GREEN 2018

The Third International Conference on Green Communications, Computing and Technologies

ISBN: 978-1-61208-667-5

September 16 - 20, 2018

Venice, Italy

GREEN 2018 Editors

Birgit Gersbeck-Schierholz, Leibniz Universität Hannover, Germany

GREEN 2018

Forward

The Third International Conference on Green Communications, Computing and Technologies (GREEN 2018), held between September 16, 2018 and September 20, 2018 in Venice, Italy, continued a series continued the series focusing on current solutions, stringent requirements for further development, and evaluations of potential directions. The event targets are bringing together academia, research institutes, and industries working towards green solutions.

The expected economic, environmental and society wellbeing impact of green computing and communications technologies led to important research and solutions achievements in recent years. Environmental sustainability, high-energy efficiency, diversity of energy sources, renewable energy resources contributed to new paradigms and technologies for green computing and communication.

Economic metrics and social acceptability are still under scrutiny, despite the fact that many solutions, technologies and products are available. Deployment at large scale and a long term evaluation of benefits are under way in different areas where dedicated solutions are applied.

We take here the opportunity to warmly thank all the members of the GREEN 2018 technical program committee, as well as all the reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated their time and effort to contribute to GREEN 2018. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

We also gratefully thank the members of the GREEN 2018 organizing committee for their help in handling the logistics and for their work that made this professional meeting a success.

We hope that GREEN 2018 was a successful international forum for the exchange of ideas and results between academia and industry and to promote further progress in the field of green communications, computing and technologies. We also hope that Venice, Italy provided a pleasant environment during the conference and everyone saved some time to enjoy the unique charm of the city.

GREEN 2018 Chairs

GREEN Steering Committee
Carlos Becker Westphall, Federal University of Santa Catarina, Brazil
Wilfried Elmenreich, University of Klagenfurt, Austria
Bo Nørregaard Jørgensen, University of Southern Denmark, Denmark
Coral Calero, University of Castilla-La Mancha, Spain

GREEN Research/Industry Committee
Steffen Fries, Siemens AG, Germany

Daisuke Mashima, Advanced Digital Sciences Cener, Singapore
Nanpeng Yu, University of California, Riverside, USA
Enrique Romero-Cadaval, University of Extremadura, Spain

GREEN 2018
Committee

GREEN Steering Committee
Carlos Becker Westphall, Federal University of Santa Catarina, Brazil
Wilfried Elmenreich, University of Klagenfurt, Austria
Bo Nørregaard Jørgensen, University of Southern Denmark, Denmark
Coral Calero, University of Castilla-La Mancha, Spain

GREEN Research/Industry Committee
Steffen Fries, Siemens AG, Germany
Daisuke Mashima, Advanced Digital Sciences Cener, Singapore
Nanpeng Yu, University of California, Riverside, USA
Enrique Romero-Cadaval, University of Extremadura, Spain

GREEN 2018 Technical Program Committee
Emad Abd-Elrahman, National Telecommunication Institute, Cairo, Egypt
Kouzou Abdellah, Djelfa University, Algeria
Naji Abdenouri, Université Cadi Ayyad, Morocco
Afrand Agah, West Chester University of Pennsylvania, USACarlos A. Astudillo Trujillo, University of Campinas, Brazil
Baris Aksanli, San Diego State University, USA
Zacharoula Andreopoulou, Aristotle University of Thessaloniki, Greece
Mounir Arioua, Abdelmalek Essaadi University, Morocco
Sheheryar Arshad, The University of Texas at Arlington, USA
Figen Balo, Firat University, Turkey
Carlos Becker Westphall, Federal University of Santa Catarina, Brazil
Rachid Benchrifa, Mohammed V University, Morocco
Rémi Bonnefoi, CentraleSupélec/IETR, France
Abdelhak Bouchakour, Applied research unit in renewable energy, Ghardaia, Algeria
Mohsine Bouya, Rabat International University, Morocco
Coral Calero, University of Castilla-La Mancha, Spain
William Campbell, Birmingham City University, UK
Massimo Canonico, University of Piemonte Orientale, Italy
Fernando Jose Castor de Lima Filho, Federal University of Pernambuco, Brazil
Cicek Cavdar, Royal Institute of Technology (KTH), Sweden
David (Bong Jun) Choi**,** SUNY Korea / Stony Brook University, Korea
Giuseppe D'Aniello, University of Salerno, Italy
Sanjoy Das, Kansas State University, USA
Monika dos Santos, University of South Africa, South Africa
Ahmed El Oualkadi, Abdelmalek Essaadi University, Morocco
Wilfried Elmenreich, University of Klagenfurt, Austria
Youssef Errami, Chouaïb Doukkali University, El Jadida, Morocco

Steffen Fries, Siemens AG, Germany
Song Fu, University of North Texas, USA
Saurabh Garg, University of Tasmania, Australia
Arnob Ghosh, Purdue University, USA
Chris Gniady, University of Arizona, USA
Tong Guan, SUNY at Buffalo, USA
Marco Guazzone, University of Piemonte Orientale, Italy
Burhan Gulbahar, Ozyegin University, Turkey
Poria Hasanpor, KTH Royal Institute of Technology, Sweden
Jianhui Hu, Shanghai Jiao Tong University, China
Song Huang, University of North Texas, USA
Ali Hurson, Missouri University of Science and Technology, USA
Mansoureh Jeihani, Morgan State University, USA
Khalil Kassmi, Mohamed Premier University, Morocco
Firdous Kausar, Sultan Qaboos University, Muscat, Sultanate of Oman
Shaian Kiumarsi, Universiti Sains Malaysia (USM), Malaysia
Sedef Akinli Kocak, Ryerson University, Canada
Christina Krenn, STENUM GmbH, Austria
Mohamed Latrach, University of Rennes 1, France
Stephen Lee, University of Massachusetts, Amherst, USA
Tao Li, Nankai University, China
Daisuke Mashima, Advanced Digital Sciences Cener, Singapore
Mª Ángeles Moraga, University of Castilla-La Mancha, Spain
Masayuki Murata, Osaka University Suita, Japan
Bo Nørregaard Jørgensen, University of Southern Denmark, Denmark
Antonio Padula, Universidade Federal do Rio Grande do Sul, Brazil
Kartik Palani, University of Illinois at Urbana Champaign, USA
Kandaraj Piamrat, LS2N | University of Nantes, France
Manuel Prieto-Matias, Complutense University of Madrid, Spain
Rahim Rahmani, Stockholm University, Sweden
D. Rekioua, University of Béjaia, Algeria
Enrique Romero-Cadaval, University of Extremadura, Spain
Camille Salinesi, Université Paris 1 Panthéon - Sorbonne, France
Sattar Sattary, University of Southern Queensland, Brisbane, Australia
Sandra Sendra, Universidad de Granada, Spain
Izzet Fatih Senturk, Bursa Technical University, Turkey
Vinod Kumar Sharma, Italian National Agency for New Technologies, Energy and Sustainable Economic Development (ENEA), Italy
Sabu M. Thampi, Indian Institute of Information Technology and Management - Kerala (IIITM-K), India
Barbara Tomaszewska, AGH University of Science and Technology, Poland
Onder Turan, Anadolu University, Turkey
John Vardakas, Iquadrat Informatica, Barcelona, Spain
Julian Leonard Weber, Advanced Telecommunications Research Institute (ATR) - Kyoto, Japan

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

The Management of Risks Related to Innovation and Technologies

Application to the Ecosystem of Renewable Energy

Julie Chehaita, Eddie Soulier
University of technology of Troyes,
ICD, CNRS UMR 6281 Tech-CICO
Troyes, France
E-mail: Julie.chehaita@utt.fr
E-mail : eddie.soulier@utt.fr

Raed Kouta
University of technology of Belfort-Montbeliard
UTT ICD/LM2S
Troyes, France
Email: raed.kouta@utbm.fr

Abstract- **Energy is a strategic sector in France. The concept of energy transition has favored the development of renewable energies. In the traditional energy model, a few large groups, resulting into an oligopoly situation, control the market. Systems based on renewable energies, on the other hand, can be deployed in a decentralized way. The concept of Renewable Energy Ecosystem (REE) is then relevant in order to think about the transition. This emerging ecosystem relies heavily on innovation, including start-ups. However, the risks posed by innovation and technology are significant; they favor start-up failures and limit the investments of venture capitalists and business angels in the ecosystem. We propose the principles of a risk assessment approach related to innovation, which takes into consideration the characteristics of the ecosystem, the role of economic models, and finally, the role of digital in accelerating the energy transition.**

Keywords- Ecosystems; Renewable Energies; Ecosystem Model; Risk indicators; Evaluation of Innovation.

I. INTRODUCTION

In 2015, the energy sector represented 2.0% of the Produit Intérieur Brut (PIB) [English: Gross Domestic Product] in France, and the energy bill represented 1.8% of PIB. In 2015, primary energy consumption (at the production level) was 47.6% of fossil fuels (of which 30.1% was petroleum products, 14.2% natural gas, and 3.3% coal) and 42.5% of non-renewable primary electricity (nuclear + pumping production - electricity exporter balance). Electricity accounted for 24.7% of the final energy consumption in France in 2015. The electricity produced in 2016 comes 72.3% from nuclear energy, 17.8% from renewable sources (mainly hydroelectric generation), and 11.1% and 8.6% from fossil thermal power plants. In fact, France's energy bill in the third quarter of 2016 amounted to 31.4 billion Euros according to the general commission for sustainable development [1].

Synthetically, the reserves of oil and natural gas are limited, while, due to demographic and economic factors, the consumption of oil, gas and coal will always be higher in the future as compared to today's consumption. The energy transition, largely based on renewable energies, is therefore imperative. Nevertheless, many experts estimate that it will be possible to cover only 30% to 40% of the needs with the help of renewable energies by 2050 if an energy transaction is taken into consideration.

Innovation, especially on the technological level, appears to be a determining factor in the competitiveness of renewable energies. According to a recent study by Irena (International Renewable Energy Agency), "All renewable technologies will be competitive with fossil fuels in 2020" [3]. Also, it must be added to the dimension cost, the non-cost competitiveness factors, such as quality, innovation, technological level, reliability, services, etc. This paper is structured as follows. In Sections II and III, we present the situation of renewable energy in France, and recall the concepts of ecosystems, business models and innovation. Sections IV and V are dedicated to the explanation of the hypothesis and the elaborated research model. We conclude in Section VI with discussions and perspectives.

II. RENEWABLE ENERGY

At the turn of the 2000s, many countries are engaged in the energy transition sector. This is one of the components of the ecological transition. The latter results from technical developments, prices, and the availability of energy resources. It also depends on the political will of governments, populations, and businesses, willing to reduce the negative effects of this sector on the environment. In France by 2050, the objective is to cut the greenhouse gas emission by 4 to 5%, and reduce by 2025 the share of nuclear energy to 50% of the French electricity production. This is done by developing renewable energies and seeking all forms of energy efficiency and therefore energy savings.

Renewable energies (RE) are energy sources whose natural renewal is fast enough that they are considered inexhaustible at the human time scale. Their renewable nature depends partly on the speed at which the source is consumed and on the other hand on the rate at which it regenerates. Renewable energies are divided into hydropower, wind, solar, biomass, biogas, renewable urban waste, geothermal energy, and marine renewable energies.

 ISBN: 978-1-61208-667-5

Primary production of renewable energy has been steadily increasing since the mid-2000s, particularly with the development of wind power, photovoltaics, biofuels and biogas. In the third quarter of 2016, the electricity produced in France by the renewable energy sector in one year equals 95 TWh. This volume covers 20.1% of the country's consumption. This share is up to 6.6% in 2015 by the same time of the year. According to the reports, renewable energy accounts for about 12% of final energy consumption (at the consumer level) in 2015 [1].

The energy sector is organized in France around a limited number of very large companies. In fact, TOTAL is the main player in the French oil sector, and the first French market capitalization on February 27, 2014, and EDF, by far the leading producer, carrier, distributor and supplier of electricity in France, and finally, ORANO for the nuclear professions. These companies are at the heart of a first ecosystem of energy production. The recent opening to competition driven by the European Union and the diffusion of the concept of energy transition has forced these major groups to invest in the field of renewable energies. Renewable energies have thus become for some of these players an investment priority [4].

III. ECOSYSTEMS, BUSINESS MODELS AND INNOVATION

Our first hypothesis is that the current and traditional actors of energy represent the first classical ecosystem, we will call it the Energy Ecosystem (EE). Next to this EE, a second ecosystem is being articulated, more specifically focusing on the renewable energies, as a clean and partly emerging segment, the "Renewable Energy Ecosystem" (REE). The EE/REE dynamic is the factor of complexity of the field of analysis.

First, the concept of business ecosystem is now well known, notably through its introduction by the work of Moore [5]. In particular, this concept was defined as "All the relations between heterogeneous actors guided by the promotion of a common resource and an ideology that drives the development of shared competencies (ecosystem skills) [6].

