次

1. 概説	1
1.1 逐次版とスレッド並列版の使い分け	1
1.2 BLAS	2
1.3 LAPACK	2
1.4 ScaLAPACK	3
1.5 BLAS, LAPACK 及び ScaLAPACK の構成	4
2. 呼出し方法	6
2.1 書籍	6
2.2 WWW アドレス	6
3. BLAS, LAPACK の使用例.....	7
3.1 問題	7
3.2 逐次版	7
3.3 スレッド並列版	8
付録 A ルーチン一覧.....	11
A.1 BLAS	11
A.2 LAPACK	16
A.3 ScaLAPACK	33
A.3.1 ScalAPACK	33
A.3.2 PBLAS	36
A.3.3 BLACS	37
A.3.4 ツールルーチン	38
A.4 BLAS の拡張機能	39
A.4.1 半精度版 BLAS ルーチン	39
A.4.2 スレッド並列ライブラリ内の逐次 BLAS ルーチン	39
A.4.3 行列のコピー・転置ルーチン	42
A.4.3.1 Fortran インタフェース	42
A.4.3.2 C/C++ インタフェース	44
A.4.4 SIMD 長単位の実部虚部分離配列対応の行列行列積ルーチン	45
付録 B ライセンス	47
B.1 XBLAS	47
B.2 LAPACK	48
B.3 PLASMA	49
B.4 ScaLAPACK	50

図目次

図 1 BLAS, LAPACK 及び ScaLAPACK の構成.....	5
--	---

表目次

付表 1 Level 1 BLAS ルーチン一覧	11
付表 2 Level 2 BLAS ルーチン一覧	12
付表 3 Level 3 BLAS ルーチン一覧	12
付表 4 Sparse BLAS ルーチン一覧(Fortran)	13
付表 5 Level 1 XBLAS ルーチン一覧(Fortran/C 共通).....	13
付表 6 Level 2 XBLAS ルーチン一覧(Fortran/C 共通)(続く).....	13
付表 7 Level 3 XBLAS ルーチン一覧(Fortran/C 共通).....	15
付表 8 BLAS のスレーブルーチン	16
付表 9 LAPACK ドライバ及び計算用ルーチン一覧(続く).....	16
付表 10 LAPACK 補助ルーチン一覧(続く).....	19
付表 11 C インタフェース LAPACK ルーチン一覧(続く).....	26
付表 12 PLASMA 簡易インタフェースルーチン一覧(続く).....	29
付表 13 PLASMA メインルーチン一覧.....	31
付表 14 互換のため含まれている旧ルーチン	32
付表 15 インタフェースが変更になったルーチン(LAPACK version 3.1.1)	32
付表 16 インタフェースが変更になったルーチン(LAPACK version 3.2)	32
付表 17 ScaLAPACK ドライバ及び計算用ルーチン一覧(続く)	33
付表 18 ScaLAPACK 補助ルーチン一覧(続く).....	34
付表 19 Level 1 PBLAS ルーチン一覧	36
付表 20 Level 2 PBLAS ルーチン一覧	36
付表 21 Level 3 PBLAS ルーチン一覧	37
付表 22 BLACS ルーチン一覧(続く)	37
付表 23 ツールルーチン	38
付表 24 REAL*2 BLAS ルーチン一覧	39
付表 25 Level 1 BLAS ルーチン一覧	40
付表 26 Level 2 BLAS ルーチン一覧	41
付表 27 Level 3 BLAS ルーチン一覧	41
付表 28 REAL*2 BLAS 逐次専用ルーチン一覧	42
付表 29 行列のコピー・転置逐次専用ルーチン一覧	42

1. 概説

本製品は Netlib (<http://www.netlib.org/>)で公開されている BLAS (Basic Linear Algebra Subprograms), LAPACK (Linear Algebra PACKAGE), ScaLAPACK (Scalable LAPACK)と同一のユーザインタフェースを提供するものです。BLAS, LAPACK は一般的な逐次 Fortran プログラムだけでなく、OpenMP を使って並列化されたプログラムからも利用できるようスレッドセーフになっています。また、OpenMP Fortran により並列化されたスレッド並列版も提供します。ScaLAPACK は Fortran 言語から使用できるよう記述されており、ユーザプログラムから CALL 文で利用できます。

1.1 逐次版とスレッド並列版の使い分け

a) 逐次版

逐次版 BLAS, LAPACK は、一般的な逐次 Fortran プログラムから call できます。

また、スレッドセーフな作りになっているため、OpenMP Fortran で書かれたユーザプログラムからも call できます。OpenMP Fortran で記述されたユーザプログラムから利用することで、同じルーチンに独立した複数の問題を与え、複数スレッドで独立に、しかも同時に解くことができます。たとえば、行列積ルーチンを複数スレッドから利用する場合、1 つのスレッドが 1 つの行列積を計算します。

スレッドセーフな BLAS, LAPACK を利用する目的は、独立した複数組のデータを与えて同時に計算させる場合のターンアラウンドを短縮することです。従来の逐次処理では複数組のデータに対してあるサブルーチンを使う場合、データを変えては繰り返し call するというものでした。これに対しスレッドセーフな BLAS, LAPACK では、1 つのスレッドが一組のデータを担当して、複数スレッドがサブルーチンの中を同時に実行することを許します。発生させたスレッドの数に相当する分の並列処理が行われ、時間の短縮が図れます。もちろん、CPU またはコアが複数個備わっていることが条件です。プログラムの例については“3. BLAS, LAPACK の使用例”を参照してください。

b) スレッド並列版

スレッド並列版 BLAS, LAPACK では 1 つの問題を複数スレッドを使って並列に解くことができます。これにより、問題を解く時間を短縮します。たとえば、行列積ルーチンを利用する場合、1 つの行列積を引数に与えると、ルーチン内部でスレッドを生成し、複数スレッドを使って並列に計算します。スレッドの発生、計算の分担、同期化、そして消滅は OpenMP Fortran 仕様で記述しています。

スレッド並列版のルーチン名及び呼び出しインターフェースは従来の BLAS, LAPACK と同じです。ユーザプログラムを変更することなく、スレッド並列版ライブラリを結合するだけで、スレッド並列版を使用することができます。また、スレッドセーフなつくりになっており、OpenMP で並列化したユーザプログラムからも利用することができます。

なお、1 つのプログラムの中で逐次版とスレッド並列版を同時に使用することはできません。

OpenMP で並列化したスレッド並列版 LAPACK に加えて、pthread で並列化した PLASMA を提供します。PLASMA は LAPACK とは異なるルーチン名になっており、1 つのユーザプログラムの中で、PLASMA とスレッド並列版 LAPACK を併用することができます。

1.2 BLAS

BLAS はベクトル演算や行列演算を行うライブラリです。本製品では Netlib で公開されている Level 1 BLAS, Level 2 BLAS, Level 3 BLAS(1998-07-03 版)の他に Sparse BLAS および XBLAS を提供します。これらは以下の計算を行います。

Level 1 BLAS	:	ベクトル演算
Level 2 BLAS	:	行列一ベクトル演算
Level 3 BLAS	:	行列一行列演算
Sparse BLAS	:	スパースベクトル演算
XBLAS Version 1.0.248	:	精度拡張版 BLAS
CBLAS 2/23/03 版	:	BLAS の C インタフェース

BLAS には 81 種類のルーチンが用意されており、各精度を合わせると約 660 本になります。

並列化されているルーチンについて

スレッド並列版では BLAS の使用頻度の高いルーチンについて並列化しています。これら以外のルーチンは逐次ルーチンが結合されます。具体的なルーチン名については“A.1 BLAS”を参照してください。

本製品では Netlib で公開されている BLAS の機能の他に、以下の拡張機能を提供しています。これらの機能の詳細については“A.4 BLAS の拡張機能”を参照してください。

- 半精度 REAL*2 版 BLAS ルーチン
- スレッド並列ライブラリ内の逐次 BLAS ルーチン
- 行列のコピー・転置ルーチン
- SIMD 長単位で格納した配列の複素行列行列積ルーチン

1.3 LAPACK

LAPACK は線形代数の問題を解くライブラリです。本製品では Netlib で公開されている LAPACK version 3.5.0 のルーチンを提供します。

また、LAPACK を pthread で並列化した PLASMA も提供します。本製品では公開されている PLASMA version 2.7.1 のルーチンを提供します。

LAPACK では以下の分野の問題を解く機能を提供します。

- 連立一次方程式
- 線形最小 2 乗問題
- 固有値問題
- 特異値分解

LAPACK には約 400 種類のルーチンが用意されており、各精度で合わせると約 1700 本になります。

LAPACK にはドライバルーチン、計算用ルーチン及び補助ルーチンがあります。ドライバルーチンは連立一次方程式など一般問題のモデルを扱います。計算用ルーチンは LU 分解などのように一般問題を解くのに構成するアルゴリズムを扱います。補助ルーチンは補助的な計算や、共通に使用される低いレベルの計算を行います。

並列化されているルーチンについて

OpenMP で並列化したスレッド並列版では、以下のルーチンが並列化されています。具体的なルーチン名については“A.2 LAPACK”を参照してください。また、これら以外のルーチンについても、並列化された BLAS を呼び出しているため、多くのルーチンが並列に実行されます。

- ・ 一般行列, 正値対称/エルミート行列及び対称/エルミート行列の連立一次方程式 (simple driver 及び expert driver)
- ・ 対称/エルミート行列の固有値問題 (divide and conquer)
- ・ 対称/エルミート行列の一般固有値問題 (divide and conquer)
- ・ 線形最小 2 乗問題
- ・ 特異値分解

1.4 ScaLAPACK

ScaLAPACK は線形代数分野の計算をメッセージパッシングで並列化したライブラリです。連立一次方程式、逆行列、固有値問題、特異値分解などの問題を解くことができます。本製品では Netlib で公開されている ScaLAPACK 及び MPI ベースの BLACS を提供します。

ScaLAPACK の作りとしては、行列演算を並列処理で行う PBLAS と呼ばれるレイヤー、そして、まとめた単位でデータ転送を行う BLACS と呼ばれるレイヤーからなります。

ScaLAPACK, PBLAS, BLACS では以下の機能を提供します。

ScaLAPACK	: 連立一次方程式 固有値問題	線形最小 2 乗問題 特異値分解
PBLAS	: Level 1 PBLAS Level 2 PBLAS Level 3 PBLAS	ベクトル演算 行列一ベクトル演算 行列一行列演算
BLACS	: MPI を使用してベクトルや行列をプロセス間で転送するための ライブラリ	

本製品の ScaLAPACK の各レイヤーは Netlib で公開されている以下のソフトウェアを元に開発しています。

ScaLAPACK	: Version 2.0.2
PBLAS	: ScaLAPACK Version 2.0.2 に含まれるソース
BLACS	: Version 1.1

最上位のレイヤーとしての ScaLAPACK にはドライバールーチン、計算用ルーチン及び補助ルーチンがあります。

すべてのルーチンを合わせると約 200 種類のルーチンが用意されています。各精度で合わせると約 700 本のルーチンを使用することができます。各ルーチンは Fortran 言語で記述されており、ユーザプログラムから CALL 文で呼び出すことができます。

PBLAS レイヤーは BLAS の MPI 並列版であり、BLAS で提供されるルーチンのサブセットの機能を提供します。また、ScaLAPACK レイヤーは LAPACK の MPI 並列版であり、LAPACK で提供されるルーチンのサブセットの機能を提供します。

ScaLAPACK では、複数プロセス間に渡るデータ分割は極めて柔軟な方法を提供していますが、性能面からいえば、ルーチンごとに最適な配置、不得意な配置が存在します。詳細は Netlib で公開されているので参照してください。こうした注意を払えば、ScaLAPACK はスケーラビリティ(プロセス数とデータ規模の増加に伴って性能が上がる)を実現できます。

ScaLAPACK は BLAS、LAPACK を呼び出しており、逐次版またはスレッド並列版 BLAS、LAPACK を選択して結合することができます。スレッド並列版 BLAS、LAPACK と組み合わせて結合すれば、ScaLAPACK レイヤーでは MPI で並列に計算し、各プロセスでは BLAS、LAPACK がマルチスレッドにより並列計算します。一般に、ノード内ではスレッド並列にプロセッサを割り当てたほうが計算バランスがよく、転送が不要になるため高速ですが、計算の一部を BLAS、LAPACK を使わずに ScaLAPACK レイヤーで並列計算を行っているルーチ

