



Paving the Way Toward Energy-Aware and Automated Datacentre

Conference Paper**Author(s):**

Bartolini, Andrea; Beneventi, Francesco; Borghesi, Andrea; [Cesarini, Daniele](#) ; Libri, Antonio; [Benini, Luca](#) ; Cavazzoni, Carlo

Publication date:

2019

Permanent link:

<https://doi.org/10.3929/ethz-b-000388104>

Rights / license:

[In Copyright - Non-Commercial Use Permitted](#)

Originally published in:

ACM International Conference Proceeding Series, <https://doi.org/10.1145/3339186.3339215>

Paving the Way Toward Energy-Aware and Automated Datacentre

Andrea Bartolini
DEI, University of Bologna
a.bartolini@unibo.it

Daniele Cesarini
DEI, University of Bologna
daniele.cesarini@unibo.it

Francesco Beneventi
DEI, University of Bologna
francesco.beneventi@unibo.it

Antonio Libri
a.libri@iis.ee.ethz.ch
IIS, ETHZ Zurich

Carlo Cavazzoni
SCAI, CINECA
c.cavazzoni@cineca.it

Andrea Borghesi
DISI, University of Bologna
andrea.borghesi3@unibo.it

Luca Benini
lbenini@iis.ee.ethz.ch
IIS, ETHZ Zurich

ABSTRACT

Energy efficiency and datacentre automation are critical targets of the research and deployment agenda of CINECA and its research partners in the Energy Efficient System Laboratory of the University of Bologna and the Integrated System Laboratory in ETH Zurich. In this manuscript, we present the primary outcomes of the research conducted in this domain and under the umbrella of several European, National and Private funding schemes. These outcomes consist of: (i) the *ExaMon* scalable, flexible, holistic monitoring framework, which is capable of ingesting 70GB/day of telemetry data of the entire CINECA datacentre and link this data with machine learning and artificial intelligence techniques and tools. (ii) The exploitation of *ExaMon* to evaluate the viability of machine-learning based job scheduling, power prediction and deep-learning based anomaly detection of compute nodes. (iii) The viability of scalable, out-of-band and high-frequency power monitoring in compute nodes, by leveraging low cost and open source embedded hardware and edge-computing, namely *DiG*. (iv) Finally, the viability of run time library to exploit communication regions in large-scale application to reduce the energy consumption without impairing the execution time, namely *COUNTDOWN*.

KEYWORDS

HPC, Energy Efficiency, Quantum Espresso, Big Data, Anomaly Detection, Artificial Intelligence, Datacentre automation

ACM Reference Format:

Andrea Bartolini, Francesco Beneventi, Andrea Borghesi, Daniele Cesarini, Antonio Libri, Luca Benini, and Carlo Cavazzoni. 2020. Paving the Way Toward Energy-Aware and Automated Datacentre. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference'17, July 2017, Washington, DC, USA

© 2020 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM... \$15.00
<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 INTRODUCTION

On the race toward exascale, high-performance computing systems are facing essential challenges that limit their efficiency. Among all, power and energy consumption fueled by the end of Dennard's scaling start to show their impact on limiting supercomputers peak performance and cost-effectiveness. Also, the reliability and the security of the computing systems, Hardware (HW) components as well as Software (SW) components pose novel challenges on the management of the system at large. Overall, it is a daunting task for system administrators, and users to optimise supercomputer/jobs performance and power consumption, identify anomalous behaviours, faulty situations, and guarantee systems operating in optimal conditions[27].

Datacentre automation aims at combining control theory, artificial intelligence and big data technologies toward the automation of the datacentre management process. To pave the way toward datacentre automation a set of steps must be undertaken.

The first step to obtain datacentre automation is to implement a monitoring framework with a high level of detail and granularity, that can be used to characterise the target system. The system-level data collection infrastructure has to be scalable, capable of handling a large amount of information, thus big-data oriented but also suitable to be connected with information extraction level. With the wealth of collected data, it is possible to create a virtual model that describes and behaves similarly to its physical counterpart, and that can be used for automated processes and predictive, maintenance. For examples, if we infuse reasoning capability (via Artificial Intelligence (AI) techniques such as Machine Learning) in it, we can automatically detect faults or anomalous conditions disrupting the normal behaviour of the supercomputer. Moreover, AI approaches can also be used for improving the general system management, i.e., improving the scheduling and resource allocation policies based on predicted evolution of the system. The monitoring infrastructure and the added AI can be hosted on the supercomputer itself, thus creating a self-monitoring and -adapting system."

The second step is to obtain a job-level monitoring framework and run-time suitable to intercept the application characteristics and leverage it to reduce energy consumption. This job-level monitoring framework needs to be low overhead, transparent to the user and thus capable of preserving the normal application flow.

In this paper, we report the concrete action and research outcome performed to improve the energy efficiency and the automation of the CINECA datacentre. Finally, we will conclude describing how these lessons learnt have shaped its future agenda.

In Section 2, we introduce the *ExaMon* holistic and scalable monitoring framework. In Section 3, we present how AI can be coupled with *ExaMon* to deliver datacentre automation. In Section 4, we introduce edge computing as a way to overcome technological challenges in dealing with fine-grain data and real-time analysis and artificial intelligence. In Section 5, we focus on job level energy efficiency. Finally in Section 6, we describe how these research results entered in the CINECA agenda.