Specific work on the concept of the ecosystem applied to renewable energies is still rare. Yet the various reports of the International Energy Agency show that 2016 was a tipping point in the sense that investments in renewable energy, especially electricity, exceeded those made in coal, oil and gas [7]. Renewable energies thus become a massive investment deployment sector underpinned by important innovations. Renewable energies have benefited in 2016 from a series of technical advances that promise to make sustainable energy more and more efficient and affordable (artificial photosynthesis, CO2 storage, etc.).

Our second hypothesis is that this sector or, rather, this ecosystem in the phase of creation, relies largely on innovation, especially technological innovation. In the field of renewable energies, a large number of startups are currently developing. In fact, the Observatory of French start-up cleantech has observed 952 start-up companies cumulated since the creation of the Observatory in 2011. The studies conducted by the Observatory confirm the importance of belonging to an ecosystem. As a result, 62% of start-ups belong to a competitiveness cluster, 32% to an incubator and 27% to an accelerator [8]. New financing methods are unfolding strongly, like crowdfunding (or crowdfunding), which tends to show that the ecosystem of renewable energies has different characteristics from the traditional energy ecosystem. This creates complex networks, linking renewable energy start-ups with major groups in addition to citizens and other stakeholders.

IV. A EXPLORATORY MODEL FOR RISK EVALUATION

"In our model (see figure 1)," innovation plays the role of intermediate variable between the REE, as it tends to be structured today, and the desired performance by investors (or dependent variable), which can be societal, ecological, technological and financial. To set the stakes, the RE fundraisers collected by French cleantech companies during the year 2017 amounted to € 529 million, according to GreenUnivers. Our research problem is articulated around the innovation risk evaluation of the renewable energy ecosystem (REE), in particular the risk factors related to the dynamic of this ecosystem from its structuration and innovation point of view.

Our contribution is the proposition of a risk evaluation system for both business angels and venture capitalist of this ecosystem, based on risk indicators related to innovation; assessment of risks through models inspired by dependability, and finally through management measures that tend to improve the control of these risks.

Our third hypothesis is that there is a link between failures of innovations in start-ups and the notion of economic model. It is customary to share the risks of innovation in current theories [9] in three fields of uncertainty; the human need (desirability), technology (feasibility) and economic potential (viability).

Recent work that examines companies' performance in terms of their business model and the ecosystem structure to which they belong, shows that the business model / business ecosystem pair is a significant determinant of profitability [10].

As pointed out in the introduction of a special issue of Industrial Economics Review devoted to the links between business model, business ecosystem and innovation: "The use of the model of business as an independent variable has a significant link to the performance of the business [11]. It is therefore an important element in predicting the success of a business in a given ecosystem. Stability of the ecosystem (and the economic environment in general) is an important determinant of business survival [12].

 ISBN: 978-1-61208-667-5

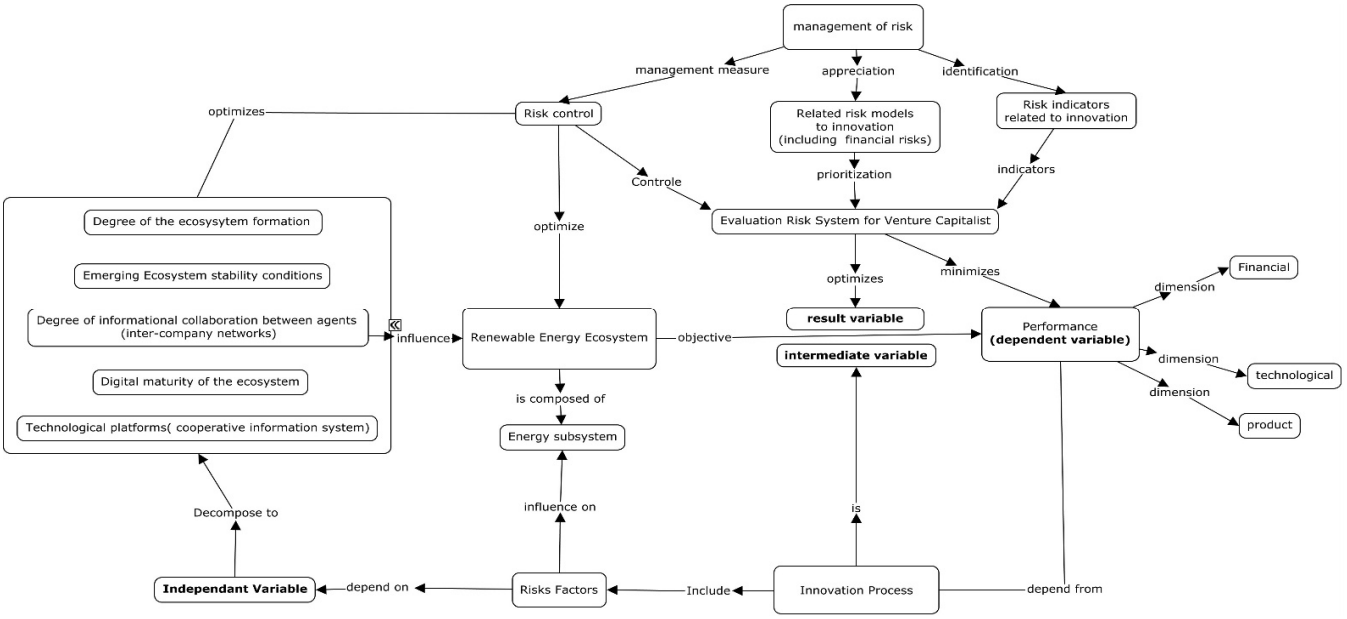

Figure 1. Research model

A large number of failures could be avoided if the company paid more attention to stakeholders and to the interactions between the company and its ecosystem. To the still too introverted view of the risks associated with a business model, new dimensions must therefore be added, especially those related to both the increasingly open nature of innovations in ecosystems and the consideration of value sharing within the value network in the ecosystem [13].

Many risk factors can explain the failure of companies in the renewable energy ecosystem. However, by focusing on the relation between the components of the innovation, and on identifying sub-variables related to technology, it is possible to identify an independent variable to explore. In fact, the relation between these components can lead to a presentation of a business model on one hand, and the structural and dynamic characteristics of a business ecosystem on the other hand. As a result, it is possible to identify a dependent variable composed of three risk classes and six sub-variable classes. The first class of risks regroup the sub-variables related to the ecosystem and covers: 1) the formation degree of the ecosystem 2) stability 3) state of inter-company networks. The second class of risk concerns the economic model, knowing that it depends strongly on the concerned energetic sub-system such as (solar, hydraulic, wind, etc.). Finally, the third class gathers the sub-variables related to the digital, its function and deployment within the ecosystem. The two-identified variables are the digital maturity of the ecosystem and the quality of platform responsible of the collaboration and the cooperation between the different agents.

V. TAKING THE DIGITAL IN CONSIDERATION IN ACCELERATING THE ENERGETIC TRANSITION

We do not tackle all the risk factors in this short paper, rather, we limit our study to what the concept “digital maturity of the ecosystem” covers. In fact, the problematic of digital maturity takes part in the problematic of digital transformation. Based on the Annual Conference of the Renewable Energies Union (SER) that was held on 8 February 2018, energy transition is now based on positive trends, such as the cost of renewable technologies that continue to decline dramatically, and the solar system that gets anchored permanently in the French landscape. In addition, innovation is a crucial factor in this transition and takes many forms. However, in the time when the acceleration of the energy transition will take place through the digital world, these positive trends do not consider digitalization as an essential way to achieve the objective of the energy transition at the best cost.

In one of the first reference books on digital transformation [14], the authors propose a model of digital transformation. This very thorough study of the strategy of 400 major global groups, such as Nike and Pernod Ricard in France, gives the keys to a successful digital transformation in terms of business models, management, customer experience, leadership, and the mobilization of employees. The main conclusion of this study is that thanks to digital technologies, higher level of profits, productivity, and performance, the masters of digital exist but they are rare. The proposed approach consist in diagnosing the digital maturity of the company based on its digital capacities (both managerial and digital), and then realign its competitive strategy according to its digital maturity in order to be able to plan for the energetic transformation within the company. The second variable related to digital, which is not tackled here, considers the ecosystem of renewable energies as a new digital ecosystem. While some companies tend to integrate the next generation of ecosystem, others are designing platforms themselves and creating their own ecosystems to position themselves at the center. In most of the studied ecosystems, the platforms have rapidly became the central hubs of ecosystems that are themselves more and more digital. Providing platforms or participating in existing offers is a strategic alternative, but it is also, from our risk perspective, a structuring and stabilizing factor of the ecosystem or, on the contrary, a new factor of risk, which

will have to be taken into account. A significant example is energy-purchasing platforms that are gaining importance, especially in fuel oil where the risk of uberization is real with actors such as fioulmarket.

VI. DISCUSSION AND PERSPECTIVES

Until now, limited studies have been undertaken, linking the performance of companies to their business model and the ecosystem structure to which they belong. These studies focus on the value proposition (innovation, target customer segments), its architecture (the ecosystem) and the model of incomes (value distribution mechanisms).

However, these models seem less suitable for emerging ecosystems such as the ecosystem of renewable energies (REE), which is articulated less around dominant firms and more around eco-citizen dynamics, start-ups. Second characteristic, this REE maintains a strong dependence on technological innovation, unlike other ecosystems more sensitive to non-technological innovations. Therefore, the question of risk management induced by innovation is crucial, while it is not taken in to consideration in these models. Thirdly, the actors of the REE do not seem to place the digital transformation at the heart of the energy transition and the decentralization of the productive model, whereas industry specialists consider that the acceleration of the energy transition will be done through digital technology.

We recall that the fundraising of renewable energies are 529 Million euros in 2017 and that the transaction market in the renewable energy sector is dynamic. In fact, in France, in the first semester of 2017, 19 transactions were initiated for a total value of 1.2 billion euros [15]. As a result, investors expect to be secure against technological challenges in order to increase the volume of transactions in the renewable energy sector, and thus accelerate the progress of the market towards the energy transition. The framework outlined here aims to develop a more integrated approach explicitly taking into account, besides the business model and the ecosystem, the risks related to innovation within the REE as well as the place of the digital in accelerating the energy transition.

REFERENCES

[1] S. Moreau, “Key numbers of Energy,” Datalab, p. 72, 2016.

[2] S. Merceron, M. Theulière and Insee, "Household energy expenditure for 20 years," Insee Prem., no. 1315,' pp. 10', 2010.

[3] International Renewable Energy Agency (IRENA), Technology Roadmap. 2014.

[4] Jean-Michel Bezat, “TOTAL is streinghtening in Renewable Energies,” Le Monde Econ., 'pp. 4', 2017.

[5] J. F. Moore, “The Death of Competition: Leadership and Strategy in the Age of Business Ecosystems,” Leadership, p. 297, 1996.

[6] C. Fourcade, “Agri-food systems as collective modalities,” Rev. française Gest, vol. 32, no. 167, pp. 183–202, 2006.

[7] IRENA, “Renewable Energy and Jobs: Annual Review 2018,” international renewable energy agency,2018.

[8] Green univers, “ How do startups of the energy transition evolve”,2018.

[9] T. Brown and B. Katz, “Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation”, 2009.

[10] C. Zott and R. Amit, “The fit between product market strategy and business model: implications for firm performance,” Strateg. Manag. J. Strat. Mgmt. J, vol. 29, no. 1, pp. 1–26, 2008.

[11] A. Attour and T. Burger-Helmchen, “Ecosystems and business models: introduction,” http://journals.openedition.org/rei, no. 146, pp. 11–25, June 2018.

[12] G. Dosi and R. R. Nelson, “The evolution of technologies: an assessment of the state-of-the-art,” Eurasian Bus. Rev., vol. 3, no. 1, pp. 3–46, 2013.

[13] V. Chanal, “ Why we must rethink the business models of innovations” , Grenoble University Press, pp. 15-23,2011.

[14] G. Westerman, D. Bonnet, A. McAfee, “Digital Transformation: A Roadmap for Billion-Dollar Organizations,” MIT Center for Digital Business and Cap Gemini Consulting, 2011.

[15] Thierry Iochem, “M&A : and if you specialize in renewable energies?,” 2018. [Online]. Available: https://news.efinancialcareers.com/fr 2018/07/19.

