PARALLEL NUMERICAL MODELLING OF SOLIDIFICATION ON A PC-BASED CLUSTER

R. WYRZYKOWSKI\textsuperscript{1}, N. SCZYGIOL\textsuperscript{2}, T. OLAS\textsuperscript{3}
\textsuperscript{1,3} Instytut Matematyki i Informatyki
\textsuperscript{2} Instytut Mechaniki i Podstaw Konstrukcji Maszyn
Politechnika Częstochowska
ul. Dąbrowskiego 73, 42-201 Częstochowa

SUMMARY

The paper concerns theoretical and practical aspects of parallelising the finite element computations used for modelling non-linear heat transfer, which occurs in solidifying castings. To parallelise the resulting FE model, a finite element mesh is divided into a set of sub-meshes. They are then solved concurrently over different processors, which communicate with each other. Examples of computer simulation were performed on a cluster, built this year at Technical University of Częstochowa.

Key words: parallel computing, cluster, solidification, FEM, mush partitioning

1. INTRODUCTION

Design in many branches of engineering requires numerical analysis. Parallel computing permits the engineer to undertake such an analysis in a considerably shorter time, reducing the time from the initial design concept to its implementation as a product. Among these branches is the modelling of solidification by the use of FEM.

Based on the object-oriented technique, an environment for the parallel FEM modelling on clusters of PC-s has been developed at the Technical University of Częstochowa [1]. This year in the Institute of Mathematics and Computer Science a cluster with 9 nodes and 18 processors has been built. It is called ACCORD, which

\textsuperscript{1} dr hab. inż., prof. P.Cz., roman@k2.pcz.czest.pl
\textsuperscript{2} dr hab. inż., norbert.sczygiol@imipkm.pcz.czest.pl
\textsuperscript{3} mgr inż., olas@k2.pcz.czest.pl
stands for Academic Cluster of Częstochowa Research and Development [2]. This cluster allows us to achieve a considerable shortening of computing time in the numerical modelling of solidification.

2. SOLIDIFICATION MODEL

Solidification is governed by a quasi-linear heat conduction equation containing the heat source term, which describes the rate of latent heat evolution. For parallel numerical modelling the apparent heat capacity formulation of this equation has been used, which belongs to the class of enthalpy formulations. It has the following form [3]

$$\nabla \cdot (\lambda \nabla T) = c^* \frac{\partial T}{\partial t}$$

(1)

where $\lambda$ is the heat conductivity and $c^*$ is the effective heat capacity defined as

$$c^*(T) = c_p - \rho_s L \frac{\partial f_s}{\partial T}$$

(2)

where $c$ is the specific heat, $\rho$ is the density (subscript $s$ refers to the solid phase), $L$ is the latent heat of solidification and $f_s$ is the solid phase fraction.

Eq. (1) was solved by FEM. After semi-discretisation the two-step Dupont II scheme was applied for integration over time. A two-step scheme was used because of the dependence of casting material properties on temperature. The application of a two-step scheme requires the use of a one-step scheme in the first time step. For this reason the modified Euler-backward scheme was used in the first time step. The final form of Eq. (1) after semi-discretisation and the application of the modified Euler-backward scheme, is as follows

$$\left( M^n + \Delta t K^n \right) T^{n+1} = M^n T^n + \Delta t b^{n+1}$$

(3)

while the application of the Dupont II scheme gives

$$\left( M^O + \frac{3}{4} \Delta t K^O \right) T^{n+2} = M^O T^{n+1} - \frac{1}{4} \Delta t K^O T^n + \frac{1}{4} \Delta t \left( 3b^{n+2} + b^n \right)$$

(4)

where $M$ is the mass matrix, $K$ is the conductivity matrix, $T$ is the temperature vector and $b$ is the boundary conditions vector. The superscript $(O)$ denotes that the thermal conductivity coefficient is calculated for an extrapolated temperature.
The effective heat capacity is present in the mass matrix in this formulation. It can be approximated by some schemes. In this paper the Del Giudice scheme is used, in which

\[ c^* = \frac{H_i N_{i,\alpha} T_j N_{j,\alpha}}{T_k N_{k,\beta} T_l N_{l,\beta}} \]  

where \( i, j = 1, \ldots, w \), and \( w \) is the number of nodes in a particular finite element, and \( \alpha \) and \( \beta \) are space co-ordinates. The enthalpy needed in Eq. (6) is defined as

\[ H(T) = \int_{T_{ref}}^{T} c^* \, dT + \rho_s L (1 - f_s(T)) \]

3. PARALLEL FINITE ELEMENT MODELLING

Parallel implementation of large and time consuming computational tasks on parallel computers or clusters is based on the concept of their division into smaller and more manageable sub-problems which may be solved efficiently. Using this approach a finite element mesh is divided into a finite number of sub-domains (sub-meshes) such that the computational load per sub-mesh is approximately the same. These sub-domains are then solved concurrently over different processors. The task of partitioning the meshes for parallel finite element computations is rather complicated since not only does the load per sub-domain have to be kept the same, but also the inter-processor communications have to be minimised.