2 THE EXAMON FRAMEWORK

In this section, we give a high-level description of the scalable monitoring infrastructure namely *ExaMon* [6, 7]. The model-learning framework described in this paper is composed of several components. With the help of the hierarchical view showed in Figure 1 we can distinguish four main groups. Starting from the bottom:

Data Collection. We collect two kinds of data: i) physical data measured with sensors and ii) workload information obtained from the job dispatcher. These are the low-level components having the task of reading the data from several sensors scattered across the system and deliver them, in a standardized format, to the upper layer of the stack. These software components are composed of two main objects, the Message Queue Telemetry Transport (MQTT) API and the Sensor API object. The former implements the MQTT protocol functions and it is the same among all the collectors while the latter implements the custom sensor functions related to the data sampling and is unique for each kind of collector. Considering the specific sensor API object, we can distinguish collectors that have direct access to hardware resources like PMU (Performance Monitoring Unit), IPMI (Intelligent Platform Management Interface), HW accelerators, HW interconnects like I2C and PMBUS and collectors that sample data from other applications as batch schedulers (Altair's PBS and Slurm Workload Manager) and tools such as perf, PAPI, xCat/Confluent, Nagios and Ganglia.

The second typology of data regards the jobs running in the system and its workload. To gather this data, we need to extend the job scheduler adding a software component that collects the information and sends it as an MQTT message to the upper layers of the framework. Current state-of-the-art schedulers usually expose a set of APIs that can be used by developers to add custom functions and behaviours. For example, the *hooks* in PBS and *plugins* in Slurm.

Communication Layer. The framework is built around the MQTT protocol. MQTT implements the "publish-subscribe" messaging pattern and requires three different agents to work: (i) The "publisher", having the role of sending data on a specific "topic". (ii) The "subscriber", that needs certain data so it subscribes to the appropriate topic. (iii) The "broker", that has the functions of (a) receiving data from publishers, (b) making topics available to subscribers, (c) delivering data to subscribers. The basic MQTT communication mechanism is as follows. When a publisher agent sends some data having a certain topic as a protocol parameter, the topic is created and available at the broker. Any subscriber to that topic will receive

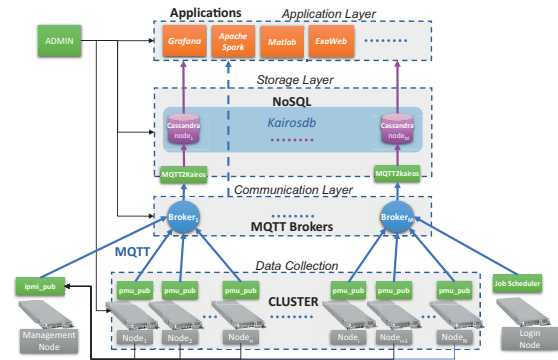


Figure 1: The *ExaMon* Monitoring framework

the associated data as soon as it is available to the broker. In this scenario, collector agents have the role of "publishers".

Storage Layer. The monitoring framework provides a mechanism to store metrics mainly for visualization and analysis of historical data. We use a distributed and scalable time series database (KairosDB) that is built on top of a NoSQL database (Apache Cassandra) as back-end. A specific MQTT subscriber (MQTT2Kairos) is implemented to provide a bridge between the MQTT protocol and the KairosDB data insertion mechanism. The bridge leverages the particular MQTT topics structure of monitoring framework to automatically form the KairosDB insertion statement. This gives a twofold advantage: first, it lowers the computational overhead of the bridge since it is reduced to a string parsing operation per message; and secondly, it makes easy to form the database query starting only from the knowledge of the matching MQTT topic.

Applications Layer. The data gathered by the monitoring framework can serve multiple purpose, as presented in the application layer. For example, machine learning techniques can be applied to extract predictive models or devise online fault detection mechanisms as shown in Section 3.

2.1 ExaMon Results

Today *ExaMon* monitors at multi-granularity the CINECA *Galileo*, *Marconi*, and *D.A.V.I.D.E.* systems [19] collecting processors elements, node level and job level statistics from 7393 computing nodes in total and ingesting daily 70GB/day of Data. In the next sections, we will show practical results conducted on top of the *ExaMon* collected data.

3 AI FOR DATACENTRE AUTOMATION

In this section, we describe two use cases of combining Artificial Intelligence (AI) techniques toward the datacentre automation.

The data collected through the *ExaMon* system is used for two purposes: (1) correlating multi-scale metrics with the user's computational demand and resource usage (e.g., power/energy, job's request, efficiency). (2) correlate the multi-scale metrics between them self to probe the status of the systems.

By profiling the user needs and demands, system-level tools can be made aware of the user while optimizing datacentre level policies.

With this perspective in last years, we used the EURORA[19] and the *D.A.V.I.D.E.* system to prototype job-level power predictors combined with predictive-aware job schedulers. The research questions that were answered by this research were threefolds: is it possible to predict the power consumption of jobs before their execution? To which degree optimization algorithms can be integrated into the job schedulers? How proactive (based on predictions of state evolution) power management approaches behave w.r.t. reactive (based on current state) power management at the job level? We answer these research questions in Section 3.1.

On the contrary, the same monitoring infrastructure can be used to correlate time-traces and vital signs of the supercomputer and nodes to spot out anomalous behaviours. With this perspective, we conducted research focused on understanding the main challenges in applying artificial intelligence approaches to the anomaly detection problem. The research questions that were answered by this research were twofolds: how to overpass the lack of faulty data in production systems? Is it possible to train anomaly detection models without domain knowledge? We answer these research questions in the Section 3.2.

3.1 AI drive power capping

A common approach to curtail the excessive power demands of supercomputers is to hard-bound their consumption, *power capping*. Most HPC systems that strive to enforce power capping adopted HW-based solutions, such as Dynamic Voltage Frequency Scaling (DVFS)[20, 24] or Intel's RAPL[18]. These technologies have the advantages of being well-known and requiring minimal effort by users. However, HW-based methods operate by exchanging computational power for energy consumption, and this typically implies that the duration of the HPC applications increases. This poses a challenge for the accounting system [12], as in nowadays most supercomputers the price paid by users depends on the duration of their jobs. Hence, alternative means for enforcing power capping have been widely studied. AI can help in this regard, as it allows for smart scheduling and allocation decisions, with the possibility to better use the system resources through careful planning.