ンでは、逆に遅くなる場合があります。このためプロセスとスレッドにバランス良くプロセッサを割り当てる必要があります。

ScaLAPACK はスレッドセーフではないため、OpenMP Fortran プログラムで複数のスレッドから同時に利用することはできません。

1.5 BLAS, LAPACK 及び ScaLAPACK の構成

BLAS, LAPACK 及び ScaLAPACK の各レイヤーの構成を図 1 に示します。

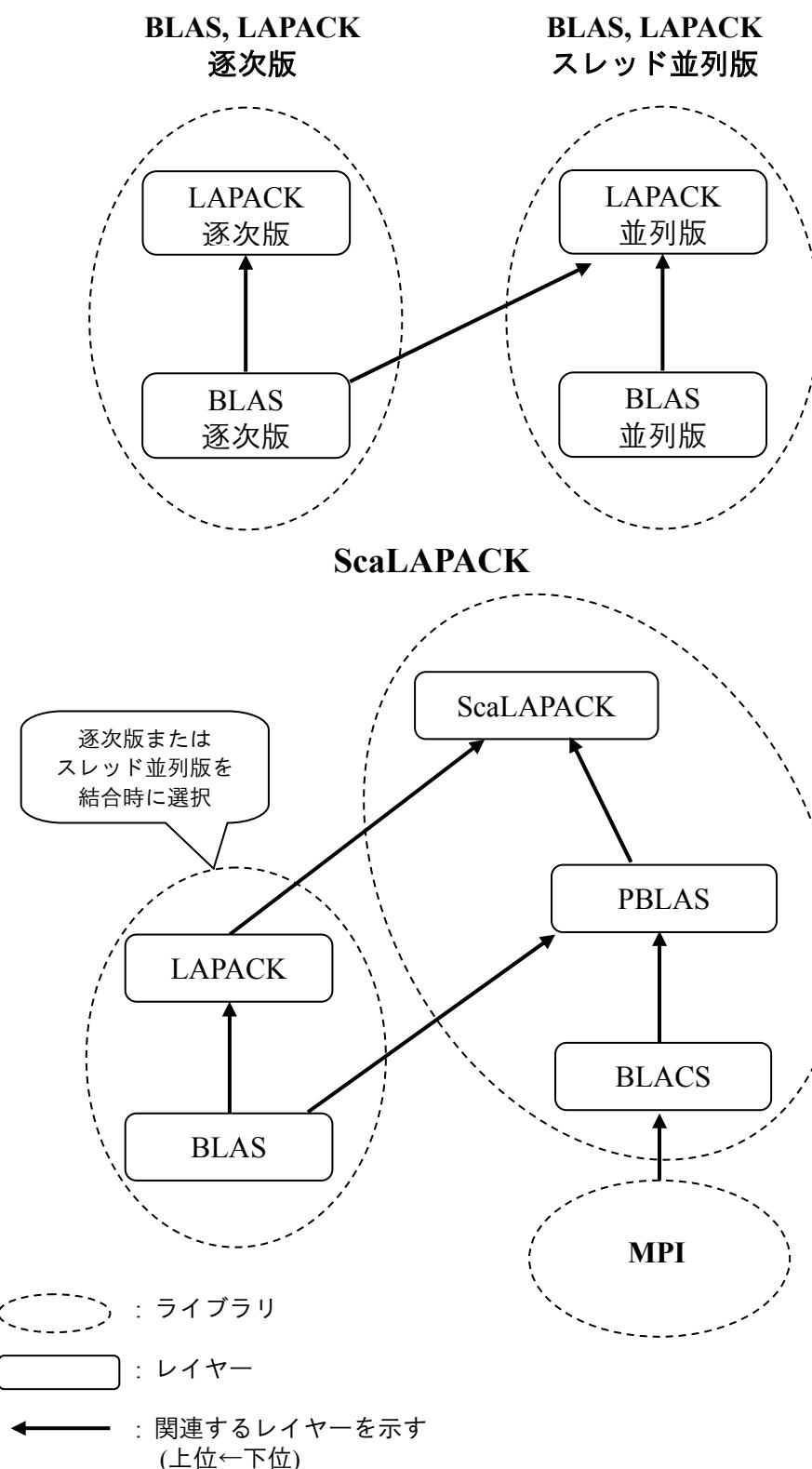


図1 BLAS, LAPACK 及び ScaLAPACK の構成

2. 呼出し方法

本製品に含まれるルーチンの呼出し方法は、Netlib で公開されている BLAS, LAPACK, ScaLAPACK の該当するルーチンと同じです。各ルーチンの呼出し方法については、Netlib のルーチンに関する以下の書籍や WWW で公開されている情報を参照してください。

2.1 書籍

LAPACK, ScaLAPACK に関して以下の書籍が販売されています。

- *LAPACK Users' Guide, Third Edition* (SIAM, 1999)
本書は LAPACK の使用方法について解説しています。ドライバルーチン及び計算用ルーチンの呼出し方法や計算内容、引数の説明、性能、精度、LAPACK が検出するエラーなどに関する詳細な説明が記述されています。また、BLAS のクイックリファレンスも含んでいます。
- *ScaLAPACK Users' Guide* (SIAM, 1997)
本書は ScaLAPACK の使用方法について解説しています。行列を分割して格納する方法、ドライバルーチン及び計算用ルーチンの呼出し方法や計算内容、引数の説明、性能、精度、ScaLAPACK が検出するエラーなどに関する詳細な説明が記述されています。また、PBLAS 及び BLACS のクイックリファレンスも含んでいます。関連するドキュメントやプログラムが多数収録された CD-ROM が付属しています。

2.2 WWW アドレス

BLAS, LAPACK, ScaLAPACK に関する情報は <http://www.netlib.org/> で公開されています。特に、以下のアドレス(2019年2月現在)には BLAS, LAPACK 及び ScaLAPACK の使用方法(呼出し方法)が示されているので、参照してください。

- BLAS
<http://www.netlib.org/blas/>
- XBLAS
<http://www.netlib.org/xblas/>
- CBLAS
<http://www.netlib.org/blas/blast-forum/>
- LAPACK
<http://www.netlib.org/lapack/>
- LAPACKE (LAPACK の C インタフェース)
<http://www.netlib.org/lapack/lapacke.html>
- PLASMA
<https://bitbucket.org/icl/plasma>
- ScaLAPACK
<http://www.netlib.org/scalapack/>

3. BLAS, LAPACK の使用例

OpenMP Fortran プログラムから BLAS, LAPACK を呼び出す方法を、連立一次方程式の計算を例に挙げて説明します。

3.1 問題

n 元の連立一次方程式を考えます。

$$Ax=b$$

ここで, A は n 次元の実行列, b は n 次の右辺ベクトル, そして x は n 次の解ベクトルです。今、右辺ベクトルが複数あってそれぞれの解ベクトルを求めたいとします。

$$AX=B$$

ここで, B の各列が右辺ベクトル b_i ($i=1,2,\dots,m$) であり, X の各列が解ベクトル x_i ($i=1,2,\dots,m$) です。

3.2 逐次版

ここではスレッドセーフ性を利用した使い方を述べます。そのために、最初、普通の逐次版 BLAS, LAPACK を使用するプログラムの例を示します。

a) 逐次プログラムからの利用

200 元の連立一次方程式を、80 本の異なる右辺ベクトルに対して解くとします。ドライバールーチン DGESV を使って一度に行なうことができますが、ここでは、との説明のために DGETRF(LU 分解)と DGETRS(求解)を使います。プログラム中, inita(...)は配列 a を初期化するサブルーチン, initb(...)は 80 本の右辺ベクトルを初期化するルーチンです。

```
implicit real*8 (a-h,o-z)
parameter(maxn=200,m=80,k=maxn+1)
real*8 a(k,maxn),b(k,m)
integer ip(maxn)
C =====
C Define the matrix
C =====
n=maxn
call inita(a,k,n)
call initb(b,k,n,m)
C =====
C LU decomposition
C =====
call dgetrf(n,n,a,k,ip,info)
if(info.ne. 0) then
    print *,'The given problem seems to be not normal'
    stop
endif
C =====
C Solution
C =====
call dgetrs('N',n,m,a,k,ip,b,k,info)
:
```

b) OpenMP Fortran プログラムからの利用

上の例において、サブルーチン DGETRS は 80 本の右辺ベクトルに対して、80 本の解ベクトルを求めます。各右辺ベクトルに対する解ベクトルは、理論上、独立に計算できるので複数スレッドで並列処理できます。今、4 本ずつを 1 スレッドに分け与え、DGETRS で同時に解くとします。このために OpenMP のディレクティブを使って、DGETRS への呼び出し部分を並列化します。以下はその例です。

```
implicit real*8 (a-h,o-z)
parameter(maxn=200,m=80,mbblk=4,k=maxn)
real*8 a(k,maxn),b(k,m)
integer ip(maxn)
C =====
C Define the matrix
C =====
n=maxn
call inita(a,k,n)
call initb(b,k,n,m)
C =====
C LU decomposition
C =====
call dgetrf(n,n,a,k,ip,info)
if(info.ne. 0) then
    print *, 'The given problem seems to be not normal'
    stop
endif
C =====
C Solution
C =====
!$OMP PARALLEL DO PRIVATE(mb,info)
do i=1,m,mbblk
    mb=min(mbblk,m-i+1)
    call dgetrs('N',n,mb,a,k,ip,b(1,i),k,info)
end do
!$OMP END PARALLEL DO
:
```

解説

1. 翻訳指示行 !\$OMP PARALLEL と !\$OMP END PARALLEL で囲まれたブロックが、複数のスレッドで実行されます。 !\$OMP PARALLEL と書かれた位置が複数のスレッドの発生起点になります。PARALLEL 指示行のオプションの PRIVATE(...) は、指定した変数/配列はスレッドごとに別領域にコピーを持たせることを指示しています。これ以外のブロック内の変数/配列は、この例の場合、スレッド間で共有されます。すなわち、メモリには 1 つの実体しかありません。
2. 翻訳指示行 !\$OMP DO は、直下の DO 文について、DO インデックスの各値に対して、DO 内の一連の実行文は独立に計算してよいことを示し、スレッド間で手分けして計算させる意味を持ちます。
3. さらに特徴的なことは、上のプログラムを、-Kopenmp オプションを指定せずに翻訳し、BLAS, LAPACK を結合すれば、処理は遅くなりますが正常に動作し、同じ計算結果を得ることができます。これは、上のプログラムが !\$OMP の翻訳指示行を除けば普通の逐次処理プログラムと同じであること、その翻訳指示行が普通の Fortran コンパイラにはコメント文として扱われるためです。

3.3 スレッド並列版

スレッド並列版 BLAS, LAPACK を使用するプログラムの例を示します。最初に逐次プログラムから呼び出す例を、次に OpenMP Fortran プログラムで並列化されたプログラムから呼び出す例を説明します。

a) 逐次部分からの利用

3000 元の連立一次方程式を、800 本の異なる右辺ベクトルに対して解くとします。使うサブルーチンは、LU 分解から求解までを行う DGESV とします。以下のプログラムをスレッド並列版と結合すると、DGESV の内部でスレッドを生成し、複数スレッドで問題を解きます。(このような逐次 Fortran プログラムからの呼び出しだけでなく、OpenMP Fortran プログラムの逐次部分から呼び出した場合も同様です。)

```
implicit real*8 (a-h,o-z)
parameter(maxn=3000,m=800,k=maxn+1)
real*8 a(k,maxn),b(k,m)
integer ip(maxn)
C =====
C Define the matrix
C =====
n=maxn
call inita(a,k,n)
call initb(b,k,n,m)
C =====
C LU decomposition and Solution
C =====
call dgesv(n,m,a,k,ip,b,k,info)
if(info.ne. 0) then
    print *, 'The given problem seems to be not normal'
    stop
endif
:
:
```

b) OpenMP Fortran プログラムからの利用例

今度は、全く異なる 2 つの連立一次方程式を、複数のスレッドを使って同時に解きます。各問題をサブルーチン DGESV に与えます。DGESV の内部では、さらに各問題を並列処理で解きます。以下のプログラムでは、1 つの問題あたり 2 スレッドを使わせるため環境変数 OMP_NUM_THREADS に 2 を設定します。問題が 2 つなので、このプログラム全体を 4 コアのシステムで実行すると、1 つの問題を解くのに 2 コア を使うことになります。

```
implicit real*8 (a-h,o-z)
parameter(maxn=3000,m=800,k=maxn,np=2)
real*8 a(k,maxn,np),b(k,m,np)
integer nsize(np)
data nsize/2800,3000/
integer ip(maxn,np)
!$OMP PARALLEL DO DEFAULT(SHARED) PRIVATE(n,info)
    do ii=1,np
C =====
C Define the matrix
C =====
n=nsize(ii)
call inita(a(1,1,ii),k,n)
call initb(b(1,1,ii),k,n,m)
C =====
C LU decomposition and Solution
C =====
call dgesv(n,m,a(1,1,ii),k,ip(1,ii),b(1,1,ii),k,info)
if(info.ne. 0) then
    print *, 'The given problem seems to be not normal'
    stop
endif
end do
!$OMP END PARALLEL DO
:
:
```

解説

1. パラレルリージョンの内側から BLAS または LAPACK のスレッド並列版を呼び出す場合は、環境変数 OMP_NESTED または OMP_SET_NESTED サブルーチンでパラレル

リージョンのネストを有効にします。また、生成するスレッドの数だけ CPU またはコアが備わっていないと期待どおりの性能は得られません。

付録 A ルーチン一覧

以下に本製品で提供されるルーチンの一覧を示します。

A.1 BLAS

BLAS で使用可能なサブルーチン一覧を付表 1～付表 7 に示します。また、付表 8 に BLAS に含まれるスレーブルーチン一覧を示します。

「ルーチン名」の#の部分には、データ型及び演算精度を示す文字が入ります。各文字は以下の意味を持ちます。

- S : 単精度実数型
- D : 倍精度実数型
- C : 単精度複素数型
- Z : 倍精度複素数型

また、精度の組み合わせは複数の精度を使用することを意味します。たとえば、SCNRM2 の'SC'は単精度複素数の入力に対し、単精度の実数を返す関数であることを示しています。

並列版の欄の記号は以下の意味を持ちます。

- : スレッド並列版で並列化されているルーチン。
- : スレッド並列版で並列化されていない(逐次計算を行う)ルーチン。

付表1 Level 1 BLAS ルーチン一覧

Fortran BLAS ルーチン名	CBLAS ルーチン名	サポートする精度 (CBLAS では小文字)	並列版
#ROTG	cblas_#rotg	S, D	-
#ROTMG	cblas_#rotmg	S, D	-
#ROT	cblas_#rot	S, D	○
#ROTM	cblas_#rotm	S, D	○
#SWAP	cblas_#swap	S, D, C, Z	○
#SCAL	cblas_#scal	S, D, C, Z, CS, ZD	○
#COPY	cblas_#copy	S, D, C, Z	○
#AXPY	cblas_#axpy	S, D, C, Z	○
#DOT	cblas_#dot	S, D, DS	○
#DOTU	cblas_#dotu_sub	C, Z	○
#DOTC	cblas_#dotc_sub	C, Z	○
##DOT	cblas_##dot	SDS	○
#NRM2	cblas_#nrm2	S, D, SC, DZ	○
#ASUM	cblas_#asum	S, D, SC, DZ	○
I#AMAX	cblas_i#amax	S, D, C, Z	○