A Fast Heuristic for Tasks Assignment in Manycore Systems with Voltage-Frequency Islands

Shervin Hajiamini*, Behrooz Shirazi*, Hongbo Dong+
*School of Electrical Engineering and Computer Science, +Department of Mathematics
Washington State University
Pullman, WA, U.S.A.
Email: {shervin.hajiamini, shirazi, hongbo.dong}@wsu.edu

***Abstract*—Dynamic Voltage/Frequency Scaling (DVFS) is a well-known technique that dynamically scales cores' Voltage/Frequency (V/F) levels to save energy or minimize the application's execution time in manycore systems. This paper proposes an optimization framework that provides a DVFS-based cost- and energy-efficient methodology to balance time-energy tradeoffs in manycore systems with Voltage/Frequency Island (VFI) architectures. The proposed methodology has two steps: 1) formulating a Mixed Integer Linear Programming (MILP) problem that populates islands through the task-to-island assignments given that the islands are symmetric, 2) formulating an Integer Linear Programming (ILP) problem that computes the V/F levels of cores in each island per execution phase of parallel applications. The first step, which is performed at compile-time, considers the per execution phase computational characteristics of the tasks for islanding while the second step optimizes the V/F levels of formed islands at runtime. As solutions time of the proposed task-to-island assignment problem increases significantly with a large number of tasks or islands, this paper presents a fast heuristic that only requires a sorting procedure in its most time-consuming step and obtains near-optimal solutions. The proposed framework's energy efficiency is compared to an optimal, per-core islanding that establishes the best energy-time solutions for the experimented applications. Using Energy-Delay Product as performance metric, experimental results show that the framework's energy efficiency, at the worst case, is within 13% of the per-core DVFS. The results also show that this framework utilizes the idle times of low Central Processing Unit (CPU)-intensive benchmarks to increase energy saving with reasonable performance loss.**

Keywords-Manycore System; Task Partitioning; Dynamic Voltage-Frequency Scaling; Voltage-Frequency Islands; Optimization Framework; Energy Efficiency.

I. INTRODUCTION

Large-scale computer systems have become more pervasive by providing computing resources to solve complex applications. Parallel computing utilizes the multiprocessing aspect of the computing resources (e.g., CPU cores) to perform simultaneous computational processes (tasks) in order to increase the speed of the system. To strengthen the Operating System (OS) capability for running the user tasks in parallel, applications are instrumented by parallel programming techniques to take advantage of the increasing cores' computing power, which are interconnected and used as shared resources within a single computer system. As the number of cores continues to scale in manycore systems, excessive energy consumption has become a primary concern for the system designers and devising effective energy-aware techniques that are sustainable with the applications' computational demands is an important research area.

The Dynamic Voltage and Frequency Scaling (DVFS) is a method for executing high performance applications on manycore systems while maintaining the system energy consumption below a user-defined energy budget [6]. There are three approaches to apply the DVFS for energy efficiency optimization in the manycore systems. (1) Running an application on a chip-wide DVFS, where a common Voltage/Frequency (V/F) level is assigned to all the cores [1]. This method does not scale with the varying applications' computational demands. (2) In the per-core DVFS approach, the V/F level for each core is adjusted throughout the program execution, resulting in the best energy efficiency, but at the cost of hardware complexity and complicated system level control [2]. (3) As a compromise, a more flexible Voltage and Frequency Island (VFI) approach has been adopted, where cores in an island share the same V/F level, which may vary during the program execution based on the program characteristics [3].

Nowadays, large high performance computing applications have changing computational behavior during the applications runtime. Fixing the VFIs' V/F levels for the entire application execution limits exploiting opportunities to speed up or down the islands speed to gain high performance or energy saving depending on the application characteristics [8]. To address this limitation, this work applies the DVFS on islands where each VFI's V/F level can be configured individually during the program runtime.

The traditional approach for islanding is to group cores executing similar tasks across the application's execution phases (intervals) [14]. A more effective approach to perform the partitioning is to identify the similarities of tasks within the individual execution phases of the applications. This way, the system energy efficiency can be further improved by incorporating the tasks computational variations, across and within the execution phases, into the problem's optimization objectives.

The VFIs may have the same size or number of cores (symmetric) or may be asymmetric in size [4]. To simplify the task-to-island assignment problem, this paper assumes a symmetric system, where the VFIs sizes (the number of cores in a VFI that execute the assigned tasks) are the same.

This paper presents a framework for optimizing the task-to-island assignments (tasks partitioning) and the VFIs' V/F level assignments. Using this framework, this paper's goal is to minimize the applications' total execution times

 ISBN: 978-1-61208-667-5

(makespans) without exceeding user-defined energy budgets/limits.

The framework proposed in this paper has the following contributions compared to our previous work [4], which also discussed a method for an energy-constrained, optimized makespan VFI-based system:

- The VFIs' V/F levels are dynamically changed per execution phase of the experimented applications.
- The same V/F level can be assigned to different VFIs in each execution phase to achieve an overall better energy/time tradeoff.
- To improve the system's energy efficiency, application tasks, with similar computational characteristics, are assigned to the VFIs before applying DVFS.
- This paper demonstrates the extent to which the energy efficiency is maximized considering the applications with different compute/memory intensive workloads.
- Compared to [4], the proposed heuristic is faster and more scalable for larger applications or system sizes.

This paper is structured as follows. Section II summarizes related works followed by our contributions. Section III describes a system model and a model for executing applications on the system. Section IV explains the proposed two-step framework. Section V and Section VI present experimental setup and the results, respectively. Section VII concludes this paper.

II. RELATED WORK

The multi/manycore processing is a form of parallel computing where a parallelized application uses the shared hardware resources (CPU cores) to simultaneously execute the applications' threads and shorten the application runtime [5]. Increasing the number of cores in a chip may improve the application speedup but it overheats the chip due to the energy consumed by cores during the idle and busy periods. The DVFS is a well-known method that has been used to address this problem with two mainstream techniques. The per-core DVFS is a resource-demanding technique where a separate voltage regulator is allocated to each core to adjust its V/F level at runtime (lowering energy during idle periods and increasing it during compute phases). As a second method, the VFI-based systems provide a less complex and economical alternative where the V/F level of an island of cores is tuned by a single regulator. The VFI-based systems are cost-effective and provide reasonable energy saving opportunities with acceptable application execution delay [6][7]. The following summarizes the VFI-based works that are related to this paper.

The VFIs' V/F levels are determined either statically (at compile-time) or adjusted dynamically (at runtime) to account for the applications' computational variations. For example, Duraisamy et al. [8] used the cores' number of instructions per cycle and inter-core data transfers for per VFI static V/F level assignment, while Ogras et al. [9] used a feedback controller to dynamically adjust the V/F levels of a Network-on-Chip (NoC)-based VFI system using the occupancy levels of inter-VFI queues.

In terms of VFIs formation, both the symmetric and asymmetric partitioning of cores has been deployed. David et al. [10] partitioned 24-tile Intel's single-chip cloud computer into 6 VFIs, each one containing 4 tiles (symmetric). Jin et al. [11] used asymmetric VFIs whose sizes are reconfigured once by adding cores that were not assigned to the same VFI through multiple static optimizations of VFIs formation.

The prior research works have solved one or both of the islanding and V/F level assignment problems. Ozen et al. [12] used two VFIs with corresponding fixed V/F levels in a NoC, where cores' slack times were used to run the under-loaded VFIs with lower V/F levels to minimize the energy consumption. Ogras et al. [13] performed the islanding and V/F level assignment iteratively by merging two VFIs, which resulted in reducing the system energy consumption while maintaining the performance constraints.

The islanding and V/F level assignment problems have been solved by heuristics or linear programming-based (LP) techniques. Ghosh et al. [14] used ILOG CPLEX, an Integer LP-based technique, for determining the physical locations of cores on NoC-based VFIs and their respective V/F levels. Jin et al. [15] used a statistical heuristic that used the probability distributions of the tasks' execution times and energy consumptions under different V/F levels. The VFIs' V/F levels were determined such that tasks with large energy variations are assigned more slack and run with lower V/F level to maximize the energy saving.

A number of works have addressed the task scheduling (or task assignment) when formulating energy efficiency objectives for systems with homogenous and heterogeneous compute nodes. Leung et al. [17] proposed a list scheduling algorithm to compute the tasks priorities, executed on NoC-based equally-sized islands, based on the links communication delays. Chou et al. [18] devised an iterative task mapping heuristic that identified and grouped the neighboring idle cores of a NoC, with pre-defined V/F levels, for the application tasks assignment. Oxley et al. [19] analyzed the robustness of a set of heuristics, used for the static assignment of tasks to heterogeneous nodes, in terms of meeting makespan deadlines or energy budgets considering the stochastic tasks execution time.

The research contributions of this paper include:

- Formulating a MILP for the task-to-island assignment problem that forms the symmetric islands of tasks with similar computation behavior. In a sense, the proposed formulation aims at forming per execution phase islands based on measuring the tasks characteristics for each execution phase of the applications.
- Formulating an ILP for the VFIs' V/F level assignment problem that performs DVFS on the islands per execution phase in order to minimize the applications makespans under the user-defined energy budgets.
- Proposing a fast and low-cost heuristic to solve the task-to-island assignment (tasks partitioning) problem. The experimental results show that when

 ISBN: 978-1-61208-667-5

using the heuristic for task-to-island assignments, the system energy efficiency, measured by the Energy-Delay Product (EDP) metric, is, at worst, within 13% of the optimal per-core DVFS across the experimented benchmarks. Furthermore, the results show that the proposed framework efficiently maximizes the energy saving of low CPU-intensive benchmarks.

III. MANYCORE SYSTEM CONFIGURATION

This section presents assumptions about the multiple-VFI manycore system setup and the execution model of applications running on this system. This section also explains an applications profiling strategy that provides the task-level application characteristics that are utilized by the VFI-based optimization framework to measure the energy-performance tradeoff.

A. VFI-based Manycore System Design

This paper assumes an N-core manycore system $C = \{c_1 \dots c_N\}$, where cores are arranged in a $\sqrt{N} \times \sqrt{N}$ mesh of homogenous cores. It is assumed that the system is partitioned into a fixed number of symmetric islands, $I = \{i_1 \cdots i_K\}$ where there are $Q = N/K$ cores per island. For example, Figure 1 shows a partitioned system with $K = 3$. Also, $Q = 1$ represents a manycore system with the most fine-grained islands. The cores in a VFI operate under a common V/F level, which is determined by the V/F level assignment step of the framework. These V/F levels are attained from a range of available CPU performance states: $S = \{s_1 \dots s_L\}$ where s_1 and s_L correspond to the lowest and highest V/F levels, respectively. Any two VFIs may have the same or different V/F levels, which impact the system's overall energy efficiency. Each core has a local non-unified L1 cache and all cores share a unified L2 cache.

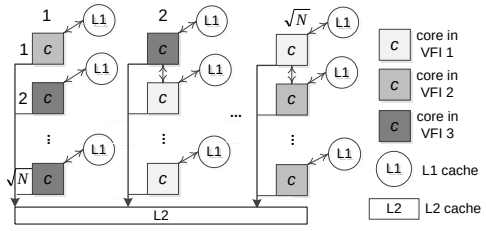

Figure 1. A manycore system with three islands

B. Application Execution Model

This paper considers multithreaded applications chosen from benchmark suites, which will be explained in Section V, where each thread of execution runs on a particular core and is not re-assigned to another core during the application execution. The execution of these applications follows Single Program Multiple Data (SPMD) parallelization technique wherein the same program is split up among cores to perform tasks on different data. These benchmarks are developed and utilized in a shared memory system that facilitates inter-core/thread data exchange at runtime [16]. The execution runs, which are used to evaluate the optimization goals, encompass a unique section inside the benchmark's source codes known as Region Of Interest (ROI).

ROIs, representing the parallel sections of the applications, are divided into multiple tasks, according to the SPMD model, and are assigned to cores/threads for the parallel execution. Because of the changing workloads of the applications (benchmarks), the execution of the ROIs represents distinct application characteristics in the form of phases or execution windows during the runtime. During the applications execution, some of threads produce data while the others consume it. To ensure that the consumer threads obtain the correct data before executing the next phase of the applications, the benchmarks' ROIs are instrumented by synchronization routines (such as barriers), which resolve, among the cores, data memory access delays within the phases, as well as data transfers across the phases of the applications. The execution of a number of instructions between two consecutive synchronization points defines a distinct computational phase of the benchmark, which are represented as the cores' parallel tasks within that execution phase. Figure 2 shows an example of an application with P execution phases where within each phase gray portions show the computation periods of cores executing their tasks and black portions show the core's idle periods. These periods, representing execution overheads, may be created by memory access delays (or data transfers) resulting in idle periods upon reaching synchronization points at the end of each phase.

Task model

An application consists of a set of tasks sets $T = \{T_1 \dots T_P\}$ defined over the P execution phases where T_j denotes a task set executed in phase j ($1 \le j \le P$) of the application. Each task set T_j is composed of tasks executed by cores in the corresponding application phase where $\tau_{j,i}$ denotes task i ($1 \le i \le N$) in phase j. Thus, it is assumed that each core executes one task in the application phase. As indicated above, the execution of a task set in the next phase is dependent on the completion of a task set in the previous phase. As such, the assignment of tasks to islands represents typical application task graphs, assuming a negligible/zero memory access delays between the dependent tasks (because the memory access delays for data transfers among the task sets are already accounted for in the tasks execution time).

The tasks partitioning formulation considers the similarity of the tasks' workloads in an execution phase to perform the task-to-island placements. The outcome of the task-to-island assignments guides the VFIs' V/F level assignment formulation to improve the system's energy efficiency by slowing down VFIs with lower workloads and speeding up the highly loaded VFIs.