The scheduling and allocation problems consist in deciding when to start the execution of jobs submitted by users and where to execute them (i.e., choosing the set of resources to be allocated to the application). In recent years, AI-based job dispatchers for supercomputers have been devised, using the EURORA and *D.A.V.I.D.E.* systems as test cases; these works combined Constraint Programming (CP) and heuristic algorithms to find optimal scheduling plans[5, 13, 14], with the goal of maximizing the machine utilization and decreases response times (aiming at satisfying both system owners and users). Another area where supercomputer can benefit from AI contributions is power consumption. To make well-informed decisions regarding the facility, knowing the workload power consumption before its execution is critical. Machine Learning (ML) methods address this issue, by leveraging a large amount of historical data to learn the power consumption model of HPC applications. The key is to have large data sets containing fine-grained data about the power consumption of previous jobs run on the target system, that is, the data collected via *ExaMon*. With such data, in both EURORA and *D.A.V.I.D.E.* we created ML models based on Random

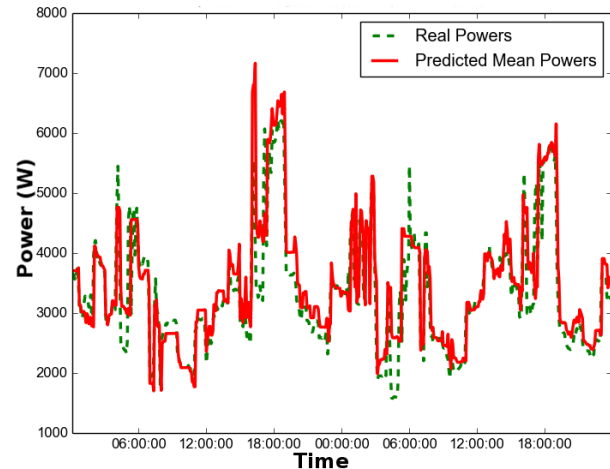


Figure 2: Predicted VS Real Power

Forest that can predict with high accuracy the power consumption of HPC jobs, using the only job-related information provided at submission time (e.g., the requested resources, the user, the submission queue, etc.)[9]. The accuracy of this model was very good, with an average error around 8-9%, as shown in Figure 2 that compares the predicted trend to the real power consumption trend in two days on the EURORA system.

Exploiting the ML model an AI enhanced power capping feature was developed, by creating a job scheduler capable of limiting a supercomputer power consumption acting only on the workload scheduling[11], without trading off performance for power consumption – it relies on proactive planning and is based on a combination of Constraint Programming (CP) plus a heuristic algorithm. The scheduler is awakened at whenever a new job is submitted, or a running job terminates; at each activation, we build a full schedule and resource assignment for all the waiting jobs, but then we dispatch only those jobs that are scheduled for immediate execution. By taking into account future jobs, we avoid making dispatching decisions with undesirable consequences; by starting only the ones scheduled for immediate execution, the system can manage uncertain execution times. The power is treated as an additional resource and constrained never to exceed the given budget; since the dispatching decisions take place before the execution of the jobs, the ML method to predict the power consumption aforementioned is a fundamental component. The AI-enhanced method improves the solutions from the state-of-the-art (SoA) by around 8.5%, in terms of average waiting time[11].

3.2 AI driven anomaly detection

Another area where AI can offer significant advantages in the HPC context is anomaly detection. Large supercomputers are composed of numerous components that risk to break down or behave in unwanted manners. Identifying broken or wrongly configured components is a daunting task for system admins. Hence an automated tool would be a boon. The current SoA relies on *supervised* ML methods that learn to distinguish between healthy and faulty states

after a training phase during which the supercomputer is subjected to different conditions, e.g. normal behaviour and a set of anomalies. However, it is quite hard to obtain the large, unbalanced and labelled training set necessary for this learning methodology¹, since in supercomputers data is very abundant but labels are scarce and that most of the time these systems are in normal states.

We showed in previous works [10, 15] on the *D.A.V.I.D.E.* supercomputer that an easier to implement approach can be adopted: we devised a *semi-supervised* method that relies on the abundance of normal data collected during the lifetime of the machine to create a model of the normal state; this model can then be used to discern between normal or anomalous conditions in an online fashion. The key components of the proposed approach are I) the data collected in a 2 months period via Examon and guaranteed to correspond to *D.A.V.I.D.E.* in normal state and a II) a type of neural network called *autoencoder*[22] (from the deep learning field) that learns the correlations among the input features (i.e., the various metrics collected by Examon) that characterize the normal behaviour of *D.A.V.I.D.E.* computing nodes. After having learnt the normal correlations, the autoencoder can notice representation changes that underlie anomalous conditions, thus detecting the anomalies.

To test our approach, we deployed the trained autoencoder network onto the embedded monitoring boards that gather sensor measurements collected with *ExaMon*, and then we injected faults (in a controlled way) on a subset of *D.A.V.I.D.E.* computing nodes. The live data from the measuring sensors is then fed to the autoencoder that decides if the current set of measurements correspond to normal or anomalous behaviour; the decision is made with negligible overhead and in under 11ms. The detection rate is very precise, as on average, the accuracy is around 92%, which is 12% higher than the accuracy of other semi-supervised methods from the literature.

4 EDGE COMPUTING AND FINE-GRAIN MONITORING

While the previous section focuses on the coarse grain sensors and in aggregating the information in job level and node level models, this section focuses on fine-grain sensor data and the research challenges related to the usage of them. The scientific questions that this research answers are: can we leverage sub-second power measurement for datacentre automation? Are these metrics of interest? How to process and integrate them into the *ExaMon* framework?