付表2 Level 2 BLAS ルーチン一覧

Fortran BLAS ルーチン名	CBLAS ルーチン名	サポートする精度 (CBLAS では小文字)	並列版
#GEMV	blas_#gemv	S, D, C, Z	○
#GBMV	blas_#gbmv	S, D, C, Z	—
#HEMV	blas_#hemv	C, Z	○
#HBMV	blas_#hbmv	C, Z	—
#HPMV	blas_#hpmv	C, Z	○
#SYMV	blas_#symv	S, D	○
#SBMV	blas_#sbmv	S, D	—
#SPMV	blas_#spmv	S, D	○
#TRMV	blas_#trmv	S, D, C, Z	○
#TBMV	blas_#tbtmv	S, D, C, Z	—
#TPMV	blas_#tpmv	S, D, C, Z	—
#TRSV	blas_#trsv	S, D, C, Z	○
#TBSV	blas_#tbsv	S, D, C, Z	—
#TPSV	blas_#tpsv	S, D, C, Z	—
#GER	blas_#ger	S, D	○
#GERU	blas_#geru	C, Z	○
#GERC	blas_#gerc	C, Z	○
#HER	blas_#her	C, Z	○
#HPR	blas_#hpr	C, Z	—
#HER2	blas_#her2	C, Z	○
#HPR2	blas_#hpr2	C, Z	—
#SYR	blas_#syr	S, D	○
#SPR	blas_#spr	S, D	—
#SYR2	blas_#syr2	S, D	○
#SPR2	blas_#spr2	S, D	—

付表3 Level 3 BLAS ルーチン一覧

Fortran BLAS ルーチン名	CBLAS ルーチン名	サポートする精度 (CBLAS では小文字)	並列版
#GEMM	blas_#gemm	S, D, C, Z	○
#SYMM	blas_#symm	S, D, C, Z	○
#HEMM	blas_#hemm	C, Z	○
#SYRK	blas_#syrk	S, D, C, Z	○
#HERK	blas_#herk	C, Z	○
#SYR2K	blas_#syr2k	S, D, C, Z	○
#HER2K	blas_#her2k	C, Z	○
#TRMM	blas_#trmm	S, D, C, Z	○
#TRSM	blas_#trsm	S, D, C, Z	○

付表4 Sparse BLAS ルーチン一覧(Fortran)

ルーチン名	サポートする精度	並列版
#AXPYI	S, D, C, Z	—
#DOTI	S, D	—
#DOTCI	C, Z	—
#DOTUI	C, Z	—
#GTHR	S, D, C, Z	—
#GTHRZ	S, D, C, Z	—
#SCTR	S, D, C, Z	—
#ROTI	S, D	—

付表5 Level 1 XBLAS ルーチン一覧(Fortran/C 共通)

ルーチン名	精度(#)	精度(\$)	並列版
BLAS_#dot_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#sum_\$	s	x	—
	d	x	—
	c	x	—
	z	x	—
BLAS_#axpby_\$	s	x	—
	d	s, s_x, x	—
	c	s, s_x, x	—
	z	c, c_x, d, d_x, x	—
BLAS_#waxpby_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—

付表6 Level 2 XBLAS ルーチン一覧(Fortran/C 共通) (続く)

ルーチン名	精度(#)	精度(\$)	並列版
BLAS_#gemv_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#gemv2_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x,	—

付表 6 Level 2 XBLAS ルーチン一覧(Fortran/C 共通) (続く)

ルーチン名	精度(#)	精度(\$)	並列版
BLAS_#gemv2_\$	z	d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#gbmv_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#gbmv2_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#spmv_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#hpmv_\$	c	c_s, c_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#symv_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#symv2_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#hemv_\$	c	c_s, c_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#hemv2_\$	c	c_s, c_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#sbmv_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—

付表6 Level 2 XBLAS ルーチン一覧(Fortran/C 共通) (続き)

ルーチン名	精度(#)	精度(\$)	並列版
BLAS_#sbmv_\$	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#hbmv_\$	c	c_s, c_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#trsv_\$	s	x	—
	d	s, s_x, x	—
	c	s, s_x, x	—
	z	c, c_x, d, d_x, x	—
BLAS_#tbsv_\$	s	x	—
	d	s, s_x, x	—
	c	s, s_x, x	—
	z	c, c_x, d, d_x, x	—
BLAS_#trmv_\$	s	x	—
	d	s, s_x, x	—
	c	s, s_x, x	—
	z	c, c_x, d, d_x, x	—
BLAS_#tpmv_\$	s	x	—
	d	s, s_x, x	—
	c	s, s_x, x	—
	z	c, c_x, d, d_x, x	—
BLAS_#ge_sum_mv_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—

付表7 Level 3 XBLAS ルーチン一覧(Fortran/C 共通)

ルーチン名	精度(#)	精度(\$)	並列版
BLAS_#gemm_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#symm_\$	s	x	—
	d	d_s, d_s_x, s_d, s_d_x, s_s, s_s_x, x	—
	c	c_s, c_s_x, s_c, s_c_x, s_s, s_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, d_d, d_d_x, d_z, d_z_x, x, z_c, z_c_x, z_d, z_d_x	—
BLAS_#hemm_\$	c	c_s, c_s_x, x	—
	z	c_c, c_c_x, c_z, c_z_x, x, z_c, z_c_x, z_d, z_d_x	—

付表8 BLAS のスレーブルーチン

ルーチン名	項目
DCABS1, LSAME, XERBLA	公開されている BLAS に含まれるスレーブルーチン

A.2 LAPACK

LAPACK で使用可能なサブルーチン一覧を付表 9～付表 10 に示します。

表に示したルーチン名は单精度実数及び单精度複素数ルーチンの名前です。倍精度実数のルーチン名は单精度実数ルーチン名の先頭の'S'を'D'に置き換え、倍精度複素数のルーチン名は单精度複素数のルーチン名の先頭の'C'を'Z'に置き換えてください。

補助ルーチンのうち XERBLA および XERBLA_ARRAY はデータの型に依存せず、ただ 1 つだけ存在するルーチンです。

ドライバ及び計算用ルーチンの表の並列版の欄の記号は以下の意味を持ちます。

- : スレッド並列版で並列化されているルーチン。
- : スレッド並列版で並列化されていないルーチン。ただし、多くのルーチンは BLAS のスレッド並列版を使用することにより並列に実行されます。

ルーチン名の後に “+” が付いているものは Netlib 版 LAPACK が version3.2.2 から version3.4.1 になったときに新規に追加されたルーチンを意味します。

また、ルーチン名の後ろに “*” が付いているものは Netlib 版 LAPACK が version3.4.1 から version3.5.0 になったときに新規に追加されたルーチンを意味します。

付表9 LAPACK ドライバ及び計算用ルーチン一覧(続く)

実数ルーチン	複素数ルーチン	並列版	実数ルーチン	複素数ルーチン	並列版
DSGESV	ZCGESV	-	SGEBAL	CGEBAL	-
DSPOSV	ZCPOSV	-	SGEBRD	CGEBRD	○
SBBCSD ⁺	CBBCSD ⁺	-	SGECON	CGECON	-
SBDSDC	-	○	SGEEQU	CGEEQU	○
SBDSQR	CBDSQR	-	SGEES	CGEES	-
SDISNA	-	-	SGEESX	CGEESX	-
SGBBRD	CGBBRD	-	SGEEQUB	CGEEQUB	-
SGBCON	CGBCON	-	SGEEV	CGEEV	-
SGBEQU	CGBEQU	-	SGEEVX	CGEEVX	-
SGBEQUB	CGBEQUB	-	SGEHRD	CGEHRD	○
SGBRFS	CGBRFS	-	SGEJSV	-	-
SGBRFSX	CGBRFSX	-	SGELQF	CGELQF	○
SGBSV	CGBSV	-	SGELS	CGELS	○
SGBSVX	CGBSVX	-	SGELSD	CGELSD	○
SGBSVXX	CGBSVXX	-	SGELSS	CGELSS	-
SGBTRF	CGBTRF	-	SGELSY	CGELSY	-
SGBTRS	CGBTRS	-	SGEMQRT ⁺	CGEMQRT ⁺	-
SGEBAK	CGEBAK	-	SGEQLF	CGEQLF	○

付表 9 LAPACK ドライバ及び計算用ルーチン一覧(続く)

実数ルーチン	複素数ルーチン	並列版	実数ルーチン	複素数ルーチン	並列版
SGEQP3	CGEQP3	—	SHSEQR	CHSEQR	—
SGEQRF	CGEQRF	○	SLARTGS ⁺	—	—
SGEQRFP	CGEQRFP	—	SLA_GERPVGRW ⁺	CLA_GERPVGRW ⁺	—
SGEQRT ⁺	CGEQRT ⁺	—	SOPGTR	CUPGTR	—
SGEQRT2 ⁺	CGEQRT2 ⁺	—	SOPMTR	CUPMTR	—
SGEQRT3 ⁺	CGEQRT3 ⁺	—	SORBDB ⁺	CUNBDB ⁺	—
SGERFS	CGERFS	○	SORBDB1 *	CUNBDB1 *	—
SGERFSX	CGERFSX	—	SORBDB2 *	CUNBDB2 *	—
SGERQF	CGERQF	○	SORBDB3 *	CUNBDB3 *	—
SGESDD	CGESDD	○	SORBDB4 *	CUNBDB4 *	—
SGESV	CGESV	○	SORBDB5 *	CUNBDB5 *	—
SGESVD	CGESVD	—	SORBDB6 *	CUNBDB6 *	—
SGESVJ	—	—	SORCSD ⁺	CUNCSD ⁺	—
SGESVX	CGESVX	○	SORCSD2BY1 *	CUNCSD2BY1 *	—
SGESVXX	CGESVXX	—	SORGBR	CUNGBR	○
SGETRF	CGETRF	○	SORGHR	CUNGHR	—
SGETRI	CGETRI	—	SORGLQ	CUNGLQ	○
SGETRS	CGETRS	—	SORGQL	CUNGQL	○
SGGBAK	CGGBAK	—	SORGQR	CUNGQR	○
SGGBAL	CGGBAL	—	SORGRRQ	CUNGRQ	○
SGGES	CGGES	—	SORGTR	CUNGTR	○
SGGESX	CGGESX	—	SORMBR	CUNMBR	○
SGGEV	CGGEV	—	SORMHR	CUNMHR	—
SGGEVX	CGGEVX	—	SORMLQ	CUNMLQ	○
SGGGLM	CGGGLM	○	SORMQL	CUNMQL	○
SGGHRD	CGGHRD	—	SORMQR	CUNMQR	○
SGGLSE	CGGLSE	○	SORMRQ	CUNMRQ	○
SGGQRF	CGGQRF	○	SORMRZ	CUNMRZ	—
SGGRQF	CGGRQF	○	SORMTR	CUNMTR	○
SGGSVD	CGGSVD	—	SPBCON	CPBCON	—
SGGSVP	CGGSVP	—	SPBEQU	CPBEQU	—
SGTCON	CGTCON	—	SPBRFS	CPBRFS	—
SGTRFS	CGTRFS	—	SPBSTF	CPBSTF	—
SGTSV	CGTSV	—	SPBSV	CPBSV	—
SGTSVX	CGTSVX	—	SPBSVX	CPBSVX	—
SGTTRF	CGTTRF	—	SPBTRF	CPBTRF	—
SGTTRS	CGTTRS	—	SPBTRS	CPBTRS	—
—	CHEEQUB	—	SPFTRF	CPFTRF	—
—	CHERFSX	—	SPFTRI	CPFTRI	—
—	CHESVXX	—	SPFTRS	CPFTRS	—
SHGEQZ	CHGEQZ	—	SPOCON	CPOCON	○
SHSEIN	CHSEIN	—	SPOEQU	CPOEQU	○

付表 9 LAPACK ドライバ及び計算用ルーチン一覧(続く)

実数ルーチン	複素数ルーチン	並列版	実数ルーチン	複素数ルーチン	並列版
SPOEQUB	CPOEQUB	—	SSPGVX	CHPGVX	—
SPORFS	CPORFS	○	SSPRFS	CSPRFS	—
SPORFSX	CPORFSX	—	—	CHPRFS	—
SPOSV	CPOSV	○	SSPSV	CSPSV	—
SPOSVX	CPOSVX	○	—	CHPSV	—
SPOSVXX	CPOSVXX	—	SSPSVX	CSPSVX	—
SPSTF2	CPSTF2	—	—	CHPSVX	—
SPSTRF	CPSTRF	—	SSPTRD	CHPTRD	—
SPOTRF	CPOTRF	○	SSPTRF	CSPTRF	—
SPOTRI	CPOTRI	—	—	CHPTRF	—
SPOTRS	CPOTRS	—	SSPTRI	CSPTRI	—
SPPCON	CPPCON	—	—	CHPTRI	—
SPPEQU	CPPEQU	—	SSPTRS	CSPTRS	—
SPPRFS	CPPRFS	—	—	CHPTRS	—
SPPSV	CPPSV	—	SSTEBC	—	—
SPPSVX	CPPSVX	—	SSTEBC	CSTEBC	○
SPPTRF	CPPTRF	—	SSTEGR	CSTEGR	—
SPPTRI	CPPTRI	—	SSTEIN	CSTEIN	—
SPPTRS	CPPTRS	—	SSTEQR	CSTEQR	—
SPTCON	CPTCON	—	SSTERF	—	—
SPTEQR	CPTEQR	—	SSTEV	—	—
SPTRFS	CPTRFS	—	SSTEVD	—	—
SPTSV	CPTSV	—	SSTEVR	—	—
SPTSVX	CPTSVX	—	SSTEVX	—	—
SPTTRF	CPTTRF	—	SSYCON	CSYCON	—
SPTTRS	CPTTRS	—	—	CHECON	—
SSBEV	CHBEV	—	SSYCON_ROOK *	CSYCON_ROOK *	—
SSBEVD	CHBEVD	—	—	CHECON_ROOK *	—
SSBEVX	CHBEVX	—	SSYCONV ⁺	CSYCONV ⁺	—
SSBGST	CHBGST	—	SSYEQUB	CSYEQUB	—
SSBGV	CHBGV	—	SSYEV	CHEEV	—
SSBGVD	CHBGVD	—	SSYEVD	CHEEVD	○
SSBGVX	CHBGVX	—	SSYEVR	CHEEVR	—
SSBTRD	CHBTRD	—	SSYEVX	CHEEVX	—
SSPCON	CSPCON	—	SSYGST	CHEGST	—
—	CHPCON	—	SSYGV	CHEGV	—
SSPEV	CHPEV	—	SSYGV	CHEGV	○
SSPEVD	CHPEVD	—	SSYGV	CHEGV	—
SSPEVX	CHPEVX	—	SSYGVX	CHEGVX	—
SSPGST	CHPGST	—	SSYRFS	CSYRFS	○
SSPGV	CHPGV	—	SSYRFSX	CSYRFSX	—
SSPGVD	CHPGVD	—	—	CHERFS	○
			SSYSV	CSYSV	○

