Architectural Patterns for Parallel Systems

Jorge Luis Ortega Arjona
Dept. of Computer Science
University College London
Gower Street
London WC1E 6BT, U.K.
+44 (0) 171 419 36 79
J.Ortega-Arjona@cs.ucl.ac.uk

KEYWORDS
Parallel Systems, architectural software patterns, simulation models, performance characteristic estimation, creation and coordination of elements.

1 INTRODUCTION

Software design is a critical feature in parallel systems programming because the process is significantly more complex than for programming single processor computers. More processors means more instructions to be programmed and executed, which also means that errors can appear more frequently. Examples can be found in the history of parallel software design where scientific codes (such as numerical analysis, fluid dynamics, etc.) have been hand-crafted precisely for machines and problems, at immense expense. Furthermore, it is not clear if programmers have properly addressed concerns unique to parallel processing, such as synchronisation, load balancing, etc.

The main goal of this research is to propose a method to estimate the performance charactersitic of parallel systems, based on architectural patterns and simulation models for parallel systems.

2 MOTIVATION FOR THE WORK

The objetive of a parallel software design should not just be to optimize performance (in terms of speed) as the only metric of parallel systems. A good parallel design must consider other important metrics of software quality, such as simplicity, understandability, portability, reusability, and others.

However, it is important to recognize that from a pragmatic point of view the objective of most scientists and engineers when using parallel systems is performance. Therefore, if performance cannot be considered as the most important metric in parallel programming, it should be considered among the most important.

The need for performance estimation for parallel systems does not come from a single source. It comes from various needs of estimation that can be divided into three areas: design, procurement and optimization.

- In the design and development of new parallel computer systems, performance estimation plays an important and continuing role. System engineers and programmers should use projections of performance to guide the development of new parallel systems. This area of perfomance estimation includes new processor design and communication components, as well as software systems.
- The decision to purchase or lease a parallel system, or even a new component of hardware or software, is another area where performance estimation matters. Usually, when performance estimation is undertaken when buying new systems, they are operational and available for testing, and the options are limited to a small number of products. For these reasons, performance estimation for a purchasing decision relies more on costly activities of mesuring and testing.
- A third area of performance estimation is concerned with the optimization of a specific parallel system. The nature of this performance estimation involves incremental and often reversible changes on an existing parallel system. There is the opportunity for testing and monitoring the performance of a parallel system. Inefficiencies in the system can be found, new solutions can be proposed and tried.

3 EXPECTED CONTRIBUTION

The main expected contribution is expressed in the following hypothesis:

Is it possible to estimate, with a certain degree of accuracy, the performance of a parallel system based on patterns of creation and coordination of its elements – expressed in the form of simulation models of architectural patterns – during the initial stages of parallel soft-

ware design?

This work has two purposes: first, to present a set of architectural patterns that are commonly used for parallel programming, classified according to the patterns of data access or request of actions and the requirements of order of data and operations; and second, to discuss if these architectural patterns can be used as a basis for building scale models that simulate their behaviour and can be used for performance estimation.

Parallel systems are based on concurrent execution of sequential elements. However, the performance of a particular parallel implementation depends not only on the correct and efficient execution of its sequential elements, but also in the form in which its elements cooperate among themselves. One of the main contributions of this work is based on this idea. The features of a parallel application are based on the characteristics of their sequential elements, but more important, they are related with how these elements are arranged, and their behaviour when cooperating as a single architectural structure. Therefore, it is possible to state that the performance of a parallel application depends, to certain extent, on its characteristic of creation and coordination of elements. The result of this dissertation tries to estimate that extent.

The creation of an architectural simulation model that experimentally depicts expected characteristics of a real parallel application, is a central issue of the research work development. A good simulation model should be able to explain available observations and aid in estimating performance, while abstracting away unimportant details. However, as parallel systems present a large number of possible variations, the models should be complete and flexible enough to be used in the simulation of different configurations of software.

4 METHOD FOR CARRYING OUT RESEARCH

4.1 Plan for evaluating research work

- *Propose Architectural patterns commonly used in parallel programming.* In parallel programming the nature of the problem to be solved is tightly related with the structure and behaviour of the program that solves it. In particular, the patterns of access to data, request of actions, and the characteristic of order of both actions and data, are indicators of how easy or difficult the development of the program will be. The research work then is undertaken with the aim of using architectural patterns that relate characteristics of order in data or operations of the problem with a proven solution.
- *Development of simulation models.* In building architecture practice, it is common the use of scale models to represent expected properties of a projected building before initiating a costly construction. The creation of such models is a basic and important activity during the early real architecture development. These models should always reflect relevant characteristics of the building. Most of these characteristics are based on architectural patterns, and come together with explanations of initial actions for the construction. In the same way, software scale models can be used to simulate certain expected properties of a software system.
- *Validation and evaluation of models.* The best way to learn aspects of performance estimation, as with any skill, is to apply the basic concepts to real problems. However, for a dissertation, this is too expensive and time consuming. Instead of this, and to corroborate the hypothesis, it is possible to get for each architectural pattern some already developed parallel applications, and obtain for each one a simulation model. Hardware and software characteristics are constrained by the application to be modelled. For the purposes of this work, modelling parallel systems focuses on the creation and coordination of multiple parallel components of an architectural pattern.
- *Analysis of performance characteristic estimation.* The effort spent developing one of these scale models, using it in the context of a measurement experiment to collect data, and finally critically analyzing the resultant data is probably the most effective way to learn the rudiments of performance estimation for parallel systems. The analysis focuses on accuracy of the estimation obtained from the model, in multiple cases. Also, it will include a search of possible causes and parameters of accuracy in the estimation.

4.2 Presentation of evidence of results to the research community

So far, work has been done proposing an initial set of architectural patterns commonly used in parallel programming, paying attention to the characteristic of order of data or operations on the data.

Currently, more work about architectural development is being carried out, involving modelling of the proposed architectural patterns for parallel systems and the creation of a simulation model that experimentally depicts expected characteristics of a parallel application. Parallel applications that reflect the proposed architectural patterns, or new ones, are being sought and studied according to the feasibility of being tested.

Future work includes finishing and validating models for at least two case studies per pattern, the evaluation of tests on performance of models and applications, and the analysis of estimation and accuracy.