4.1 DiG: Dwarf-In-a-Giant

Modern HPC systems still have limited power introspection capabilities. Indeed, at node level *ExaMon* is lacking fine-grain and accurate measurements of the power consumption of the node, which are only available from the IPMI sensors data or CPU-level performance counters such as Intel-RAPL [18] as well as dedicated systems for live edge analysis. To bridge this gap, in the *D.A.V.I.D.E.* [2, 6] system we developed DiG (i.e., Dwarf in a Giant) [26], an enabler framework for green computing, predictive maintenance and security of supercomputers. DiG provides high-quality monitoring of power and energy consumption of HPC nodes. It is completely out-of-band and can be deployed in any hardware architecture/large-scale

¹A partial solution not yet explored could be *transfer learning*, i.e. applying to other systems the ML models trained with the data from a specific supercomputer

datacentre at a low cost. It supports: (i) fine-grain power monitoring up to 20ms (50x improvement in resolution than state-of-the-art - SoA); (ii) below 1% (σ) of uncertainty on power measurements, which makes it suitable for the most rigorous requirements of HPC ranking lists (i.e., Top500); (iii) high-precision time-stamping (sub-microsecond), which is three orders of magnitude better than SoA; (iv) real-time profiling, useful for debugging energy-aware applications; and (v) possibility for edge analytics via machine learning algorithms, with no impact on the HPC computing resources and no additional load to the *ExaMon* monitoring infrastructure. The latter feature ensures scalability for large-scale installations.

Figure 3 sketches the three main components of the DiG system: (i) a dedicated power sensor to measure the whole node power consumption at a high resolution, (ii) an embedded computer (i.e., Beaglebone Black - BBB) to carry out edge analytics on the HPC's power and performance measurements - along with the high-resolution power measurements we collect performance measurements from integrated out-of-band telemetries, such as IBM Amester and IPMI -, and (iii) a scalable interface to *ExaMon* (i.e., MQTT), to carry out cluster-level analytics on large-scale systems.

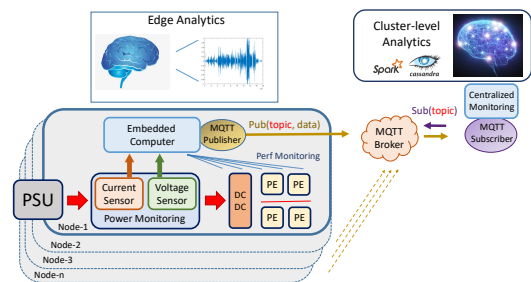


Figure 3: DiG monitoring system architecture.

4.2 Edge Analytics

At the ExaScale the burden of executing signal processing or data analytics tasks required for the datacentre's automation can easily become the bottleneck of the holistic multiscale monitoring system. For this reason, we leverage the DiG platform as a vehicle to embed these features together with the out-of-band telemetry system of each computing node (e.g., IBM Amester) [15]. As an example, we exploit the same autoencoder models described in the previous section in combination with the out-of-band monitoring system and the embedded monitoring boards (BBB) to execute the inference online and detect anomalies thanks to edge computing. We installed TensorFlow on the BBB and took advantage of the NEON accelerator (SIMD architecture). On each BBB we load the trained autoencoder of the corresponding node, then we feed it in real-time with new data coming from the monitoring framework. The results of the detection were presented in Section 3.2. Here we want to point out that we process on edge a batch of input data (the set of 166 features) in just 11ms, which is a negligible overhead considering the sampling rate of several seconds.

In addition to the out-of-band telemetry data, DiG samples the node's power consumption at an SoA time-granularity for HPC

systems in production (20 microseconds). We use this information to compute locally and in real-time signal processing tasks, useful for machine learning inference on the edge [26]. Indeed, the embedded computers used on DiG features HW extensions to accelerate signal processing workloads and thus perform lively the Power Spectral Density (PSD) of the nodes' power consumption.

Figure 4 reports an example of the PSDs computed by DiG in a time window of 40 milliseconds, while we were running on a node different applications. Goal of this test is not to analyze the reasons behind the peaks, but instead to show that different patterns emerge in the power spectrum with different workloads. These patterns can be used as input features for machine learning algorithms (e.g., Deep Neural Networks) targeting specific applications, such as energy efficiency, maintenance and security of supercomputers [26]. In particular, comparing the first plot which portrays the PSD of the computing node in idle, with the second and third plots that depict respectively a memory bound synthetic benchmark and a real scientific application (i.e., Quantum Espresso - QE), we can clearly see three different patterns (peaks highlighted with dark / light circles to indicate stronger / weaker magnitude).

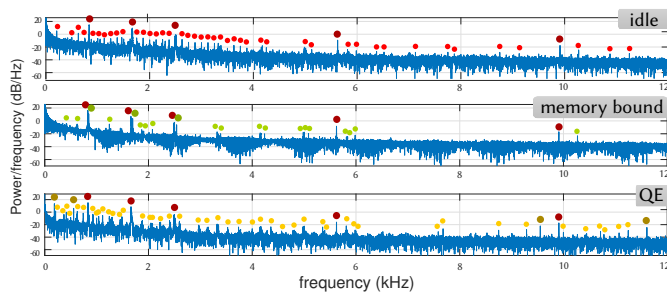


Figure 4: Example of PSD patterns of real bottlenecks and applications that can be captured with DiG.

To conclude and answer the research questions, frameworks based on embedded systems, like DiG, can (i) have the form factor and computation power to enhance the out-of-band telemetry integrated in computing nodes (e.g., IBM Amester) and (ii) ease the centralized monitoring system (e.g., *ExaMon*), while (iii) deploying localized artificial-intelligence analysis and datacentre-automation tasks. Our practical experience shows that the DiG system can leverage the fine-grain (sub-ms) telemetry to capture key spectral features of real computing applications, opening new opportunities for learning algorithms on power management, maintenance and security of supercomputers.

5 JOB LEVEL ENERGY REDUCTION

While previous Sections focuses on automating the maintenance tasks of the computing infrastructure, not a lot is done in terms of increasing its efficiency. The *ExaMon* framework enables users to assess the energy consumed by their running job, but let them control the energy consumed by their job could be detrimental to the supercomputing capacity and TCO [12].

