Lineární algebra v softwarových knihovnách

Lineární algebra

Úlohy

zde: řešení soustav lineárních algebraických rovnic
A . x = b, A čtvercová matice (N krát N prvků), x vektor řešení, b vektor pravé strany (oba N prvků)

jindy: vlastní čísla a vektory, (lineární) metoda nejmenších čtverců, metoda singulárního rozkladu

Časová složitost metod

	matice	metody	časová složitost	užitečnost
přímé metody				
	plná	faktorizace	$O(n^3)$	ano
	trojúhelníková	dopředná/zpětná substituce	$O(n^2)$	ano
	pásová (šířka pásu m)	faktorizace a substituce	$O(n.m^2)$	ano
	řídká (nnz_i nonzero prvků na ř.)	faktorizace a substituce	$O(\Sigma(nnz_i)^2)$	ano
iterační metody				
	řídká (nnz nonzero prvků)	Jacobiova/Gaussova-Seidelova	$O(nnz.n^2)$	málo
	řídká (nnz prvků)	superrelaxační (SOR)	$O(nnz.n^2)$ až $O(nnz.n)$	někdy
	řídká (nnz prvků)	krylovovské (CG/BCG, GMRES)	$O(nnz.n)$	ano
	řídká	multigridová (MG)	$O(nnz)$	ano

zde: faktorizace (LU dense, LU band, LU sparse) a krylovovské metody (BCG a GMRES) pro nesymetrickou matici

Knihovny a překladače

Přehled knihoven

low-level libraries	BLAS a LAPACK (Linear Algebra Package)	3.4.0
vendor libraries	Intel MKL (Math Kernel Library)	10.2/10.3
	ACML (AMD Core Math Library)	4.4.0/5.1.0
GPU libraries	CUBLAS a CULA Tools (CUDA-based LA)	4.0/4.1 a R13/R14, Sparse S1/S2
all-in-one libraries	IMSL (Int. Mathematics and Statistics Library)	FNL (Fortran Numerical Library) 6.0/7.0
	NAG (Numerical Algorithms Group)	FL (Fortran Library) Mark 23

zde: LAPACK, MKL, ACML, CULA a IMSL
další řešiče: Hypre, MAGMA (pro GPU), MUMPS, PARDISO (je v MKL), SPOOLES, SUPERLU, UMFPACK aj.

Knihovna BLAS (1979)
specifikace rozhraní pro základní operace mezi vektory a maticemi
obvykle volán prostřednictvím řešičů, přímo jen násobení gemv/gemm (general matrix-vector/matrix multiplication)
referenční fortranské zdrojové kódy: netlib.org
optimalizované implementace: Intel MKL, ACML, ATLAS (automatically tuned LA), (Kazushige) Goto BLAS aj.
implementace pro GPU: NVIDIA CUBLAS, včetně procedur pro přesun dat mezi pamětmi CPU a GPU

Překladače

zde: komerční Intel Fortran ifort 11.1/12.1, PGI (Portland) Fortran pgfortran 12.2/12.3
volně dostupné GNU gfortran 4.4/4.6, (one-man show) g95 0.93

Operační systémy

zde: Linux (Ubuntu 10.04 64bit), Windows (XP 32bit, Vista a 7 64bit)

Testovací úloha

Laplaceova a Poissonova rovnice ve 2D

$\Delta u(X) = f(X)$, $X = (x, y)$ nebo (r, θ)

Obdélníková oblast v kartézských souřadnicích nebo kruhová oblast v polárních souřadnicích
Dirichletovy, Neumannovy nebo periodické okrajové podmínky
Řešení hrubou silou: konečné diference 2. řádu (FD2) na síti J×K vnitřních bodů, j=1..J, k=1..K
Matice: 2D kartézsky – symetrická blokově třídiagonální (5 diagonál, na ekvidistantní síti s konstantními hodnotami)
2D polárně – nesymetrická blokově třídiagonální (5 diagonál s nekonstantními hodnotami)
počet rovnic N = J.K, počet nenulových prvků nnz ~ 5.N, šířka pásu m = 2.J+1 (rovnice řazeny podle j, pak k)

Struktura zdrojových kódů

1. inicializace sítě, okrajových podmínek a pravé strany
2. inicializace matice: plná (GE) nebo pásová (GB) pro LAPACK (MKL/ACML/CULA Dense)
   compressed sparse row (CSR) formát pro PARDISO/Sparse BLAS (MKL)/CULA Sparse
3. konfigurace a volání řešiče

