Composability of Cloud Accelerators in Virtual World Simulations

Dionysios Diamantopoulos, Burkhard Ringlein, Beat Weiss, Mark Lantz, François Abel
IBM Research Europe, Säumerstrasse 4, 8803, Rüschlikon, Switzerland
{*did, ngl, wei, mla, fab*}*@zurich.ibm.com*

***Abstract*—Immersive and interactive experiences offered by virtual world simulations (VWS) are becoming increasingly indispensable in today's digital era, particularly in domains such as gaming, engineering, and education. Cloud computing provides a compelling solution for VWS, with its scalability, cost-effectiveness, and accessibility, combined with efficient processing capabilities of hardware accelerators such as FPGAs and GPUs. This paper presents APEIRON, an in-progress distributed library built on the Ray framework, that aims to facilitate the effective composition of hardware accelerators on public clouds and emerging disaggregated platforms. Additionally, this work introduces a novel composability metric, analyzing the intricate relationship between resource utilization and workload performance. This study focuses on theoretical composability analyses, laying groundwork for optimized utilization of hardware resources to enhance efficiency, thereby potentially reducing operational costs.**

***Index Terms*—computer vision, cloud, accelerators, GPUs, FPGAs, disaggregated systems, composable systems**

I. Introduction

Advances in computer vision (CV), powered by Artificial Intelligence (AI) and machine learning techniques, have revolutionized our interaction with the physical environment. The technology underpinning CV now enables us to process visual data with an unprecedented efficiency and speed [1]. CV has become a pillar in numerous sectors including education, food industry, civil engineering, arts, medicine, agriculture, and robotics, among others, serving to automate and enhance human vision.

CV is also pivotal in virtual world simulations (VWS), primarily developed by the movie and game industry for complex storytelling. Apart from entertainment, these photorealistic virtual environments serve as valuable tools for research by simulating hard-to-replicate real-world conditions, such as extreme weather phenomena. They enable stress-testing of vision algorithms under hazardous conditions or conducting costly robotics training. Anticipating the surge in virtual world applications, including digital twins [2], augmented reality [3], and the yet-to-be-defined "Metaverse" [4], the demand for efficient processing of CV analytics is set to increase significantly. A VWS example is illustrated in Fig. 1.

As CV processing traditionally relies on GPUs, the future demands of VWS applications necessitate innovative solutions for performance, efficiency, and flexibility. We're currently developing APEIRON — a distributed library, employing the Ray framework [5], that aims to meet these requirements. APEIRON is designed to provide a unified interface for VWS, facilitating both task-parallel computations using Ray's dynamic execution engine. Our approach with APEIRON is to utilize hardware accelerators on the public cloud for deployment, eliminating the need for specialized on-prem hardware. This library also advocates a disaggregated and composable approach, dynamically combining resource pools to tailor to VWS applications. Preliminary tests reveal promising results, with APEIRON showing potential for better scaling performance compared to existing specialized systems. It's expected that, with a disaggregated platform, APEIRON might deliver equivalent performance or energy efficiency using fewer resources. In our ongoing work, we anticipate the following contributions:

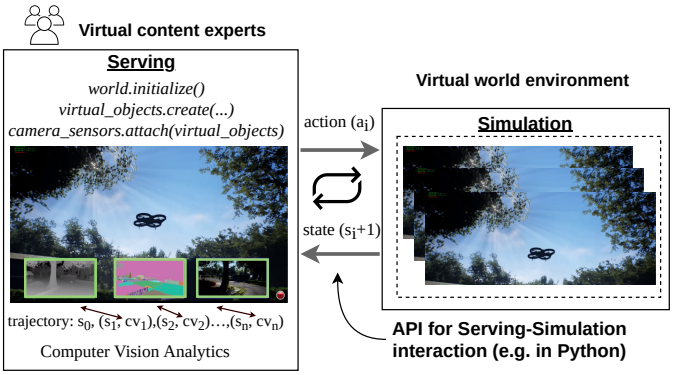

Fig. 1: Example of a Virtual World Simulation System (AirSim).

- We conduct the first study of Ray-based clusters with CPUs, GPUs, PCIe-attached Field Programmable Gate Arrays (FPGAs), and network-attached FPGAs in publicly available clouds. Our study covers both the architecture and the programming model.
- We introduce a novel metric, *composability*, to quantify the suitability of different cloud resources for VWS workloads.

II. Background and Motivation

A. Virtual World Simulation

In a VWS system, as depicted in Fig. 1, experts interact with a computing environment simulating a *virtual world* in distinct states over time. The world is initialized with a state, and virtual objects are added using actions. The goal is to provide a sequence of states similar to those in the real world, given physical rules and an initial state. For computer vision analytics, experts create a mapping from the environment's state to an action that controls the simulation. Unlike conventional computer simulations, CV analytics for virtual worlds demands a synergistic approach towards serving and simulation within a single application, subject to stringent latency constraints. Current methods often employ multiple specialized frameworks, leading to restrictive data flow and limited throughput [6]. Currently, we are developing a distributed library to address these challenges. Our work-in-progress aims to enhance both serving and simulation functionalities within a singular application while ensuring optimal data flow and high throughput.

B. Cloud Infrastructure Evolution

1) Hyper-converged Infrastructure: Cloud computing, with its cost advantages, scalability, and usage-based pricing, appeals to traditional Fortune 500 companies. To accommodate growing workloads, infrastructures are either enhanced (scale-up) or expanded (scale-out). Many cloud service providers use hyper-converged servers with accelerators like GPUs and FPGAs for performance and flexibility [7]. However, the requirements of modern data centers, like the resource allocation for the pay-as-you-go pricing model, have challenged the heterogeneity, simplicity, and agility of the hyper-converged architecture.

