

Distributed event-based computing

Andrew BROWN^{a1}, David THOMAS^b, Jeff REEVE^a, Ghaith TARAWNEH^c,
Alessandro DE GENNARO^c, Andrey MOKHOV^c, Matthew NAYLOR^d and Tom
KAZMIERSKI^a

^a*Department of Electronics, University of Southampton, SO17 1BJ, UK*

^b*Electrical and Electronic Engineering, Imperial College London, SW7 2AZ, UK*

^c*Electrical and Electronic Engineering, University of Newcastle, NE1 7RU, UK*

^d*Computer Laboratory, University of Cambridge, CB3 0FD, UK*

Abstract. As computing systems get larger in capability - a good thing - they also get larger in ways less desirable: cost, volume, power requirements and so on. Further, as the datastructures necessary to support large computations grow physically, the proportion of wallclock time spent communicating increases dramatically at the expense of the time spent calculating. This state of affairs is currently unacceptable and will only get worse as exa-scale machines move from the esoteric to the commonplace. As the unit cost of non-trivial cores continues to fall, one powerful approach is to build systems that have immense numbers of relatively small cores embedded (both geometrically and topologically) in a vast distributed network of stored state data: take the compute to the data, rather than the other way round. In this paper, we describe POETS - **P**artially **O**rdered **E**vent **T**riggered **S**ystems. This is a novel kind of computing architecture, built upon the neuromorphic concept that has inspired such machines as SpiNNaker[1,2] and BrainScaleS[3]. The central idea is that a problem is broken down into a large set of interacting *devices*, which communicate asynchronously via small, hardware brokered *packets* (the arrival of which is an *event*). The set of devices is the **task graph**. You cannot take a conventional codebase and port it to a POETS architecture; it is necessary to strip the application back to the underlying mathematics and reconstruct the algorithm in a manner sympathetic to the solution capabilities of the machine. However, for the class of problems for which this approach is suitable, POETS has already demonstrated solution speedups of a factor of 200 over conventional techniques.

Keywords. Multicore/manycore systems, heterogeneous systems, accelerators

Introduction

Previously in this conference series[4] we have described the SpiNNaker system[1,2] consisting (when fully assembled) of just over a million ARM-9 cores, interconnected via a bespoke, high-speed, asynchronous network. The network is implemented as a flat triangular mesh of nodes (with 16 application cores at each node) with opposite edges identified such that the entire structure configures as a torus. SpiNNaker is a neural network simulator: the design intent is that each core hosts ~1000 neuron models, which communicate with each other using short (32 bit) hardware messages. The mapping of the (problem specific) neural network topology onto the *a priori* fixed communications network is achieved by a pre-processing stage

¹ Corresponding Author.