Special section on functional paradigm for high performance computing

Aleksander Byrski, Katarzyna Rycerz, John Hughes, Kevin Hammond

PII: S0167-739X(17)32083-6
DOI: http://dx.doi.org/10.1016/j.future.2017.09.035
Reference: FUTURE 3691

To appear in: *Future Generation Computer Systems*

Special Section on Functional Paradigm for High Performance Computing

Aleksander Byrski[1], Katarzyna Rycerz[1], John Hughes[2], Kevin Hammond[3]

[1] *AGH University of Science and Technology, Krakow, Poland*
{olekb,kzajac}@agh.edu.pl

[2] *Chalmers University, Göteborg, Sweden*
rjmh@chalmers.se

[3] *University of St. Andrews, Scotland, UK*
kevin@kevinhammond.net

Abstract

This paper is a foreword for the Special Section of Future Generation Computing Systems journal on Functional Paradigm for High-Performance Computing connected with Lambda Days 2017 Conference. In this paper the substance of the special section is located in the current state of the art and the overviews of the four papers constituting this special section are given.

Keywords: functional programming, high-performance computing

1. Introduction

Efficient programming for parallel execution on many core architectures and on emerging distributed hardware platforms leads to issues related to a proper synchronization between threads and processes, development of robust communication protocols, maintaining scalability and resilience of the developed applications and software systems [1, 2] and has been developed for over the last decade with increasing frequency (c.f., e.g. [3, 4, 5, 6]). Many mechanisms to address these issues are available in the functional programming paradigm. Inherent capabilities of functional languages, such as referential transparency, lazy evaluation, and the absence of (or restricted) side-effects in purely functional languages make them perfect as a means for implementation of concurrent algorithms on distributed systems [7]. Moreover, many functional languages provide their users a reliable asynchronous