CINERGY: Reasoning over the Worst Case Power Consumption of Cloud Virtual Machines

Pierre Jacquet, Camille Coti, Marcos Dias de Assunção, Romain Rouvoy

CINERGY: Reasoning over the Worst Case Power Consumption of Cloud Virtual Machines

Pierre JACQUET,* *Camille* COTI,* *Marcos* DIAS DE ASSUNÇÃO,* *Romain* ROUVOY†
[1] École de Technologie Supérieure, Université du Québec, Montréal, Canada
[2] Univ. Lille, Inria, CNRS, UMR 9189 CRIStAL, France
{pierre.jacquet, camille.coti, marcos.dias-de-assuncao}@etsmtl.ca,
romain.rouvoy@univ-lille.fr

***Abstract*—Energy consumption has become a critical concern in *Information and Communication Technologies* (ICT), pressing for more accurate measurements. While the power consumption of physical servers can be physically monitored, organizations are increasingly adopting virtual environments, such as cloud computing, rendering physical measurements impractical in operational contexts. The state-of-the-art approaches to estimating this "virtual" consumption mostly consist of assigning server power consumption shares among hosted processes, guided by various system metrics. Unfortunately, such a bottom-up approach is highly sensitive in a multi-tenant environment, thus failing to report stable measurements to stakeholders. For example, the same activity performed by one *Virtual Machine* (VM) may lead to different power consumption traces, depending on the activity of the co-hosted VMs.**

As cloud customers have only control over their provisioned virtual resources, we propose a new method to model the power consumption of their virtual appliances, enabling context-agnostic tracking of their environmental impact. This framework, called CINERGY, is designed to be more predictable than the state-of-the-art power models, while still exposing the gains from consolidation. We evaluate its accuracy against ground-truth measurements, often lacking in the literature. We show that CINERGY is deterministic and accurate, with an average error of 6.6%.

I. INTRODUCTION

The electric consumption of hyperscale *Data Center*s (DCs) remains a key concern, due to its continuous increase [1], and despite significant improvements in infrastructure, particularly cooling systems [2]. To effectively reduce the environmental impact of these infrastructures, optimizations must now be complemented by software optimizations. While environmental considerations have driven changes in software design methodologies [3] and DC design, these efforts typically address only a single perimeter: either the customer's perspective (e.g., software optimization) or the cloud provider's perspective (e.g., hardware optimization).

From the customer's perspective, monitoring the power consumption on shared computing infrastructure, such as public cloud computing infrastructures, remains a challenge. In this setting, there is no *Running Average Power Limit* (RAPL) interface for virtualized resources to easily track their usage. From the cloud provider's perspective, efficiency gains are mostly promoted through indicators, such as the *Power Usage Efficiency* (PUE), *Water Usage Effectiveness* (WUE), and others. While these metrics can be used to compare providers based on a given standard (e.g., ISO/IEC 30134-2:2016 for the PUE), their scope ends at the server level. Nevertheless, how can the power saved by resource virtualization be properly advertised?

Researchers and practitioners have developed various process power meters to tackle this challenge [4], [5], [6]. The primary purpose of a software-defined power meter is to estimate the share of a server's power consumption attributable to hosted processes. These tools generally operate in a similar way. First, they capture the server's global power consumption. Then, they build a power model to estimate how much each process contributes to this total, based on carefully selected usage metrics. While this approach is intuitive and allows for estimating software's impact on current power consumption, it faces challenges, notably in shared environments [7]:

- First, software power consumption is highly contextual. This variability is problematic in shared environments, where cloud providers may report different readings for the same *Virtual Machine* (VM) activity.
- Second, there is no definitive ground truth to evaluate the model, as individual software power consumption is not physically measurable.

To bridge the gap between customers' and cloud providers' optimization efforts, we advocate for a common framework that effectively captures and harmonizes both efficiencies. This paper introduces a power framework, called CINERGY, to expose both efficiency indicators to customers. Rather than using a *bottom-up* design to assign a share of the current server's power consumption, we propose a *top-down* approach that estimates the power the VM would consume if operated in isolation. Interestingly, this design delivers the *Worst-Case Power Consumption* (WCPC) for a given software service, when it operates at the lowest CPU utilization rate. Furthermore, this configuration delivers more deterministic estimations, as the VM activity is the only one responsible for any power variation. Finally, our power framework introduces a new metric to compare this isolated power consumption and the runtime one, hence highlighting the savings obtained from virtualization and consolidation.