Indeed, low power design strategies enable computing resources to trade-off their performance for power consumption by mean of

low power modes of operation. These states obtained by dynamic and voltage frequency scaling (DVFS) (P-states [1]), clock gating or throttling states (T-states), and idle states which switch off unused resources (C-states [1]). Power states transitions are controlled by hardware policies [23, 28], operating system (OS) policies, and with an increasing emphasis in recent years, at user-space by the final users [3, 20, 21, 24] and at execution time [25, 30]. However, exploring the EtS (Energy-to-Solution)-TtS (Time-to-Solution) Pareto curve at run-time has a limited potential in the current supercomputing scenario: slowing down applications is often detrimental to the total cost of ownership (TCO) due to the large contribution related to the depreciation cost of the IT equipment [12].

Several approaches have shown that it is possible to limit the performance degradation while cutting the IT energy wasted by reducing the performance of the processing elements when the application is in a region with communication slack available [16, 17, 25, 29, 30]. These approaches try to isolate, at execution time, regions of the application execution flow which can be executed at a reduced P-state (DVFS) without impacting the application performance (not in the critical task).

To explore and evaluate these approaches with production runs, the *ExaMon* framework is not suitable, as it has no introspection on the application flow, nor is capable of injecting core-level power management actions selectively in code regions. The research challenges lay in being capable of extracting and intercepting the application flow without causing overheads and isolating the right computing phase to be executed at a reduced performance.

5.1 Reactive and Proactive Power Management

Message Passing Interface (MPI) libraries implement idle-waiting mechanisms, but these are not used in practice to avoid performance penalties caused by the transition times in and out of low-power states [23]. To avoid changing frequency in fast MPI primitives, which can induce high overhead and low energy saving, it is possible to adopt two different strategies: using (i) proactive mechanisms, which try to identify MPI primitives (through learning mechanisms) where is possible to reduce the core's frequency with a limited or negligible impact on the execution time, or implementing (ii) reactive mechanisms to impose a predetermined action to filter-out fast and costly MPI primitives in term of overhead.

5.2 COUNTDOWN - A Reactive Approach

For this purpose we designed COUNTDOWN. This library instruments the application intercepting MPI primitives, it uses a timeout strategy [8] to avoid changing the power state of the cores during fast application and MPI context switches avoiding performance overhead without significant energy and power reduction. Each time the MPI library asks to enter in low power mode, COUNTDOWN defers the decision for a defined amount of time. If the MPI phase terminates within this amount of time, COUNTDOWN does not enter in the low power states, filtering out short MPI phases which are costly in terms of overheads and with a negligible impact of energy saving. This strategy is purely reactive, and it is triggered by the MPI primitives called by the application.

COUNTDOWN implements the timeout strategy through the standard Linux timer APIs, which expose the system calls: *setitimer()*

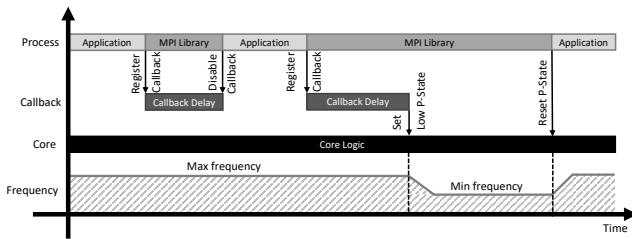


Figure 5: Timer strategy implemented in COUNTDOWN.

and `getitimer()` to manipulate user’s space timers and register callback routines. This methodology is depicted in Figure 5. When COUNTDOWN encounters an MPI phase, in which opportunistically can save energy by entering in a low power state, COUNTDOWN registers a timer callback in the *prologue* routine (`Event(start)`), after that the execution continues with the standard workflow of the MPI phase. When the timer expires, a system signal is raised, the “normal” execution of the MPI code is interrupted, the signal handler triggers the COUNTDOWN *callback*, and once the callback returns, execution of MPI code is resumed at the point, it was interrupted. If the “normal” execution returns to COUNTDOWN (termination of the MPI phase) before the timer expiration, COUNTDOWN disables the timer in the *epilogue* routine and the execution continues as nothing happened.

COUNTDOWN is a profiling and fine-grain power management run-time C library. It implements profile capabilities, and it can inject run-time code in the application to inspect and react to the MPI primitives. The library exposes the same interface of a standard MPI library, and it can intercept all MPI calls from the application. COUNTDOWN implements two wrappers to intercept MPI calls: i) one for C/C++ MPI libraries, ii) one for FORTRAN MPI libraries. This is mandatory due to C/C++, and FORTRAN MPI libraries produce assembly symbols that are not application binary (ABI) compatible. The FORTRAN wrapper implements a marshaling and unmarshaling interface to bind MPI FORTRAN handlers incompatible MPI C/C++ handlers. This allows COUNTDOWN to interact with MPI libraries in FORTRAN applications.

The library targets the instrumentation of applications through dynamic linking without user intervention. When dynamic linking is not possible COUNTDOWN has also a fall-back, a static-linking library, which can be used in the toolchain of the application to inject COUNTDOWN at compilation time. However, dynamic linking allows to instrument every MPI-based application without any modifications of the source code nor the toolchain. Linking COUNTDOWN to the application is straightforward: it is enough to configure the environment variable `LD_PRELOAD` with the path of COUNTDOWN library and start the application as usual.

Moreover, COUNTDOWN is endowed with profiler capabilities which allow a detailed analysis of the application which relies on the raw HW performance counter of Intel CPUs. The profiler uses the Intel Running Average Power Limit (RAPL) registers to monitor the energy/power consumed by the CPU.

