FRONTIER: Use of HPCN Technologies*

J.M.R.Shaw, D.C.Spicer

British Aerospace Defence Ltd, Preston, PR4 1AX, Lancashire, U.K.
email: John.Shaw@bae.co.uk, David.Spicer@dial.pipex.com

Abstract. Business trends are leading companies to work collaboratively both internally and externally. Data exchange technology has been driven by this trend, and the growth of World Wide Web technology and usage, supported by HPCN capabilities, is opening further possibilities for organisations to carry out product design and optimisation cooperatively. The FRONTIER project seeks to demonstrate the viability of collaborative design optimisation by putting into place a system whose architecture and technology addresses these needs.

1 Introduction

The FRONTIER project is concerned with *design optimisation*. It addresses directly the problems of tradeoff of *multiple conflicting objectives*, and includes 6 case studies from a range of engineering sectors.

The current industrial trend is towards Integrated Product Definition, involving collaborative consideration of all aspects of design, rather than via traditional tasks based on single-function responsibilities. This applies to inhouse design and manufacturing teams; and also to equivalent close working arrangements between collaborators, partners and customers. In I.T. terms, this places emphasis on systems integration, data sharing, and distributed operations.

In practice, an engineer's aim in optimising a design is to be able to consider as wide a range of design cases as possible. Firstly, to identify the best concept; and then to refine it sufficiently, within the time available. HPCN technologies are essential to support this. However, the successful industrial exploitation of HPCN's potential requires more than parallelised solvers and a respectable HPC platform to run them on. In order for design analysis systems to play their full part, they need to be able to operate responsively within the design and manufacturing framework, and integrate in particular with CAD and product definition systems.

* This work has been supported by the EU, ESPRIT project 20082

2 FRONTIER Technology

In FRONTIER, we want to minimise n objective functions $F_1(\tilde{X})$, $F_2(\tilde{X})$, $F_3(\tilde{X})$, $F_n(\tilde{X})$ simultaneously with respect to the design variables $\tilde{X}$, subject to a number, m , of constraints $c_i(\tilde{X}) \geq 0$; $i = 1, m$. In practice, these objectives are in conflict, so they need to be traded off against each other.

Tradeoffs are viewed by considering the performance of design options plotted in *objective space*. The region of feasible designs is limited by a notional tradeoff boundary, the *Pareto frontier*. This is illustrated in Figure 1. To identify the boundary, several approaches are possible. They include: *explicit multiobjective search* [1], conversion of multiple objectives into a single objective via suitable decision support, and fixing objectives as constraints. FRONTIER caters for all of these. A repertoire of probabilistic *Genetic Algorithms* [2] and hill climbing algorithms is being provided to address search strategies, as well as allowing full user participation if required.

In optimisation, robust *global search* has traditionally been a major problem. FRONTIER incorporates genetic algorithms to address this. G.A.'s lend themselves naturally to generalisation to cover the case of more than one objective function. Also, natural parallelism is available when using a generational-based G.A. The G.A. technology being used in FRONTIER, which includes new inheritance techniques and automatic parameter setting, is currently under development by Trieste University.

It is also crucial to provide efficient hill-climbing techniques to address single criterion *constrained optimisation*. Within the FRONTIER project this will be provided by Quasi-Newton, Sequential Quadratic Programming and gradient-free Multi-Directional Search (MDS) algorithms. Parallel versions of these are being developed by Manchester University within the PINEAPL project [3].

The aim is to combine the robustness of the G.A. approach with the superior convergence performance of hill climbers in a *hybrid optimisation algorithm* which switches between the two approaches when the region of the global minimum can confidently be identified.

Optimisation involves many design analyses, particularly where tradeoff boundaries need to be explored. *Decision support tools* are being provided, to enable the user to input tradeoff preference information as the optimisation proceeds. This allows several disparate objective functions to be combined into a single objective, expressed as a 'utility function'. The interactive coupling of these '*Multi Criteria Decision Making*' [4] tools within the main optimisation allows attention to be focussed early on the part of the boundary which is of most interest. This focussing operation is important in limiting computing resource demands, because it avoids making a large expenditure on solutions around the entire tradeoff boundary before facing up to the tradeoff decision. These tools are being developed in FRONTIER by Newcastle University.

3 Architectural Approach

The FRONTIER conceptual architecture is shown in Figure 2. Optimisation involves proposing designs to consider, by creating the design variable sets which define them (Optimisation layer); then translating, where necessary, these design variable parameters into real geometry (Parametrisation layer); and evaluating the designs (Design Evaluation layer). A *heterogeneous* collection of hardware platforms and operating systems will typically be involved, in dispersed geographical locations. *Design evaluation* will be carried out using the appropriate HPC platforms for the analysis codes concerned. *Geometry parametrisation* is likely to require repeated use of CAD systems (CATIA, Euclid, I-DEAS, ...) running within the main organisational CAD platform setup. Pre and postprocessing operations are needed on the user's desktop machine.