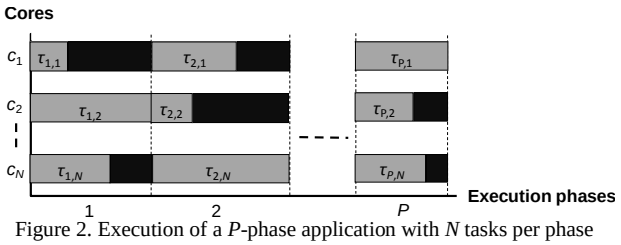

Figure 2. Execution of a P-phase application with N tasks per phase

C. Application Profiling Methodology

The optimization framework has a priori knowledge of the benchmarks/applications execution. The profiling data used for the static optimization of the task-to-island assignments and their V/F levels include the execution times, energy consumptions, and workloads of the task set for each execution phase of the benchmark collected at each possible V/F level. This paper uses a profiling strategy that runs the benchmark on the manycore system once per V/F level and collect the pertinent per phase execution time, energy usage, and workload information of all tasks in that phase. Here, the execution time corresponds to the computational period of a task in the execution phase before reaching the barrier (black portions in Figure 2). The energy consumption means the rate of the task power usage during its execution in the corresponding phase. The workload is defined as the ratio of the task's busy (computation) cycles to the total cycles (the summation of the busy and idle cycles) in the execution phase.

IV. TWO STEP TASK-TO-ISLAND ASSIGNMENT AND V/F LEVEL ASSIGNMENT TECHNIQUE

The task-to-island assignment and V/F level assignment steps are formulated in this section. To reduce the computation time of solutions obtained by the optimization framework, this paper solves the above steps sequentially. The islanding step uses the tasks' workloads to identify the groups (islands) of tasks with similar computational similarities per execution phase. The V/F level assignment step considers the execution time and energy usage of the islands under multiple V/F levels to make the best performance-energy tradeoff that minimizes the benchmarks makespan given an energy budget.

A. Task-to-Island Assignment

As mentioned above, partitioning the tasks among the islands is based on the similarity of tasks. To measure the degree of similarity among tasks, this formulation computes the percentage difference ratio between a task workload and the maximum workload in an island to which the task may be assigned. To find the maximum similarity among the tasks, this optimization step minimizes the ratio that indicates the wasted workload.

The following are the problem's objective and constraints:

$$\underset{y_{i,k},F_k,c_k,z_{k,j},x_{i,k}}{Minimize} \quad Y_j = \sum_{i=1}^{N}\sum_{k=1}^{K} y_{i,k} \quad \forall T_j \in T \tag{1}$$

$$y_{i,k} \geq x_{i,k} - t_i \cdot F_k \quad \forall \tau_{j,i} \in T_j, \forall i_k \in I \tag{2}$$

$$c_k = \sum_{j=1}^{r} a_j \cdot z_{k,j} \quad \forall i_k \in I \tag{3}$$

$$F_k = \sum_{j=1}^{r} \left(\frac{1}{a_j}\right) \cdot z_{k,j} \quad \forall i_k \in I \tag{4}$$

$$t_i \cdot x_{i,k} \leq c_k \quad \forall \tau_{j,i} \in T_j, \forall i_k \in I \tag{5}$$

$$\min(t_1 \cdots t_N) \leq c_k \leq \max(t_1 \cdots t_N) \quad \forall i_k \in I \tag{6}$$

$$\sum_{k=1}^{K} x_{i,k} = 1 \quad \forall \tau_{j,i} \in T_j \tag{7}$$

$$\sum_{i=1}^{N} x_{i,k} = Q \quad \forall i_k \in I \tag{8}$$

$$\sum_{j=1}^{r} z_{k,j} = 1 \quad \forall i_k \in I \tag{9}$$

$$z_k = \{z_{k,1} \cdots z_{k,r}\} \quad \forall i_k \in I \text{ (SOS-2 variable set)} \tag{10}$$

$$y_{i,k} \geq 0 \quad \forall \tau_{j,i} \in T_j, \forall i_k \in I \tag{11}$$

$$F_k \geq 0 \quad \forall i_k \in I \tag{12}$$

$$c_k \geq 0 \quad \forall i_k \in I \tag{13}$$

$$x_{i,k} \in \{0,1\} \quad \forall \tau_{j,i} \in T_j, \forall i_k \in I \tag{14}$$

$$z_{k,j} \geq 0 \quad \forall i_k \in I, 1 \leq j \leq r \tag{15}$$

The task-to-island assignment formulation aims at minimizing the wasted workloads of islands for every execution phase. The island's maximum workload is not known before solving the above optimization problem. Therefore, the problem objective (the percentage wasted workload) becomes non-linear. The non-linear functions are typically linearized to obtain optimum solutions more efficiently. The non-linear curve of a function representing the island's maximum workload, is linearized by a mathematical technique, known as the piece-wise linear function [19], which approximates the actual value of the non-linear function. For the linearization, this technique divides the function's non-linear curve (such as the objective function in this paper) into multiple segments of straight lines that each can be represented by a linear function.

Y_j denotes the total wasted workload in execution phase j. $y_{i,k}$ is the wasted workload of task $\tau_{j,i} \in T_j$ ($1 \leq i \leq N$) in island i_k. c_k is the approximation of island's maximum workload. F_k approximates $1/c_k$. These approximations use Special Ordered Set (type 2) variables (SOS-2), $z_{k,j}$, where each variable indicates how likely it is that a line segment, connected by two adjacent points (i.e., a_j and a_{j+1}), approximates c_k or $1/c_k$. Technically, the SOS-2 variables transform the piece-wise linear functions to a form that can be used by linear programming methods to solve optimization problems. t_i is the workload of task $\tau_{j,i}$. $x_{i,k}$ shows where task $\tau_{j,i}$ is assigned to island i_k. Q denotes the number of tasks assigned per island. r is the number of adjacent points that form the line segments.

Constraint (1) minimizes the total amount of wasted workloads for a task set across all the islands. Constraint (2) computes the wasted workload if a task is assigned to an island. Constraints (3) and (4) approximate c_k and $1/c_k$, respectively. Constraint (5) determines c_k (the maximum workload of an island). Constraint (6) ensures that the island's maximum workload is within the minimum and maximum values of tasks workload in an execution phase. Constraint (7) shows that a task is assigned to only one island. Constraint (8) indicates that all islands have an equal size. For all the SOS-2 variables defined in (10), only two of

them are non-zero. These non-zero variables, which have to be adjacent, indicate the two end points of a line segment.

Figure 3 shows an application running on a system with 2 execution phases ($P = 2$) and 4 tasks per phase ($|T_j| = 4$, $1 \leq j \leq 2$) before (3(a)) and after (3(b)) applying the task-to-island assignment formulation. For two symmetric islands ($K = 2$), it is observed from 3(b) that in the first execution phase, $i_1 = \{\tau_{1,1}, \tau_{1,3}\}$ and $i_2 = \{\tau_{1,2}, \tau_{1,4}\}$ whereas for the second execution phase, $i_1 = \{\tau_{2,1}, \tau_{2,4}\}$ and $i_2 = \{\tau_{2,2}, \tau_{2,3}\}$. For example, for the first phase in Figure 3, the wasted workload, Y_1, is computed based on i_1 and i_2 where the task pair in each island has the most similar execution workloads. It should be noted in Figure 3 that i_1 and i_2 can be executed on any combination of 4 cores in each execution phase because 1) it is assumed that the system consists of homogenous cores, and 2) the islanding is performed independently per execution phase due to the synchronization of threads at the end of the phase.

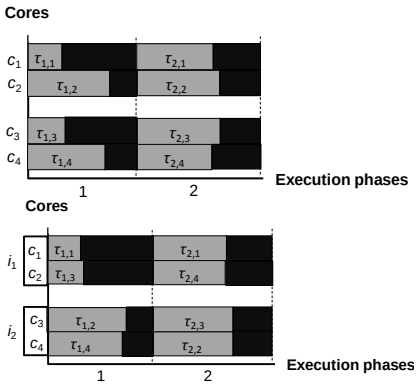

Figure 3. Application task sets with $N = 4$ and $K = 2$ showing (a) default and (b) optimized tasks assignment

B. VFIs' V/F Level Assignment

The goal of islanding step, discussed above, is to separate the islands with different workloads using the tasks computational similarity. For a given V/F level, any two islands with different workloads may have different execution performance. Such performance gap among the islands is utilized by the V/F level assignment step to maximize the system energy saving while increasing the performance within the allocated energy budget. This is accomplished by slowing down islands with low workloads and speeding up the ones with high workloads.

Running the islands (VFIs) under the fixed V/F level for the entire application execution may improve energy-performance tradeoff for applications with steady workloads but it has poor performance outcomes for applications with changing workloads at runtime. The second step in the optimization framework addresses this concern by adjusting the islands' V/F levels per execution phase of applications based on the workloads intensity of islands in the corresponding application phase.

The following are the objective and constraints for formulating the V/F level assignment problem:

$$\underset{\theta_j, a_{k,l,j}}{Minimize} \quad \theta = \sum_{j=1}^{P} \theta_j \tag{16}$$

$$\sum_{l=1}^{L} d_{k,l,j} \cdot a_{k,l,j} \leq \theta_j \quad \forall i_k \in I, \forall T_j \in T \tag{17}$$

$$\sum_{l=1}^{L} a_{k,l,j} = 1 \quad \forall i_k \in I, \forall T_j \in T \tag{18}$$

$$\sum_{j=1}^{P} \sum_{k=1}^{K} \sum_{l=1}^{L} e_{k,l,j} \cdot a_{k,l,j} \leq EB \quad EB \geq 0 \tag{19}$$

$$a_{k,l,j} \in \{0, 1\} \ \forall i_k \in I, \forall T_j \in T, \forall s_l \in S \tag{20}$$

Where, Θ is the makespan of application. Θ_j is the execution time of phase j, which is determined by the maximum finish time among islands in that phase. $d_{k,l,j}$ and $e_{k,l,j}$ are the execution time and energy consumption of a core running a task, assigned to VFI i_k, under V/F level l at the execution phase j, respectively. $a_{k,l,j}$ states whether the V/F level l is assigned to i_k in phase j. EB constrains the system energy consumption for the application execution.

The problem objective (16) minimizes the benchmark's makespan, defined by the execution times of application phases. Constraint (17) determines the execution time of a phase. Constraint (18) affirms that only one V/F level is assigned to an island per execution phase. Constraint (19) ensures that the system's energy consumption, computed by the energy usage of VFIs across all execution phases, does not exceed the user-defined energy budget.

Figure 4 depicts an example of V/F level assignment step for the same application task sets shown in Figure 3. It is observed from Figure 4 that in the first execution phase, V/F levels s_2 and s_4 are assigned to i_1 and i_2, respectively. Since i_1 has a lower computational workload than i_2 in the first phase, running it with the lower V/F level (s_2) results in saving more energy while running i_2 with the higher V/F level (s_4) improves the performance. For the second execution phase, i_1 and i_2 have comparable workloads. Thus, s_3 is assigned to both islands.

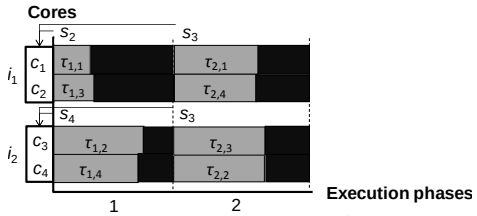

Figure 4. Per phase V/F levels assignment for islands i_1 and i_2

C. Fast Heuristic for Task-to-Island Assignment

The task-to-island assignment problem is an NP-hard problem due to its growing complexity when experimenting with larger task sets size or the number of islands per execution phase. To reduce the computation time of solving this problem, this section presents a fast, practical heuristic that only requires a sorting procedure in its most time-consuming step.

This heuristic performs the following two steps per application phase: 1) tasks are sorted in the increasing order of their execution workloads. In other words, the sorting procedure orders the tasks (i.e., from small to large tasks) based on their computational workloads. As mentioned in

Section III-C, the execution workload refers to the task utilization measured over an application phase's time span and is computed as the ratio of the core's busy cycles to the total execution cycles. The utilization values do not significantly change when running cores/islands with different V/F levels at runtime. Therefore, this performance measure was chosen for the task-to-island assignments in each execution phase. 2) every $Q = N/K$ consecutive sorted tasks are assigned to an island (N and K are number tasks and islands, respectively). The time complexity of step (1) increases with $O(N \cdot \log(N))$ in the best case while step (2) is performed in constant time.

It should be mentioned that the assignment of tasks to islands implies that Q cores are allocated to the corresponding Q tasks assigned to an island because the application execution model (see Section III-B) assumes that a core executes only one task in each phase.

D. Real-life Realization of VFI-based System

The application of the proposed optimization framework in embedded systems is useful when multicore processors are designed to run specific applications many times given system configurations that are pre-optimized once at compile-time. To use this framework for such cases, the applications are first profiled using the profiling method explained in Section III-C. At compile-time, the islanding step assigns tasks to islands and the V/F level assignment step determines the VFIs' V/F levels. The per VFI, per execution phase V/F levels are then stored in a look-up table to be used later at runtime when at the start of each execution phase the OS fetches the V/F levels from the table and uses special registers to communicate the V/Fs with DVFS controllers that tune the islands' performance.

