Component-Based Computation-Energy Modeling for Embedded Systems

A. Seewald[1], U.P. Schultz[1], J. Roeder[2], B. Rouxel[2], and C. Grelck[2]

[1]University of Southern Denmark; [2]University of Amsterdam
Correspondence to: ads@mmmi.sdu.dk
https://bitbucket.org/adamseew/powprofiler

Context

We are interested in parallel software featuring independent tasks that runs as a result of different computations being executed together. Component–based development can reduce overall complexity in such a scenario, although, its adoption in embedded domain remains marginal [2], as non–functional properties satisfiability poses a major obstacle [1]. We here present a component–based energy modeling approach abstracting per–component energy in a dataflow computational network executed according to a given scheduling, with the goal of predicting and optimizing the energy efficiency of power–critical devices.

Research Questions

? How precise can an empirical energy model be, when based on profiling of different computational units, such as CPU and GPU?

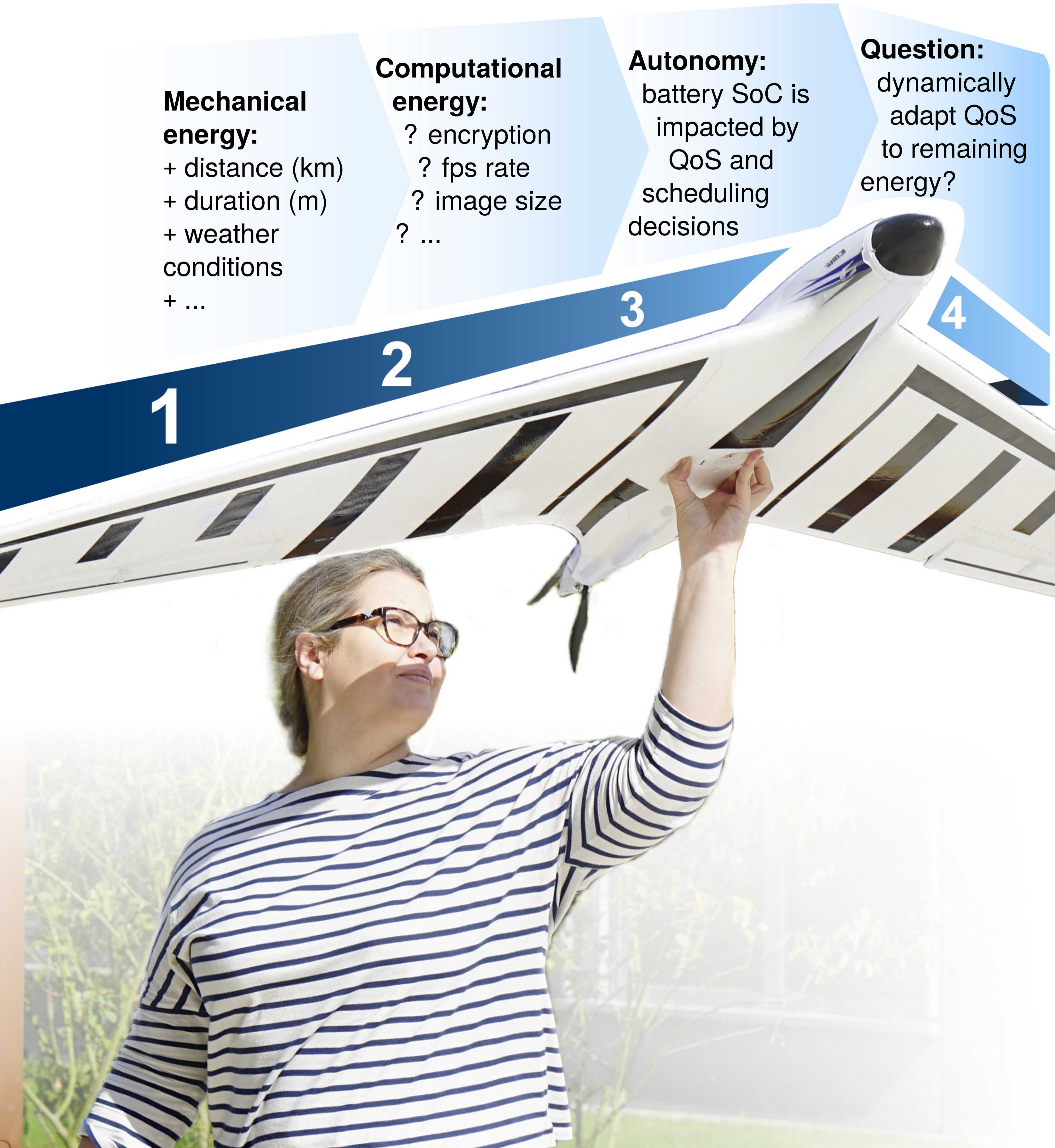
? What is the best way of doing energy modeling, i.e., at application level (coarse–grained) or at component level (component–based)?

