mpC Workshop

Integrated Parallel Programming System
for Heterogeneous Networks of Personal Computers

Advanced Technical Services (ATS)
June 2002

Abstract

The **mpC Workshop** for Windows is a joint project of Advanced Technical Services (ATS) and Institute for System Programming of Russian Academy of Sciences (ISP-RAS).

The **mpC Workshop** is integrated development environment for the **mpC** language for parallel programming of heterogeneous networks. It allows easy developing of efficient parallel applications intended for running on usual office network. Such network of many diverse personal computers is an attractive alternative to a specialized parallel system.

Office network has one principal difference from specialized parallel system. As a rule it consists of computers with different performances. This heterogeneity is the main obstacle to exploit full performance potential of this network because the execution time of usual parallel application on heterogeneous parallel system is limited by the power of weakest computer. **mpC** is designed especially for those who face this problem in their work.

The purpose of this white paper is to examine benefits of using Windows PCs networks for parallel computing, to outline the strengths of **mpC** and **mpC Workshop** and to explain how they can provide a competitive advantage for customers.

Introduction

To keep competitive ability every business must carry out a lot of tasks requiring a huge amount of computer power. It is usual to find this power in specialized parallel system. More natural is to find it in office network of personal computers since they are inactive most of the time and their hardware – processors and network equipment - is often the same as in clusters.

Network versus cluster

Cluster is a specially designed parallel system, which, unlike a local network, has only one job flow.

There are two major classes of a cluster configurations (1) capacity cluster and (2) capability cluster. A capacity cluster is targeted for solutions of multiple problems, each running on a dedicated single CPU with minimum communication between individual servers. A capability cluster is used as a collective computational power of several computational nodes for solution of a single problem as rapidly as possible.

The **mpC** programming environment is intended for use of office network as capability cluster.

Position of office network in cluster classification

Capability cluster configurations can be divided the into two distinct subclasses, (1) high-end clusters consisting of number of powerful multiprocessor nodes connected within a simple but powerful and mostly proprietary network topology, and (2) low-end clusters employing considerable number of single- or two-CPU nodes connected via inexpensive commercially available networks.

Hardware of usual office network, as a rule does not essentially differ from the hardware of low-end cluster.

Flexibility, local control, relatively low cost combined with suitable performance makes clusters a popular choice, especially for small-sized companies and departments. *Using of office network as cluster when necessary is very attractive because it combines advantages of personal computers and specialized parallel systems.* It is effective to use office network for parallel computing even in no-dedicated mode that is when desktops are used for usual office work.

Implicit and explicit approaches to parallel programming

Implicit approach assumes a presence of a tool, which transforms an algorithm description, for example, sequential program to parallel program. As a rule such tools generate very ineffective parallel program. It is the serious deficiency of implicit approach because parallel computing is used in those cases where efficiency is the key issue.

Explicit approach makes programmer responsible for distribution of data and computation, synchronization of processes of parallel program and so on. Gradually it became the preferred approach. De-facto standard for explicit parallel programming is MPI (Message Passing Interface). It is low-level library with bindings to Fortran and C that provides, in fact, the assembler level of parallel programming for networks of computers. The low level of their parallel primitives makes the writing of complex and useful parallel applications using MPI tedious and error-prone.

Till recently there was a gap between availability of hardware and software for parallel computing. Using local networks for this purpose was restrained by absence of appropriate system software only. In the second half of 90th in Institute for System Programming **mpC**, a high-level language with explicit parallelism, was developed. It is designed specially to develop portable adaptable applications for heterogeneous networks of computers. The **mpC Workshop** for Windows is the Integrated Development Environment for the **mpC** language. It is aimed to simplify the programming process, shorten the program development time and potentially raise the program quality.

The mpC language overview

The main idea underlying mpC is that an mpC application explicitly defines an abstract network and distributes data, computations and communications over the network. The mpC programming system uses this information to map the abstract network to any real executing network in such a way that ensures efficient running of the application on this real network. This mapping is performed in run time and based on information about performances of processors and links of the real network, dynamically adapting the program to the executing network.

The mpC language is an ANSI C superset that introduces a new kind of managed resource, computing space, defined as a set of virtual processors of different performances connected with links of different communication speeds. At run time, actual processes of the particular running parallel application represent the virtual processors. The programmer manages the computing space by means of creating and discarding regions of the computing space, named network objects, just like he/she manages storage, creating and discarding data objects (regions of storage). The networks are used to distribute data, compute expressions and execute statements.

The mpC language relates to MPI in parallel programming as C language relates to Assembler language in sequential programming. C allows much easy developing applications with practically the same performance as Assembler does and Assembler has very restricted area of application. As well mpC allows much easy developing parallel applications with practically the same performance as MPI does.