5.3 Fermata - A Proactive Approach

To understand the benefit of the reactive COUNTDOWN policy it is useful to compare it with the SoA proactive *Fermata* [29, 30] algorithm. *Fermata* implements a simple algorithm to reduce the cores’ P-state in communication regions. It uses a prediction algorithm to decide when scaling down the P-state; the prediction is determined by the amount of time spent in communication during the previous call. If the duration is greater than or equal to twice the switching threshold, *Fermata* sets a timeout to expire at the threshold time. The threshold time is empirically set to 100ms. Calls are identified as specific MPI primitives in the application code through the hash of the pointer that makes up the stack trace. The hash is generated when the application encounters an MPI primitive; hence, each MPI primitive in the code is uniquely identified. The information about the last call is stored in a look-up table used to choose if to set the timer in the next call.

5.4 Reactive vs Proactive

In this Section, we evaluate the performance of both approaches using the NAS Parallel Benchmarks (NPB) [4]. NAS is a set of kernels and dwarf applications developed by the NASA Advanced Supercomputing division. The NPB consist of benchmarks widely used in different scientific areas such as spectral transform, fast Fourier transform, fluid dynamics, and so on. We use the NPB version 3.3.1 with the dataset E. We executed the NAS on 29 compute nodes with a total core count of 1024 cores. We use 1024 cores due to the execution time of the application run using dataset E is, on average, ten minutes for each benchmark.

In our experimental setup, we used the GALILEO tier-1 HPC system. Its compute nodes are equipped with 2 Intel Broadwell E5-2697 v4 CPUs, with 18 cores at 2.3 GHz nominal clock speed and 145W TDP and interconnected with an Intel QDR (40Gb/s) Infiniband high-performance network. We use the complete software stack of Intel systems for real production environments. We use Intel *MPI Library 5.1* as the runtime for communication and Intel *ICC/IFORT 18.0* in our toolchain. We select the Intel software stack because it is currently used in our target systems as well is supported in most of HPC machines based on Intel architectures.

We run NAS with and without instrumentation of COUNTDOWN and *Fermata* and we compare the results. COUNTDOWN reports an average overhead of 3.85%, while *Fermata* shows an average overhead of 4.21%. In term of energy and power saving, COUNTDOWN reports in average respectively 14.67% and 17.93% while *Fermata* reports an energy and a power saving of 9.95% and 13.64%. We can notice that COUNTDOWN outperforms *Fermata* with lower overhead and higher energy and power saving respectively of 4.72% and 4.29% of gain. We must remark that COUNTDOWN logic guarantees that no transition to low-power states are triggered for MPI phases shorter than 500us, for which the latency of the CPU’s internal power controller would cause uncertainty in the applied low-power state. These results suggest that it is possible to decrease the energy consumption of supercomputing machines with reduced overhead. However, how will this behave in a real production run?

5.5 COUNTDOWN on Quantum ESPRESSO

After we prove that the proposed reactive policy has advantage over proactive one to reduce power consumption in MPI communication inducing lower overhead, we scale our experiments on a real production run using Quantum ESPRESSO (QE) with COUNTDOWN.

QE is a suite of packages for performing Density Functional Theory based simulations at the nanoscale, and it is widely employed to estimate ground state and excited state properties of materials *ab initio*. The code used for the experimental setup is PWscf (Plane-Wave Self-Consistent Field) which is used to solve the self-consistent Kohn and Sham (KS) equations and obtain the ground state electronic density for a typical case study. The code uses a pseudo-potential and plane-wave approach and implements multiple hierarchical levels of parallelism implemented with a hybrid MPI+OpenMP approach. As of today, OpenMP is generally used when MPI parallelism saturates, and it can improve the scalability in the highly parallel regime. Nonetheless, in the following, we will only refer to data obtained with pure MPI parallelism without significantly impairing the conclusions reported later.

We run QE v6.1.0 on 96 compute nodes, using 3456 cores and 12 TB of DRAM. We used an input dataset capable of scaling on such number of cores, and we configured QE to avoid network bottlenecks, which would have limited the scalability. We run an instance of the application with and without COUNTDOWN on the same nodes, and we compared the results.

Figure 6 shows the total time spent in the application and in MPI phases, which are shorter and longer than 500us, which is the reaction time of the HW power controller [23]. On the x-axis, the figure reports the Id of the MPI rank, while in the y-axis reports in the percentage of the total time spent in phases longer and shorter than 500us. We recall that 500us is the latency time of the internal power controller logic of the GALILEO CPUs [23]. We can immediately see that in this real and optimized run, the application spends a negligible time in phases shorter than 500us. In addition, the time spent in the MPI library and the application is not homogeneous among the MPI processes. This is an effect of the workload parameters chosen to optimize the communications, which distribute the workload in subsets of MPI processes to minimize broadcast and All-to-All communications. When the COUNTDOWN library is preloaded our experimental results report 2.88% of overhead with an energy saving of 22.36% and a power saving of 24.53%².

The results of COUNTDOWN are encouraging, showing that it is possible to leverage communication slacks in an application for energy saving at a reduced overhead. In future work, we will extend the COUNTDOWN algorithm with critical path information to nullify the application overhead of this solution. As a conclusion, we must remark that job level energy-management is a feasible way toward more energy-efficient datacentre and that promising algorithms and basics buildings blocks exist to enable it.