付表9 LAPACK ドライバ及び計算用ルーチン一覧(続き)

実数ルーチン	複素数ルーチン	並列版	実数ルーチン	複素数ルーチン	並列版
—	CHESV	○	STBRFS	CTBRFS	—
SSYSV_ROOK *	CSYSV_ROOK *	○	STBTRS	CTBTRS	—
—	CHESV_ROOK *	○	SSTEMR	CSTEMR	—
SSYSVX	CSYSVX	○	STFTRI	CTFTRI	—
SSYSVXX	CSYSVXX	—	STGEVC	CTGEVC	—
—	CHESVX	○	STGEXC	CTGEXC	—
SSYTRD	CHETRD	○	STGSEN	CTGSEN	—
SSYTF2_ROOK *	CSYTF2_ROOK *	—	STGSJA	CTGSJA	—
—	CHETF2_ROOK *	—	STGSNA	CTGSNA	—
SSYTRF	CSYTRF	○	STGSYL	CTGSYL	—
—	CHETRF	○	STPCON	CTPCON	—
SSYTRF_ROOK *	CSYTRF_ROOK *	○	STPMQRT ⁺	CTPMQRT ⁺	—
—	CHETRF_ROOK *	○	STPQRT ⁺	CTPQRT ⁺	—
SSYTRI	CSYTRI	—	STPQRT2 ⁺	CTPQRT2 ⁺	—
—	CHETRI	—	STPRFS	CTPRFS	—
SSYTRI_ROOK *	CSYTRI_ROOK *	—	STPTRI	CTPTRI	—
—	CHETRI_ROOK *	—	STPTRS	CTPTRS	—
SSYTRI2 ⁺	CSYTRI2 ⁺	—	STRCON	CTRCON	—
—	CHETRI2 ⁺	—	STREVC	CTREVC	—
SSYTRI2X ⁺	CSYTRI2X ⁺	—	STREXC	CTREXC	—
—	CHETRI2X ⁺	—	STRRFS	CTRFFS	—
SSYTRS	CSYTRS	○	STRSEN	CTRSEN	—
—	CHETRS	—	STRSNA	CTRSNA	—
SSYTRS_ROOK *	CSYTRS_ROOK *	—	STRSYL	CTRSYL	—
—	CHETRS_ROOK *	—	STRTRI	CTRTRI	—
SSYTRS2 ⁺	CSYTRS2 ⁺	—	STRTRS	CTRTRS	—
—	CHETRS2 ⁺	—	STZRZF	CTZRZF	—
STBCON	CTBCON	—			

付表10 LAPACK 補助ルーチン一覧(続く)

実数ルーチン	複素数ルーチン
DLAT2S	—
—	CLACGV
—	CLACRM
—	CLACRT
—	CLAESY
—	CROT
—	CSPMV
—	CSPR
—	CSROT

付表 10 LAPACK 補助ルーチン一覧(続く)

実数ルーチン	複素数ルーチン
—	CSYMV
—	CSYR
—	ICMAX1
ILACLC	—
ILACLR	—
ILADIAG	—
ILADLC	—
ILADLR	—
ILAENV	—
ILAPREC	—
ILASLC	—
ILASLR	—
ILATRANS	—
ILAUPLO	—
ILAZLC	—
ILAZLR	—
ILAVER	—
IPARMQ	—
LSAME	—
LSAMEN	—
—	SCSUM1
SGBTF2	CGBTF2
SGEBD2	CGEBD2
SGEHD2	CGEHD2
SGELQ2	CGELQ2
SGEQL2	CGEQL2
SGEQR2	CGEQR2
SGERQ2	CGERQ2
SGEQR2P	CGEQR2P
SGESC2	CGESC2
SGETC	CGETC2
SGETF2	CGETF2
SGSVJ0	—
SGSVJ1	—
SGTTS2	CGTTS2
SISNAN	—
SLABAD	—
SLABRD	CLABRD
SLACN2	CLACN2
SLACON	CLACON
SLACPY	CLACPY
SLADIV	CLADIV
SLAE2	—
SLAEBZ	—
SLAED0	CLAED0

付表 10 LAPACK 補助ルーチン一覧(続く)

実数ルーチン	複素数ルーチン
SLAED1	—
SLAED2	—
SLAED3	—
SLAED4	—
SLAED5	—
SLAED6	—
SLAED7	CLAED7
SLAED8	CLAED8
SLAED9	—
SLAEDA	—
SLAEIN	CLAEIN
SLAEV2	CLAEV2
SLAEXC	—
SLAG2	—
SLAG2D	CLAG2Z
SLAGS2	—
SLAGTF	—
SLAGTM	CLAGTM
SLAGTS	—
SLAGV2	—
SLAHQR	CLAHQR
SLAHR2	CLAHR2
SLAHRD	CLAHRD
SLAIC1	CLAIC1
SLAISNAN	—
SLALN2	—
SLALS0	CLALS0
SLALSA	CLALSA
SLALSD	CLALSD
SLAMCH	—
SLAMRG	—
SLANEG	—
SLANGB	CLANGB
SLANGE	CLANGE
SLANGT	CLANGT
SLANHS	CLANHS
SLANSB	CLANSB
SLANSF	CLANHF
—	CLANHB
SLANSP	CLANSP
—	CLANHP
SLANST	CLANHT
SLANSY	CLANSY
—	CLANHE

付表 10 LAPACK 補助ルーチン一覧(続く)

実数ルーチン	複素数ルーチン
SLANTB	CLANTB
SLANTP	CLANTP
SLANTR	CLANTR
SLANV2	—
SLAPLL	CLAPLL
SLAPMR ⁺	CLAPMR ⁺
SLAPMT	CLAPMT
SLAPY2	—
SLAPY3	—
SLAQGB	CLAQGB
SLAQGE	CLAQGE
SLAQP2	CLAQP2
SLAQPS	CLAQPS
SLAQR0	CLAQR0
SLAQR1	CLAQR1
SLAQR2	CLAQR2
SLAQR3	CLAQR3
SLAQR4	CLAQR4
SLAQR5	CLAQR5
SLAQSB	CLAQSB
SLAQSP	CLAQSP
SLAQSY	CLAQSY
SLAQTR	—
SLAR1V	CLAR1V
SLAR2V	CLAR2V
SLARF	CLARF
SLARFB	CLARFB
SLARFG	CLARFG
SLARFGP	CLARFGP
SLARFT	CLARFT
SLARFX	CLARFX
SLARGV	CLARGV
SLARNV	CLARNV
SLARRA	—
SLARRB	—
SLARRC	—
SLARRD	—
SLARRE	—
SLARRF	—
SLARRJ	—
SLARRK	—
SLARRR	—
SLARRV	CLARRV
SLARSCL2	CLARSCL2
SLASCL2	CLASCL2

付表 10 LAPACK 補助ルーチン一覧(続く)

実数ルーチン	複素数ルーチン
SLARTG	CLARTG
SLARTGP ⁺	—
SLARTV	CLARTV
SLARUV	—
SLARZ	CLARZ
SLARZB	CLARZB
SLARZT	CLARZT
SLAS2	—
SLASCL	CLASCL
SLASD0	—
SLASD1	—
SLASD2	—
SLASD3	—
SLASD4	—
SLASD5	—
SLASD6	—
SLASD7	—
SLASD8	—
SLASD9	—
SLASDA	—
SLASDQ	—
SLASDT	—
SLASET	CLASET
SLASQ1	—
SLASQ2	—
SLASQ3	—
SLASQ4	—
SLASQ5	—
SLASQ6	—
SLASR	CLASR
SLASRT	—
SLASSQ	CLASSQ
SLASV2	—
SLASWP	CLASWP
SLASY2	—
SLASYF	CLASYF
—	CLAHEF
SLASYF_ROOK *	CLASYF_ROOK *
—	CLAHEF_ROOK *
SLATBS	CLATBS
SLATDF	CLATDF
SLATPS	CLATPS
SLATRD	CLATRD
SLATRS	CLATRS
SLATRZ	CLATRZ

付表 10 LAPACK 補助ルーチン一覧(続く)

実数ルーチン	複素数ルーチン
SLAUU2	CLAUU2
SLAUUM	CLAUUM
SLAZQ3	—
SLAZQ4	—
SLA_GBAMV	CLA_GBAMV
SLA_GBRCOND	CLA_GBRCOND_C
—	CLA_GBRCOND_X
SLA_GBRFSX_EXTENDED	CLA_GBRFSX_EXTENDED
SLA_GRPVGRW	CLA_GRPVGRW
SLA_GEAMV	CLA_GEAMV
SLA_GERCOND	CLA_GERCOND_C
—	CLA_GERCOND_X
SLA_GERFSX_EXTENDED	CLA_GERFSX_EXTENDED
—	CLA_HEAMV
—	CLA_HERCOND_C
—	CLA_HERCOND_X
—	CLA_HERFSX_EXTENDED
—	CLA_HRPVGRW
SLA_LIN_BERR	CLA_LIN_BERR
SLA_PORCOND	CLA_PORCOND_C
—	CLA_PORCOND_X
SLA_PORFSX_EXTENDED	CLA_PORFSX_EXTENDED
SLA_PORPVGRW	CLA_PORPVGRW
SLA_RPVGRW	CLA_RPVGRW
SLA_SYAMV	CLA_SYAMV
SLA_SYRCOND	CLA_SYRCOND_C
—	CLA_SYRCOND_X
SLA_SYRFSX_EXTENDED	CLA_SYRFSX_EXTENDED
SLA_SYRPVGRW	CLA_SYRPVGRW
SLA_WWADDW	CLA_WWADDW
SORG2L	CUNG2L
SORG2R	CUNG2R
SORGL2	CUNGL2
SORGR2	CUNGR2
SORM2L	CUNM2L
SORM2R	CUNM2R
SORML2	CUNML2
SORMR2	CUNMR2
SORMR3	CUNMR3
SPBTF2	CPBTF2
SPOTF2	CPOTF2
SPTTS2	CPTTS2
SRSCL	CSRSC1
SSFRK	CHFRK
—	CHLA_TRANSTYPE

付表 10 LAPACK 補助ルーチン一覧(続き)

実数ルーチン	複素数ルーチン
SSYGS2	CHEGS2
SSYSWAPR ⁺	CSYSWAPR ⁺
—	CHESWAPR ⁺
SSYTD2	CHETD2
SSYTF2	CSYTF2
—	CHETF2
STFSM	CTFSM
STFTTP	CTFTTP
STFTTR	CTFTTR
STGEX2	CTGEX2
STGSY2	CTGSY2
STPRFB ⁺	CTPRFB ⁺
STPTTF	CTPTTF
STPTTR	CTPTTR
STRTI2	CTRDI2
STRRTF	CTRRTF
STRRTP	CTRRTP
XERBLA	—
XERBLA_ARRAY	—

C インタフェースの LAPACK ルーチン一覧を付表 11に示します.

