ANSYS AND HIGH PERFORMANCE COMPUTING

Ian P Jones[1]

ANSYS UK,
West Central 127, Milton Park,
Abingdon,
Oxfordshire OX14 4SA,
UK.

Email ian.jones@ansys.com

SUMMARY

High Performance Computing (HPC) is now a routine part of the product development process in industry, achieving significant process compression and consequent industrial benefits. This paper summarises some of the developments within ANSYS which have enabled this to happen, by providing efficient and scalable software on the latest hardware. The paper also outlines some of the developments which are currently taking place, to make the benefits of HPC accessible to non experts, and open up significant new capability for industrial simulation.

1. INTRODUCTION

ANSYS develops, markets, and supports engineering simulation software used to predict how product designs will operate and how manufacturing processes will behave in real-world environments. Its products [1] include ANSYS® Mechanical™, for performing Finite Element stress Analysis (FEA), and ANSYS CFD, which includes ANSYS® CFX® and ANSYS® FLUENT®, for Computational Fluid Dynamics (CFD). While these are the core products, there are a number of other products, including meshing solutions, CAD integration, electromagnetics, Design Of Experiments (DOE) as well as specialist simulation products, all incorporated into a single simulation environment, ANSYS® Workbench™, providing seamless integration of the different technologies. High Performance Computing (HPC) is an integral part of the product portfolio, but in line with the theme of this lecture series, the paper focuses mainly on the CFD solutions, CFX and FLUENT, but it does include aspects of ANSYS Mechanical, so as to give an overview of the different considerations that arise when solving a different class of engineering problems, and when combining the technologies in a multi-physics simulation.

The ANSYS software products have a wide range of functionality, with different solution strategies and functionality. It is not possible to provide a comprehensive overview of this functionality within this paper, but some of the important differentiating features of the solvers which influence the parallelisation strategies are given in Table 1 below.

[1] Also, Visiting Professor, University of Nottingham

FEATURES	ANSYS® CFX®	ANSYS® FLUENT®	ANSYS® Mechanical™
Application	CFD	CFD	Stress Analysis
Formulation	Control Volume Finite Element Method (Cell Vertex).	Finite Volume (Cell Centred).	Finite Element.
Solver Methodology	Coupled Iterative Algebraic Multigrid Method, see Raw [2].	Principally Segregated Solver, using the Algebraic MultiGrid Method, although other methods are available.	Direct Solver (Sparse Gaussian Elimination) + various iterative linear solvers. [3]
Communications Libraries[2]	MPI and PVM.	Principally MPI.	MPI and shared memory (SMP) options.
Partitioning	Metis [2], Recursive Coordinate Bisection and other methods.	Principal axis, Metis and other methods.	Metis and other methods.

Table 1: Principal Features of the ANSYS simulation solvers.

The paper includes:

- An overview of requirements for the development of commercial software to be run in an HPC environment.
- A description of the solution approaches in the ANSYS products.
- Some technical considerations in the detailed implementation of parallel processing.
- Typical HPC results from the products.
- Future trends in HPC, including the potential influence of Grid computing.

While reading this paper, it should be borne in mind that the technology is changing rapidly, and the intention of the paper is to give an overview of the topic, and that it is not intended to be a comprehensive survey. The reader is therefore advised to check separately to get up-to-date information, when looking at appropriate hardware for running ANSYS software.

2. REQUIREMENTS FOR PARALLEL COMPUTING

The design requirements of commercial and general purpose software are very different from academic research oriented codes or application specific codes. Some of the guiding principles adopted by ANSYS [3] can be summarised as:

1) A new parallel implementation cannot reduce or limit the simulation capability of the code. Each new version must maintain backward compatibility with an ever increasing array of capabilities and maintain the familiarity, ease of use and full functionality that users expect. This requires an evolutionary approach to a revolutionary change.

[2] MPI includes variants such as MPICH and vendor specific versions such as HP-MPI.