Energy optimization on MPSoC with convex framework

E. Nogues, A. Mercat, F. Arrestier, M. Pelcat, D. Menard, T. Amestoy
UEB - INSA Rennes - IETR, France

Context and motivations

- Energy optimization in modern MPSoCs
  - Availability of advanced power management techniques
    - Dynamic Voltage and Frequency Scaling (DVFS)
    - Dynamic Power Management (DPM)
- Objective : provide guidelines gathering in a **single framework**
  - **Real-time** requirements
  - Low power mechanisms
    - **Frequency scaling** and **deep sleep modes**
  - **Parallel** programming and application scheduling properties

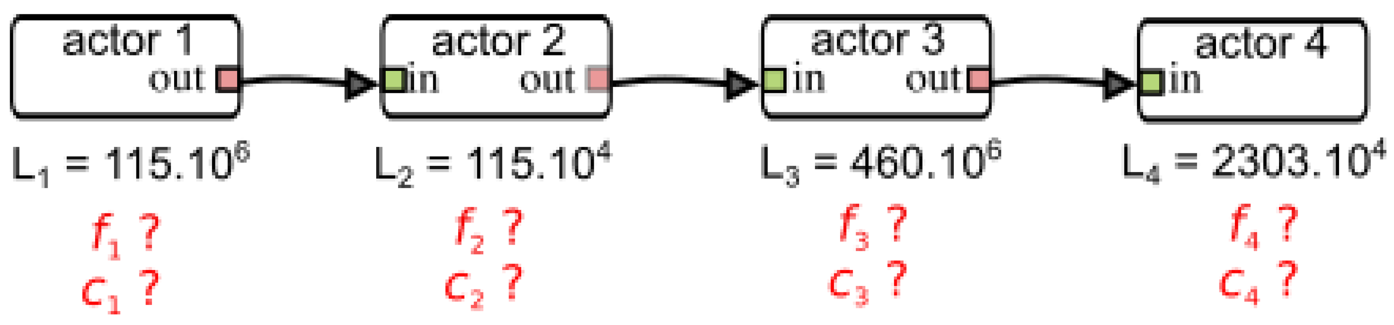
Optimization Problem in the General Case

- Minimization of the total energy E_{tot} under real-time constraint D

$$\min_{f,c} \quad E_{tot}(f,c) \qquad \text{subject to} \qquad T_{tot} \leq D$$

- f : **processing frequency** , $f_{min} \leq f \leq f_{max}$
- c : **number of cores**, $c_{min} \leq c \leq c_{max}$
- T_{tot} : application execution time
- D : deadline

- Optimization Problem for Streaming Signal Processing Apps
  - Sequence of N *actors* with a known load $L_{i|i=1..N}$ in cycle count



⇒ Find the best frequency f_i and parallelism level c_i for each *actor* i

- Use of normalized values ($]0,1]$) for f_i and c_i

- Total execution time model

$$T_{tot} = \sum_{i=1}^{N} \frac{L_i}{f_i . S_i(c_i)}$$

- S_i Speed-up model for actor i
  - Perfect speed-up : $S_i(c_i) = c_i . N_c$ with N_c the maximal number of cores
  - Not-perfect speed-up model : $SU(c) = k_0 . c^{0.25}$ with k_0 from the app.

Energy model

- Total Energy model

$$\sum_{i=1}^{N} \frac{L_i}{f_i . S_i(c_i)} . E_{cycle}(f_i, c_i)$$

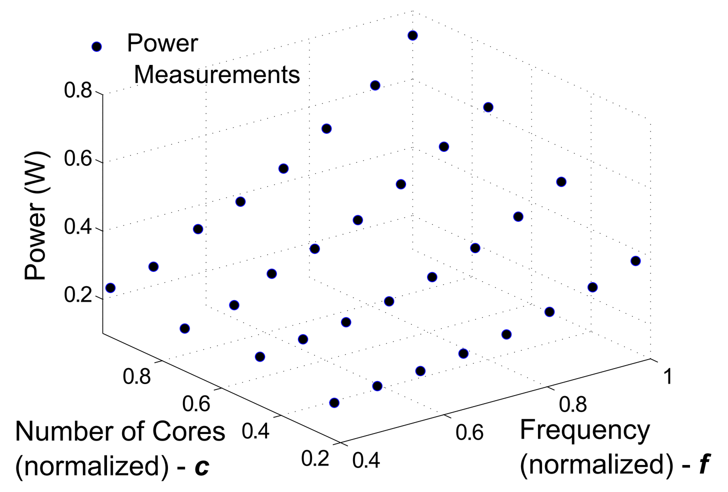
- **Energy** per cycle count (DPM) processing at f (DVFS) on c cores in parallel