付表11 C インタフェース LAPACK ルーチン一覧(続く)

実数ルーチン	複素数ルーチン	実数ルーチン	複素数ルーチン
LAPACKE_ilaver *	—	LAPACKE_sgeqrfp	LAPACKE_cgeqrfp
LAPACKE_dsgesv	LAPACKE_zgesv	LAPACKE_sgeqrt	LAPACKE_cgeqrt
LAPACKE_dsposv	LAPACKE_zposv	LAPACKE_sgeqrt2	LAPACKE_cgeqrt2
LAPACKE_dtpqrt	LAPACKE_ctpqrt	LAPACKE_sgeqrt3	LAPACKE_cgeqrt3
LAPACKE_sbbcsd	LAPACKE_cbbcsd	LAPACKE_sgerfs	LAPACKE_cgerfs
LAPACKE_sbdsdc	—	LAPACKE_sgerfsx	LAPACKE_cgerfsx
LAPACKE_sbdsqr	LAPACKE_cbdsqr	LAPACKE_sgerfqf	LAPACKE_cgerfqf
LAPACKE_sdisna	—	LAPACKE_sgesdd	LAPACKE_cgesdd
LAPACKE_sgbbrd	LAPACKE_cgbbrd	LAPACKE_sgesv	LAPACKE_cgesv
LAPACKE_sgbcon	LAPACKE_cgbcon	LAPACKE_sgesvd	LAPACKE_cgesvd
LAPACKE_sgbequ	LAPACKE_cgbequ	LAPACKE_sgesvj	—
LAPACKE_sgbequb	LAPACKE_cgbequb	LAPACKE_sgesvx	LAPACKE_cgesvx
LAPACKE_sgbrfs	LAPACKE_cgbrfs	LAPACKE_sgesvxx	LAPACKE_cgesvxx
LAPACKE_sgbrfsx	LAPACKE_cgbrfsx	LAPACKE_sgetf2	LAPACKE_cgetf2
LAPACKE_sgbsv	LAPACKE_cgbsv	LAPACKE_sgetrf	LAPACKE_cgetrf
LAPACKE_sgbsvx	LAPACKE_cgbsvx	LAPACKE_sgetri	LAPACKE_cgetri
LAPACKE_sgbsvxx	LAPACKE_cgbsvxx	LAPACKE_sgetrs	LAPACKE_cgetrs
LAPACKE_sgbtrf	LAPACKE_cgbtrf	LAPACKE_sggbak	LAPACKE_cggbak
LAPACKE_sgbtrs	LAPACKE_cgbtrs	LAPACKE_sggbal	LAPACKE_cggbal
LAPACKE_sgebak	LAPACKE_cgebak	LAPACKE_sgges	LAPACKE_cgges
LAPACKE_sgebal	LAPACKE_cgebal	LAPACKE_sggesx	LAPACKE_cggesx
LAPACKE_sgebrd	LAPACKE_cgebrd	LAPACKE_sggev	LAPACKE_cggev
LAPACKE_sgecon	LAPACKE_cgecon	LAPACKE_sggevxx	LAPACKE_cggevxx
LAPACKE_sgeequ	LAPACKE_cgeequ	LAPACKE_sgglm	LAPACKE_cggglm
LAPACKE_sgeequeb	LAPACKE_cgeequeb	LAPACKE_sgghrd	LAPACKE_cgghrd
LAPACKE_sgees	LAPACKE_cgees	LAPACKE_sgglse	LAPACKE_cgglse
LAPACKE_sgeesx	LAPACKE_cgeesx	LAPACKE_sgqrdf	LAPACKE_cggqrdf
LAPACKE_sgeev	LAPACKE_cgeev	LAPACKE_sggrqf	LAPACKE_cggrqf
LAPACKE_sgeevx	LAPACKE_cgeevx	LAPACKE_sggsvd	LAPACKE_cggsvd
LAPACKE_sgehrd	LAPACKE_cgehrd	LAPACKE_sggsvp	LAPACKE_cggsvp
LAPACKE_sgejsv	—	LAPACKE_sgtcon	LAPACKE_cgtcon
LAPACKE_sgelsq2	LAPACKE_cgelsq2	LAPACKE_sgtrfs	LAPACKE_cgtrfs
LAPACKE_sgelfqf	LAPACKE_cgelfqf	LAPACKE_sgtsv	LAPACKE_cgtsv
LAPACKE_sgels	LAPACKE_cgels	LAPACKE_sgtsvx	LAPACKE_cgtsvx
LAPACKE_sgelsd	LAPACKE_cgelsd	LAPACKE_sgttrf	LAPACKE_cgtrrf
LAPACKE_sgellss	LAPACKE_cgellss	LAPACKE_sgttrs	LAPACKE_cgtrrs
LAPACKE_sgelsy	LAPACKE_cgelsy	—	LAPACKE_chfrk
LAPACKE_sgempqr	LAPACKE_cgempqr	LAPACKE_shgeqz	LAPACKE_chgeqz
LAPACKE_sgeqlf	LAPACKE_cgeqlf	LAPACKE_shsein	—
LAPACKE_sgeqp3	LAPACKE_cgeqp3	LAPACKE_shseqr	LAPACKE_chseqr
LAPACKE_sgeqpff	LAPACKE_cgeqpff	—	LAPACKE_chsein
LAPACKE_sgeqr2	LAPACKE_cgeqr2	—	LAPACKE_clacgv
LAPACKE_sgeqrff	LAPACKE_cgeqrff	—	—

付表 11 C インタフェース LAPACK ルーチン一覧(続く)

実数ルーチン	複素数ルーチン	実数ルーチン	複素数ルーチン
LAPACKE_slacpy	LAPACKE_clacpy	LAPACKE_spbstf	LAPACKE_cpbstf
LAPACKE_slag2d	LAPACKE_clag2z	LAPACKE_spbsv	LAPACKE_cpbsv
LAPACKE_slamch	—	LAPACKE_spbsvx	LAPACKE_cpbsvx
LAPACKE_s lange	LAPACKE_clange	LAPACKE_spbtrf	LAPACKE_cpbtrf
LAPACKE_s lansy	LAPACKE_clansy	LAPACKE_spbtrs	LAPACKE_cpbtrs
—	LAPACKE_clanhe	LAPACKE_spftrf	LAPACKE_cpftrf
LAPACKE_s lantr	LAPACKE_clantr	LAPACKE_spftri	LAPACKE_cpftri
LAPACKE_s lapmr	LAPACKE_clapmr	LAPACKE_spftrs	LAPACKE_cpftrs
LAPACKE_s lapy2	—	LAPACKE_spocon	LAPACKE_cpocon
LAPACKE_s lapy3	—	LAPACKE_spoequ	LAPACKE_cpoequ
LAPACKE_s larfb	LAPACKE_clarfb	LAPACKE_spoequb	LAPACKE_cpoequb
LAPACKE_s larfg	LAPACKE_clarfg	LAPACKE_sporfs	LAPACKE_cporfs
LAPACKE_s larft	LAPACKE_clarft	LAPACKE_sporfsx	LAPACKE_cporfsx
LAPACKE_s larfx	LAPACKE_clarfx	LAPACKE_sposv	LAPACKE_cposv
LAPACKE_s larnv	LAPACKE_clarnv	LAPACKE_sposvx	LAPACKE_cposvx
LAPACKE_s lartgp	—	LAPACKE_sposvxx	LAPACKE_cposvxx
LAPACKE_s lartgs	—	LAPACKE_spotrf	LAPACKE_cpotrf
LAPACKE_s laset	LAPACKE_claset	LAPACKE_spotri	LAPACKE_cpotri
LAPACKE_s lasrt	—	LAPACKE_spotrs	LAPACKE_cpotrs
LAPACKE_s laswp	LAPACKE_claswp	LAPACKE_sppcon	LAPACKE_cppcon
LAPACKE_s lauum	LAPACKE_clauum	LAPACKE_sppequ	LAPACKE_cppequ
LAPACKE_s opgr	—	LAPACKE_spprfs	LAPACKE_cpprfs
LAPACKE_s opmtr	—	LAPACKE_sppsv	LAPACKE_cppsv
LAPACKE_s orbdb	LAPACKE_cunbdb	LAPACKE_sppsvx	LAPACKE_cppsvx
LAPACKE_s oresd	LAPACKE_cuncsd	LAPACKE_spptrf	LAPACKE_cpptrf
LAPACKE_s orgbr	LAPACKE_cungbr	LAPACKE_spptri	LAPACKE_cpptri
LAPACKE_s orghr	LAPACKE_cunghr	LAPACKE_spptrs	LAPACKE_cpptrs
LAPACKE_s orglq	LAPACKE_cunglq	LAPACKE_spstrf	LAPACKE_cpstrf
LAPACKE_s orgql	LAPACKE_cungql	LAPACKE_sptcon	LAPACKE_cptcon
LAPACKE_s orgqr	LAPACKE_cungqr	LAPACKE_spteqr	LAPACKE_cpteqr
LAPACKE_s orgrq	LAPACKE_cungrq	LAPACKE_sptrfs	LAPACKE_cptrfs
LAPACKE_s orgtr	LAPACKE_cungtr	LAPACKE_sptsv	LAPACKE_cptsv
LAPACKE_sormbr	LAPACKE_cunmbr	LAPACKE_sptsvx	LAPACKE_cptsvx
LAPACKE_sormhr	LAPACKE_cunmhr	LAPACKE_spttrf	LAPACKE_cpttrf
LAPACKE_sormlq	LAPACKE_cunmlq	LAPACKE_spttrs	LAPACKE_cpttrs
LAPACKE_sormql	LAPACKE_cunmql	LAPACKE_ssbev	LAPACKE_chbev
LAPACKE_sormqr	LAPACKE_cunmqr	LAPACKE_ssbevd	LAPACKE_chbevd
LAPACKE_sormrq	LAPACKE_cunmrq	LAPACKE_ssbevx	LAPACKE_chbevx
LAPACKE_sormrz	LAPACKE_cunmrz	LAPACKE_ssbgst	LAPACKE_chbgst
LAPACKE_sormtr	LAPACKE_cunmtr	LAPACKE_ssbgv	LAPACKE_chbgv
LAPACKE_spbcon	LAPACKE_cpbcon	LAPACKE_ssbgvd	LAPACKE_chbgvd
LAPACKE_spbequ	LAPACKE_cpbequ	LAPACKE_ssbgvx	LAPACKE_chbgvx
LAPACKE_spbrfs	LAPACKE_cpbrfs	LAPACKE_ssbt rd	LAPACKE_chbtrd

付表 11 C インタフェース LAPACK ルーチン一覧(続く)

実数ルーチン	複素数ルーチン	実数ルーチン	複素数ルーチン
LAPACKE_ssfrk	—	LAPACKE_ssygst	LAPACKE_chegst
LAPACKE_sspcon	LAPACKE_cspcon	LAPACKE_ssygv	LAPACKE_chegv
—	LAPACKE_chpcon	LAPACKE_ssygvd	LAPACKE_chegvd
LAPACKE_sspev	LAPACKE_chpev	LAPACKE_ssygvx	LAPACKE_chegvx
LAPACKE_sspevd	LAPACKE_chpevd	—	LAPACKE_csyr
LAPACKE_sspevx	LAPACKE_chpevx	LAPACKE_ssyrfs	LAPACKE_csyrf
LAPACKE_sspdst	LAPACKE_chpgst	—	LAPACKE_cherfs
LAPACKE_ssppgv	LAPACKE_chpgv	LAPACKE_ssyrfsx	LAPACKE_csyrf
LAPACKE_ssppgd	LAPACKE_chpgvd	—	LAPACKE_cherfsx
LAPACKE_ssppgvx	LAPACKE_chpgvx	LAPACKE_ssysv	LAPACKE_csysv
LAPACKE_ssprfs	LAPACKE_csprfs	—	LAPACKE_chesv
—	LAPACKE_chprfs	LAPACKE_ssysv_rook *	LAPACKE_csysv_rook *
LAPACKE_sspsv	LAPACKE_cspsv	LAPACKE_ssysvx	LAPACKE_csysvx
—	LAPACKE_chpsv	—	LAPACKE_chesvx
LAPACKE_sspsvx	LAPACKE_cspsvx	LAPACKE_ssysvxx	LAPACKE_csysvxx
—	LAPACKE_chpsvx	—	LAPACKE_chesvxx
LAPACKE_ssptrd	LAPACKE_chptrd	LAPACKE_ssystapr	LAPACKE_csyswapr
LAPACKE_ssptrf	LAPACKE_csptrf	—	LAPACKE_cheswapr
—	LAPACKE_chptrf	LAPACKE_ssytrd	LAPACKE_chetrd
LAPACKE_ssptri	LAPACKE_csptri	LAPACKE_ssytrf	LAPACKE_csytrf
—	LAPACKE_chptri	—	LAPACKE_chetrf
LAPACKE_ssptrs	LAPACKE_csptrs	LAPACKE_ssytri	LAPACKE_csytri
—	LAPACKE_chptrs	—	LAPACKE_chetri
LAPACKE_sstebz	—	LAPACKE_ssytri2	LAPACKE_csytri2
LAPACKE_sstedc	LAPACKE_cstedc	—	LAPACKE_chetri2
LAPACKE_ssteigr	LAPACKE_csteigr	LAPACKE_ssytri2x	LAPACKE_csytri2x
LAPACKE_sstein	LAPACKE_cstein	—	LAPACKE_chetri2x
LAPACKE_sstemr	LAPACKE_cstemr	LAPACKE_ssytrs	LAPACKE_csytrs
LAPACKE_ssteqr	LAPACKE_csteqr	—	LAPACKE_chetrs
LAPACKE_ssterf	—	LAPACKE_ssytrs2	LAPACKE_csytrs2
LAPACKE_sstev	—	—	LAPACKE_chetrs2
LAPACKE_sstevd	—	LAPACKE_stbcon	LAPACKE_ctbcon
LAPACKE_sstevr	—	LAPACKE_stbrfs	LAPACKE_ctbrfs
LAPACKE_sstevx	—	LAPACKE_stbtrs	LAPACKE_ctbtrs
LAPACKE_ssycon	LAPACKE_csycon	LAPACKE_stfsm	LAPACKE_ctfsm
—	LAPACKE_checon	LAPACKE_stftri	LAPACKE_ctftri
LAPACKE_ssyconv	LAPACKE_csyconv	LAPACKE_stftfp	LAPACKE_ctftfp
LAPACKE_ssyequb	LAPACKE_csyequb	LAPACKE_stfttr	LAPACKE_ctfttr
—	LAPACKE_cheequb	LAPACKE_stgevc	LAPACKE_ctgevc
LAPACKE_ssyev	LAPACKE_cheev	LAPACKE_stgexc	LAPACKE_ctgexc
LAPACKE_ssyevd	LAPACKE_cheevd	LAPACKE_stgsen	LAPACKE_ctgsen
LAPACKE_ssyevr	LAPACKE_cheevr	LAPACKE_stgsja	LAPACKE_ctgsja
LAPACKE_ssyevx	LAPACKE_cheevx	LAPACKE_stgsna	LAPACKE_ctgsna
		LAPACKE_stgtsy	LAPACKE_ctgtsy

付表 11 C インタフェース LAPACK ルーチン一覧(続き)

