ECP PowerSteering: Updated power model

1. Introduction

This document describes the updates to the power model used by the configuration space exploration logic of the ECP PowerSteering runtime project.

2. Model description

Assume we have generated the quadratic model for the Execution time-power correlation of the processors. For all processors, the execution time-power relationship can be described in the following form:

$$P = A_\alpha(1/E)^2 + C_\alpha,$$

where, *P* represents power and *E* represents execution time, the coefficients are specific to applications. Factor *α* is a real number in [0, 1] representative of each processor. We use α = 1 and α = 0 to represent the best/worst-performing processor respectively. Based on the assumption that the relative ranking of the processors stays constant over different processor power limits and on different computation kernels, we assume

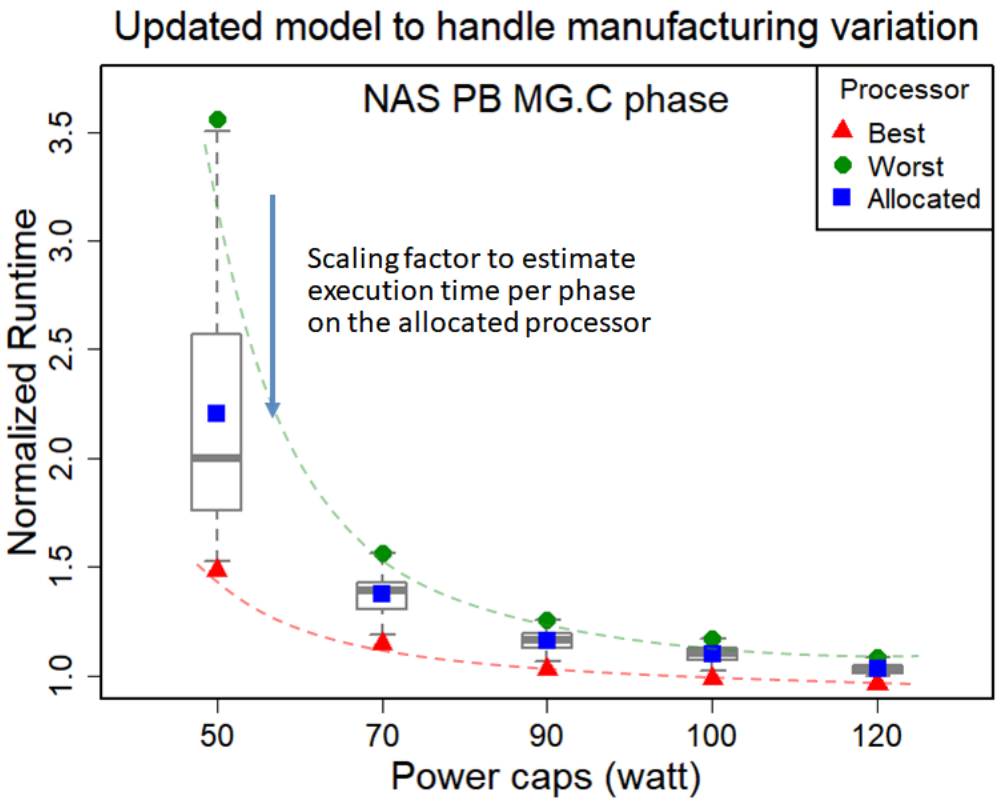
$$A_\alpha = \alpha A_1 + (1 - \alpha)A_0,$$

$$C_\alpha = \alpha C_1 + (1 - \alpha)C_0$$

Our target problem is to partition a fixed total power *P* for an application to the processors in a system with *m* processors, and simultaneously try to minimize the total execution time of each computation phase on all allocated processors.

3. Updates to the Configuration Exploration step of Conductor

a) **Description**: The following plot shows the distribution of execution times of an arbitrary processor in the job allocation relative to the best-case and worst-case processor on Quartz which is a Broadwell-based cluster at LLNL. The plot shows the main compute phase of MG.C. The runtime system uses a non-linear performance/power model generated based on empirical data over 5 applications at different power caps. Each MPI rank uses this non-linear model to adjust the performance data it obtains (through MPI_Allgather) from other MPI ranks during the configuration exploration phase.



b) **Model accuracy**: Since the model is based on offline empirical data, the accuracy of the model for unseen phases is not very high especially at lower power caps. Improving the accuracy is work-in-progress: the system currently uses scaling factor relative to the worst-case processor (green) than the best-case processor (red) as it results in better accuracy. This needs to be changed to a classification-based model based on the distribution of the allocated processors.

c) **Updates to the configuration space exploration logic**: We observed that a relatively small error in performance prediction can result in a significant degradation in performance of the associated phase at scale (repetitive degradation of all time steps). To minimize this degradation, we modified Conductor's configuration exploration logic to update the predicted performance of the application phase in-place if the observed performance is higher or lower than the predicted performance by more than 10%. This threshold is configurable and needs to be configured for each application. We plan to improve this mechanism to automatically select a threshold for each application based on on-line monitoring of phases.