II. Related Work

Current generic software-defined power meters typically use a bottom-up approach to distribute a server's total energy consumption across the software stack based on various metrics. An effective allocation metric strongly correlates with the server's energy consumption and can be measured on a per-process basis. Examples of such metrics include CPU time, which is utilized by tools, like SCAPHANDRE [4] and JOULARJX [8]. Other tools, such as SMARTWATTS [6], [9], which is integrated into the POWERAPI framework [10], [11], employ performance counters. Finally, KEPLER proposes a hybrid approach that combines both methods [5].

All these bottom-up approaches deliver insights into hotspots in a server's current power consumption. However, they face certain limitations as they are non-deterministic. For example, power consumption is non-linear; a process that consistently uses 20% of the CPU will affect a server's power consumption differently depending on its current load. The impact will drastically change if a server is already loaded at 0% compared to one that is at 80% utilization.

III. Principles

In this section, we describe the principles of our framework. Specifically, we aim for top-down power estimations, where the power consumption of an isolated system should be deducted from the software activity. We also aim to evaluate energy efficiency gains from virtualization.

A. Building a Top-Down Power Model

Measuring the power consumption of a given software hosted in isolation can limit two potential biases. The first one is caused by noisy neighbors—i.e., threads that noticeably impact the system's power consumption. The second one is the kernel activity variation, as it creates a comparable baseline by default setting the system—i.e., hardware and operating system. We aim to model the power measurements in this default setting.

As cloud VMs tend to be small [12], it is essential to model their isolated power consumption on larger servers, particularly during low usage phases. We focus our power consumption modeling on a specific usage range to address this challenge. Specifically, when modeling the usage of a VM of n cores, we select the range as follows: $[0; \frac{n}{\text{host cores}} \times \alpha]$, α being a safety factor to ensure an accurate trend in the curve. In this study, we determined α based on the number of cores in physical servers, aiming to limit the range to low CPU usage phases (up to 40%). For instance, $\alpha = 3$ is appropriate for VMs with 4 cores on hosts with 32 cores.

Our models are constructed through simple polynomial regressions from this usage phase's CPU usage and power readings.

B. Workload consolidation efficiency

While top-down models allow us to estimate software's isolated power consumption, our objective is also to incorporate the energy efficiency benefits derived from workload consolidation. The isolated VM power consumption represents a worst-case scenario, one that is rarely achieved in practice, as servers are typically shared among multiple instances. This sharing increases resource utilization, improving energy efficiency due to the server's logarithmic power consumption profile. By making this behavior transparent to clients, we aim to emphasize the significant energy savings enabled by virtualization and orchestration techniques.

$$\text{CINERGY ratio}_{VM} = \frac{\text{bottom-up model}_{\text{VM}}}{\text{top-down model}_{\text{VM}}} \quad (1)$$

We introduce a consolidation factor known as the CINERGY ratio[1] (Equation 1). It is derived by dividing the runtime power consumption estimation (bottom-up) by the isolated power consumption estimation (top-down). Since isolated power consumption is typically higher, this ratio usually falls between 0 and 1.

For cloud customers, the CINERGY ratio reflects the efficiency gains that cloud providers achieve by reducing individual power consumption.

For instance, a CINERGY ratio of 0.4 indicates that only 40% of the power expected in an isolated scenario is consumed, representing a 60% power saving. Thus, a lower CINERGY ratio highlights greater energy efficiency than the standalone setting. The CINERGY ratio notably promotes the savings resulting from server consolidation, possibly even in oversubscribed environments that keep being seen negatively despite their potential to make cloud computing greener [13], [14], [15].

IV. Evaluation

Cloud data centers operate various processors from different vendors and generations, each possessing unique energy characteristics. To assess the broad applicability of our approach, we evaluated it across multiple processors, paying careful attention to covering a wide diversity of architectures.

The evaluation process comprises three stages. First, we developed power models in a controlled environment for each type of hardware (cf. Subsection IV-A). Next, we measured the activity and corresponding power consumption of a VM on a dedicated server to establish a ground truth reference (cf. Subsection IV-B). Finally, we assessed our model's ability to replicate this ground truth measurement in a cloud-like scenario, where the same VM ran alongside other processes on its host (cf. Subsection IV-C).

A. Building Power Models

We created power models for each server in a controlled environment. Workloads of varying intensity levels were created using `stress-ng`, a widely-used benchmarking tool, with its default stressor settings. We, then, gradually increased the host load by running threads that used 50% of a CPU core.