実数ルーチン	複素数ルーチン	実数ルーチン	複素数ルーチン
LAPACKE_stpcon	LAPACKE_ctpcon	LAPACKE_strrfs	LAPACKE_ctrrfs
LAPACKE_stpmqrt	LAPACKE_ctpmqrt	LAPACKE_strsen	LAPACKE_ctrsen
LAPACKE_stpqrt2	LAPACKE_ctpqrt2	LAPACKE_strsna	LAPACKE_ctrsna
LAPACKE_stprfb	LAPACKE_ctprfb	LAPACKE_strsyl	LAPACKE_ctrsyl
LAPACKE_stprfs	LAPACKE_ctprfs	LAPACKE_strtri	LAPACKE_ctrtri
LAPACKE_stptri	LAPACKE_ctptri	LAPACKE_strtrs	LAPACKE_ctrtrs
LAPACKE_stptrs	LAPACKE_ctptrs	LAPACKE_strttf	LAPACKE_ctrttf
LAPACKE_stpttf	LAPACKE_ctpttf	LAPACKE_strttp	LAPACKE_ctrttp
LAPACKE_stptrr	LAPACKE_ctptrr	LAPACKE_stzrzf	LAPACKE_ctzrzf
LAPACKE_strcon	LAPACKE_ctrcon	—	LAPACKE_cupgtr
LAPACKE_strevc	LAPACKE_ctrevc	—	LAPACKE_cupmtr
LAPACKE_strexc	LAPACKE_ctrexc		

PLASMA のルーチン一覧を付表 12～付表 13に示します。これらのルーチンは C と Fortran で共通です。ただし、ルーチン名の後に “*” がついているものは C 言語からのみ利用可能です。

付表12 PLASMA 簡易インタフェースルーチン一覧(続く)

実数ルーチン	複素数ルーチン
PLASMA_sgebrd	PLASMA_cgebrd
PLASMA_sgecfi	PLASMA_cgecfi
PLASMA_sgecfi_Async	PLASMA_cgecfi_Async
PLASMA_sgecon	PLASMA_cgecon
PLASMA_sgelfqf	PLASMA_cgelfqf
PLASMA_sgels	PLASMA_cgels
PLASMA_sgemm	PLASMA_cgemm
PLASMA_sgeqp3	PLASMA_cgeqp3
PLASMA_sgeqr	PLASMA_cgeqr
PLASMA_sgeqr	PLASMA_cgeqr
PLASMA_sgesdd	PLASMA_cgesdd
PLASMA_sgesv	PLASMA_cgesv
PLASMA_sgesv_incpiv	PLASMA_cgesv_incpiv
PLASMA_sgesvd	PLASMA_cgesvd
PLASMA_sgetmi	PLASMA_cgetmi
PLASMA_sgetmi_Async	PLASMA_cgetmi_Async
PLASMA_sgetrf	PLASMA_cgetrf
PLASMA_sgetrf_incpiv	PLASMA_cgetrf_incpiv
PLASMA_sgetrf_nopiv	PLASMA_cgetrf_nopiv
PLASMA_sgetrf_tntpiv	PLASMA_cgetrf_tntpiv
PLASMA_sgetri	PLASMA_cgetri
PLASMA_sgetrs	PLASMA_cgetrs

付表 12 PLASMA 簡易インタフェースルーチン一覧(続き)

実数ルーチン	複素数ルーチン
PLASMA_sgetrs_incpiv	PLASMA_cgetrs_incpiv
PLASMA_slacpy	PLASMA_clacpy
PLASMA_s lange	PLASMA_c lange
PLASMA_s lansy	PLASMA_c lansy
—	PLASMA_c lanhe
PLASMA_s lantr	PLASMA_c lantr
PLASMA_s laset	PLASMA_c laset
PLASMA_s laswp	PLASMA_c laswp
PLASMA_s laswpc	PLASMA_c laswpc
PLASMA_s lauum	PLASMA_c lauum
PLASMA_s orglq	PLASMA_c unglq
PLASMA_s orgqr	PLASMA_c ungqr
PLASMA_s ormlq	PLASMA_c unmlq
PLASMA_s ormqr	PLASMA_c unmqr
PLASMA_s plgsy	PLASMA_c plgsy
—	PLASMA_c plghe
PLASMA_s plrnt	PLASMA_c plrnt
PLASMA_s pltmg	PLASMA_c pltmg
PLASMA_s pocon	PLASMA_c pocon
PLASMA_s posv	PLASMA_c posv
PLASMA_s potrf	PLASMA_c potrf
PLASMA_s potri	PLASMA_c potri
PLASMA_s potrs	PLASMA_c potrs
PLASMA_ssye v	PLASMA_c heev
PLASMA_ssye vd	PLASMA_c heevd
PLASMA_ssye vr	PLASMA_c heevr
PLASMA_ssygst	PLASMA_c hegst
PLASMA_ssygv	PLASMA_c chegv
PLASMA_ssygvd	PLASMA_c chegvd
PLASMA_ss ymm	PLASMA_c symm
—	PLASMA_chemm
PLASMA_ssyr2k	PLASMA_c syr2k
—	PLASMA_cher2k
PLASMA_ssyrk	PLASMA_c syrk
—	PLASMA_cherk
PLASMA_ssytrd	PLASMA_chetrd
PLASMA_strmm	PLASMA_ctrmm
PLASMA_strsm	PLASMA_ct rsm
PLASMA_st rsmpl	PLASMA_ct rsmpl
PLASMA_st rsmrv	PLASMA_ct rsmrv
PLASMA_st rtri	PLASMA_ct rtri
PLASMA_d gesv	PLASMA_z gesv
PLASMA_d sposv	PLASMA_z posv
PLASMA_d sungen sv	PLASMA_z cungesv

付表13 PLASMA メインルーチン一覧

実数ルーチン	複素数ルーチン
PLASMA_Alloc_Workspace_sgebrd	PLASMA_Alloc_Workspace_cgebrd
PLASMA_Alloc_Workspace_sggev	PLASMA_Alloc_Workspace_cggev
PLASMA_Alloc_Workspace_sgehrд	PLASMA_Alloc_Workspace_cgehrд
PLASMA_Alloc_Workspace_sgels	PLASMA_Alloc_Workspace_cgels
PLASMA_Alloc_Workspace_sgeqrф	PLASMA_Alloc_Workspace_cgeqrф
PLASMA_Alloc_Workspace_sgelqf	PLASMA_Alloc_Workspace_cgelqf
PLASMA_Alloc_Workspace_sgesdd	PLASMA_Alloc_Workspace_cgesdd
PLASMA_Alloc_Workspace_sgесv_incpiv	PLASMA_Alloc_Workspace_cгесv_incpiv
PLASMA_Alloc_Workspace_sgесvд	PLASMA_Alloc_Workspace_cгесvд
PLASMA_Alloc_Workspace_sgетrf_incpiv	PLASMA_Alloc_Workspace_cгетrf_incpiv
PLASMA_Alloc_Workspace_ssуev	PLASMA_Alloc_Workspace_chеev
PLASMA_Alloc_Workspace_ssуevд	PLASMA_Alloc_Workspace_chеevд
PLASMA_Alloc_Workspace_ssуevr	PLASMA_Alloc_Workspace_chеevr
PLASMA_Alloc_Workspace_ssуgv	PLASMA_Alloc_Workspace_chевg
PLASMA_Alloc_Workspace_ssуgvд	PLASMA_Alloc_Workspace_chевgд
PLASMA_Alloc_Workspace_ssуtrd	PLASMA_Alloc_Workspace_chetrд
PLASMA_Dealloc_Handle	—
PLASMA_Desc_Create	—
PLASMA_Desc_Destroy	—
PLASMA_Disable	—
PLASMA_Enable	—
PLASMA_Finalize	—
PLASMA_Get	—
PLASMA_Init	—
PLASMA_Init_Affinity *	—
PLASMA_Sequence_Create *	—
PLASMA_Sequence_Destroy *	—
PLASMA_Sequence_Flush *	—
PLASMA_Sequence_Wait *	—
PLASMA_Set	—
PLASMA_Version	—

付表 14に示す旧ルーチンは, LAPACK version 3.0 および LAPACK version 3.2.2 より同様の機能を持つ最新のルーチンに置き換えられました。旧ルーチンは互換のためにライブラリに含めていますが、新ルーチンの使用が推奨されています。

付表14 互換のため含まれている旧ルーチン

旧ルーチン	新ルーチン	旧ルーチン	新ルーチン
#GEGS	#GGES	#TZRQF	#TZRZF
#GEGV	#GGEV	#LATZM	#ORMRZ, #UNMRZ
#GELSX	#GELSY	#LAZQ3 *	#LASQ3 *
#GEQPF	#GEQP3	#LAZQ4 *	#LASQ4 *

付表 15および付表 16に示すルーチンは, Netlib 版 LAPACK version 3.1.1 および LAPACK version 3.2.2 よりインタフェースが変更になっています。これらのルーチンを直接 call しているプログラムは新しいインタフェースを確認し、プログラムを修正してください。

付表15 インタフェースが変更になったルーチン(LAPACK version 3.1.1)

実数ルーチン	複素数ルーチン	実数ルーチン	複素数ルーチン
SLAR1V	—	SLARRF	—
SLARRB	—	SLARRV	—
SLARRE	—		

付表16 インタフェースが変更になったルーチン(LAPACK version 3.2)

実数ルーチン	複素数ルーチン
—	ZCGESV
SLASQ3	—
SLASQ4	—

A.3 ScaLAPACK

ScaLAPACK ライブラリに含まれているルーチンの一覧を、ScaLAPACK, PBLAS, BLACS の各レイヤーごとに示します。

なお、ルーチン名の後に “+” が付いているものは、Netlib 版 ScaLAPACK が version 1.7 から version 1.8 になったときに新規に追加されたルーチンを意味します。

また、ルーチン名の後に “*” が付いているものは、Netlib 版 ScaLAPACK が version 1.8 から version 2.0 になったときに新規に追加されたルーチンを意味します。

A.3.1 ScaLAPACK

ScaLAPACK で使用可能なサブルーチン一覧を付表 17～付表 18 に示します。

表に示したルーチン名は单精度実数及び单精度複素数ルーチンの名前です。倍精度実数のルーチン名は单精度実数ルーチン名の 2 文字目または先頭の'S'を'D'に置き換え、倍精度複素数のルーチン名は单精度複素数のルーチン名の 2 文字目または先頭の'C'を'Z'に置き換えてください。

補助ルーチンのうち PXERBLA はデータの型に依存せず、ただ 1 つだけ存在するルーチンです。

付表17 ScaLAPACK ドライバ及び計算用ルーチン一覧(続く)

実数ルーチン	複素数ルーチン	実数ルーチン	複素数ルーチン
BSTREXC *	—	PSGETRF	PCGETRF
PSDBSV	PCDBSV	PSGETRI	PCGETRI
PSDBTRF	PCDBTRF	PSGETRS	PCGETRS
PSDBTRS	PCDBTRS	PSGGQRF	PCGGQRF
PSDTSV	PCDTSV	PSGGRQF	PCGGRQF
PSDTTRF	PCDTTRF	PSHSEQR *	—
PSDTTRS	PCDTTRS	PSORGLQ	PCUNGLQ
PSGBSV	PCGBSV	PSORGQL	PCUNGQL
PSGBTRF	PCGBTRF	PSORGQR	PCUNGQR
PSGBTRS	PCGBTRS	PSORGRRQ	PCUNGRQ
PSGEBAL *	—	PSORMBR	PCUNMBR
PSGEBRD	PCGEFRD	PSORMHR	PCUNMHR
PSGECON	PCGECON	PSORMLQ	PCUNMLQ
PSGEEQU	PCGEEQU	PSORMQL	PCUNMQL
PSGEHRD	PCGEHRD	PSORMQR	PCUNMQR
PSGELQF	PCGELQF	PSORMRQ	PCUNMRQ
PSGELS	PCGELS	PSORMRZ	PCUNMRZ
PSGEQLF	PCGEQLF	PSORMTR	PCUNMTR
PSGEQPF	PCGEQPF	PSPBSV	PCPBSV
PSGEQRF	PCGEQRF	PSPBTRF	PCPBTRF
PSGERFS	PCGERFS	PSPBTRS	PCPBTRS
PSGERQF	PCGERQF	PSPOCON	PCPOCON
PSGESV	PCGESV	PSPOEQU	PCPOEQU
PSGESVD	PCGESVD ⁺	PSPORFS	PCPORFS
PSGESVX	PCGESVX	PSPOSV	PCPOSV

付表 17 ScaLAPACK ドライバ及び計算用ルーチン一覧(続き)

実数ルーチン	複素数ルーチン	実数ルーチン	複素数ルーチン
PSPOSVX	PCPOSVX	PSSYGST	PCHEGST
PSPOTRF	PCPOTRF	PSSYGVX	PCHEGVX
PSPOTRI	PCPOTRI	PSSYNTRD	PCHENTRD
PSPOTRS	PCPOTRS	PSSYNGST	PCHENGST
PSPTSV	PCPTSV	PSSYTRD	PCHETRD
PSPTTRF	PCPTTRF	PSSYTTRD	PCHETTRD
PSPTTRS	PCPTTRS	PSTRCON	PCTRCON
PSSTEBZ	—	—	PCTREVC
PSSTEDC	—	PSTRORD *	—
PSSTEIN	PCSTEIN	PSTRRFS	PCTRDFS
PSSYEV	—	PSTRSEN *	—
PSSYEVD	PCHEEVD	PSTRTRI	PCTRTRI
PSSYEVR *	PCHEEVR *	PSTRTRS	PCTRTRS
PSSYEVX	PCHEEVX	PSTZRZF	PCTZRZF

付表18 ScaLAPACK 補助ルーチン一覧(続く)

