Memory Management I

- Approach 3: block algorithms (nb = 256)

```
for j=1:nb:n
  for k=1:nb:n
    for jj=j:j+nb-1
      for kk=k:k+nb-1
        c(:,jj) = a(:,kk)*b(kk,jj)+c(:,jj);
      end
    end
  end
end
```

Memory Management II

In MATLAB, `1:256:1025` means 1, 257, 513, 769

- Note that we calculate

$$\begin{bmatrix} A_{11} & \cdots & A_{14} \\ & \vdots & \\ A_{41} & \cdots & A_{44} \end{bmatrix} \begin{bmatrix} B_{11} & \cdots & B_{14} \\ & \vdots & \\ B_{41} & \cdots & B_{44} \end{bmatrix}$$

$$= \begin{bmatrix} A_{11}B_{11} + \cdots + A_{14}B_{41} & \cdots \\ \vdots & \ddots \end{bmatrix}$$

Memory Management III

- Each block: 256×256

$$
\begin{aligned}
C_{11} &= A_{11}B_{11} + \cdots + A_{14}B_{41} \\
C_{21} &= A_{21}B_{11} + \cdots + A_{24}B_{41} \\
C_{31} &= A_{31}B_{11} + \cdots + A_{34}B_{41} \\
C_{41} &= A_{41}B_{11} + \cdots + A_{44}B_{41}
\end{aligned}
$$

- For each (j, k), $B_{k,j}$ is used to add $A_{:,k}B_{k,j}$ to $C_{:,j}$

Memory Management IV

- Example: when $j = 1, k = 1$

$$
\begin{aligned}
C_{11} &\leftarrow C_{11} + A_{11}B_{11} \\
&\vdots \\
C_{41} &\leftarrow C_{41} + A_{41}B_{11}
\end{aligned}
$$

- Use Approach 2 for $A_{:,1}B_{11}$
- $A_{:,1}$: 256 columns, $1024 \times 256/65536 = 4$ pages. $A_{:,1}, \ldots, A_{:,4} : 4 \times 4 = 16$ page faults in calculating $C_{:,1}$
- For A: 16×4 page faults
- B: 16 page faults, C: 16 page faults

LAPACK I

- BLAS defines only operations such as matrix-matrix products. How about operations like LU factorization for solving linear systems?
- LAPACK – Linear Algebra PACKage, based on BLAS
- Routines for solving

  Systems of linear equations

  Least-squares solutions of linear systems of equations

  Eigenvalue problems, and

LAPACK II

Singular value problems.

- Subroutines in LAPACK classified as three levels:
- driver routines: each solves a complete problem, for example solving a system of linear equations
- computational routines: each performs a distinct computational task, for example an LU factorization
- auxiliary routines: subtasks of block algorithms, commonly required low-level computations, a few extensions to the BLAS
- LAPACK provides both single and double versions

LAPACK III

- Naming: All driver and computational routines have names of the form XYYZZZ
- X: data type, S: single, D: double, C: complex, Z: double complex
- YY, indicate the type of matrix, for example

GB	general band
GE	general (i.e., unsymmetric, in some cases rectangular)

LAPACK IV

Band matrix: a band of nonzeros along diagonals

$$\begin{bmatrix} \times & \times & & & \\ \times & \times & \times & & \\ & \times & \times & \times & \\ & & \times & \times & \times \\ & & & \times & \times \end{bmatrix}$$

- ZZZ indicates the computation performed. For example,

LAPACK V

SV	simple driver of solving general linear systems
TRF	factorize
TRS	use the factorization to solve Ax = b by forward or backward substitution
CON	estimate the reciprocal of the condition number

- SGESV: simple driver for single general linear systems

  SGBSV: simple driver for single general band linear systems

LAPACK VI

- Now optimized BLAS and LAPACK available on nearly all platforms

  For example, Intel MKL (Math Kernel Library)

Block Algorithms in LAPACK I

- From LAPACK manual Third edition; Table 3.7
  `http://www.netlib.org/lapack/lug`
- LU factorization DGETRF: $O(n^3)$
- Speed in megaflops (10^6 floating point operations per second)

Block Algorithms in LAPACK II

	No. of CPUs	Block size	n 100	n 1000
Dec Alpha Miata	1	28	172	370
Compaq AlphaServer DS-20	1	28	353	440
IBM Power 3	1	32	278	551
IBM PowerPC	1	52	77	148
Intel Pentium II	1	40	132	250
Intel Pentium III	1	40	143	297
SGI Origin 2000	1	64	228	452
SGI Origin 2000	4	64	190	699
Sun Ultra 2	1	64	121	240
Sun Enterprise 450	1	64	163	334

Block Algorithms in LAPACK III

- 100 to 1000: number of operations 1000 times
- Block algorithms are not very effective for small-sized problems
- Clock speed of Intel Pentium III: 550 MHz
- Thus by block algorithms good performance can be achieved

ATLAS: Automatically Tuned Linear Algebra Software I

- Web page: `http://math-atlas.sourceforge.net/`
- Programs specially compiled for your architecture

  That is, things related to your CPU, size of cache, RAM, etc. are considered