SAND2018-7791C

Learned P-States for High Performance Computing

Peter Feghali, Sam Green

Mentor: Craig Vineyard

Overview

As High Performance Computing (HPC) continues to scale, new problems begin to arise. One of the most challenging is power, since as computing power increases more electrical power is needed to run these systems. While the rise in power requirements is inevitable, we must lower power wherever possible while maintaining high performance. We have decided to examine dynamically modifying processor P-states to lower power requirements during run-time.

Historically this has been done through either explicit modification of code regions or simple, heuristic based approaches. These options require substantial effort, training, and slow development cycles. We have decided to approach this problem through reinforcement learning, a process which allows the system to explore its environment and find an optimal power policy by itself. This process allows for the learning algorithm to implicitly examine the HPC nodes and learn how to predict optimal P-states, potentially learning to consider patterns and correlations that humans could not.

Approach

Deep reinforcement learning is a method of teaching systems to learn based off of rewards. The system learns a ‘policy’, which in this case is a deep neural network, to make decisions and interact in its environment

It learns this policy by slowly learning which actions are ‘good’ and which are not. Initially, the policy will randomly choose actions. As it progresses through training, the policy will move the system through the action space, and hopefully determine a policy which brings it to equilibrium.

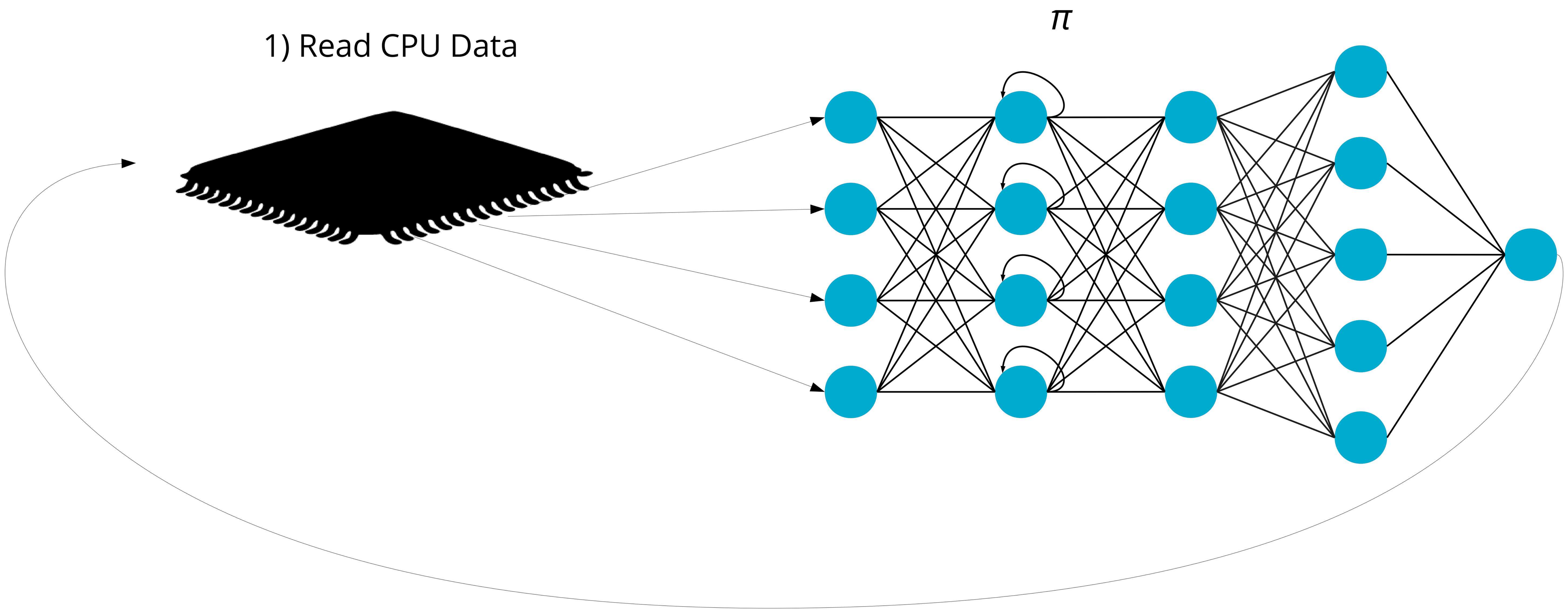
For this problem our action space is a discrete number of set p-states, defined by each individual system. The reward for the policy is a function of runtime performance – measured by time to completion – and power used.