The collected energy consumption data includes RAPL measurements, including power usage for each CPU package.

[1]The name is derived from "Cirrus," a type of cloud, and "Synergy," which denotes an interaction that produces a whole greater than the sum of its parts.

We aggregated the reported readings to compute the total power consumption of multi-socket servers. We, then, applied a polynomial regression of order 5 to convert the CPU load into its corresponding power consumption (in Watts).

B. Ground Truth Measurements

We launch a VM that runs a specific application under various CPU usage levels while monitoring the power consumption measured by RAPL domains. This setup serves as the baseline since we can fully attribute the power consumption recorded by RAPL to the system's single VM.

Our evaluation executed on a VM with four vCPUs, while running the social network application from the *DeathStarBench* (DSB) [16]. The DSB features a microservices-based architecture, incorporating various components such as load balancers, proxies, message queues, databases, caches, and services. We chose this architecture because of its complexity and diverse components, which make it a strong candidate to benefit from energy monitoring capabilities. Additionally, the DSB includes a workload generator based on `wrk2`, enabling us to simulate realistic usage patterns during our evaluation.

The VM's activity was driven by the wrk2 workload generator, simulating several load phases: *low* (100 rps), *medium* (2,000 rps), *high* (5,000 rps), and *very high* (10,000 rps). We selected these load levels based on the CPU utilization they generated. Under the low-load scenario, the VM was primarily idle, while the highest load completely engaged the CPU.

The CPU load of the VM and the corresponding CPU activity observed from the host were continuously monitored over several hours to ensure accurate and representative readings.

C. Models Evaluation in a Cloud-like Environment

Measurements of VM activity were conducted in a cloud-like environment, where a VM ran alongside additional processes to simulate background noise. The same VM workload, generated using `wrk2` under identical parameters, was repeated during these tests. Throughout the experiments, the VM activity was continuously monitored while background processes were gradually introduced, varying the host's load from mostly idle to fully saturated.

To evaluate the accuracy of our model under different conditions, we applied various load patterns (*low*, *medium*, *high*, and *very high*) to the VM in a cyclical sequence. This approach enabled us to evaluate the model's power estimations across a range of VM workloads and host load combinations.

Figure 1 depicts our WCPC projections across the studied architectures. The ground truth measurements, discussed in Subsection IV-B, are depicted in grey, while our power model's predictions, which were trained using the generated data discussed in Subsection IV-A and the metrics collected from this cloud-like environment, are plotted in orange. While the VM activity leads to different host CPU usage depending on its size, results are displayed based on VM CPU activity (x-axis) to normalize the data.

The static power consumption was allocated based on resource distribution in a non-oversubscribed setting for the ground truth and the power model predictions.

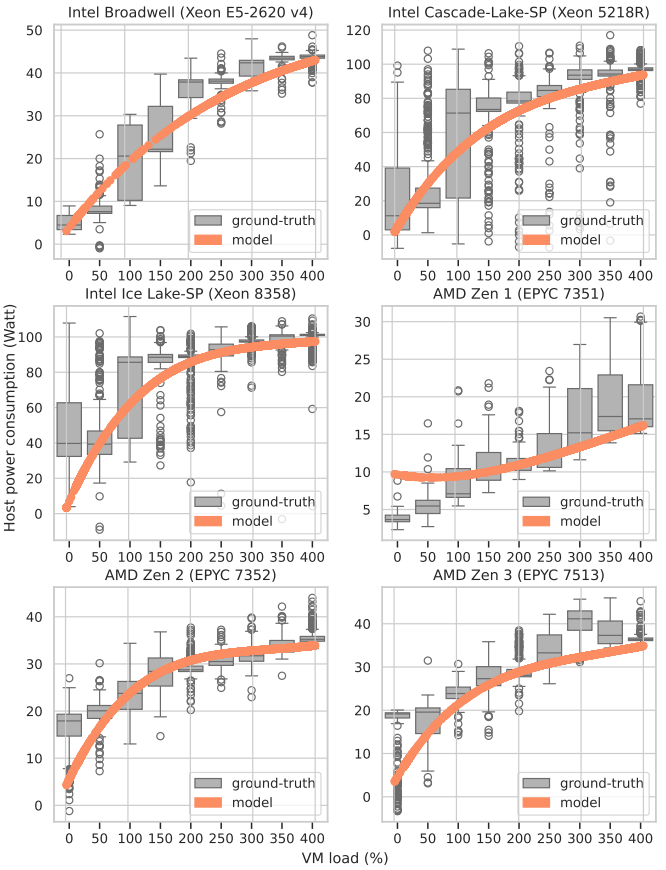

Fig. 1. CINERGY Top-down power models on various hardware

Despite some inherent variability in power consumption measurements [17], our power models report high accuracy in predicting power usage across different architectures. Overall, the models perform well across all tested architectures. The average MAPE for all evaluated processors is approximately 6.6%, indicating a low mean relative error in the predictions. This supports the feasibility of accurate isolated power estimation in diverse hardware environments.

