Parallel Skeletons
In
Glasgow Distributed Haskell

Yong Lu

Master of Computer Science Department
Heriot-Watt University
2003

Abstract

Parallel programming is an important tool when trying to handle large problems that can not be dealt with using conventional single processors systems; either parallel machines or clusters of machines are used to provide the computational power needed. However parallel programming is an inherently difficult task and any tool or language that can provide suitable abstractions making the programming of these systems easier, without sacrificing the much needed performance, is of greatest importance. Skeleton programming provides a high level of abstraction, where common patterns (or skeletons) of programmes are constructed and later instantiated with the actual problem. "Skeleton Parallelism" is analogously skeletons for parallel systems.

Haskell is functional language at the cutting edge of language development, at the same time as it is a mature language system with a great range of libraries and support tools. There are several variants of Haskell that supports parallel and distributed programming and we are going to investigate how two of these can bee used with Skeleton programming. The two Haskell variants are Glasgow parallel Haskell (GpH) and Glasgow distributed Haskell (GdH), both are conservative extensions of Haskell but with the difference that the parallelism is implicitly controlled in GpH as opposed to the explicit control of GdH.

We have implemented two skeletons, Map and Divide and Conquer, in both GpH and GdH and for Map we have measured and analysed the runtimes of several variants on both regular and irregular data sets. For the Divide and Conquer skeleton we have performed the experiments only on regular data. The experiments suggested that GdH scales better with the number of processing elements but is at the same time less easy to use since it introduces more new concepts into the language.

Acknowledgements

Firstly, I would like to thank my supervisor, Dr Phil Trinder and J.H. Nystrom for all the invaluable help and advice he gave me during the course of the research reported in this dissertation.

Secondly, many thanks also to my family and friends for giving me support and their care and patient throughout this MSc.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(Yong Lu)