$$E_{cycle}(f,c) = \frac{1}{T.f} . \int_0^T P_{tot}(t,f,c)\,\mathrm{d}t$$

⇒ Energy model has convexity properties

- Polynomial approximation of the power from measured data
  - Curve fitting with linear regression
  - Constrain the energy model as a *posynomial*

$$P(c,f) = \sum_{i=0}^{N} \sum_{j=0}^{M} a_{i,j} c^{\alpha_i} f^{\beta_j}$$



- $a_{i,j} \in \mathbb{R}_+$, $\alpha_i \in \mathbb{R}$, $\beta_j \in \mathbb{R}$.
- Platform : Exynos 5420

Final Optimization Problem

- Not-perfect scaling

$$\begin{aligned} \underset{f_i, c_i}{\text{minimize}} \quad & \sum_{i=1}^{N} \frac{L_i}{k_0\, c_i^{\,0.25}} (a_0 \frac{1}{f_i} + a_1 \sqrt{f_i} c_i^{1.5} + a_2 + a_3 f_i^2 c_i + a_4 f_i^6 c_i) \\ \text{subject to} \quad & \sum_{i=1}^{N} \frac{L_i}{k_0\, c_i^{\,0.25}} . \frac{1}{f_i f_{max}} \leq D \\ & f_i \geq \frac{f_{min}}{f_{max}},\ f_i \leq 1 \\ & c_i \geq \frac{c_{min}}{c_{max}},\ c_i \leq 1 \end{aligned}$$

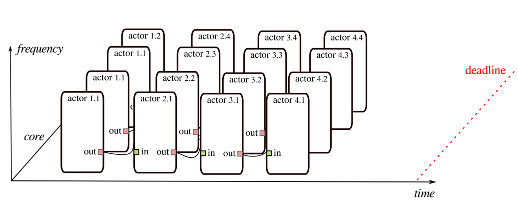
$$a_{0..4} = [0.0313, 0.2057, 0.0815, 0.2515, 0.1242]$$

- Geometric Programming
  - Transform to convex optimization problem via change of variables → Use of logarithm with **Geometric Programming**

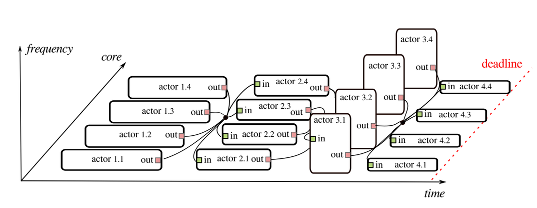
Experiment on a streaming application with 4 actors