D. Discussion & Limitations

Our evaluation was conducted on production-class servers, which may lead to an impression that the power consumption of a VM is significant. This approach was chosen to assess the accuracy of our models on representative hardware.

The selection of an appropriate server baseline is context-dependent and was not explored in this paper. Instead, our research focuses on the energy efficiency gains made possible by these large servers. This gain is computed using the CINERGY ratio, as highlighted in the next subsection.

E. Applying the CINERGY ratio

To illustrate the usage of the CINERGY ratio, we conducted an experiment where a single VM power consumption was measured under different host CPU loads. We computed power consumption estimates from our top-down model and a bottom-up-like model using Equation 2.

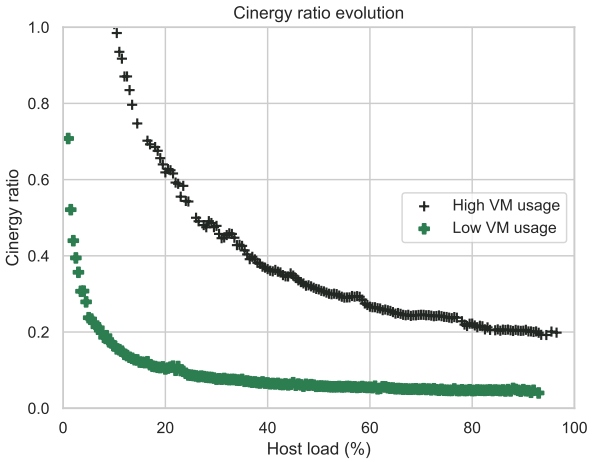

Fig. 2. CINERGY ratio evolution

$$\mathcal{P}_{\text{VM}}^{\text{bottom-up}} = \frac{\text{CPU}_{\text{VM}}^{\text{usage}}}{\text{CPU}_{\text{total}}^{\text{usage}}} \times \mathcal{P}_{\text{RAPL}} \tag{2}$$

We analyzed the CINERGY ratio variation for this VM as other processes introduced additional host load. In Figure 2, we distinguished two VM usage phases: *low usage* (below 10%) and *high usage* (above 85%). The results show that energy efficiency gains become evident even at low levels of host utilization. At a 20% host load, the CINERGY ratio demonstrates significant energy efficiency gains for both VMs. The high-usage VM operates at 60% of its baseline ($r = 0.6$), while the low-usage VM sees even greater benefits, consuming 10% ($r = 0.1$) of its initial power. As the host load approaches full capacity, the high-usage VM's consumption decreases further to 20% of its isolated power consumption ($r = 0.2$), while the low-usage VM trends toward 0.

We believe that the power readings delivered by our top-down framework enable cloud developers to track their environmental impact clearly and deterministically (*client scope*). At the same time, the CINERGY ratio offers a metric to share the energy efficiency savings achieved from consolidation (*provider scope*). To maintain the determinism of our approach when applied to worst-case power consumption readings, the average CINERGY ratio across the cluster should be applied.

V. CONCLUSION

This paper presented a novel framework for estimating process power consumption, referred to as CINERGY. Unlike existing bottom-up approaches which allocate the power consumption of live servers across processes using various models, we propose a top-down method that estimates the *Worst Case Power Consumption* (WCPC) of a process. In a cloud context, this approach offers the advantage of better determinism, as the estimated power consumption of virtual resources relies solely on a single actor. It also allows for an accurate evaluation of process power models, as ground-truth data can be easily obtained. This approach[2] demonstrated high accuracy through evaluation among a diverse set of cloud hardware, achieving an average error of 6.6%. Moreover, we introduced a new metric, called the CINERGY ratio, to assess energy efficiency gains from consolidation.