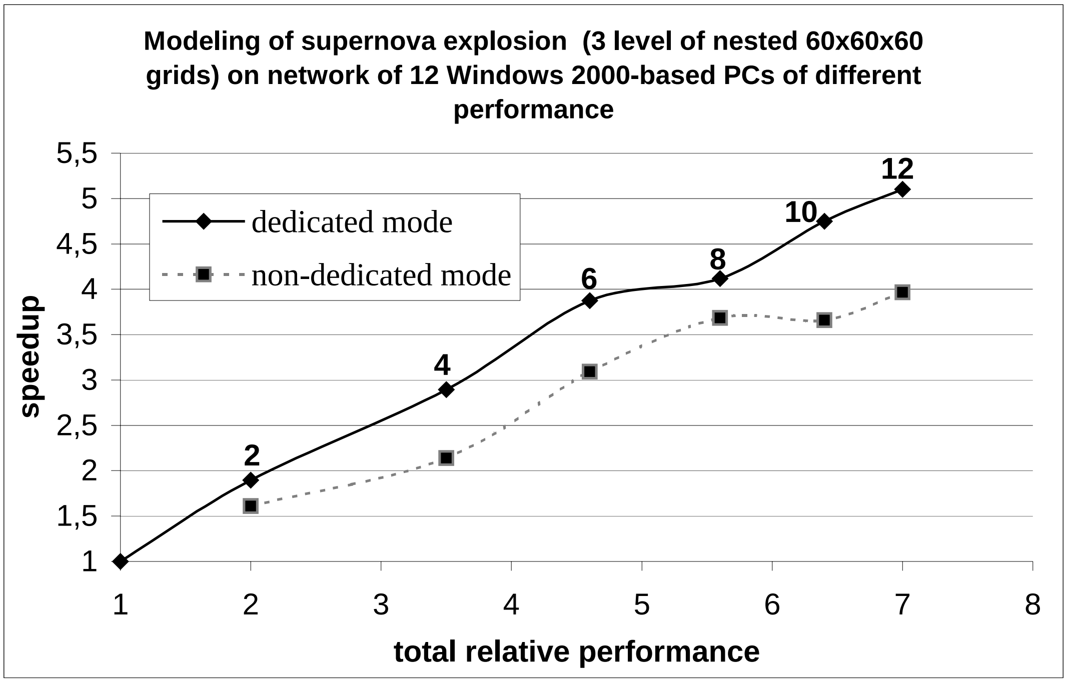
An example of mpC application

> A sequential three-dimensional application developed in Institute for Applied Mathematics of Russian Academy of Sciences for modeling of a type-II supernova was rewritten in mpC using mpC Workshop. The goal of modeling is an investigation of large-scale convection in the protoneutron core as a result of no equilibrium neutronization of matter. Since it is a very time-consuming application, investigation of all effects is not possible without use of parallel system.
>
> The application solves the system of partial differential equations using a sequence of increasingly finer nested grids. All grids are of the same dimensions and each nested grid is twice finer then the coarser one. Grids are thickened near point of location of initial distribution of entropy.

First, the mpC application makes several steps of algorithm with homogeneous data distribution (that means every processor processes equal subgrid). The relative processor performances are calculated basing on their execution time. After that data are redistributed between processors according to their performances and simulation is carried out further.

It may be surprising but execution of this application on usual office network also makes sense even in non-dedicated mode. We use a network consisting of 12 diverse uniprocessor Windows 2000 personal computers ranging from Athlon 1700+ to Pentium III 533 MHz interconnected via Fast Ethernet switch. The relative performance of the fastest computers is three times as large as that of the slowest ones. Such a ratio between performances of different computers is typical for an ordinary office network.

We estimated speedup that is possible to obtain on different subnetworks of the network. Every time we increase the number of processes we add weaker computers. The figure below presents the speedup achieved on different subnetworks of the network in both dedicated and non-dedicated modes. It demonstrates that on usual office network computationally and communicationally intensive simulation is carried out five times faster in dedicated and four times faster in non-dedicated mode. Note that in non-dedicated mode the desktops were used for software development and work with documents as they are usually used and users did not feel any influence of parallel application run.



The mpC Workshop overview

The **mpC Workshop** is an Integrated Development Environment for Microsoft Windows platform supporting development of parallel programs, written in mpC.

mpC Workshop is an Integrated Environment for development of parallel applications for Microsoft Windows. mpC Workshop uses the client-server model with GUI environment on the client side and mpC command-line environment on the server side. On the client side you create and edit mpC source files, specify settings for building executables and initiate building and debugging. Client sends commands to server and server performs them. Executable is built on one machine and broadcasted to others that will take part in execution of the parallel program.

mpC Workshop includes:

- *full-featured syntax-oriented editor*
- *source-level parallel mpC debugger*

  The main advantage of the mpC parallel debugger is that it allows to see the parallel program as a whole, but not as a set of separate processors communicating with each other. Debugger allows:

  - seeing a lot of useful information without entering commands. The mpC parallel debugger displays all the important information about a single process, showing the source code, stack trace, and stack frame for the process.

  - debugging remote programs over network, even over internet.
  - handling executable so as to control execution of any group of processes and to keep track of values of variables in any process of the parallel program.
- *Virtual Parallel Machine management tool*

  A Virtual Parallel Machine represents a network of computers on which an mpC program will be executed.
- *Adviser*

  This tool provides additional semantic information making possible to find out some semantic bugs before program execution.
- *mpC compiler*
- *run-time support system*
- *project management tool*

System requirements

mpC Workshop Server:

- Operating System: Windows 2000 SP2 (or later)
- C Compiler: Microsoft Visual C++ 6.0 SP4 (or later)
- MPI: MPI Pro 1.6.3
- Disk space: at least 25Mb

mpC Workshop Client:

- Operating System: Windows 2000 SP2 (or later)
- Disk space: at least 40Mb