実数ルーチン	複素数ルーチン	実数ルーチン	複素数ルーチン
BSLAAPP *	—	PSLACP3	PCLACP3
BSLAEXC *	—	PSLACPY	PCLACPY
—	PCLACGV	PSLAED0	—
—	PCMAX1	PSLAED1	—
—	PSCSUM1	PSLAED2	—
PILAENVX *	—	PSLAED3	—
PILAVER *	—	PSLAEDZ	—
PIPARMQ *	—	PSLAEVSWP	PCLAEVSWP
PMPCOL *	—	PSLAHQQR	PCLAHQR
PMPIIM2 *	—	PSLAHRD	PCLAHRD
PSDBTRSV	PCDBTRSV	PSLAIECT (注 1)	—
PSDTTRSV	PCDTTRSV	PSLAMCH	—
PSGEBD2	PCGEBD2	PSLAMR1D	PCLAMR1D
PSGEHD2	PCGEHD2	PSLAMVE *	—
PSGELQ2	PCGELQ2	PSLANGE	PCLANGE
PSGEQL2	PCGEQL2	PSLANHS	PCLANHS
PSGEQR2	PCGEQR2	PSLANSY	PCLANSY
PSGERQ2	PCGERQ2	—	PCLANHE
PSGETF2	PCGETF2	PSLANTR	PCLANTR
PSLABAD	—	PSLAPIV	PCLAPIV
PSLABRD	PCLABRD	PSLAQGE	PCLAQGE
PSLACHKIEEE	—	PSLAQR0 *	—
PSLACON	PCLACON	PSLAQR1 *	—
PSLACONSB	PCLACONSB	PSLAQR2 *	—
PSLACP2	PCLACP2	PSLAQR3 *	—

付表 18 ScaLAPACK 補助ルーチン一覧(続き)

実数ルーチン	複素数ルーチン	実数ルーチン	複素数ルーチン
PSLAQR4 *	—	PSORML2	PCUNML2
PSLAQR5 *	—	PSORMR2	PCUNMR2
PSLAQSY	PCLAQSY	PSPBTRSV	PCPBTRSV
PSLARED1D	—	PSPTTRSV	PCPTTRSV
PSLARED2D	—	PSPOTF2	PCPOTF2
PSLARF	PCLARF	PSROT *	—
PSLARFB	PCLARFB	PSRSCL	PCSRSCl(注 2)
—	PCLARFC	PSSYGS2	PCHEGS2
PSLARFG	PCLARFG	PSSYTD2	PCHETD2
PSLARFT	PCLARFT	PSTRTI2	PCTRTI2
PSLARZ	PCLARZ	PXERBLA	—
PSLARZB	PCLARZB	SDBTF2	CDBTF2
—	PCLARZC	SDBTRF	CDBTRF
PSLARZT	PCLARZT	SDTTRF	CDTTRF
PSLASCL	PCLASCL	SDTTRSV	CDTTRSV
PSLASSET	PCLASSET	SLAMOV *	CLAMOV *
PSLASMSUB	PCLASMSUB	SLAMSH	—
PSLASNBT	—	SLAQR6 *	—
PSLASSQ	PCLASSQ	SLAR1VA *	—
PSLASRT	—	SLAREF	—
PSLASWP	PCLASWP	SLARRB2 *	—
PSLATRA	PCLATRA	SLARRD2 *	—
PSLATRD	PCLATRD	SLARRE2 *	—
PSLATRS	PCLATRS	SLARRE2A *	—
PSLATRZ	PCLATRZ	SLARRF2 *	—
—	PCLATTRS	SLARRV2 *	—
PSLAUU2	PCLAUU2	SLASORTE	—
PSLAUUM	PCLAUUM	SLASRT2	—
PSLAWIL	PCLAWIL	SPTTRSV	CPTTRSV
PSORG2L	PCUNG2L	SSTEGR2 *	—
PSORG2R	PCUNG2R	SSTEGR2A *	—
PSORGL2	PCUNGL2	SSTEGR2B *	—
PSORGR2	PCUNGR2	SSTEIN2	—
PSORM2L	PCUNM2L	SSTEQR2	—
PSORM2R	PCUNM2R		

注 1) 倍精度のルーチン名は PDIAIECTB

注 2) 倍精度複素数のルーチン名は PZDRSCL

A.3.2 PBLAS

ScaLAPACK で使用可能な PBLAS ルーチンの一覧を付表 19～付表 21 に示します。

「ルーチン名」の#の部分には、データ型及び演算精度を示す文字が入ります。各文字は以下の意味を持ちます。

- S : 単精度実数型
- D : 倍精度実数型
- C : 単精度複素数型
- Z : 倍精度複素数型

また、精度の組み合わせは複数の精度を使用することを意味します。たとえば、PSCNRM2 の'SC'は単精度複素数の入力に対し、単精度の実数を返す関数であることを示しています。

付表19 Level 1 PBLAS ルーチン一覧

ルーチン名	サポートする精度
P#SWAP	S, D, C, Z
P#SCAL	S, D, C, Z, CS, ZD
P#COPY	S, D, C, Z
P#AXPY	S, D, C, Z
P#DOT	S, D
P#DOTU	C, Z
P#DOTC	C, Z
P#NRM2	S, D, SC, DZ
P#ASUM	S, D, SC, DZ
P#AMAX	S, D, C, Z

付表20 Level 2 PBLAS ルーチン一覧

ルーチン名	サポートする精度
P#GEMV	S, D, C, Z
P#HEMV	C, Z
P#SYMV	S, D
P#TRMV	S, D, C, Z
P#TRSV	S, D, C, Z
P#GER	S, D
P#GERU	C, Z
P#GERC	C, Z
P#HER	C, Z
P#HER2	C, Z
P#SYR	S, D
P#SYR2	S, D

付表21 Level 3 PBLAS ルーチン一覧

ルーチン名	サポートする精度
P#GEMM	S, D, C, Z
P#SYMM	S, D, C, Z
P#HEMM	C, Z
P#SYRK	S, D, C, Z
P#HERK	C, Z
P#SYR2K	S, D, C, Z
P#HER2K	C, Z
P#TRAN	S, D
P#TRANU	C, Z
P#TRANC	C, Z
P#TRMM	S, D, C, Z
P#TRSM	S, D, C, Z

A.3.3 BLACS

ScaLAPACK で使用可能な BLACS ルーチンの一覧を付表 22 に示します。

「ルーチン名」の#の部分には、データ型及び演算精度を示す文字が入ります。各文字は以下の意味を持ちます。

- S : 単精度実数型
- D : 倍精度実数型
- C : 単精度複素数型
- Z : 倍精度複素数型
- I : 整数型

付表22 BLACS ルーチン一覧(続く)

内容	ルーチン名	サポートする精度
初期化処理	BLACS_PINFO BLACS_SETUP BLACS_GET BLACS_SET BLACS_GRIDINIT BLACS_GRIDMAP	
終了処理	BLACS_FREEBUFF BLACS_GRIDEXIT BLACS_ABORT BLACS_EXIT	
送信	#GESD2D #GEBS2D #TRSD2D #TRBS2D	S, D, C, Z, I

付表 22 BLACS ルーチン一覧(続き)

内容	ルーチン名	サポートする精度
受信	#GERV2D #GEBR2D #TRRV2D #TRBR2D	S, D, C, Z, I
演算	#GAMX2D #GAMN2D #GSUM2D	S, D, C, Z, I
情報通知・その他	BLACS_GRIDINFO BLACS_PNUM BLACS_PCOORD BLACS_BARRIER DCPUTIME00 DWALLTIME00 KSENDID KRECVID KBSID KBRID blacs2sys_handle sys2blacs_handle free_blacs_system_handle	

A.3.4 ツールルーチン

ScalAPACK ではプロセスグリッドの初期化や、分割された行列の操作のためにいくつかのツールルーチンが用意されています。ツールルーチンの一覧を付表 23 に示します。

「ルーチン名」の#の部分には、データ型及び演算精度を示す文字が入ります。各文字は以下の意味を持ちます。

- S : 単精度実数型
- D : 倍精度実数型
- C : 単精度複素数型
- Z : 倍精度複素数型
- I : 整数型

付表23 ツールルーチン

ルーチン名	サポートする精度	内容
SL_INIT	—	プロセスグリッドの初期化
DESCINIT	—	ディスクリプタの初期化
P#GEMR2D P#TRMR2D	S, D, C, Z, I	行列の分割方法を変更します。
P#LAREAD	S, D, C, Z	ファイルから行列を読み込み、各プロセスに配分します。
P#LAWRITE	S, D, C, Z	分割された行列をファイルに書き出します。

A.4 BLAS の拡張機能

本製品には Netlib で公開されている BLAS ルーチン以外に、本製品固有の BLAS 機能を拡張したルーチンがあります。

A.4.1 半精度版 BLAS ルーチン

本製品では一部の BLAS ルーチンについて半精度版ルーチンを提供しています。引数の並びは従来の単・倍精度ルーチンと同じで、実数型の引数が半精度実数型になります。提供しているルーチンを付表 24 に示します。

付表24 REAL*2 BLAS ルーチン一覧

	Fortran BLAS ルーチン名	CBLAS ルーチン名	機能	並列 版
Level 1	FJBLAS_SWAP_R16	fjblas_swap_r16	$x \leftrightarrow y$	○
	FJBLAS_SCAL_R16	fjblas_scal_r16	$x \leftarrow ax$	○
	FJBLAS_COPY_R16	fjblas_copy_r16	$y \leftarrow x$	○
	FJBLAS_AXPY_R16	fjblas_axpy_r16	$y \leftarrow ax + y$	○
	FJBLAS_DOT_R16	fjblas_dot_r16	$\text{dot} \leftarrow x^T y$	○
	FJBLAS_ASUM_R16	fjblas_asum_r16	$\text{asum} \leftarrow \ \text{re}(x)\ _1 + \ \text{im}(x)\ _1$	○
	FJBLAS_AMAX_I32_R16	fjblas_amax_i32_r16	$\text{amax} \leftarrow \text{argmax } x_k$	○
Level 2	FJBLAS_GEMV_R16	fjblas_gemv_r16	$y \leftarrow aAx + by$	○
	FJBLAS_GER_R16	fjblas_ger_r16	$A \leftarrow axy^T + A$	○
Level 3	FJBLAS_GEMM_R16	fjblas_gemm_r16	$C \leftarrow aAB + bC$	○

A.4.2 スレッド並列ライブラリ内の逐次 BLAS ルーチン

本製品ではスレッド並列版 BLAS 等のルーチンを使用しているプログラムの中でも、逐次版 BLAS ルーチンを使用することができます。例えばスレッド並列版ライブラリを結合し、スレッド並列版 BLAS, LAPACK ルーチンを使用しているプログラムの中で、OpenMP で並列化した部分の中で BLAS を使用する場合などです。逐次専用ルーチンはスレッド並列化されているルーチンだけでなく、スレッド並列化されていないルーチンについても提供しています。

このような逐次専用ルーチンは 1 つのプログラムの中で呼び分けることができるよう従来の BLAS とはルーチン名が異なります。

Fortran BLAS の单精度実数、倍精度実数、单精度複素数、倍精度複素数ルーチンの逐次専用ルーチンは、従来の BLAS ルーチンのルーチン名の 2 文字目と 3 文字目に L_ を入れたものになります。引数の並びは従来 BLAS と同じです。

例 1)

```
CALL DGEMM(TRANSA, TRANSB, M, N, K, ...)
```

```
CALL DL_GEMM(TRANSA, TRANSB, M, N, K, ...) ! 逐次専用ルーチン
```

例 2)

```
I=IDAMAX(N, DX, INCX)
```

```
I=IL_DAMAX(N, DX, INCX) ! 逐次専用ルーチン
```

Fortran BLAS の半精度ルーチンの場合はルーチン名の先頭に SS_をつけます.

例 3)

```
CALL FJBLAS_GEMM_R16(TRANSA, TRANSB, M, N, K, ...)
```

```
CALL SS_FJBLAS_GEMM_R16(TRANSA, TRANSB, M, N, K, ...) // 逐次専用ルーチン
```

CBLAS では、先頭に型を表す文字(cblas_の次の文字)に続けて l_をつけます.

例 4)

```
cblas_dgemm(layout, transa, transb, ...)
```

```
dl_cblas_dgemm(layout, transa, transb, ...) // 逐次専用ルーチン
```

例 5)

```
cblas_idamax(n, x, incx)
```

```
il_cblas_idamax(n, x, incx) // 逐次専用ルーチン
```

CBLAS の半精度ルーチンおよび行列のコピー・転置の場合は、ルーチン名の先頭に ss_をつけます.

例 6)

```
fjblas_gemm_r16(layout, transa, transb, ...)
```

```
ss_fjblas_gemm_r16(layout, transa, transb, ...) // 逐次専用ルーチン
```

逐次専用 BLAS ルーチンの一覧を付表 25～付表 29に示します.