V. Experimental Setup

To measure the energy efficiency of the proposed framework, General Execution-driven Multiprocessor simulator (GEM5) [20], a full-system simulator, is used to model 64 cores that are arranged as a 8x8 mesh structure of homogenous cores, where each core has 64KB L1 instruction and data caches and a shared 8MB L2 cache. All the benchmarks are run 4 times using the following V/F levels: s_1: 0.5V/ 1.25GHz, s_2: 0.667V/ 1.666GHz, s_3: 0.834V/ 2.083GHz, s_4: 1.0V/ 2.5GHz, which are within a nominal range of states that provide stable performance and power data. The per execution phase task sets workload and execution time are collected as explained in Section III-C. To obtain the phases' energy consumption, the GEM5's performance outputs are fed to Multicore Power, Area, and Timing (McPAT) [21] that generates the energy consumption for the task sets. The time/energy overheads caused by V/F level switching are not incorporated in the optimization objectives and constraints because they are only about a few hundreds of nano seconds/Joules order of magnitude [6].

The proposed two-step optimization framework is tested on three benchmarks, namely Fast Fourier Transform (FFT), Lower and Upper triangular matrices (LU), and Cache-Aware Annealing (CANNEL) [22][23]. These benchmarks are used in different application domains and represent applications with high or low CPU-intensiveness: the percentage of compute intensity of FFT, LU, and CANNEAL is 96%, 92%, and 85%, respectively where FFT and CANNEAL are high and low CPU-intensive benchmarks, respectively.

Similar to [8], the 64-core system, used in this paper, is partitioned into 4 islands ($K = 4$) where each island has 16 ($Q = 16$) tasks, whose assignments to islands are defined by the islanding formulation in Section IV-A. This configuration was chosen to assign sufficient tasks per island in each execution phase.

The formulations, discussed in Section IV, are implemented with a modeling language, Algebraic Language for Mathematical Programming (AMPL) [24], which is used for modeling large-scale constrained optimization problems. To find solutions that make the best energy-performance tradeoff, Gurobi [25], a solver included in the AMPL software package, is used to solve the islanding and V/F level assignment problems. The heuristic is implemented and solved in MATLAB. All experiments for the symmetric VFI-based system are conducted on a CentOS workstation with Intel dual Core x86, 3.3 GHz processor and 3.6 GB RAM. The time and energy usage of workstation's physical cores when running AMPL/Gurobi are not included in the formulations since the problems are solved pre-runtime.

VI. Experimental Results

The experimental results consist of four parts. The first part presents the performance (execution time) of benchmarks under the proposed VFI-based optimization framework compared to the optimal performance obtained by the per-core DVFS VFIs. The second part demonstrates the framework's impact on system energy efficiency using two well known metrics. The third part explains the VFIs' V/F level assignment outcomes. The fourth part discusses the optimality of heuristic islanding and VFIs' V/F level assignments.

Figure 5 and Figure 6 refer to the per-core DVFS as Fine-Grained (FG) since $K = N$ (K and N are the number of islands and tasks, respectively) and dynamically tuned VFI system as DCG (Dynamic Coarse-Grained) because the V/F levels of a group of cores are adjusted per execution phase. To constrain the energy budget, the MILP-based formulation considers three levels for EB (19): High (EB(H)), Medium (EB(M)), and Low (EB(L)), which correspond to 7.5%, 22.5%, and 37.5% energy reductions from the benchmarks' energy consumption when all cores run at the fastest V/F level (s_4 in Section V).

There is a large body of research that use (meta) heuristics, greedy, and machine learning techniques for assigning tasks to cores and determining the cores' V/F levels to obtain the best objective values [26]. Instead of comparing the proposed framework performance to such a wide range of existing techniques in the literature, it is compared to the per-core DVFS, which is considered as the most energy-efficient method in high performance computing platforms. Moreover, the degree to which the VFI-based system's energy efficiency is close to the per-core

 ISBN: 978-1-61208-667-5

DVFS indicates how close the proposed framework's outcomes are to the optimal solutions.

A. Execution Time Comparison

The ILP-based formulation minimizes the performance (16) of running symmetric coarse-grained islands under the energy budget levels. Figure 5 evaluates DCG vs. FG performance (execution time) relative to the non-DVFS baseline, when all cores operate at the fastest V/F level (s_4), using the following criteria:

1) Energy Budget

Intuitively, decreasing the energy budget increases the benchmarks execution times because the islands are slowed down to consume less energy below the energy budgets. Interestingly, for EB(H) in Figure 5, the performance of DCG is comparable to FG. The reason is that for EB(H) the execution time of islands with high workloads dominate the execution time of under-loaded ones. Thus, scaling up the V/F levels of highly loaded islands in DCG improves the system performance while slowing down the under-loaded islands not only has a negligible impact on the overall benchmark execution time but also increases energy saving. By further decreasing the energy budget, the highly loaded islands have to run slower, resulting in a noticeable execution time increase for EB(M) and EB(L).

2) Benchmarks CPU-intensiveness

Regarding the impact of benchmarks CPU intensity on DCG, Figure 5 shows that for CANNEAL the system performance penalty stays below 18% across the energy budgets. This is due to the low CPU-intensiveness of CANNEAL whose execution time is not degraded by lowering the energy budget. As such, for CANNEAL, the DCG performance is closer to FG compared to FFT and LU. Since LU has low CPU-intensiveness in some phases, it is observed from Figure 5 that in EB(H) DCG performance is

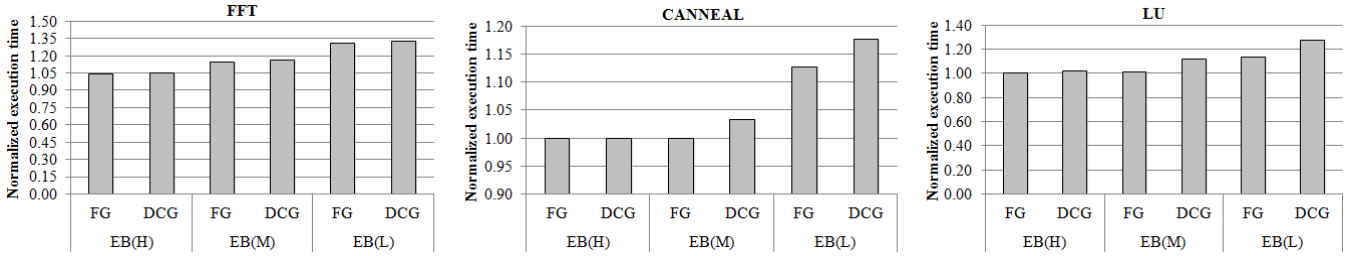

Figure 5. Execution time of Fine-Grained (FG) and Dynamic Coarse-Grained (DCG) system configurations over High (H), Medium (M), and Low (L) Energy Budgets (EB). The execution times are normalized to non-DVFS baseline.

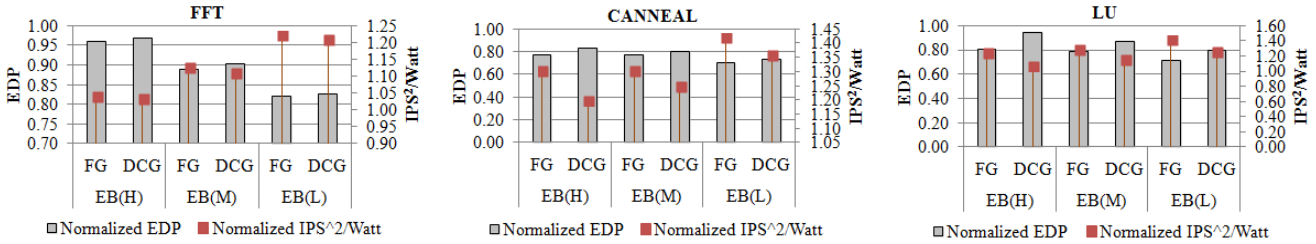

Figure 6. EDP and IPS^2/Watt of Fine-Grained (FG) and Dynamic Coarse-Grained (DCG) system configurations over High (H), Medium (M), and Low (L) Energy Budgets (EB). The EDP and IPS^2/Watt are normalized to non-DVFS baseline.

close to FG. Clearly, for a CPU-intensive benchmark like FFT, DCG has the poorest performance when the VFIs run slower in the lower energy budgets.

B. Energy Efficiency Metrics Comparison

Besides measuring the framework impact on application performance, the following metrics are used to evaluate the system energy efficiency: 1) Energy-Delay Product (EDP) and 2) Instructions Per Second, per Watt (IPS^2/Watt) [27]. The former measures the amount of energy saving obtained despite performance loss while the latter specifies the amount of throughput gained in exchange for consuming power for running a number of instructions in a time period (e.g., execution phase in the application model). Lower values for EDP and higher values for IPS^2/Watt are desirable.

Figure 6 shows the framework impact on EDP and IPS^2/Watt resulting from the application of the FG and DCG configurations normalized to the corresponding EDP and IPS^2/Watt of non-DVFS for the same benchmarks.

Figure 6 suggests that CANNEAL, compared to FFT and LU, obtains the best (lowest) EDP across the energy budget. Especially, in EB(H), DCG utilizes the CANNEAL's memory access times to maximize energy saving without losing performance while, as a CPU-intensive benchmark, most of the FFT's execution run consists of floating-point instructions, which provide less opportunity for energy saving and cause the EDPs of FG and DCG to be close to one another in EB(H). For LU, compared to FFT and CANNEAL, the EDP gap between FG and DCG is larger, which can be explained by the LU's workload variations across its execution phases. Overall, the average EDP improvements of DCG, compared to non-DVFS, are within

 ISBN: 978-1-61208-667-5

1%, 5%, and 13% of the best EDP improvements obtained by FG for FFT, CANNEAL, and LU, respectively.

IPS^2/Watt is inversely proportional to EDP. Thus, the relative energy efficiency of FG and DCG in terms of IPS^2/Watt is similar to EDP. Figure 6 specifies finer scaling range for the IPS^2/Watt axis compared to EDP to show a clearer difference between the energy-efficient solutions obtained by these two configurations across the energy budgets. Of note, in Figure 6, the upper bound limit of EDP axis is set to 1 to show the EDP improvements against the non-DVFS baseline across the studied benchmarks.

C. VFIs' V/F Levels

As mentioned in Section IV-B, the V/F assignment step tunes the VFIs performance to increase the system energy saving by lowering the V/F levels of less loaded islands and increasing the V/Fs for the heavily loaded ones. The extent to which the islands V/F levels are scaled up or down, depends on the overall characteristics of benchmarks.

Table I shows the V/F states distribution among all islands and across all the execution phases of FFT, LU, and CANNEAL at the high energy budget (EB(H)). This table suggests that the highest V/F level (s_4) constitutes the largest percentage of assignments for FFT and LU (68% for FFT and 61% for LU). This observation matches the high CPU-intensiveness of these benchmarks having highly loaded islands and their V/F states are scaled up to maximize the performance. On the other hand, for CANNEAL, a lower V/F state (s_3) is assigned to 65% of islands, which again corroborates with the low CPU-intensiveness of CANNEAL since lowering V/F levels for such benchmarks saves energy without significant performance loss. Table I also shows that for LU s_1 and s_2 are used for the V/F assignment. That's because some execution phases of LU have less amount of computation, which are utilized by the V/F level optimization step for slowing down the VFIs and saving energy.

TABLE I. PERCENTAGE OF V/F LEVELS ASSIGNED TO VFIS

Benchmark	s_1	s_2	s_3	s_4
FFT	0	0	31	69
LU	16	10	12	62
CANNEAL	0	0	65	35

D. Optimality Analysis of Solutions obtained by Heuristic and ILP-based Formulation

Section IV-C explained a heuristic for the MILP-based formulation of islanding problem. To find out the extent to which the heuristic solutions are close to optimal, the MILP-based formulation, which provides optimal solutions, is solved for a number of execution phases of the experimented benchmarks. To solve the associated problems, the heuristic task-to-island assignments (Section IV-C), are used as initial solutions. For larger problems size, the experiments are run for a week after which it was observed that the differences of solver's objective values (1) were negligible (less than a percent) compared to the objective values obtained by the heuristic and used as the initial seeds to solve the MILP-based formulation. Considering such minimal difference, the islands, obtained by the MILP-based formulation and heuristic, were found to be identical, indicating that the proposed heuristic performs optimally to solve the islanding problem. For $N = 64$, $K = 4$, and $r = 10$ used for Section IV-A, the MILP-based formulation has 560 variables and 644 constraints per application's execution phase.

The computation complexity of solving ILP-based V/F level assignment problem (Section IV-B) depends on the number of islands (K), number of V/F levels (L), and number of execution phases (P). To solve the V/F level assignment problem for DCG (coarse-grained VFIs), $K = 4$, $L = 4$, and P is set to 8, 15, and 31 for FFT, LU, and CANNEAL, respectively. Using the above parameter values, the ILP-based problems are optimally solved within a minute, from which the associated performance and energy efficiency results are obtained as shown in Figure 5 and Figure 6.