6 LESSON LEARNED, AND VISION

CINECA is going to deploy the future HPC systems to a new datacentre in the Bologna science park, where the ECMWF datacentre is going to be relocated as well. The datacentre includes 890 sqm

²These numbers are obtained comparing the measured Time-to-Solution and Energy-to-Solution measured by mean of RAPL Intel counters

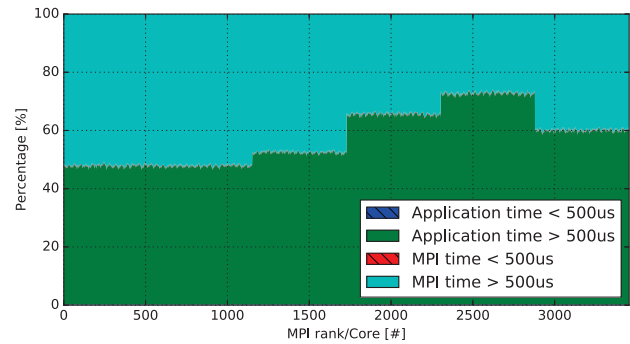


Figure 6: Sum of the time spent in phases longer and shorter than 500us for QE. Phases shorter than 500us the minority.

of data hall, 350 sqm of data storage, and electrical, cooling and ventilation systems, as well as offices and ancillary spaces, and is designed for extreme energy efficiency, targeting a PUE less than 1.1. This HPC area can be increased by 700 sqm if needed. The facility is designed for 20 MW IT, but in the first phase of operation (2020-2025), it will be equipped with an infrastructure capable of 10 MW IT. As programmed by the national roadmap, in a subsequent phase of operation, the site is therefore capable of hosting a full exascale system following an upgrade of the electricity distribution and cooling infrastructures to match 20 MW IT.

The HPC system CINECA is planning to install there will be intrinsically energy efficient; it will be co-designed with the hardware integrators for direct liquid cooling with warm water, extracting 80% of the heat produced. That, combined with dry coolers available in the datacentre will guarantee an annual PUE of less than 1.1.

To achieve the primary goal of maximizing efficiency and sustainability, and with a projected PUE of 1.1, the HPC solution to be deployed will focus on energy efficiency and power management. The following objectives are of particular interest for CINECA: (i) Enable correlation between power consumption and system workload; (ii) Enable dynamic power capping with graceful performance degradation of the system; (iii) Provide capability to optimize the job execution environment for better energy efficiency; (iv) Provide energy accounting mechanism; (v) Allow energy profiling of applications to enable EtS optimization without TtS degradation.

Thus, the HPC solution should provide reliable power and energy measurement at different level (CPU, node, rack), and interfaces allowing integration with the resource scheduler to provide energy accounting mechanism and power capping capability.

The datacentre will be equipped with an energy management system (EMS) to monitor, measure and control the loads. The energy management system will also be used to centrally control cooling devices (HVAC type, etc.) and lighting systems. EMS will be equipped with measurement, submetering and monitoring functions that allow the energy manager to access data and information on the site's energy activities. The EMS system will be monitored via wall screens inside the Control room.

The EMS system will monitor in real time and record via the EMS server at least the following functions: (i) All the status information and power measurement of the MV switchgear switch, of the

substation switch and the LV system control unit interruption; (ii) All states of the PDU main switch and measurement information; (iii) All transformer temperature alarms and all generators status information and alarms; (iv) All UPS system and battery status information and alarms; (v) All overvoltage suppression alarms; (vi) Power factor correction equipment; (vii) Multi-function counters for all general electrical distribution panels/equipment, single systems, supplies, etc. transformer power supplies and UPS output panels; (viii) Power quality analyser meter - on all major LV panels.

As shown in the paper, CINECA together with its research partners is paving the way to using high-frequency power monitoring in combination with out-of-band performance monitoring to improve datacentre automation and resilience, which are of primary concern in exascale class systems. With this in mind, CINECA will leverage *ExaMon* to build an automated pipeline to model, discover and improve the maintenance and optimization of the datacentre.

In the new datacentre CINECA will bring all its background knowledge, with the possibility to improve significantly the efficiency, also thanks to a new brand equipment, where a lot of attention will be dedicated to the quality of the monitoring and management functionalities, as well as energy efficiency. Together with the equipment, CINECA will acquire adequate site management software. For the HPC system, CINECA will rely on the monitoring and management system provided by the vendor, but among the required feature of the HPC system to be procured, there will be the provision of an energy monitoring and management system, with functionalities similar or better than those available in the PRACE PCP systems (e.g. high frequency energy sampling). In this case CINECA will as well plan deploy *ExaMon* on the system for an improved profiling, monitoring, management and reporting of the workload and system utilization.

ACKNOWLEDGMENTS

Work supported by the EU FETHPC project ANTAREX (g.a. 671623), EU ERC Project MULTITHERMAN (g.a. 291125), and CINECA research grant on Energy-Efficient HPC systems.

