Technische Universität München
Institut für Informatik
Prof. Dr. Thomas Huckle
Martin Buchholz

WT 2009/2010

05.02.2010

Parallel Numerics

Exam

A handwritten sheet of paper (size A4, front and back page) may be used during the exam as mnemonic as well as the MPI operation reference distributed during the tutorials. No other material is allowed. The exam is to be solved within 90 minutes, and the answers are to be written in German or English. Please try to answer all parts of the questions precisely and briefly. For passing the exam you will need 17 out of 40 credits. The exam consists of four problems on three pages.

Problem 1: LU-decomposition and Performance Metrics **($\approx$ 10 credits)**
For matrices A we want to compute an LU-factorization on a parallel computer. The algorithm uses a block decomposition of the following form

$$\begin{pmatrix} \mathbf{L}_{11} & 0 & 0 \\ \mathbf{L}_{21} & \mathbf{L}_{22} & 0 \\ \mathbf{L}_{31} & \mathbf{L}_{32} & \mathbf{L}_{33} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{U}_{11} & \mathbf{U}_{12} & \mathbf{U}_{13} \\ 0 & \mathbf{U}_{22} & \mathbf{U}_{23} \\ 0 & 0 & \mathbf{U}_{33} \end{pmatrix} = \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \mathbf{A}_{13} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \mathbf{A}_{23} \\ \mathbf{A}_{31} & \mathbf{A}_{32} & \mathbf{A}_{33} \end{pmatrix}$$

and is based on two subproblems

- (P1) LU-decompositions of small submatrices (simplified assumption: strictly sequential)
- (P2) parallel triangular solves and matrix multiplications (simplified assumption: fully parallel)

a) Describe a parallelisation algorithm where the blocks of A, L and U are used rowwise.
($\approx$ 4 credits)

b) Assume that a given problem with a specific size needs a runtime of t_1 for P1 and t_2 for P2 on a single processor. Now we consider the computation of exactly the same problem using p processors. What are the values for speedup and efficiency, considering the parallelisability of P1 and P2
($\approx$ 2 credits)

c) With the assumptions from b), a maximum speedup of 10 is achieved. Give a value for the fraction t_1/t_2
($\approx$ 2 credits)

d) Assume that Gustafsson's law is used to estimate the parallel efficiency of the algorithm. Is gives an efficiency of 0.8. Using the formulas for efficiency and speedup, derive the fraction of the total runtime which the subproblems P1 and P2 need.
($\approx$ 2 credits)

Problem 2: MPI Operations (≈ 8 **credits)**
Given is the code fragment

```
20: if(rank==0){
21:   a = 1;
22:   MPI_...(&a, 1, MPI_INT, 1, 1, ...);
23:   fileAccess();
24:   a = 2;
25:   MPI_...(&a, 1, MPI_INT, 1, 1, ...);
26:   a = 3;
27:   MPI_...(&a, 1, MPI_INT, 1, 1, ...);
28:   b = 4;
29:   MPI_...(&b, 1, MPI_INT, 1, 1, ...);
30:   fileAccess();
31: }
32: else if(rank==1){
33:   fileAccess();
34:   MPI_Recv(&a, 1, MPI_INT, 0, 1, ...);
35:   processData(a);
36:   MPI_Recv(&a, 1, MPI_INT, 0, 1, ...);
37:   processData(a);
38:   MPI_Recv(&a, 1, MPI_INT, 0, 1, ...);
39:   processData(a);
40:   fileAccess();
41:   MPI_Recv(&a, 1, MPI_INT, 0, 1, ...);
42:   processData(a);
43: }
```

where the function `fileAccess()` accesses a resource which is shared by the two processesand and the function `processData()` does some operations with the recieved. Before this code fragment, all necessary variable are declared, the MPI environment is initialised and `rank` contains the rank of the processes. The MPI-statments in line 22, 25, 27 and 29 are incomplete. You will need the four MPI-statements `MPI_Send`, `MPI_Isend`, `MPI_Ssend` and `MPI_Issend` (in a special order) to complete them.

a) Specify what each of the given MPI send-commands does.
(≈ 2 credits)

b) The process with rank 1 should execute the function `processData()` with `a` equal to one, two, three and four in exactly that order. Give the correct order of the four MPI send commands to ensure this order without risking any conflicts when accessing shared resources. For each of the commands, give a short explanation why it was used at the chosen position
(≈ 4 credits)

c) Another send-command ist `MPI_Bsend`. Give an example where `MPI_Send` is not sufficient and `MPI_Bsend` has to be used instead.
(≈ 2 credits)

Problem 3: Gaussian Elimination (≈ 6 **credits**)
We consider the real linear system $Ax = b$ of the form

$$A = \begin{pmatrix} a_{11} & 0 & a_{13} \\ 0 & a_{22} & a_{23} \\ a_{31} & 0 & a_{33} \end{pmatrix} \cdot x = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

a) We consider addition, subtraction, multiplication, and division as elementary operations. Give the elementary operations of the Gauss Elimination algorithm for solving the above linear equations.
(≈ 3 credits)

b) If each elementary operation can be executed in one timestep, how many timesteps does the solution need

- sequentially
- in parallel

(≈ 3 credits)

Problem 4: Sparse Matrices (≈ 16 **credits**)
We consider the 2D Poisson problem $-\Delta u(x,y) = -\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} = 1$ with Dirichlet boundary conditions on a quadratic grid with unknowns ordered lexicographically $u_{ij}, i, j = 1, ..., n$.

a) Show that the stencil $\begin{bmatrix} & & -1 & & \\ & & -1 & & \\ -1 & -1 & 8 & -1 & -1 \\ & & -1 & & \\ & & -1 & & \end{bmatrix}$ leads to a discretization of the above PDE.

(Hint: Taylor expansion $f(a+h) = f(a) + h * f_x(a) + h^2 f_{xx}(a)/2 + ...$)
(≈ 3 credits)

b) Show a typical linear equation in the discretization of the given PDE based on the stencil from a). Describe the resulting full matrix A.
(≈ 2 credits)

c) Give the sparsity graph $G(A)$ for matrix A.
(≈ 2 credits)

d) How many colours are necessary for $G(A)$ such that no neighboring vertices have the same colour.
(≈ 2 credits)

e) Write the full matrix A for a 3×3 grid, based on

- lexicographical ordering
- ordering relative to the colouring derived in (d)

(≈ 5 credits)

f) Describe the sparse coordinate form for the matrix with the lexicographical ordering from above
(≈ 2 credits)