VII. CONCLUSION

This paper presented a framework that optimizes the tasks partitioning and VFIs' V/F levels to minimize the benchmarks makespan without exceeding the user-defined allocated energy budget. Furthermore, this paper proposed a fast, low-cost heuristic that has optimal performance for the experimented problems sizes. The energy efficiency of the coarse-grained VFI-based system was compared to the optimal per-core DVFS on multiple benchmarks and with different energy budgets. While using multiple VFIs lowers the manufacturing and operating costs of manycore chips, the results showed that the VFI system's EDP, at the worst case, was within 13% of the EDP obtained by the per-core DVFS. The results also showed that the proposed framework gains greater EDP improvements for benchmarks with low CPU-intensive workloads.

According to [28], it is estimated that data centers in the U.S. are expected to consume electricity up to 73 billion kilowatt-hours per year from 2014 to 2020, which cost the American businesses $6 billion annually. Based on this report, the most efficient technologies and management practices will save energy up to 40% in 2020. Considering the system configuration used in this paper, the proposed framework saves more than 30% of energy in EB(L). Even if this framework reduces energy by 5% when it is deployed on larger system sizes, it will have a big economic impact on the energy costs of the future high performance computing.

REFERENCES

[1] C. Isci, A. Buyuktosunoglu, C.-Y. Cher, P. Bose, M. Martonosi, "An Analysis of Efficient Multi-Core Global Power Management Policies: Maximizing Performance for a Given Power Budget," *Proc. IEEE MICRO*, pp. 347-358, 2006.

[2] S. Borkar, "Thousand core chips: a technology perspective," *Proc. ACM DAC*, pp. 746-749, 2007.

[3] S. Pagani, A. Pathania, M. Shafique, J. J. Chen, J. Henkel, "Energy Efficiency for Clustered Heterogeneous Multicores," IEEE Trans. TPDS, pp. 1315-1330, 2017.

[4] S. Hajiamini, B. Shirazi, C. Cain, H. Dong, "Optimal Energy-Aware Scheduling in VFI-enabled Multicore Systems," *Proc. IEEE HPCC*, pp. 490-497, 2017.

[5] K. Greene. 2006. MIT Technology Review [online].

 ISBN: 978-1-61208-667-5

https://www.technologyreview.com/s/406760/the-trouble-with-multi-core-computers/ [retrived: July, 2018]

[6] W. Kim, M. S. Gupta, G. Y. Wei, D. Brooks, "System level analysis of fast, per-core DVFS using on-chip switching regulators," *Proc. IEEE HPCA*, pp. 123-134, 2008.

[7] U. Y. Ogras, R. Marculescu, D. Marculescu, E. G. Jung, "Design and management of voltage-frequency island partitioned networks-on-chip," IEEE Trans. VLSI, 17, pp. 330-341, 2009.

[8] K. Duraisamy et al., "Energy efficient MapReduce with VFI-enabled multicore platforms," *Proc. ACM DAC*, pp. 1-6, 2015.

[9] U. Y. Ogras, R. Marculescu, D. Marculescu, "Variation-adaptive feedback control for networks-on-chip with multiple clock domains," *Proc. ACM/IEEE DAC*, pp. 614-619, 2008.

[10] R. David, P. Bogdan, R. Marculescu, U. Ogras, "Dynamic Power Management of Voltage-Frequency Island Partitioned Networks-on-Chip using Intel Single-Chip Cloud Computer," *Proc. ACM/IEEE NOCS*, pp. 257-258, 2011.

[11] S. Jin, S. Pei, Y. Han, H. Li, "A Cost-Effective Energy Optimization Framework of Multicore SoCs Based on Dynamically Reconfigurable Voltage-Frequency Islands," *ACM Trans. Des. Autom. Electron. Syst.* 21, pp. 1-14, 2012.

[12] M. Ozen and S. Tosun, "Genetic algorithm based NoC design with voltage/frequency islands," *Proc. IEEE AICT*, pp. 1-5, 2011.

[13] P. Ghosh and A. Sen, "Efficient mapping and voltage islanding technique for energy minimization in NoC under design constraints," *Proc. SAC*, pp. 535-541, 2010.

[14] S. Jin, Y. Han, S. Pei, "Statistical energy optimization on voltage–frequency island based MPSoCs in the presence of process variations," *Elsevier J. Microelectronics*, pp. 54, 23-31, 2016.

[15] N. Barrow-Williams, C. Fensch, S. Moore, "A communication characterisation of Splash-2 and Parsec," *Proc. IEEE IISWC*, pp. 86-97, 2009.

[16] Lap-Fai Leung and C-Y. Tsui, "Energy-aware Synthesis of Networks-on-chip Implemented with Voltage Islands," *Proc. DAC*, pp. 128-131, 2007.

[17] C. L. Chou and R. Marculescu, "Incremental run-time application mapping for homogeneous NoCs with multiple voltage levels," *CODES+ISSS*, pp. 161-166, 2007.

[18] M. A. Oxley et al., "Makespan and Energy Robust Stochastic Static Resource Allocation of a Bag-of-Tasks to a Heterogeneous Computing System," IEEE Trans. TPDS, pp. 26, 2791-2805, 2015.

[19] B. Hamann and J.L. Chen, "Data point selection for piecewise linear curve approximation," in *Computer Aided Geometric Design*, 11, pp. 289-301, 1994.

[20] N. Binkert et al., "The gem5 simulator," *ACM Comput. Archit. News*, 39, pp. 1-7, 2011.

[21] S. Li et al., "McPAT: an integrated power, area, and timing modeling framework for multicore and manycore architectures," *IEEE Proc. MICRO*, pp. 469-480, 2009.

[22] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh, A. Gupta, "The SPLASH-2 programs: characterization and methodological considerations," *ACM Comput. Archit. News*, pp. 24-36, 1995.

[23] C. Bienia, S. Kumar, J. Singh, K. Li, "The PARSEC benchmark suite: characterization and architectural implications," *Proc. ACM PACT*, pp. 72-81, 2008.

[24] D. M. Gay, "The AMPL Modeling Language: An Aid to Formulating and Solving Optimization Problems," *Proc. Mathematics & Statistics*, pp. 134, 95-116, 2016.

[25] AMPL Products: Solvers. 2018. https://ampl.com/products/solvers/ [retrived: July, 2018]

[26] S. Mittal, "A survey of techniques for improving energy efficiency in embedded computing systems," *IJCAET*, 6, pp. 440-459, 2014.

[27] R. Kumar, K. I. Farkas, N. P. Jouppi, P. Ranganathan, D. M. Tullsen, "Single-ISA heterogeneous multi-core architectures: the potential for processor power reduction," *Proc. IEEE MICRO*, pp. 81-92, 2003.

[28] https://eta.lbl.gov/publications/united-states-data-center-energy [retrived: July, 2018]

 ISBN: 978-1-61208-667-5

SeDuCe: a Testbed for Research on Thermal and Power Management in Datacenters

Jonathan Pastor
IMT Atlantique - Nantes
jonathan.pastor@imt-atlantique.fr

Jean Marc Menaud
IMT Atlantique - Nantes
jean-marc.menaud@imt-atlantique.fr

***Abstract*—With the advent of Cloud Computing, the size of datacenters is ever increasing and the management of servers and their power consumption and heat production have become challenges. The management of the heat produced by servers has been experimentally less explored than the management of their power consumption. It can be partly explained by the lack of a public testbed that provides reliable access to both thermal and power metrics of server rooms. In this article, we propose SeDuCe, a testbed that targets research on energy and thermal management of servers, by providing public access to precise data about the power consumption and the thermal dissipation of 48 servers integrated in Grid'5000 as the new *ecotype* cluster. We present the chosen software and hardware architecture for the first version of the SeDuCe testbed, and propose some improvements that will increase its relevance.**

Keywords–Datacenters; Scientific testbed; Thermal management; Power management; Green computing.

I. INTRODUCTION

The advent of web sites with a global audience and the democratization of Cloud Computing have led to the construction of datacenters all over the world. Datacenters are facilities that concentrate from a few servers up to hundreds of thousands of servers hosted in rooms specially designed to provide energy and cooling for the servers. These facilities are widely used for applications from web services to *High Perfomance Computing* (HPC).

In recent years, the size of datacenters is ever increasing, which leads to new challenges such as designing fault tolerant software to manage at large scale the servers and energy management of server rooms. On the latter challenge, many research efforts have been conducted [1] [2], most of them focusing on the implementation of on demand power management systems, such as Dynamic voltage scaling (DVFS) [3] [4] and vary-on vary-off (VOVO) [5] [6]. Some work has been made to extend existing scientific testbeds with power monitoring of experiments: for example Kwapi [7] enables researchers to track the power consumption of their experiments conducted on Grid'5000.

On the other hand, the thermal management of servers has been less explored, a large part of the existing work considering only simulations [8]. This can be explained, partly, by the difficulty of conducting experiments involving thermal monitoring of servers: to ensure that the data recorded experimentally is valid, experimentions must be conducted on a testbed that contains many temperature sensors, not only positioned on cooling systems, but also at the front and the back of each server of the racks.

In addition, such a testbed must enable reproducible experimentations, by providing its users with a full control on experimental conditions like setting the temperature of the environment of their experiments and by exposing its data in a non misleading way, via a well documented Application Programming Interface (API).

Finally, as power management and temperature management of servers are related problems [9], there is a need for a testbed that enables users to access to both thermal and power data of servers.

As far as we know, there is no public testbed that enables researchers to work on both energy and thermal aspects of servers functionning. The objective of the SeDuCe - Sustainable Data Centers - project is to propose such a testbed: the SeDuCe testbed enables its users to use, in the context of the new *ecotype* cluster of the Grid'5000 infrastructure [10], 48 servers located in 5 airtight racks with a dedicated Central Cooling System (CCS) positioned inside one of the rack. In parallel of conducting the experiment by leveraging the tools provided by Grid'5000, users can get access to thermal and power data of the testbed via a web portal and a user-friendly API. The stability of experimental conditions is guaranteed by hosting the testbed in a dedicated room equipped with a secondary cooling system (SCS) that enables a precise thermoregulation of the environment outside the cluster. As resources of the testbed are made publicly available via the Grid'5000 infrastructure, all its users are able to perform reproducible research on thermal and power management of servers. The rest of the paper is structured as follows. In Section II, we describe the SeDuCe testbed, in Section III an experimental validation of the testbed is conducted. In Section IV we detail the future work on SeDuCe. Finally, we conclude in Section V.

II. TESTBED DESIGN

A. Ecotype: a Grid'5000 cluster dedicated to the study of power and thermal management of servers

In [11], we introduced our initial work on the *ecotype* cluster: we have builded the "ecotype" cluster, which contains 48 servers, and is integrated in the Grid'5000 infrastructure: any Grid'5000 user can reserve servers of the *ecotype* cluster and conduct experiments on them by using the usual Grid'5000 tools. The testbed is designed for research related to power and thermal management in datacenters: during an experiment, a user can access in real time to information regarding the temperature of the servers involved in its experiment, and get the power consumption of any parts of the testbed (servers, switches, cooling systems, etc.), or control some parameters of

 ISBN: 978-1-61208-667-5

the testbed, such as setting temperature targets for the cooling systems of the cluster.

Servers of the *ecotype* cluster are based on DELL PowerEdge R630 and contains a pair of Intel Xeon E5-2630L v4 CPUs (10 cores, 20 threads per CPU), 128GB of RAM, and 400GB Solid State Disk (SSD). The CPUs have been designed to have a lower power consumption than other CPUs of the XEON 26XX serie, with a Thermal Design Power (TDP) of 55W. Each server is connected via two 10GBe links to the Grid'5000 production network, and via a single 1GBe link to the Grid'5000 management network. For instance, the Grid'5000 production network is used for transferring the disk images required to deploy an experiment or to support communications between experimental components, while the management network is mainly used by the Grid'5000 backend to communicate with management cards of servers to turn them on and off. Additionally, each server is certified to work in hot environments where temperature can be up to 35°C. These hardware specifications will enable users to perform experiments at different levels of temperature.

The cluster is composed of 5 air-tights racks (Z1, Z2, Z3, Z4, Z5) based on the *Schneider Electric IN-ROW* model. These air-tights racks are equipped with Plexiglas doors, and create a separation between the air inside the racks and the air from outside the racks. As shown on Figure 1, one rack (Z3) is used for the cooling the cluster by hosting a dedicated Central Cooling System (CCS), while remaining racks are computing racks and are dedicated to hosting servers. The racks are connected and form two alleys: a cold alley at the front of servers and a hot alley at their back.

As depicted by Figure 1, each computing rack hosts 12 servers, and is organized following two layouts of server positions: one layout where servers are organised in a concentrated way with no vertical space between servers (Z1 and Z2), and a second layout where servers are spaced at 1U intervals (Z4 and Z5).