Formáty matice

plné matice o N řádcích, N = J.K:
(Fortran obecně) po sloupcích, prvek a_{jk} v prvku pole A(j,k) ekvivalentním s A(i), i = j+(k-1).J
pásové matice o N řádcích a šířce pásu m = ml+1+mu:
(LAPACK band) pole A(m+ml,N), sloupcový index zachován, diagonály matice v řádcích pole, prostor navíc
A(1:ml,:) pracovní prostor
A(ml+1:ml+m,:) diagonály matice (horní diagonála první shora)
(NR band) pole A(N,m) a AL(N,ml), řádkový index zachován, diagonály ve sloupcích (dolní je první zleva)
řídké matice o N řádcích a nnz nenulových prvcích:
(CSR) pole A(nnz), ICOL(nnz), IROW(N+1), v poli A nenulové prvky matice zleva po řádcích
v poli ICOL sloupcové indexy příslušné prvkům z A
v poli IROW indexy v poli A pro první nenulové prvky na řádcích matice a
př. Saad, Intel MKL Reference Manual, CULA Sparse Reference Manual
jiné formáty: coordinate format (COO), compressed sparse column (CSC), row-indexed sparse storage (NR) aj.

Řešiče pro plné matice

LU faktorizace, obvykle 2 kroky (faktorizace a substituce), někdy sloučené v 1 driver

NR kap. 2.3	ludcmp, lubksb
LAPACK driver routines:	dgesv (simple driver), dgesvx (expert driver)
computational routines:	dgetrf, dgetrs
CULA Dense	culaDgetrf, culaDgetrs
	nutná inicializace knihovny culaInitialize, vhodná finalizace culaShutdown
IMSL	stejně jako LAPACK a navíc vlastní driver lin_sol_gen nebo 2 kroky dlftrg, dlfsrg

Řešiče pro pásové matice

LU faktorizace, 2 kroky nebo 1 driver

NR kap. 2.4	bandec, banbks s maticemi v NR band formátu
LAPACK driver routines:	dgbsv (simple driver), dgbsvx (expert driver) s maticemi v LAPACK band formátu
computational routines:	dgbtrf, dgbtrs

Přímý řešič pro řídké matice

Faktorizace, 3 kroky (analýza a přeskupení, faktorizace, substituce), volba režimu pro symetrické definitní, symetrické indefinitní a nesymetrické matice.

PARDISO (MKL)	pardiso s maticí v CSR formátu

Iterační řešiče pro řídké matice

Pro symetrické definitní matice metoda sdružených gradientů (CG), pro obecné matice metoda bikonjugovaných gradientů (BCG) nebo zobecněná metoda nejmenších reziduí (GMRES) a varianty.
Uživatel řešiče obvykle na vyžádání metody provádí součin r=A.v, např. voláním Sparse BLAS. Konvergenci (rychlost poklesu normy reziduí |A.x – b|) může urychlit předpodmiňováním, tj. řešením soustavy $P^{-1}.A.x = P^{-1}.b$; knihovny pak

žádají řešení soustavy P.r=v nebo samy nabízejí předpodmínění Jacobiho P=diag(A), ILU0 (incomplete LU) s udržením struktury řídké matice v maticových faktorech, ILUT s přidáváním potřebných nových prvků do maticových faktorů aj.

NR kap. 2.7	linbcg s voláním dodaných podprogramů atimes (pro r=A.v), asolve (pro r=P^{-1}.v)
MKL	dcg s reverzním komunikačním rozhraním (RCI) a pomocnými podprogramy dcg_init, dcg_check, dcg_get
	dfgmres (flexible GMRES) a dfgmres_init, dfgmres_check a dfgmres_get
Sparse BLAS (MKL)	mkl_dcsrgemv pro násobení y=A.x s polem A v CSR (nebo jiném) formátu ad.
CULA Sparse	culaDcsrSolverPrecond, Solver: Cg|Bicg|Gmres aj., Precond: nic|Jacobi|Ilu0
	matice explicitně v CSR (COO, CSC) formátu
	nutná inicializace knihovny a parametrů řešiče i předpodmiňovače
IMSL	dpcgrc (preconditioned CG with Reverse Communication)
	dgmres, rovněž s reverzní komunikací (s možností volání Sparse BLAS)

Ověřené kombinace knihoven a překladačů

Default LAPACK

Linux gfortran/g95: default (ATLAS?), pgfortran: vlastní default

```
gfortran -O2 -llapack f.f90
g95 -O2 -llapack f.f90
pgfortran -llapack f.f90
```

Win žádný default, projekt LAPACK for Windows (icl.cs.utk.edu/lapack-for-windows) vhodný pro gfortran a g95,

```
gfortran -O2 -L. -llapack -lblas f.f90 ; MinGW gfortran nečte proměnné prostředí a vyžaduje volbu -L
g95 -O2 -llapack -lblas f.f90 ; g95 hledá knihovny na cestě dané proměnnou G95_LIBRARY_PATH
pgfortran -llapack f.f90 ; vlastní default
```