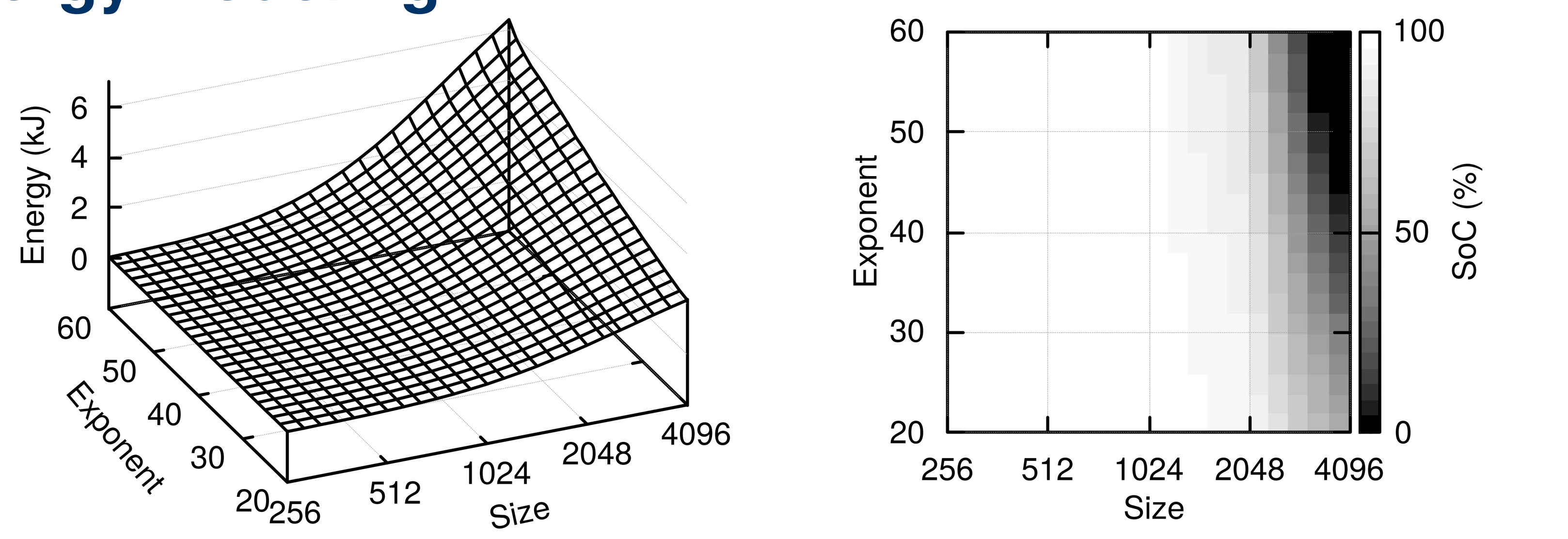
? How much can we extend the flying time of a drone by selecting the best quality of service (QoS) and scheduling policy given a flight plan and an energy budget?

Approach

We propose a component–based energy modeling approach to assess the energy efficiency of power critical devices, built upon our previous work on coarse–grained modeling [3]. The approach enables an energy–aware scheduling technique, where the scheduler generates the best possible schedule and configuration that meets the requirements. Our work, implemented in a profiling tool, is part of the TeamPlay project. TeamPlay aims to develop formally–motivated techniques to address non–functional properties satisfiability [1].

