

# 1 Dissertation Review

The two dissertations I have chosen to review are *Assessing the Performance of Optimised Primality Tests* (2016) by Cameron Curry, and *Extending ePython to support parallel Python interpreting across multiple interconnected Epiphany processors* (2017) by Dongyu Liang. I have chosen these dissertations because they are relevant to my own project. Liang's *Extending ePython* is focused on a programming language that is not typical in the HPC space, just my project will. Curry's *Optimised Primality Tests* compared implementations of a core HPC function. My project will compare my rust implementation of some HPC code, with its original C implementation. In this review, I will summarise both dissertations, and then discuss what features of the dissertation I should emulate or avoid.

Liang's *Extending ePython* chiefly aims to extend ePython, a python interpreter for the Epiphany processor, to "support parallel programming on multiple interconnected Epiphanyes". Liang first presents the technologies which they will be using: the parallela board, which hosts the ephipany and is novel for its exceptionally high GFlops per watt rat, and the ePython architecute, including its seperate interpreters and its communication mechanisms.

Liang goes on to describe the construction of their Epiphany cluster, using Parallela boards and SSHFS, which processes take place on the host, (the ARM CPU) and which processes take place on the device (the Ephiphany cores, or e-cores), and how communication between processes occurs. Communications between cores have been extended so that they are node agnostic, and include `send()`, `recv()` and `reduce()`. These ePythons methods sit upon the author's code, and MPI.

In their results section, Liang provides proof of the effectiveness of their implementation, through listings, tables and graphs. Figure 4.5 is of particular note, as it shows ePython has a very good parallel efficiency, although it is very strongly impacted by communicating with nodes on other devices.

Liang's dissertation shows that they have made a novel, and valid contribution to the territory. Their technical achievement in extending ePython's parallelism to cover many nodes should be applauded.