For each sub-mesh we distinguish three types of nodes [4,5]:
- internal nodes, which are coupled only with nodes belonging to the considered sub-mesh,
- border nodes, which are coupled with nodes in other sub-meshes; these and internal nodes are called local nodes,
- external nodes, which are coupled with local nodes.

According to the previous section, the core of the finite element modelling of solidification is repeatedly solving systems of linear equations. In parallel modelling, for every sub-domain, a local system of equations is build, the structure of which is shown in Fig. 1. A local reordering of nodes in every sub-mesh is introduced. This has important advantages, including more efficient interprocessor communication, a reduction in local indirect addressing during matrix-vector multiplication and broad possibilities to overlap computations and communications.

For solving large linear systems of equations with sparse matrices, which are usually obtained in FEM, an efficient class of iterative methods exists. Popular and
efficient members of this class are: the conjugate gradient method (CG) suitable for symmetric positive defined matrices only, GMRES and Bi-CG-STAB algorithms suitable for non-symmetric matrices [6].

Fig. 1. Structure of the system of equations for a sub-domain

Rys. 1. Struktura układu równań dla pojedynczej domeny

4. CLUSTER ARCHITECTURE AND EXAMPLES OF COMPUTER SIMULATION

The cluster is based on INTEL ISP215G server platforms. They are appropriate for assembly in 19" standard boxes (2U height) and contain two INTEL processors. The clock of the Pentium III is 750 MHz. The cluster consists of 9 servers. One of them also acts as the master. Each server has a 256 MB RAM memory, except the master, which has a 512 MB RAM memory. The total disk memory of the cluster is about 150 GB. The servers are connected via a high-efficient net called MYRINET [2].

Computer simulation on the cluster was performed using our own software, ParallelNiascaS [7], which is designed using PVM as a communication layer. The CG algorithm with the Jacobi preconditioner was used to solve linear systems of equations. Simulations of casting solidification were done for six different finite element meshes, with about 40 000 to more than 750 000 unknowns. The computations were carried out on one processor and then on from two to eight processors. Obviously, this requires the whole mesh to be divided into an appropriate number of sub-meshes (for each individual computation). The Chaco library is currently used to partition the meshes [7]. The obtained results are shown in Figs. 2 and 3. For solving systems of equations the highest
speedup (Fig. 2) and the greatest efficiency (Fig. 3) were achieved for mesh with the
greatest number of unknowns.

Fig. 2. Speedup vs. number of processors
Rys. 2. Залежność przyspieszenia od liczby procesorów

Fig. 3. Efficiency vs. number of processors
Rys. 3. Залежność efektywności wykorzystania procesorów od ich liczby

5. CONCLUDING REMARKS

This paper presents an approach to the development of a parallel software envi-
ronment for finite element modelling on a PC based cluster. This environment is an
extension of the sequential NiscaS software, developed at the Technical University of
Częstochowa. The domain decomposition technique and iterative methods for solving
large systems of linear equations are used to develop the parallel core of the computa-
tions on a cluster. The results obtained for modelling the solidification of a casting are very promising. A significant reduction in runtime can be achieved for sufficiently large tasks.

REFERENCES


The current work is sponsored by the State Committee for Scientific Research (KBN) for 1999-2002 under Grant No. 7 T07A 009 17.

RÓWNOLEGŁE MODELOWANIE NUMERYCZNE KRZEPNIĘCIA NA KLASTRZE KOMPUTERÓW PC

STRESZCZENIE

Praca dotyczy teoretycznych i praktycznych aspektów zrównoleglenia obliczeń przeprowadzanych metodą elementów skończonych w celu modelowania nieliniowego przewodzenia ciepła, które zachodzi podczas krzepnięcia odlewów. W celu zrównoleglenia końcowego modelu obliczeń za pomocą MES siatka elementów skończonych dzielona jest na pod-siatki (domeny). Są one następnie obliczane na oddzielnych procesorach, komunikujących się ze sobą. Przykładowe symulacje komputerowe wykonano na klastrze komputerów PC, zbudowanym w tym roku w Politechnice Częstochowskiej.

Recenzował Prof. Stanisław Jura