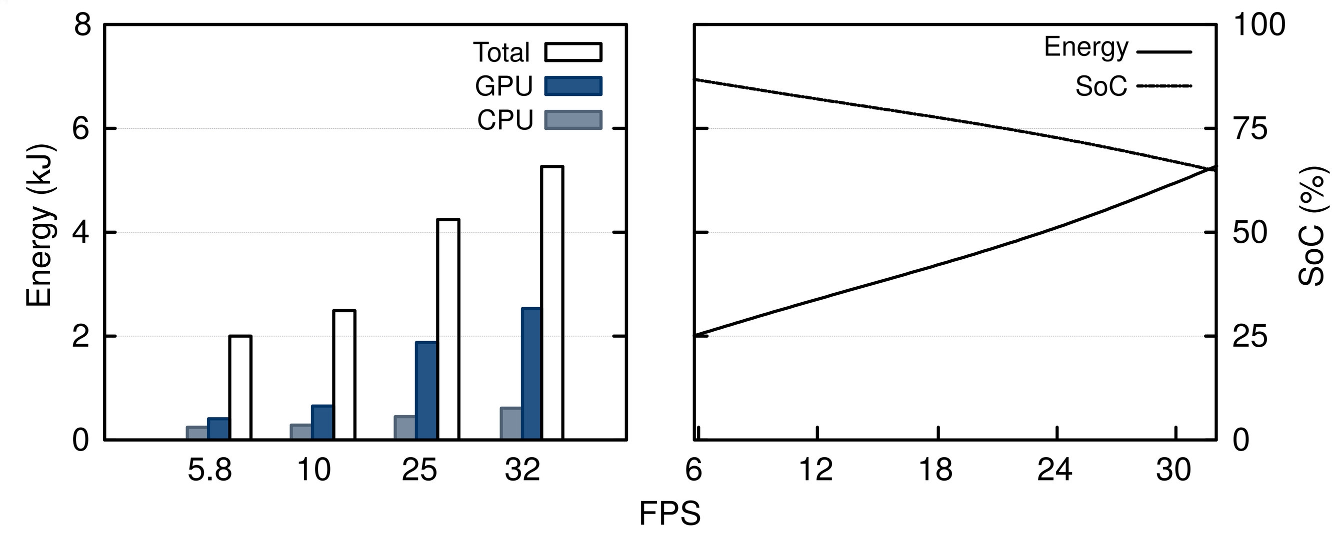
Computation-Energy Modeling





Energy and SoC. Overall energy consumption and state of charge (SoC) of a matrix exponentiation component while varying size and exponent [3]

In coarse–grained modeling a discrete set of configurations are profiled and a model is generated by varying different component configuration parameters, measuring the overall power consumption, and approximating any missing values. In component–based modeling dependencies between components are handled by a static scheduling algorithm during profile–time. The static schedule runs components sequentially and triggers profiling such that a per–component energy model can be generated.



Darknet. Overall energy consumption and state of charge of varying QoS parameter frames-per-second of the darknet deep neural network object detection algorithm [3]

Base energy can be subtracted from the per–component energy measure to obtain a component energy cost value. In this way we can considerably reduce profiling time by adding costs one on the top of each other to model components in parallel.

Opterra drone. An Opterra fixed–wing drone that can take advantage of compoent–based modeling for dynamic QoS and scheduling adjustments during flight

Use–case: drones

Energy models are of particular interest for mobile robots, as they often present a constrained power budget. To investigate this, we developed a **drone maritime** use–case based on a coarse–grained component model. Different components, like edge detection and base station communication, cooperate in the sense that an image is stored for further analysis if an object is detected by the drone flying over the sea. The modelling strategy consists of defining different acceptable quality of service parameters per component to specify along with a non–functional properties' definition.

Battery State of Charge (SoC)

A robust correlation to real scenarios is achieved by modeling the battery state of charge or SoC, as an abstraction for handling energy efficiency of power–critical devices. Our profiling tool features a SoC model that can be used for an energy–efficient scheduling policy. The choice of optimal policy can considerably impact battery life. For instance, a stable power drain on a battery can drain less energy, compared to distributing the same computation in a smaller number of power–intensive operations.

Acknowledgements

This work is supported and partly funded by the European Union's Horizon2020 research and innovation program under grant agreement No. 779882 (TeamPlay).

References

[1] Public deliverables of the TeamPlay Horizon2020 project. https://teamplay-h2020.eu/index.php?page=deliverables, 2019. Accessed: 2019-08-25.

[2] Ivica Crnkovic. Component-based software engineering for embedded systems. In *Proceedings of the 27th international conference on Software engineering*, pages 712–713. ACM, 2005.

[3] Adam Seewald, Ulrik P Schultz, Emad Ebeid, and Henrik S Midtiby. Coarse-grained computation-oriented energy modeling for heterogeneous parallel embedded systems. Manuscript submitted for publication, 2019.