We have deliberately chosen to use these two layouts: they will enable users to study the impact of the server density over the temperature and the power consumption of servers.

In addition to the servers, the cluster also contains three network switches that are in charge of connecting servers to the production network and the management network of the Grid'5000 infrastructure. Three racks (Z2, Z4, Z5) are hosting each one a network switch.

The 5 racks of the cluster are based on *Schneider Electric IN-ROW* racks. This rack model creates an inside airtight environment for servers, and guarantees that the environment outside the cluster has a limited impact on temperatures inside the racks. The temperature inside the cluster is regulated by the CCS, which is connected to a dedicated management network and implements a service that enables to remote control the cooling and to access its operating data with the Simple Network Management Protocol (SNMP) protocol. The CCS has several temperature sensors located at different parts of the racks, which are in charge of checking that the temperature inside racks is under a specified temperature threshold. It is possible to change the temperature that the CCS have to maintain inside the racks (*Cooling Temperature Target* parameter), and also change the temperature of the air injected by the CCS in the cold aisle (*Air supply Temperature Target* parameter). This will, in addition to the fact that servers have been designed to work in hot environments, enable users perform their experiments at several levels of temperature.

Regarding the temperature outside the cluster, it is regulated by the SCS which is mounted from the ceiling of the server room: the SCS is in charge of maintaining a constant temperature in the server room, and thus it prevents any event outside the racks to disturb the experiments that are conducted on the SeDuCe testbed.

Finally, we have installed several "Airflow management panels" between each pair of servers: they improve the cooling efficiency by preventing the mixing of cold air and hot air inside the racks.

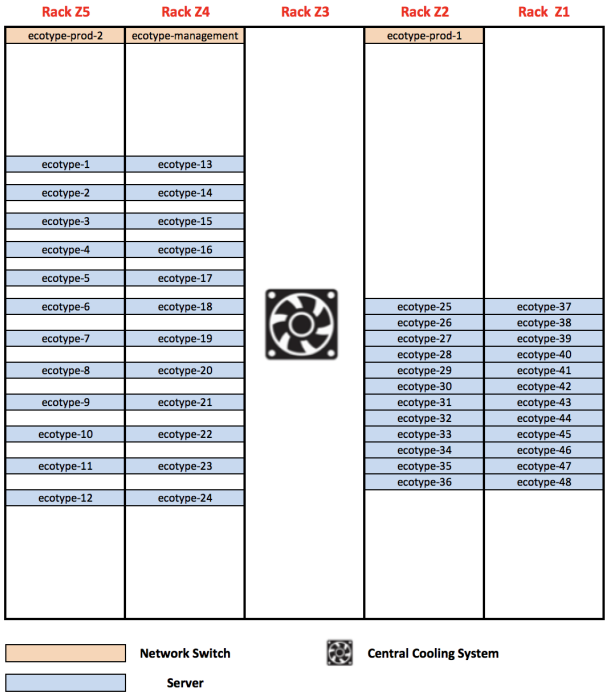

Figure 1. Layout of the ecotype cluster (front view)

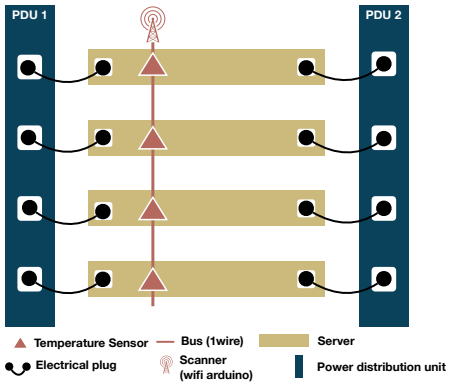

Figure 2. Back view of a server rack

B. Power Monitoring

The power consumption of each element composing the cluster (servers, network switches, cooling fans, condensators, etc.) is monitored and stored in a database, at a frequency of one hertz (one record per second).

Electrical plugs of servers and network switches are connected to Power Distribution Units (PDUs), which are in

charge of ensuring that servers and network switches can meet their power needs. Each computing racks contains two PDUs, and each server of a computing rack has two electrical plugs. As depicted in Figure 2, the two electrical plugs of a server are connected to two different PDUs, which enables servers to implement electrical redundancy. In turn, the PDUs share power consumption of servers and network switches via a dedicated service network: the power consumption of each power plug can be fetched by issuing an SNMP request to the PDU to which it is connected. In turn, the PDU provides the power consumption of each of its outlets.

The energy consumption of the CCS can similarly be fetched via SNMP requests: the CCS implements a SNMP service which is able to provide the overall power consumption and the power consumption of each of its internal part such as the condensator or the fans. On the other hand, the SCS does not implement any built-in networking access, and thus cannot share its metrics with any component over a network. To solve this problem, we instrumented several parts of the SCS by using a *Fluksometer* [12]: a *Fluksometer* is a connected device that can monitor several electrical metrics (power consumption, voltage, amperage, etc.) and expose their values over a network via a web-service at a frequency of one hertz.

Finally, we have added an additional system that tracks overall power consumption of servers, switches and the CCS. This additional system is based on the *Socomec G50 metering board* [13], and enables to check the soundness of the aforementioned source of power consumption. These additional metrics are fetched by using the modbus protocol.

C. Temperature Monitoring

To track the thermal behavior of the *ecotype* cluster, each server is monitored by a pair of temperature sensors: one sensor is positioned at the front of the server (in the cold aisle) and another sensor is positioned at the back of the server (in the hot aisle).

As depicted by Figure 2, each temperature sensor is part of a bus (based on the *1wire* protocol) connected to a *Scanner* (based on an Arduino that implements wifi communication) in charge of gathering data produced by temperature sensors of the bus. As the front and the back of each server is monitored by temperature sensors, each computing rack has in total two *Scanners* and two buses: a front bus for monitoring the cold aisle and a back bus dedicated to the hot aisle. *Scanners* fetch temperatures from their sensors at a frequency of one reading per sensor every second.

Temperature sensors are based on the DS18B20 sensor produced by "Maxim Integrated" [14] that costs approximately 3$ per sensor. According to the specifications provided by the constructor, the DS18B20 is able to provide a temperature reading every 750ms with a precision of 0.5°C between -10 °C and 85 °C.

The choice of the DS18B20 has been motivated by the fact that the DS18B20 sensor is able to work as part of an *1wire* bus. In the context of the SeDuCe infrastructure, 12 DS18B20 sensors are connected together to form an *1wire* bus, and a *Scanner*, based on an *nodeMCU* arduino with built-in wifi capabilities, fetches periodically their temperature readings. The current version of the firmware used by *Scanners* scans an 1wire bus every second, and then pushes temperature data to a *Temperature Registerer* service, as illustrated in Figure 3.

We also developed a contextualisation tool to generate firmwares for the *Scanners*. It leverages the PlatformIO framework [15] to program a *Scanner* that pushes data to a web-service. Using this contextualisation tool is simple: a developer needs to define a program template in a language close to C language and marks some parts of code with special tags to indicate that these parts need to be contextualized with additional information, such as initializing a variable with the ID of a *Scanner* device or with the address of a remote web-service (such as the one that will receive temperature records). The contextualisation tool takes this program and a context as input parameters, analyses the template program, and completes parts that requires contextualisation with information provided in the context, which results in valid C language source file. Then, the firmware is compiled and automatically uploaded to *Scanners* via their serial ports. By leveraging this contextualisation tool, we can remotely configure *Scanners* and update their firmware "on the fly".

D. SeDuCe portal

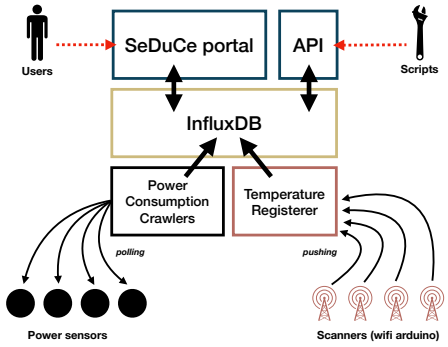

Figure 3. Architecture of the SeDuCe portal

To help users to easily access power and thermal metrics generated by the SeDuCe testbed, we developed a platform that exposes publicly two components: a web portal [16] and a documented API [17].

As illustrated by Figure 3, the web portal and the API fetch data from a Time Series Database (TSDB) based on *InfluxDB* [18]. *InfluxDB* enables to store a large quantity of immutable time series data in a scalable way. In the background, *InfluxDB* creates aggregates of data by grouping periodically data from a same series. These aggregated sets of data enable the web portal to promptly load data used for visualization.

Two kind of components are in charge of inserting data in the database : the *Power consumption crawlers* and the *Temperature Registerer*. *Power consumption crawlers* are programs that are in charge of polling data from PDUs, Socomecs, Flukso, the CCS and the SCS. In turn, this data is inserted in the database. On the other hand, the *Temperature Registerer* is a web service that receives temperature data pushed from *nodeMCU* arduino devices, and inserts it in the database.

The web portal and the API are both written in Python and leverage the "Flask" micro web framework [19]. The API component makes an extensive use of the Swagger framework [20] which automatises the generation of complete REST web services and their documentations from a single description file (written in JSON or YAML). This choice has enabled us to focus on the definition and the implementation of the API, by reducing the quantity of required boilerplate code.

 ISBN: 978-1-61208-667-5

All the components depicted in Figure 3 are implemented as micro-services. Our system is able to register 200 metrics per seconds with minimal hardware requirements (it is currently hosted on a single computer). In the case we add more sensors to our testbed, it is likely that the existing components would be sufficient. In the case that one of the component would not be able to cope with the additional workload, it would be easy to setup an high availability approach by using a load balancer such as *NGINX* that would forward requests to a pool of instances of the component.

III. EXPERIMENTATION

To illustrate the potential of the SeDuCe platform, we conducted two experiments that mix energetic and thermal monitoring. The objective of the first experiment was to reproduce a known scientific result by using both temperature and power data from the SeDuCe testbed, while the objective of the second experiment was to study the impact of parameters of the CCS over the overall power consumption.

A. First experiment: studying the impact of idle servers on cooling

The goal of this first experiment is to verify that the data produced by the SeDuCe testbed is reliable, by designing an experiment that will use both the thermal and the power data produced by the testbed. These data would be used to reproduce a scientifically validated observation, such as the impact of idle servers on the power consumption and the temperature of a server room. Such experiment has been conducted in the past [9], however as far as we know there is no public testbed that would enable researchers to reproduce this result: by reproducing this result on the SeDuCe testbed, we think that it would demonstrate the soundness of our approach and the usefulness of our testbed.

1) Description of the experiment: To illustrate the scientific relevance of our testbed, we wanted to reproduce the observations made by third party publication [9].

In [9], authors have highlighted an interesting fact: in a datacenter idle servers (i.e. servers that are turned on while not being used to execute any workload) have a significant impact on power consumption and heat production. We decided to try to reproduce this observation.

For this experiment, servers of the *ecotype* cluster are divided in three sets of servers:

- *Active servers*: servers with an even number (ecotype-2, ecotype-4, ..., ecotype-48) were executing a benchmark that generates a CPU intensive workload.
- *Idle servers*: a defined quantity (0, 6, 12, 18, 24 servers) of servers with an odd number (ecotype-1, ecotype-3, ..., ecotype-47) was remaining idle.
- *Turned-off servers*: remaining servers were electrically turned off.

and during one hour we recorded the power consumption of the CCS and the average temperature in the hot aisle of the *ecotype* cluster. The CPU intensive workload was based on the "sysbench" tool : the goal was to stress CPUs of each servers, resulting in an important power consumption and a bigger dissipation of heat. To guarantee the statistical significance of the measurements, each experimental configuration was repeated 5 times, leading to a total number of 25 experiments.

We executed two sets of experiments: one with the SCS turned-on (Figure 4) and the other while the SCS was turned off (Figure 5). The objective of turning off the SCS was to identify the impact of the SCS over the CCS.

2) Results: Figure 4 plots the cumulated power consumption of the CCS and the average temperature in the hot aisle of the cluster with the SCS enabled.

First, it is noticeable that as the number of idle nodes increases, both the energy consumed by the SCS and the temperature in the hot aisle of the rack increase. This can be explained by the fact that an idle node consumes some energy and produces some heat, which increases the workload of the CCS.

The second element highlighted by Figure 4 is that the impact of idle nodes is not linear: the red line representing the CCS consumption follows an exponential pattern and the blue line representing the average temperature in the hot aisle follows a sigmoid pattern. The exponential pattern of the power consumption of the CCS can be explained by the fact that the heat produced by a given server has an impact on the temperature of surrounding servers, thus creating hot spots in the the cluster. The CCS has it own monitoring of the temperature of servers thanks to many temperature sensors located in the racks (these sensors are independent to the ones we installed on the buses). These hot spots are detected by some of the many temperature sensors of the CCS and this leads to an activation of the CCS to reduce the temperature of hot spots to the temperature target configured in the CCS. The more the number of idle servers increases, the more the CCS must be activated to maintain its temperature target. On the other hand, the sigmoid pattern of the average temperature in the hot aisle is explained by the fact that when the number of idle servers is higher than 12, the functionning of the CCS is more intensive and thus the additional production of heat by server is absorbed by the CCS, and thus the average is growing at a slower rate.

