Improving the Performance and Energy Efficiency of HPC Applications Using Autonomic Computing Techniques

Eric RUTTEN, INRIA Grenoble, Ctrl-A team
Outline

**Project outline**

**Autonomic Computing & feedback loops in HPC**

**Adapting performance & energy in HPC**

**Preliminary results controlling RAPL**

**Perspectives more feedback loops**
Outline

**Project outline**

Autonomic Computing & feedback loops in HPC

Adapting performance & energy in HPC

Preliminary results controlling RAPL

Perspectives more feedback loops
JLESC project outline

Improving the Performance and Energy Efficiency of HPC Applications Using Autonomic Computing Techniques

Topics Advanced Architectures

Keywords autonomic computing, energy efficiency

Members

• Eric Rutten (INRIA) + B. Robu, M. Berekmeri
• Swann Perarnau (ANL) + V. Reis, K. Yoshii, …

Since 2018
JLESC project outline (ii)

Problem
• perf., power, thermal, … increasingly unpredictable
• some HW runtime mechanisms available

Approach
• SW control, application-aware, on top of HW
• feedback loop(s) : Autonomic Computing and Control Theory
Outline

Project outline

**Autonomic Computing & feedback loops in HPC**

Adapting performance & energy in HPC

Preliminary results controlling RAPL

Perspectives more feedback loops
Automated administration & regulation in reaction to variations in load, resources, … in large (Big Data) or embedded (IoT) systems

self-*: deploy, mgmt, healing, protection

promising, but challenge in developing systems need for automation & separation of concerns

Understand and design control for problems in efficiency (e.g.; energy) & assurances (e.g. crash avoidance)
Example 1: DPR FPGA control

context-aware reconfiguration management & control:

joint work with S. Gueye, J.Ph. Diguet (LabSticc, Lorient)  
[AHS17, ICAC18]

insuring: for task/operation, choice of good enough bitsream version, w.r.t. given requirements,
following measured metrics

notifying: metrics, …
in case of impossibility,
to be managed at upper level by tasks or reqs change

what: tasks/ops
how: QoS reqs
notifications metrics

bitstreams ids on/off
end metrics
Example 2: SW level management

e.g., Parallelism vs. synchronization

joint work with N. Zhou, J.F. Méhaut, G. Delaval, B. Robu

[CCPE 18]

dynamical management of trade-offs: speedup / consistency

- too much parallelism: overhead & slow down
- too low parallelism: poor performance

Example 2: SW level management

commits aborts time

decision

commit ratio (CR)

trigger range, ...

\( n_{opt} = \text{proba}_fct(CR,\ldots) \)

inc/dec #threads + mapping, profile

TinySTM Multicore
Example 2bis : SW level management

minimizing underuse of infrastructure

joint work with O. Richard (DATAMOVE), B. Robu (Gipsa-lab) [AIScience@HPDC18]

on top of grid platform with OAR scheduler

Gigri : injection of smaller/indepdt jobs, avoiding overload

runtime regulation

• measure of platform stress
• control : model-based
• considering storage

load/stress busy jobs

decision thresholds, PID, MPC

OAR grid

jobs queue
Example 3: multiple loops coordination

with S. Gueye, N. de Palma, A. Tchana, N. Berthier [FGCS 14, IEEE TSE16]

Self-sizing & self-repair & consolidation in Multi-tier Cloud

intuition avoid interference/redundancies between loops suspend downstream mgrs when upstream busy

model : activity state of mgrs (FSM)

multiple loops
Outline

Project outline

Autonomic Computing & feedback loops in HPC

*Adapting performance & energy* in HPC

Preliminary results controlling RAPL

Perspectives more feedback loops
Performance and Energy in HPC

HPC systems need power management:

- Facilities need to control for max power, or swings in power consumption
- Manufacturing variability: all nodes don’t have the same power/performance

Advanced Workloads (workflows, in-situ)

- Node level: workload might not need full CPU power
- Across nodes: workload imbalance, variability can be improved by power shifting.
Infrastructure for Control

Node Power/Performance Management:

- Node-local daemon with access to power and performance monitors and controls
- Launched by users, no root access needed ideally

General resource management design:

- Acts as a customizable control loop inside user jobs
- Can be connected to job launcher, performance APIs.
Outline

Project outline

Autonomic Computing & feedback loops in HPC

Adapting performance & energy in HPC

Preliminary results controlling RAPL

Perspectives more feedback loops
Target system considered:

- **HW**: platform with *power capping*: RAPL
  - for a power cap $P_{cap}$, given as input internal DVFS regulation around / close to $P_{cap}$
  - output: actually used power: $P_{u}$
- **SW**: application with *measure for progress*
  - based on heartbeat or iteration count
Autonomic Computing for Power Management in HPC (ii)

Controlled system:

• closing the feedback loop to regulate the Pcap according to objective

• objective: keeping $Pcap$ minimal for a maintained performance

$Pcap = f(Pr, Pu)$

SW RAPL

control

Pr

Pu
Autonomic Computing for Power Management in HPC (iii)

Designing controllers:

- simple ones, intuitive
  - scanning
  - hill-climbing
- control theory
  - PID (Proportional, Integral, Derivative)
  - MPC
    (Model predictive control)

\[
P_{\text{cap}} = f(Pr, Pu)
\]

(\text{stateful})

\[
\text{SW RAPL}
\]
Outline

- Project outline
- Autonomic Computing & feedback loops in HPC
- Adapting performance & energy in HPC
- Preliminary results controlling RAPL

Perspectives more feedback loops
Perspectives

Short term: finalize the design of controllers
range from simple intuitive algorithms, to model-based

Experimental evaluation characterize/compare w.r.t.
• ease of use / design
• gain in power consumption
• properties of the controllers (convergence, stability).
Perspectives (ii)

Longer term:

• characterize applications / controllers relationship

• coordinate multiple loops for other system features
  e.g. thermal aspects, parallelism, storage, …

• hierarchical or distributed composition of
  multiple loops for large / complex systems

(e.g. involving Dynamically Partially Reconfigurable FPGA)