Haidar et al. proved that the power consumption can be reduced by setting the power cap with minimal or no penalty in the runtime of the applications.

Appropriately adjusting the power draw of computational hardware plays a crucial role in its efficient use. While the compute power of an instance $P_i$ is given as $P_i = p_i - P_m$, we expect it to be adjusted in the operational intensity $I_m$. Therefore, the short-term power limit $l_i$ is not adjusted. The hardware classification and its variables is found in the 1.5 s interval of $0.5 \leq t$.

A processor’s state is recognizable by tracing hardware events, hence the signatures might depend on the power step $\Delta P$. Therefore, the short term power limit $B_i$ is non-adjusted. Other factors the signatures might depend on: intensity is constant, a steady state hardware energy signature like shown in fig. 4 is to be assumed. Other factors the signatures might depend on.

MEASURING TRAINING DATA FOR THE MACHINE LEARNING CLASSIFIER

**UTILITY FUNCTIONS**

We suggest the following use-cases for online power capping:

**OUTLOOK**

**LIMITS**

- Complex classifiers significantly increase the overhead of prediction.
- No data-driven approach is leading.
- No data-driven approach is leading.
- Online or cluster-wide approaches based on machine learning may enhance hardware signature and load balancing.

**COMPUTATIONALLY INFORMED SOFTWARE AGNOSTIC**

A processor’s state is recognizable by tracing hardware events, hence computationally informed algorithms, state change estimations devised on the operational performance and its problem size, in software-aware approaches will benefit even higher energy savings.

**HARDWARE SIGNATURES**

We propose the following use-cases for online power capping:

**RESULTS**

**PERFORMANCE OVERHEAD**

- Different architectures and core counts to be implemented.
- Following the target platform of multi-core architecture for computers to meet a changing compute-center wide power target.
- Minimizing energy consumption for temporally varied applications.
- Wall-time aware energy savings.
- Energy saving in a fully utilized cluster.
- Energy savings by power capping memory.
- Hardening requirements.
- High sampling rates improve the fidelity of power capping.
- Running the complete set of experiments with the same set of counters.

Our approach is a software agnostic one and we recommend an on-package classifier firmware like shown in fig. 9 which is based on fig. 2 to minimize the overhead of classifying the time series of the performance counters.

**CLASSIFIER Firmware**

Power capping is a key-topic for a power configuration of a hardware architecture, but it shows a consistent behavior in memory-bound situations.

- Idealized Power Delay Profile (PDP)
- Online power capping
- Readily available hardware counters, traced with 5 Hz
- x86 64 Bit, accelerators pending
- Socket-wide power capping with on-package classification

**CONCLUSION**

Online Power Capping by Computationally Informed Machine Learning can reduce power and energy consumption with minimal to no loss in output.

**ACKNOWLEDGEMENTS**

This research was supported by the National Science Foundation Office of Advanced Cyberinfrastructure (OAC) CSE Dir. for Comp. & Info Sci. & Eng. under Grant No. 1913800.