Practical Course in Parallel Computing - Exercise 9

Maximilian Wess, Hanna Holderied

last edited June 14, 2016

1. Exercise 1
   a) For Exercise 1, please see e1.c. To solve the problem lines 59 and 60 need to be switched.
   b) Using for instance, mpirun -n 2 -hosts c000,c001 ./e1
   c) See e1_pbs.sh
2. Exercise 2
   a) See e21.c.
   b) See e22.c
3. Exercise 3 Using e3c. the functions can be explained as follows(not that e3 was executed with mpirun -n 4 e3):
   a) MPI_Gather Arrays x and y are initialized with whitespaces. The first position of x is filled with 'a' + rank. The root rank in this case is 1. Using MPI_Gather all processes send the first character in the x array to the root process, which uses the char array y as a receive buffer. For that reason the y array contains a b c d. Moreover MPI_Gather orders the received data chunks, i.e. process 0 gets the first part of the receiving array, pprocess 1 gets the second part and so on.
   b) MPI_Allgather MPI_Allgather workds similiar to MPI_Gather except that not only the root process, but all processes receive the send buffer.
   c) MPI_Scatter The send buffer of the root process is partitioned into chunks. TGhen each chunk is sent to one process. The receivers are again ordered, i.e. process 0 gets the first chunks, process 1 the second and so on. In the example one can see that the array e f g h is partitioned into 4 chunks. Process 0 receives e, process 1 receives f,...
   d) MPI_Alltoall In each process the sendbuff is split into n chunks (n is the number of processes). Each chunk contains sendcount elements. The recvbuf is partitioned accordingly. The process j sends the k-th block of its local sendbuff to process k, which places the data in the j-th block of its recvbyuf. In the exapmle this means for instance, that for n=4 chunk 0 of process 3 (m) is send to process 0 and is placed in the recv buffer at position 3.

e) MPI_Bcast The root process sends the send buffer to every other process. In the example only the root process has character b in the sendbufgf, but after calling MPI_Bcast every other process has received the character.

4. Exercise 4

   a) See ex1_plot.pdf.

   b) See e4b.c and ex2_plot.pdf.

   c) See e4c.c and ex3_plot.pdf.

   d) See e4d1.c, e4d2.c, e4d3.c and e4d4.c.