- 3 configurations have been tested

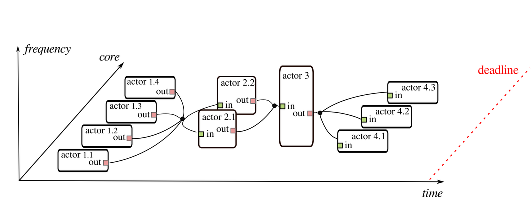
As-Fast-As-Possible scheduling



As-Slow-As-possible scheduling



proposed scheduling



- Loose Scheduling - Deadline = 1.2

Perfect

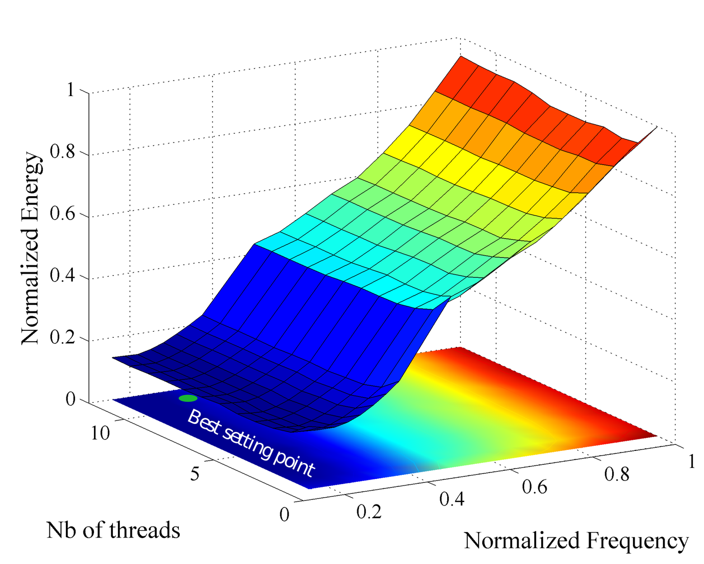
	Time (s)	Norm'd Freq.	Norm'd Cores	Energy (J|norm)
ASAP	0.60	[0.4 0.4 0.4 0.4]	[1.0 1.0 1.0 1.0]	0.2269 | 1.000
AFAP	0.25	[1.0 1.0 1.0 1.0]	[1.0 1.0 1.0 1.0]	0.2191 | 0.965
Our method	0.44	[0.5 0.5 0.5 0.5]	[1.0 1.0 1.0 1.0]	**0.1976** | 0.870

Not-Perfect

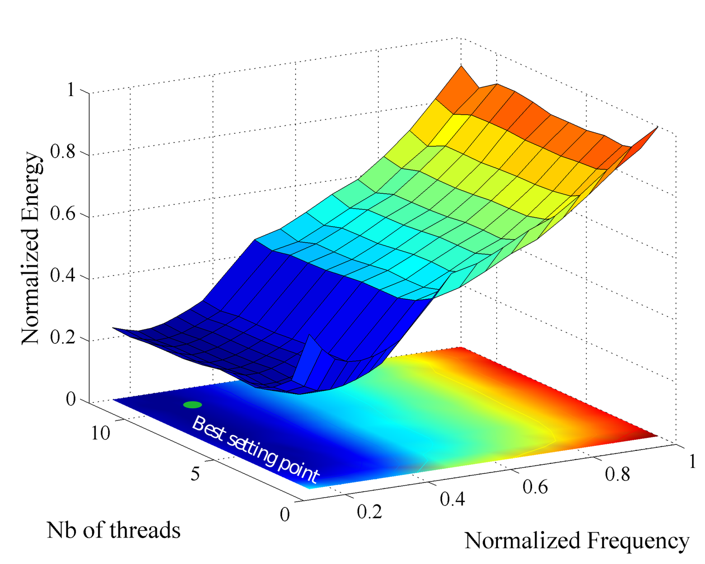
	Time (s)	Norm'd Freq.	Norm'd Cores	Energy (J|norm)
ASAP	1.20	[0.4 0.4 0.4 0.4]	[1.0 1.0 1.0 1.0]	0.4538 | 1.000
AFAP	0.50	[1.0 1.0 1.0 1.0]	[1.0 1.0 1.0 1.0]	0.4381 | 0.965
Our Method	0.82	[0.7 0.7 0.7 0.7]	[0.4 0.4 0.4 0.4]	**0.3563** | 0.785

Experiments on offline HEVC decoder

Random Access profile - Core Energy



Random Access profile - Total Energy



Optimal		Gains (%)			
f_{proc}	p_{thread}	p_{min} - f_{min}	p_{min} - f_{max}	p_{max} - f_{min}	p_{max} - f_{max}
350	11	52.3	76.8	12.7	71.5

Conclusion

- **Frequency scaling**, **Deep Sleep** modes and **Parallelization level** can be jointly optimized wrt :
  - **Real-time** requirements
  - **Low power characteristics** of the platform
  - **Parallel** programming and scheduling properties of the app.
- Gains compared to traditional approaches

PROJET FINANCÉ PAR L'ANR - PROJECT FUNDED BY THE ANR