Data Collection

Attempting to change P-states live requires a substantial amount of system data to make informed decisions. We have decided to use Model-specific register (MSR) data which can safely be read from each processor core. We can determine running frequencies for each core, power, DRAM Energy, and much more. We may use this in-band data collected in-system with out of band data collected by external monitoring solutions.

It is not only important to collect data, but also to do so at a rate which makes sense. In some cases, subprocesses could only be live for fractions of a second, while others subsequently could be running for a few seconds. In addition, the more we sample in-band data, the more we interrupt the processor. Deciding on a frequency which allows for an effective network while negligibly impacting performance is a substantial problem which will be solved through experimentation.

Process



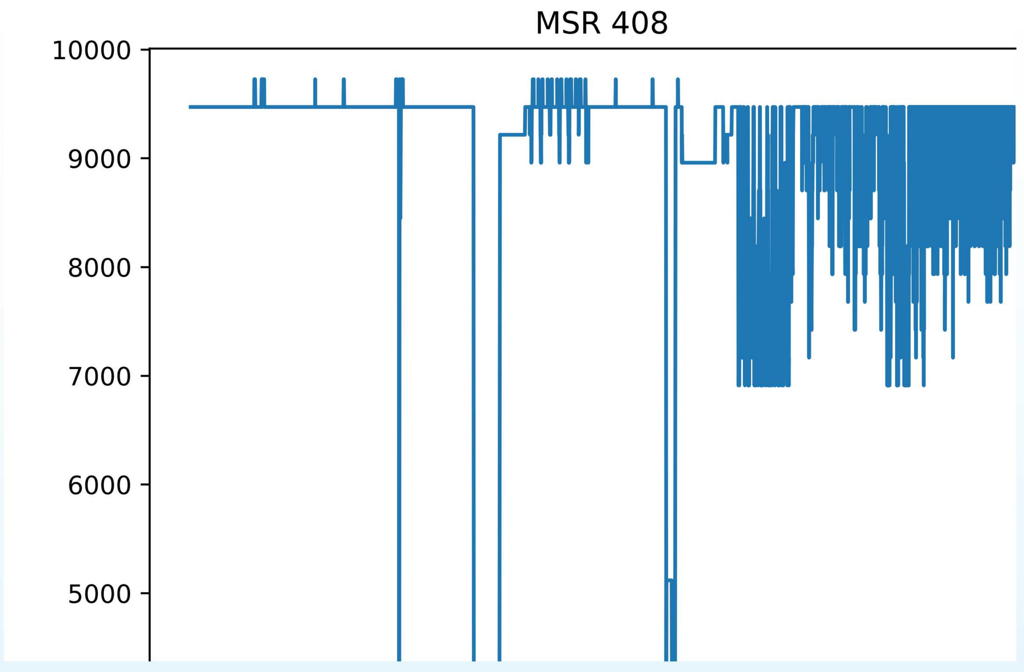
2) Next we enter the MSR data into a neural network. We are planning on using a Long Short-Term memory (LSTM) architecture and a Multilayer Perceptron (MLP) network. This is also known as the learned ‘policy’ π that the reinforcement learning agent learned. The policy will then decide on an action.

Impact

Saving power has obvious benefits, in reducing costs and increasing cluster accessibility. In addition to these benefits, if we can reduce power usage by 5%, a cluster could potentially increase it’s computing capability as a whole.

Saving power is also a preventative measure. High Performance Computing will continue to scale and will soon reach substantial power levels (~40MW). At these levels, failures in any part of the system can be catastrophic without requisite infrastructure updating.
By strengthening our power requirements and policies, we can prevent against these failures and improve computing capability.

MSR 408 – Performance Status



Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC,
a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy’s National Nuclear Security Administration under contract DE-NA0003525.
SAND No. ____________