[2]Our repository: https://github.com/jacquetpi/cinergy-models

VI. ACKNOWLEDGEMENT

This work was supported by Mitacs/OVHcloud under project IT42864. It received partial support from the "*FrugalCloud*" project by Inria and from the French government through the *Agence Nationale de la Recherche* (ANR) under the France 2030 program, with CARECLOUD (ANR-23-PECL-0003) and DISTILLER (ANR-21-CE25-0022) grants.

REFERENCES

[1] International Energy Agency, "Data centres and data transmission networks," 2021. Available at https://www.iea.org/energy-system/buildings/data-centres-and-data-transmission-networks#programmes.

[2] European Commission Joint Research Centre, "The eu code of conduct for data centres – towards more innovative, sustainable and secure data centre facilitie)," 2023. Available at https://joint-research-centre.ec.europa.eu/jrc-news-and-updates/eu-code-conduct-data-centres-towards-more-innovative-sustainable-and-secure-data-centre-facilities-2023-09-05_en.

[3] S. Georgiou, S. Rizou, and D. Spinellis, "Software development lifecycle for energy efficiency: Techniques and tools," *ACM Comput. Surv.*, vol. 52, Aug. 2019.

[4] Scaphandre, "Scaphandre documentation," 2024. Available at https://https://hubblo-org.github.io/scaphandre-documentation/.

[5] Kepler, "Kubernetes efficient power level exporter (kepler)," 2024. Available at https://sustainable-computing.io/.

[6] G. Fieni, R. Rouvoy, and L. Seinturier, "Smartwatts: Self-calibrating software-defined power meter for containers," in *2020 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID)*, pp. 479–488, IEEE, 2020.

[7] E. Cadorel and D. Saingre, "A protocol to assess the accuracy of process-level power models," in *2024 IEEE International Conference on Cluster Computing (CLUSTER)*, 2024.

[8] A. Noureddine, "Powerjoular and joularjx: Multi-platform software power monitoring tools," in *2022 18th International Conference on Intelligent Environments (IE)*, pp. 1–4, 2022.

[9] G. Fieni, R. Rouvoy, and L. Seinturier, "Selfwatts: On-the-fly selection of performance events to optimize software-defined power meters," in *CCGRID*, pp. 324–333, IEEE, 2021.

[10] M. Colmant, R. Rouvoy, M. Kurpicz, A. Sobe, P. Felber, and L. Seinturier, "The next 700 cpu power models," *Journal of Systems and Software*, vol. 144, pp. 382–396, 2018.

[11] G. Fieni, D. R. Acero, P. Rust, and R. Rouvoy, "Powerapi: A python framework for building software-defined power meters," *Journal of Open Source Software*, vol. 9, no. 98, p. 6670, 2024.

[12] P. Jacquet, T. Ledoux, and R. Rouvoy, "Cloudfactory: An open toolkit to generate production-like workloads for cloud infrastructures," in *11th IEEE International Conference on Cloud Engineering*, IC2E'23, 2023.

[13] P. Jacquet, T. Ledoux, and R. Rouvoy, "Sweetspotvm: Oversubscribing CPU without sacrificing VM performance," in *CCGrid*, pp. 148–157, IEEE, 2024.

[14] P. Jacquet, T. Ledoux, and R. Rouvoy, "SCROOGEVM: boosting cloud resource utilization with dynamic oversubscription," *IEEE Trans. Sustain. Comput.*, vol. 9, no. 5, pp. 754–765, 2024.

[15] P. Jacquet, T. Ledoux, and R. Rouvoy, "Slackvm: Packing virtual machines in oversubscribed cloud infrastructures," in *CLUSTER*, pp. 190–201, IEEE, 2024.

[16] Y. Gan, Y. Zhang, D. Cheng, A. Shetty, P. Rathi, N. Katarki, A. Bruno, J. Hu, B. Ritchken, B. Jackson, K. Hu, M. Pancholi, Y. He, B. Clancy, C. Colen, F. Wen, C. Leung, S. Wang, L. Zaruvinsky, M. Espinosa, R. Lin, Z. Liu, J. Padilla, and C. Delimitrou, "An Open-Source Benchmark Suite for Microservices and Their Hardware-Software Implications for Cloud & Edge Systems," in *ASPLOS*, pp. 3–18, ACM, 2019.

[17] Z. Ournani, M. C. Belgaid, R. Rouvoy, P. Rust, J. Penhoat, and L. Seinturier, "Taming energy consumption variations in systems benchmarking," in *Proceedings of the ACM/SPEC International Conference on Performance Engineering*, ICPE '20, p. 36–47, ACM, 2020.