It is thus apparent that the FRONTIER architecture needs to be distributed across hardware and operating systems, in particular across Unix and Windows-NT. Operations and results may also need to be shared between design partners. The infrastructure thus needs environment and communication protocol independence. The emergence of *CORBA* object request brokers [5] and the Java programming language [6], and their ability to interact relatively easily with Microsoft tools and standards makes it possible for FRONTIER to realise the distributed system needed.

Design optimisation is a major HPCN customer area of the future. However, extending operations beyond the boundaries of a single enterprise, in line with the collaborative needs mentioned earlier, will involve many contractors. Interactive design will then need *web technology* combined with HPCN facilities. This subject area is debated in [7], motivated in part by the needs of the U.S. 'Affordable Systems Optimisation Project' [8]. The needs are architecturally complex and multileveled. A goal of FRONTIER is to expose the architectural issues and adopt the appropriate strategic technologies.

4 HPCN Content

Although the FRONTIER system will make major use of HPCN resources, FRONTIER is not primarily a code parallelisation project. Rather, it can be viewed as a *systems integration project*, in which advantage is taken of the increasing number of codes already parallelised.

These include: STAR-CD, FIDAP, KIVA II and SAUNA. In addition, one 'in-house' solver, 'HISSS-D' from Daimler-Benz Aerospace, is being parallelised by Bergen University.

FRONTIER project trial cases involve optimisations of *aircraft wings*, *marine pumps*, *heat exchangers*, *electrodomestic equipment*, and *diesel engine design*. Analysis codes used are shown in Table 1, with single-analysis serial code run-times on platforms listed. Some of the codes, particularly structural optimisation codes, include their own 'private' sub-optimisations using additional design variables owned by them. Others are pure analysis codes. Computational affordability is an important element in the trials. However, in all cases significant optimisation can be

done. As an example, trials conducted on the design of a single stage of a heat exchanger using FIDAP have been run using 512 evaluations in an initial G.A. computation followed by refinement of the best solution using a hill climber. This has produced a design weight reduction of around 20%.

The platforms to be utilised in FRONTIER include workstation clusters and MPP machines. The system must allow for very flexible use of available processors, reconciling the needs of individual parallel codes with parallel design evaluations required by G.A. operations. Both generational and steady-state G.A.'s are provided, and a 'pool of tasks' approach is adopted to ensure that processors are not idle whilst individual evaluators are converging at different rates.

File transfer times between platforms are also a potential bottleneck. Each design evaluation needs a *geometry transfer*, which could take several hours, and will also produce extensive results files even after reduction and compression. Thus the overlapping of these operations with computation is important.

References

- [1] P. Hajela, and C.Y. Lin: Genetic search strategies in multicriterion optimal design. *Structural Optimisation*. 4, pp 99-107. Springer-Verlag, 1992.
- [2] L. Davis: *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, 1991.
- [3] ESPRIT Project 20018 'PINEAPL' home page. <http://www.nag.co.uk/projects/PINEAPL.html>, 1996.
- [4] P. Sen, and J.-B. Yang: Multiple-criteria Decision-making in Design Selection and Synthesis. *Journal of Engineering Design*, Vol. 6, No. 3, 1995
- [5] Object Management Group (OMG) home page. <http://www.omg.org/>, 1996.
- [6] Sun Microsystems. Java homepage. <http://java.sun.com/>, 1996.
- [7] H.W. Yau, A. Leung, W. Furmanski, and G.C. Fox: Exploration of Emerging HPCN Technologies for Web-Based Distributed Computing. *Lecture Notes in Computer Science*. 1067, pp 869-874. Springer-Verlag, 1996.
- [8] G.C. Fox: 'Implementation of Affordable Systems Optimization Process in the NII'. <http://www.npac.syr.edu/users/gcf/ASOPSept95A/fullhtml.html>, 1995.

Figure 1

Figure 2

Code	Type of code	Run time	Platform	Intensity
FIDAP	pure analysis	500 sec (one analysis)	Cray J/916	medium
HISSE-D	anal. & opt.	100 min (with optimisation)	AIX 370 AT	high
LAGRANGE	anal. & opt.	15 min (one analysis)	HP-715	medium
CODAS	anal. & opt.	10 min (one analysis)	Cray YMP	medium
STARS	anal. & opt.	3 min (one analysis)	HP9000	low
KIVA-II	pure analysis	6-8 hours	IBM RS6000	high
MODEL	pure analysis	0.5 min	IBM W/S	very low
STAR-CD	pure analysis	3 hours 6 min	SGI Power Challenge 1CPU	high

Table 1