Figure 5 plots the cumulated power consumption of the CCS and the average temperature in the hot aisle of the cluster while the SCS is disabled. This figure highlights that the power consumption of the CCS is lower when the SCS is disabled. This can be explained by the fact that the SCS was configured to maintain a temperature of 19 °C in the outside room, which is close to the maximum temperatures in the cold aisle: as the SCS does not cool down enough the outside air, by means of thermal conduction, it warms the temperature inside the racks. As a consequence, it increases the needs in term of cooling inside the cluster, leading to an higher power consumption of the CCS.

This experimental campaign has shown that idle servers have an important impact on the power consumption of cooling systems and overall racks temperature, thus it confirms the observation made in this publication [9].

B. Second experiment: Finding the optimum cooling parameters for the CCS

1) Description of the experiment: The CCS is based on the *Schneider Electric IN-ROW* cooling system. The functioning of this cooling system can be customised by changing several parameters, such as:

 ISBN: 978-1-61208-667-5

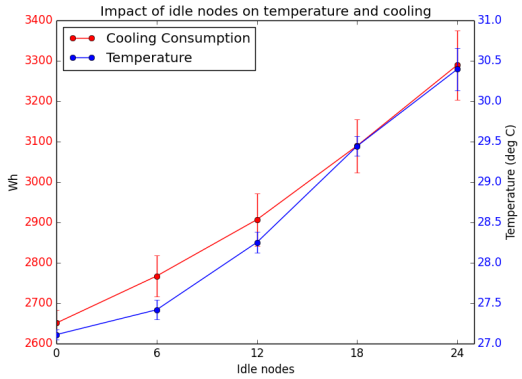

Figure 4. Central cooling consumption and average temperature in the hot

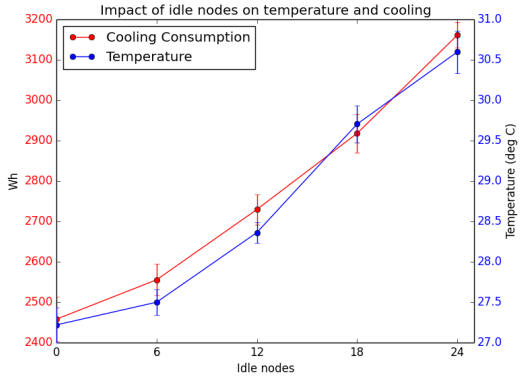

Figure 5. Central cooling consumption and average temperature in the hot aisle (SCS disabled)

- *Cooling Temperature Target*: corresponds to the temperature that the CCS tries to maintain inside the racks.
- *Air supply Temperature Target*: corresponds to the temperature of the air supplied by the CCS to the cold aisle.

In this second experiment, the impact of these two parameters is studied, in order to understand which combination of these two variables would lead to an optimal power consumption of the CCS. The experimental protocol of this second experiment is very similar to the one described in Section III-A1 as nodes where divided in two sets of servers:

- *Active servers*: these servers were executing a benchmark that generates a CPU intensive workload.
- *Turned-off servers*: these servers were electrically turned off.

We defined two configurations: an "heavy-load" configuration with 48 active servers and 0 turned-off servers, and "medium-load" configuration configuration with 24 active servers (servers with an even number) and 24 turned-off servers (servers with an odd number).

We tried several combinations of the *Cooling Temperature Target* and *Air supply Temperature Target*: the CCS's configuration was set to use these values. Each of combination has been tried in both the "heavy-load" and the "medium-load" configuration: we let the cluster in the chosen configuration for one hour, and then measured the overall power consumption of the CCS. Each experiment was repeated at least 4 times. Between each run of an experiment, we have implemented a pause step, where the CCS was configured back to its default settings (*Cooling Temperature Target* set to 23°C and *Air supply Temperature Target* set to 20°C), and all the servers were turned off until the overall temperature inside racks was under 26°C. Once the racks were cool enough, the CCS was programatically (via an SNMP API) setup to use the two cooling parameters required by the next experiment, and then the next experience would start.

2) Results: Figure 6 plots the cumulated power consumption of the CCS depending on the cooling parameters introduced in Section III-B1 in the case of a "medium-load" configuration. First, it is noticeable that as the *Air supply Temperature Target* increases, the overall power consumption of the CCS decreases: when it is set to 20°C, all the cumulated power consumption are over 2500 Wh, while they are lower than 2000 Wh when the air supply is set to be at 26°C, which corresponds to a decrease of 20%. Second, an high value for *Cooling Temperature Target* parameter seems also to have an impact on the overall power consumption of the CCS, as setting the *Cooling Temperature Target* to 31 °C leads to a power consumption that is more than 130 Wh over the consumption with lower *Cooling Temperature Target* values. We explain this observation by the fact that an important *Cooling Temperature Target* value creates hot spots in the hot aisle, which are detected by some of the many CCS's sensors and leads to an additional functioning of the CCS.

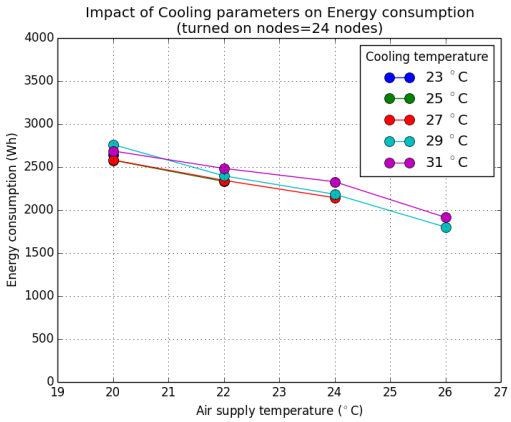

Figure 6. Central cooling consumption depending of temperature targets ("medium-load" configuration)

On the other hand, in the "heavy-load" configuration the results of the different strategies are closer. As illustrated in Figure 7, changing the values of *Air supply Temperature Target* and *Cooling Temperature Target* parameters does not seem to have an impact over the consumption of the CCS. We explain this observation by the fact that 48 active servers are an important source of heat, which requires the CCS to work continuously to maintain the target temperature, whichever cooling strategy has been used.

This second experimental campaign shows the experimental potential enabled by the the SeDuCe testbed: users can get access to thermal and power data produced during the functioning of the testbed, and they can also parameterize the configuration of the CCS. Thus, the SeDuCe testbed can help reasearchers working on the cooling of datacenters to design experiments with a fine-grained-control on experimental conditions.

 ISBN: 978-1-61208-667-5

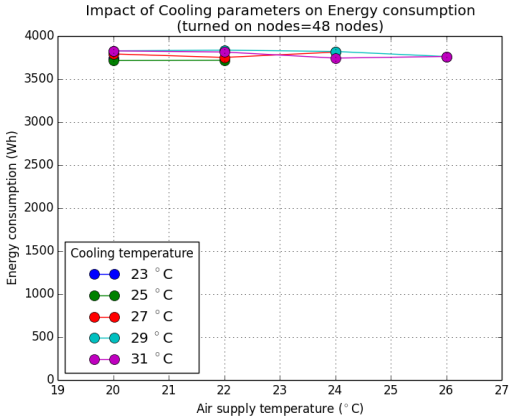

Figure 7. Central cooling consumption depending of temperature targets ("heavy-load" configuration)

IV. Future work

In this article, a first version of the SeDuCe testbed has been presented. In its current state, the SeDuCe testbed provides its users with an access to the thermal and power data generated during its functioning, and users can use this data in their work, such as studying the thermal profile of a software, or building energy efficient placement strategies. We currently see two areas of progress : one is related adding new energetic capabilities to the testbed, while the second is related to improving the precision of our thermal measurements.

Regarding the addition of new energetic capabilities to the testbed, during the next phase of building of the SeDuCe testbed (summer 2018), several solar panels will be installed on the roof of one building at *IMT Atlantique*. The energy produced by the solar panels will be used, either for supplying electricity to the testbed or for storage in batteries.Integrating these sources of renewable energy in the existing testbed will be a challenge, as we would like our users to be able to control, via an API, how the energy produced by solar panels is used, and to dynamically decide what quantity will be used by the testbed and what quantity will be stored in batteries. We think that these new capabilities will enable researchers to have access to all the elements required to experimentally study energy efficient placement strategies in datacenters.

In [11], we highlighted the fact that the accuracy of DS18B20 was not satisfactory enough in the cold aisle. Regarding the objective of improving the precision of the thermal measurements, we are currently investigating the addition of new temperature sensors based on the thermocouple approach. We have designed a prototype of an electronic card that embeds several thermocouples lines. We are working with some electronic assemblers to manufacture the cards and plan to install few of them in the cluster within September 2018.

V. Conclusion

In this article we have presented our initial work on building the SeDuCe testbed, a scientific testbed that targets research related to power and thermal management in datacenters, and which is integrated in Grid'5000 infrastructure as the new "ecotype" cluster. We have described the architecture of the testbed, which is built on buses of sensors, storage of power and thermal metrics in a time series oriented database (InfluxDB) and an user friendly web portal and a documented API. We have also detailed the components used in this first version of the SeDuCe testbed. We have illustrated the relevance of the SeDuCe testbed by performing two experimental campaigns that use the data produced by the testbed: the first experiment consisted in reproducing an existing scientific result, while the purpose of the second experiment was to illustrate the fine-grained-control that users of the SeDuCe testbed have over experimental conditions. Future work will focus on two areas: adding renewable energy capabilities to the SeDuCe testbed, and improving the precision of temperature sensors.

References

[1] F. Hermenier, X. Lorca, J.-M. Menaud, G. Muller, and J. Lawall, "Entropy: a consolidation manager for clusters," in Proceedings of the 2009 ACM SIGPLAN/SIGOPS international conference on Virtual execution environments. ACM, 2009, pp. 41–50.

[2] E. Feller, L. Rilling, and C. Morin, "Energy-aware ant colony based workload placement in clouds," in Proceedings of the 2011 IEEE/ACM 12th International Conference on Grid Computing. IEEE Computer Society, 2011, pp. 26–33.

[3] G. Von Laszewski, L. Wang, A. J. Younge, and X. He, "Power-aware scheduling of virtual machines in dvfs-enabled clusters," in Cluster Computing and Workshops, 2009. CLUSTER'09. IEEE International Conference on. IEEE, 2009, pp. 1–10.

[4] C.-M. Wu, R.-S. Chang, and H.-Y. Chan, "A green energy-efficient scheduling algorithm using the dvfs technique for cloud datacenters," Future Generation Computer Systems, vol. 37, 2014, pp. 141–147.

[5] E. Pinheiro, R. Bianchini, E. V. Carrera, and T. Heath, "Dynamic cluster reconfiguration for power and performance," in Compilers and operating systems for low power. Springer, 2003, pp. 75–93.

[6] J. S. Chase, D. C. Anderson, P. N. Thakar, A. M. Vahdat, and R. P. Doyle, "Managing energy and server resources in hosting centers," ACM SIGOPS operating systems review, vol. 35, no. 5, 2001, pp. 103–116.

[7] F. Clouet et al., "A unified monitoring framework for energy consumption and network traffic," in TRIDENTCOM-International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities, 2015, p. 10.

[8] H. Sun, P. Stolf, and J.-M. Pierson, "Spatio-temporal thermal-aware scheduling for homogeneous high-performance computing datacenters," Future Generation Computer Systems, vol. 71, 2017, pp. 157–170.

[9] J. D. Moore, J. S. Chase, P. Ranganathan, and R. K. Sharma, "Making scheduling" cool": Temperature-aware workload placement in data centers." in USENIX annual technical conference, General Track, 2005, pp. 61–75.

[10] R. Bolze et al., "Grid'5000: A large scale and highly reconfigurable experimental grid testbed," The International Journal of High Performance Computing Applications, vol. 20, no. 4, 2006, pp. 481–494.

[11] J. Pastor and J.-M. Menaud, "Seduce: Toward a testbed for research on thermal and power management in datacenters," in E2DC, vol. 18, 2018, pp. 1–7.

[12] Flukso website. [Online]. Available: https://www.flukso.net/about

[13] Socomec g50 website. [Online]. Available: https://www.socomec.fr/gamme-interfaces-communication_fr.html?product=/diris-g_fr.html

[14] Technical documentation of the ds18b20 sensor. [Online]. Available: https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf

[15] Website of the platformio project. [Online]. Available: https://platformio.org/

[16] Website of the seduce portal. [Online]. Available: https://seduce.fr

[17] Documented seduce api. [Online]. Available: https://api.seduce.fr/apidocs

[18] I. InfluxData. Website of the influxdb project. [Online]. Available: https://www.influxdata.com/

[19] A. Ronacher. Website of the flask project. [Online]. Available: http://flask.pocoo.org/

[20] Website of the swagger project. [Online]. Available: https://swagger.io/

 ISBN: 978-1-61208-667-5