Intel MKL

Linux linkovatelné se všemi 4 překladači, vláknově (OpenMP) paralelizované
Sparse BLAS pro řídké matice, MPI-paralelizovaný ScaLAPACK

```
ifort -mkl f.f90 ; pro 1vláknový běh ifort -mkl=sequential f.f90 nebo export MKL_NUM_THREADS=1
pgfortran -L$LIBMKL -lmkl_intel_lp64 -lmkl_sequential -lmkl_core f.f90
gfortran -O2 -lmkl_gf_lp64 -lmkl_sequential -lmkl_core f.f90
g95 -O2 -lmkl_gf_lp64 -lmkl_sequential -lmkl_core f.f90
```

Win linkovatelné s překladači ifort a pgfortran, vláknově paralelizované, Sparse BLAS pro řídké matice

```
ifort /Qmkl f.f90 /nologo ; set MKL_NUM_THREADS=1 nebo více
pgfortran -lmkl_intel_s_dll -lmkl_sequential_dll -lmkl_core_dll f.f90 pro neparalelizovaný běh
```

ACML

ACML je součástí instalace pgfortranu. Na webu ACML jsou nabízeny verze i pro jiné překladače. Linkování acml.lib je určeno pro 1vláknový běh, linkování acml_mp.lib pro běh vícevláknový.

Linux
```
pgfortran -L$LIBACML -lacml f.f90
```

Win
```
pgfortran -Mdll -Munix f.f90 %LIBACML%\libacml.lib
```

CULA Tools

Linux GPU hardware: geof30/40/50 14x32 jader, geof10/20/60/70/80/90 4x48 jader, karel 27x8 jader
CULA Dense (cula_lapack) i CULA Sparse (cula_sparse) linkovatelné se všemi 4 překladači

```
ifort -lcula_lapack f.f90 ; ifort -cula_sparse f.f90
pgfortran -L/usr/local/cula/lib64 -lcula_lapack f.f90 ; pgfortran -L/usr/local/culasparse/lib64 -lcula_sparse f.f90
gfortran -lcula_lapack -lcula_sparse f.f90 ; g95 stejně
```

Win GPU hardware: posluchárna, v PCL (zatím) ne
CULA Dense i Sparse linkovatelné se všemi 4 překladači, někdy potřeba i cula_core

```
ifort f.f90 cula_lapack.lib /nologo
pgfortran -lcula_lapack f.f90
gfortran -O2 -L"C:\Program Files\CULA\R13\bin" -lcula_lapack f.f90 ; g95 stejně
```

IMSL

Linux (geof30) ifort: BLAS/LAPACK z knihovny MKL (lze paralelně), volitelně CUBLAS

```
ifort $FFLAGS f.f90 $LINK_FNL          ! FNL = Fortran Numerical Library, verze 7.0
```

Win ifort: BLAS/LAPACK z knihovny MKL (lze paralelně), bez GPU

```
ifort %FFLAGS% f.f90 %LINK_FNL%    ! inicializace: fnlsetup.bat (vhodné fnlsetup7.bat, neboť verze 7.0)
```

pgfortran: referenční BLAS/LAPACK (nelze paralelně), bez GPU

```
ifort %FFLAGS% f.f90 %LINK_FNL%    ! inicializace: fnlsetup.bat (vhodné fnlsetup6.bat, neboť verze 6.0)
```

Zdrojové kódy

Řešiče pro plné matice: NR (lubksb), LAPACK general (dgetrf), CULA Dense (culaDgetrf), IMSL (dlftrg)
Řešiče pro pásové matice: LAPACK band (dgbtrf)
Přímý řešič pro řídké matice: PARDISO (pardiso)
Iterační řešiče pro řídké matice: NR (linbcg), MKL (dfgmres), CULA Sparse (Bicg, Gmres), IMSL (dgmres)

Odkazy

Lineární algebra
Golub G. H. and van Loan C. F., Matrix Computations, 3rd Ed., 1996 (PDF)
Saad Y., Iterative Methods for Sparse Linear Systems, 2nd Ed., 2000 (PDF)
Knihovny
Press W. H. et al., Numerical Recipes in Fortran 77: The Art of Scientific Computing, 2nd Ed., 1996 (PDF)
Anderson E. et al., LAPACK Users' Guide, 3rd Ed., 1999 (PDF)
Intel MKL Reference Manual (PDF k verzi 10.2)
CULA Tools Programmer's Guide & Reference Manuals: CULA Dense, CULA Sparse (PDF)
IMSL User's Guide (PDF)
Web
LAPACK: netlib.org
MKL: software.intel.com/en-us/articles/intel-mkl
ACML: developer.amd.com/libraries/acml
CULA Tools: www.culatools.com
IMSL: www.roguewave.com/products

L. H., 4. 4. 2012