付表25 Level 1 BLAS ルーチン一覧

Fortran BLAS 逐次専用ルーチン名	CBLAS 逐次専用ルーチン名	サポートする精度 (CBLAS では小文字)
#L_ROTG	#l_cblas.#rotg	S, D
#L_ROTMG	#l_cblas.#rotmg	S, D
#L_ROT	#l_cblas.#rot	S, D
#L_ROTM	#l_cblas.#rotm	S, D
#L_SWAP	#l_cblas.#swap	S, D, C, Z
#L_SCAL	#l_cblas.#scal	S, D, C, Z, CS, ZD
#L_COPY	#l_cblas.#copy	S, D, C, Z
#L_AXPY	#l_cblas.#axpy	S, D, C, Z
#L_DOT	#l_cblas.#dot	S, D, DS
#L_DOTU	#l_cblas.#dotu_sub	C, Z
#L_DOTC	#l_cblas.#dotc_sub	C, Z
#L##DOT	#l_cblas.###dot	SDS
#L_NRM2, #L_NRM2	#l_cblas.#nrm2, #l_cblas.##nrm2	S, D, SC, DZ
#L_ASUM, #L_ASUM	#l_cblas.#asum, #l_cblas.##asum	S, D, SC, DZ
IL_AMAX	#l_cblas.i#amax	S, D, C, Z

付表26 Level 2 BLAS ルーチン一覧

Fortran BLAS 逐次専用ルーチン名	CBLAS 逐次専用ルーチン名	サポートする精度 (CBLAS では小文字)
#L_GEMV	#l_cblas_#gemv	S, D, C, Z
#L_GBMV	#l_cblas_#gbmv	S, D, C, Z
#L_HEMV	#l_cblas_#hemv	C, Z
#L_HBMV	#l_cblas_#hbmv	C, Z
#L_HPMV	#l_cblas_#hpmv	C, Z
#L_SYMV	#l_cblas_#symv	S, D
#L_SBMV	#l_cblas_#sbmv	S, D
#L_SPMV	#l_cblas_#spmv	S, D
#L_TRMV	#l_cblas_#trmv	S, D, C, Z
#L_TBMV	#l_cblas_#tbmv	S, D, C, Z
#L TPMV	#l_cblas_#tpmv	S, D, C, Z
#L TRSV	#l_cblas_#trsv	S, D, C, Z
#L TBSV	#l_cblas_#tbsv	S, D, C, Z
#L TPSV	#l_cblas_#tpsv	S, D, C, Z
#L GER	#l_cblas_#ger	S, D
#L GERU	#l_cblas_#geru	C, Z
#L GERC	#l_cblas_#gerc	C, Z
#L HER	#l_cblas_#her	C, Z
#L HPR	#l_cblas_#hpr	C, Z
#L HER2	#l_cblas_#her2	C, Z
#L HPR2	#l_cblas_#hpr2	C, Z
#L SYR	#l_cblas_#syr	S, D
#L SPR	#l_cblas_#spr	S, D
#L SYR2	#l_cblas_#syr2	S, D
#L SPR2	#l_cblas_#spr2	S, D

付表27 Level 3 BLAS ルーチン一覧

Fortran BLAS 逐次専用ルーチン名	CBLAS 逐次専用ルーチン名	サポートする精度 (CBLAS では小文字)
#L_GEMM	#l_cblas_#gemm	S, D, C, Z
#L_SYMM	#l_cblas_#symm	S, D, C, Z
#L_HEMM	#l_cblas_#hemm	C, Z
#L_SYRK	#l_cblas_#syrk	S, D, C, Z
#L_HERK	#l_cblas_#herk	C, Z
#L_SYR2K	#l_cblas_#syr2k	S, D, C, Z
#L_HER2K	#l_cblas_#her2k	C, Z
#L_TRMM	#l_cblas_#trmm	S, D, C, Z
#L_TRSM	#l_cblas_#trsm	S, D, C, Z

付表28 REAL*2 BLAS 逐次専用ルーチン一覧

	Fortran BLAS ルーチン名	CBLAS ルーチン名
Level 1	SS_FJBLAS_SWAP_R16 SS_FJBLAS_SCAL_R16 SS_FJBLAS_COPY_R16 SS_FJBLAS_AXPY_R16 SS_FJBLAS_DOT_R16 SS_FJBLAS_ASUM_R16 SS_FJBLAS_AMAX_I32_R16	ss_fjblas_swap_r16 ss_fjblas_scal_r16 ss_fjblas_copy_r16 ss_fjblas_axpy_r16 ss_fjblas_dot_r16 ss_fjblas_asum_r16 ss_fjblas_amax_i32_r16
Level 2	SS_FJBLAS_GEMV_R16 SS_FJBLAS_GER_R16	ss_fjblas_gemv_r16 ss_fjblas_ger_r16
Level 3	SS_FJBLAS_GEMM_R16	ss_fjblas_gemm_r16

付表29 行列のコピー・転置逐次専用ルーチン一覧

	Fortran BLAS ルーチン名	CBLAS ルーチン名
	SS_FJBLAS_GEOCOPY_R16	ss_fjblas_geocopy_r16
	SS_FJBLAS_GEOCOPY_R32	ss_fjblas_geocopy_r32
	SS_FJBLAS_GEOCOPY_R64	ss_fjblas_geocopy_r64
	SS_FJBLAS_GEOCOPY_C32	ss_fjblas_geocopy_c32
	SS_FJBLAS_GEOCOPY_C64	ss_fjblas_geocopy_c64

A.4.3 行列のコピー・転置ルーチン

本製品では行列のコピーまたは転置を行うルーチンを提供します。

A.4.3.1 Fortran インタフェース

呼出し形式

```
call fjblas_geocopy_r16(order, trans, rows, cols, alpha, a, lda, b, ldb)
call fjblas_geocopy_r32(order, trans, rows, cols, alpha, a, lda, b, ldb)
call fjblas_geocopy_r64(order, trans, rows, cols, alpha, a, lda, b, ldb)
call fjblas_geocopy_c32(order, trans, rows, cols, alpha, a, lda, b, ldb)
call fjblas_geocopy_c64(order, trans, rows, cols, alpha, a, lda, b, ldb)
```

機能

2 次元配列に格納された行列のコピー、転置、共役コピーまたは共役転置を行います。このとき定数 alpha を掛けることができます。

$$B := \alpha * op(A)$$

入力配列と出力配列は重なりがってはいけません。

ルーチンと引数 alpha, a, b の型の対応は以下のとおりです。

fjblas_geocopy_r16 : real(2)

fjblas_geocopy_r32 : real(4)

```
fjblas_geocopy_r64 : real(8)
fjblas_geocopy_c32 : complex(4)
fjblas_geocopy_c64 : complex(8)
```

パラメタ

order	入力. character*1 行列の格納順を指定します. order='R' または 'r': row-major order='C' または 'c': column-major 一般的な Fortran の 2 次元配列の場合は'c'を指定してください.
trans	入力. character*1 a に対する操作を指定します. trans='N' または 'n': コピー trans='T' または 't': 転置 trans='R' または 'r': 共役コピー trans='C' または 'c': 共役転置
rows	入力. 整数型 b に格納する行列の行数.
cols	入力. 整数型 b に格納する行列の列数.
alpha	入力. 行列に掛ける定数.
a	入力. 2 次元配列 a(lda,*) コピー元の行列.
lda	入力. 整数型 配列 a の整合寸法
b	出力. 2 次元配列 b(lbd,*) コピー・転置した行列を格納します.
ldb	入力. 整数型 配列 b の整合寸法.

order, trans と a, b に格納される行列

order='R' または 'r'の場合 かつ trans='N', 'n', 'R' または'r'の場合,

alpha*a(1:cols,1:rows)をコピーして b(1:cols,1:rows)に格納します.

lda>=cols, ldb>=cols

order='R' または 'r'の場合 かつ trans='T', 't', 'C' または'c'の場合,

alpha*a(1:cols,1:rows)を転置して b(1:rows,1:cols)に格納します.

lda>=cols, ldb>=rows

order='C' または 'c'の場合 かつ trans='N', 'n', 'R' または'r'の場合,

alpha*a(1:rows,1:cols)をコピーして b(1:rows,1:cols)に格納します.

lda>=rows, ldb>=rows

order='C' または 'c'の場合 かつ trans='T', 't', 'C' または'c'の場合,

alpha*a(1:rows,1:cols)を転置して b(1:cols,1:rows)に格納します.

lda>=rows, ldb>=cols

A.4.3.2 C/C++ インタフェース

呼出し形式

```
#include "cblas.h"

fjcblas_geocopy_r16(order, trans, rows, cols, alpha, (__fp16 *)a, lda, (__fp16 *)b, ldb)
fjcblas_geocopy_r32(order, trans, rows, cols, alpha, (float *)a, lda, (float *)b, ldb)
fjcblas_geocopy_r64(order, trans, rows, cols, alpha, (double *)a, lda, (double *)b, ldb)
fjcblas_geocopy_c32(order, trans, rows, cols, alpha, (fcomplex *)a, lda, (fcomplex *)b, ldb)
fjcblas_geocopy_c64(order, trans, rows, cols, alpha, (dcomplex *)a, lda, (dcomplex *)b, ldb)

fcomplex, dcomplex は cblas.h の中で定義されている単精度複素数, 倍精度複素数の型です.
```

機能

2 次元配列に格納された行列のコピー, 転置, 共役コピーまたは共役転置を行います. このとき定数 alpha を掛けることができます.

$B := \alpha * op(A)$

入力配列と出力配列は重なりがってはいけません.

ルーチンと引数 alpha, a, b の型の対応は以下のとおりです.

```
fjcblas_geocopy_r16 : __fp16
fjcblas_geocopy_r32 : float
fjcblas_geocopy_r64 : double
fjcblas_geocopy_c32 : fcomplex
fjcblas_geocopy_c64 : dcomplex
```

パラメタ

order	入力. char 行列の格納順を指定します. order='R' または 'r': row-major order='C' または 'c': column-major
trans	入力. char a に対する操作を指定します. trans='N' または 'n': コピー trans='T' または 't': 転置 trans='R' または 'r': 共役コピー trans='C' または 'c': 共役転置
rows	入力. size_t b に格納する行列の行数.
cols	入力. size_t b に格納する行列の列数.
alpha	入力. 行列に掛ける定数.
a	入力. 2 次元配列 a[][][lda] コピー元の行列.
lda	入力. size_t 配列 a の整合寸法
b	出力. 2 次元配列 b[][][ldb] コピー・転置した行列を格納します.

ldb 入力. size_t
 配列 b の整合寸法.

order, trans と a, b に格納される行列

order='R' または 'r' の場合 かつ trans='N', 'n', 'R' または'r'の場合,
alpha*a[0...rows-1][0...cols-1]をコピーして b[0...rows-1][0...cols-1]に格納します.
lda>=cols, ldb>=cols

order='R' または 'r' の場合 かつ trans='T', 't', 'C' または'c'の場合,
alpha*a[0...rows-1][0...cols-1]を転置して b[0...cols-1][0...rows-1]に格納します.
lda>=cols, ldb>=rows

order='C' または 'c' の場合 かつ trans='N', 'n', 'R' または'r'の場合,
alpha*a[0...cols-1][0...rows-1]をコピーして b[0...cols-1][0...rows-1]に格納します.
lda>=rows, ldb>=rows

order='C' または 'c' の場合 かつ trans='T', 't', 'C' または'c'の場合,
alpha*a[0...cols-1][0...rows-1]を転置して b[0...rows-1][0...cols-1]に格納します.
lda>=rows, ldb>=cols

A.4.4 SIMD 長単位で格納した配列の複素行列行列積ルーチン

本製品では SIMD 長単位で実部と虚部を分離して配列に格納した複素行列の積を計算するルーチンを提供します。この機能は Armv8-A アーキテクチャの拡張である SVE を利用した機能であり、SVE をサポートしたシステムでのみ利用可能です。

ルーチンは Fortran から利用可能で、呼出し形式は以下のとおりです。

```
call fjblas_gemm_c32aososv(transa, transa, m, n, k, alpha, a, lda, b, ldb, beta, c, ldc)
call fjblas_gemm_c64aososv(transa, transb, m, n, k, alpha, a, lda, b, ldb, beta, c, ldc)
```

ルーチンと引数 alpha, beta, a, b, c の型の対応は以下のとおりです。

```
fjblas_gemm_c32aososv : complex(4)
fjblas_gemm_c64aososv : complex(8)
```

引数の並びは CGEMM/ZGEMM と同じですが、複素配列は以下のように SIMD 長単位で格納してください。引数の整合寸法には配列を複素数配列と見なしたときの整合寸法を渡してください。

```
! プロセッサがサポートしている SIMD 長を nvl とします
parameter(m=100, n=100, k=100, lda=m, ldb=k, ldc=m)
complex(8)::ca(lda,k), cb(ldb,n), cc(ldc,n)
real(8)::a(lda*2,k), b(ldb*2,n), c(ldc*2,n)
:
do ik=1,k
    ! SIMD 長単位で実部と虚部を格納します
    do im=1,m-nvl+1,nvl
        a(im*2-1:im*2+nvl-1, ik)=real(ca(im:im+nvl-1, ik))
        a(im*2+nvl-1:im*2+nvl*2-1, ik)=dimag(ca(im:im+nvl-1, ik))
```

```

end do
ims=im
nvr=m-ims+1
! 余り部分は詰めて格納します.
if(nvr>0) then
  a(ims*2-1:ims*2+nvr-1, ik)=real(ca(ims:ims+nvr-1,ik))
  a(ims*2+nvr-1:ims*2+nvr*2-1,ik)=dimag(ca(ims:ims+nvr-1,ik))
end if
end do
:
call fblas_gemm_c64aososv('n','n',m,n,k,alpha,a,lda,b,ldb,
-           beta,c,ldc)
end

```

付録 B ライセンス

B.1 XBLAS

Copyright (c) 2008-2009 The University of California Berkeley. All rights reserved.

\$COPYRIGHT\$

Additional copyrights may follow

\$HEADER\$

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer listed in this license in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

B.2 LAPACK

Copyright (c) 1992-2011 The University of Tennessee and The University
of Tennessee Research Foundation. All rights
reserved.

Copyright (c) 2000-2011 The University of California Berkeley. All
rights reserved.

Copyright (c) 2006-2012 The University of Colorado Denver. All rights
reserved.

\$COPYRIGHT\$

Additional copyrights may follow

\$HEADER\$

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

- Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer listed
in this license in the documentation and/or other materials
provided with the distribution.
- Neither the name of the copyright holders nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

The copyright holders provide no reassurances that the source code
provided does not infringe any patent, copyright, or any other
intellectual property rights of third parties. The copyright holders
disclaim any liability to any recipient for claims brought against
recipient by any third party for infringement of that parties

intellectual property rights.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

B.3 PLASMA

-- Innovative Computing Laboratory
-- Electrical Engineering and Computer Science Department
-- University of Tennessee
-- (C) Copyright 2008-2010

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the University of Tennessee, Knoxville nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,

SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

B.4 ScalAPACK

Copyright (c) 1992-2011 The University of Tennessee and The University of Tennessee Research Foundation. All rights reserved.

Copyright (c) 2000-2011 The University of California Berkeley. All rights reserved.

Copyright (c) 2006-2011 The University of Colorado Denver. All rights reserved.

\$COPYRIGHT\$

Additional copyrights may follow

\$HEADER\$

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer listed in this license in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

The copyright holders provide no reassurances that the source code

provided does not infringe any patent, copyright, or any other intellectual property rights of third parties. The copyright holders disclaim any liability to any recipient for claims brought against recipient by any third party for infringement of that parties intellectual property rights.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.