REFERENCES

- [1] Advanced Configuration and Power Interface (ACPI) Specification. [Online]; <http://www.acpi.info/spec.htm>, 2019. accessed 29 March 2019.
- [2] AHMAD, W. A., BARTOLINI, A., BENEVENTI, F., ET AL. Design of an energy aware petaflops class high performance cluster based on power architecture. In *2017 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)* (May 2017), pp. 964–973.
- [3] AUWETER, A., BODE, A., BREHM, M., ET AL. A case study of energy aware scheduling on supermuc. In *Supercomputing* (Cham, 2014), J. M. Kunkel, T. Ludwig, and H. W. Meuer, Eds., Springer International Publishing, pp. 394–409.
- [4] BAILEY, D. H. *NAS Parallel Benchmarks*. Springer US, Boston, MA, 2011, pp. 1254–1259.
- [5] BARTOLINI, A., BORGHESI, A., BRIDI, T., ET AL. Proactive workload dispatching on the eurora supercomputer. In *International Conference on Principles and Practice of Constraint Programming* (2014), Springer, pp. 765–780.
- [6] BARTOLINI, A., BORGHESI, A., LIBRI, A., ET AL. The D.A.V.I.D.E. big-data-powered fine-grain power and performance monitoring support. In *Proceedings of the 15th ACM International Conference on Computing Frontiers, CF 2018, Ischia, Italy, May 08–10, 2018* (2018), pp. 303–308.
- [7] BENEVENTI, F., BARTOLINI, A., CAVAZZONI, C., AND BENINI, L. Continuous learning of hpc infrastructure models using big data analytics and in-memory processing tools. In *Proceedings of the Conference on Design, Automation & Test in Europe* (2017), European Design and Automation Association, pp. 1038–1043.
- [8] BENINI, L., BOGLIOLO, A., AND MICHELI, G. D. A survey of design techniques for system-level dynamic power management. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 8, 3 (June 2000), 299–316.
- [9] BORGHESI, A., BARTOLINI, A., LOMBARDI, M., ET AL. Predictive modeling for job power consumption in hpc systems. In *International Conference on High Performance Computing* (2016), Springer, pp. 181–199.
- [10] BORGHESI, A., BARTOLINI, A., LOMBARDI, M., ET AL. Anomaly detection using autoencoders in high performance computing systems. *arXiv preprint arXiv:1811.05269* (2018).
- [11] BORGHESI, A., BARTOLINI, A., LOMBARDI, M., ET AL. Scheduling-based power capping in high performance computing systems. *Sustainable Computing: Informatics and Systems* 19 (2018), 1–13.
- [12] BORGHESI, A., BARTOLINI, A., MILANO, M., AND BENINI, L. Pricing schemes for energy-efficient hpc systems: Design and exploration. *The International Journal of High Performance Computing Applications* 0, 0 (0), 1094342018814593.
- [13] BORGHESI, A., COLLINA, F., LOMBARDI, M., ET AL. Power capping in high performance computing systems. In *International Conference on Principles and Practice of Constraint Programming* (2015), Springer, pp. 524–540.
- [14] BORGHESI, A., CONFICONI, C., LOMBARDI, M., AND BARTOLINI, A. Ms3: A mediterranean-stile job scheduler for supercomputers-do less when it's too hot! In *2015 International Conference on High Performance Computing & Simulation (HPCS)* (2015), IEEE, pp. 88–95.
- [15] BORGHESI, A., LIBRI, A., BENINI, L., AND BARTOLINI, A. Online anomaly detection in hpc systems. *arXiv preprint arXiv:1902.08447* (2019).
- [16] CESARINI, D., BARTOLINI, A., BONFÀ, P., ET AL. COUNTDOWN - three, two, one, low power! A run-time library for energy saving in MPI communication primitives. *CoRR abs/1806.07258* (2018).
- [17] CESARINI, D., BARTOLINI, A., BONFÀ, P., ET AL. Countdown: A run-time library for application-agnostic energy saving in mpi communication primitives. In *Proceedings of the 2Nd Workshop on Autotuning and daptivity Approaches for Energy Efficient HPC Systems* (2018), ANDARE '18, ACM, pp. 2:1–2:6.
- [18] DAVID, H., GORBATOV, E., HANE BUTTE, U. R., ET AL. Rapl: Memory power estimation and capping. In *Proceedings of the 16th ACM/IEEE International Symposium on Low Power Electronics and Design* (2010), ISLPED '10, ACM.
- [19] DONGARRA, J. J., MEUER, H. W., STROHMAIER, E., ET AL. Top500 supercomputer sites. <https://www.top500.org/lists>, 2019. accessed 29 March 2019.
- [20] EASTEP, J., SYLVESTER, S., CANTALUPO, C., ET AL. Global extensible open power manager: A vehicle for hpc community collaboration on co-designed energy management solutions. In *High Performance Computing* (2017), Springer International Publishing, pp. 394–412.
- [21] FRATERNALI, F., BARTOLINI, A., CAVAZZONI, C., ET AL. Quantifying the impact of variability on the energy efficiency for a next-generation ultra-green supercomputer. In *International Symposium on Low Power Electronics and Design, ISLPED'14, La Jolla, CA, USA - August 11 - 13, 2014* (2014), pp. 295–298.
- [22] GOODFELLOW, I., BENGIO, Y., COURVILLE, A., AND BENGIO, Y. *Deep learning*, vol. 1. MIT press Cambridge, 2016.
- [23] HACKENBERG, D., SCHÖNE, R., ILSCHKE, T., ET AL. An energy efficiency feature survey of the intel haswell processor. In *2015 IEEE International Parallel and Distributed Processing Symposium Workshop* (May 2015), pp. 896–904.
- [24] HSU, C., AND FENG, W. A power-aware run-time system for high-performance computing. In *SC '05: Proceedings of the 2005 ACM/IEEE Conference on Supercomputing* (Nov 2005), pp. 1–1.
- [25] LI, D., DE SUPINSKI, B. R., SCHULZ, M., ET AL. Hybrid mpi/openmp power-aware computing. In *2010 IEEE International Symposium on Parallel Distributed Processing (IPDPS)* (April 2010), pp. 1–12.
- [26] LIBRI, A., BARTOLINI, A., AND BENINI, L. Dig: Enabling out-of-band scalable high-resolution monitoring for data-center analytics, automation and control. In *The 2nd International Industry/University Workshop on Data-center Automation, Analytics, and Control* (2018).
- [27] MAITERTH, M., KOENIG, G., PEDRETTI, K., ET AL. Energy and power aware job scheduling and resource management: Global survey—initial analysis. In *2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)* (2018), IEEE, pp. 685–693.
- [28] ROSEDAHL, T., BROYLES, M., LEFURGY, C., ET AL. Power/performance controlling techniques in openpower. In *High Performance Computing* (Cham, 2017), J. M. Kunkel, R. Yokota, M. Tauber, and J. Shalf, Eds., Springer International Publishing, pp. 275–289.
- [29] ROUNTREE, B., LOWENTHAL, D. K., FUNK, S., ET AL. Bounding energy consumption in large-scale mpi programs. In *Proceedings of the 2007 ACM/IEEE Conference on Supercomputing* (2007), SC '07, ACM, pp. 49:1–49:9.
- [30] ROUNTREE, B., LOWENTHAL, D. K., DE SUPINSKI, B. R., ET AL. Adagio: Making dvs practical for complex hpc applications. In *Proceedings of the 23rd International Conference on Supercomputing* (2009), ICS '09, ACM, pp. 460–469.