This is the accepted version of the article published by IEEE: 10.1109/CLOUD60044.2023.00038

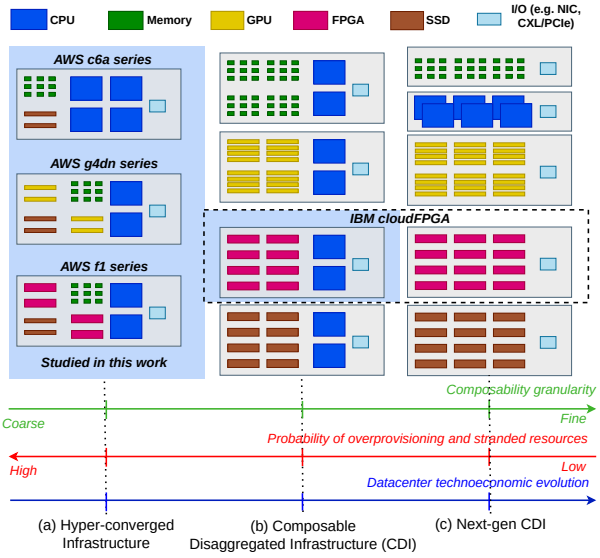

Fig. 2: Positioning of studied systems with respect to the server's techno-economic evolution that gives birth to CDIs.

2) Composable Disaggregated Infrastructure: The traditional IT infrastructure has seen recent evolution with the rise of composable disaggregated infrastructure (CDI) [8], [9], as shown in Fig. 2. CDI amalgamates compute, storage, and networking elements to create a software-defined cloud experience. This involves disaggregating servers and their resources - CPUs, GPUs, FPGAs, SSDs - into a vast, connected network pool. CDI's promise of flexible, on-demand resource allocation makes it a coveted goal for data centers [10]. To fully realize its potential, challenges need to be addressed [11]. Ensuring servers run optimally without interference calls for treating all types of devices as first-class citizens. Solutions proposed include Data Processing Units (DPUs) for workloads typically handled by CPUs, and utilizing varied interconnection mediums, like PCIe, Ethernet, and InfiniBand. IBM's[‡] cloudFPGA (cF) research platform [12] serves as a pioneering platform for studying disaggregated computing.

3) Composability on cloud: Cloud composability refers to software-managed computing, storage, and networking resources, regardless of physical location. This renders data center resources as accessible as cloud services, like AWS instances that can be provisioned through the AWS Console. Users can select different resources via AWS web interface, supporting distributed applications with MPI or frameworks like MapReduce or SPARK.

Yet, software composability faces issues like fixed resources within an instance, increased latency, and insufficient focus on hardware accelerators. These issues curb fine-grained system scalability, as depicted in Fig. 3. This study addresses these inefficiencies by creating a library for software-composed cloud resources, focusing on performance, energy efficiency, cost, and resource utilization. We'll examine various cloud environments and hardware accelerators to advance research on cloud composable systems, focusing on selected deployments including AWS and cF clusters, as shown in Fig. 2.

III. APEIRON Library and Composability

APEIRON, based on Ray [5], is a novel library designed and implemented to streamline cloud resource usage and limit overprovisioning. Initially built for distributed machine learning, Ray has been revamped to address high per-task overhead and the lack of actor abstraction, making it more efficient than similar systems like Dask [13].

In its design, APEIRON uses Ray's resource extensions and OpenCV for CPU and GPU operations. It brings in workers

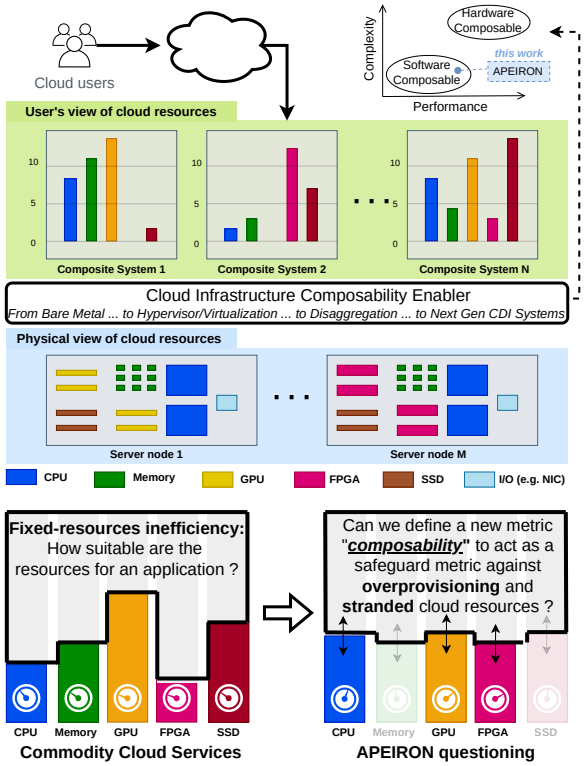

Fig. 3: Composability inefficiency as a study vehicle for efficient use of resources in software-defined composable systems.

that delegate computations to both PCIe and network-attached FPGA devices, employing Vitis Vision and cFp_Zoo[1] for compatibility with IBM's cF. This setup, outlined in a YAML description, allows users to compose systems as needed. On the implementation front, APEIRON uses a *Serving* system based on Ray with custom extensions, a *VWS system* powered by the open-source CARLA driving simulator [14], and a Python API for integrating the two systems.

Composability: The complexity of distributed systems such as VWS necessitates the evaluation of performance through diverse metrics. A fundamental part of our analysis involved understanding how Ray can enhance VWS performance by effectively distributing workloads across cloud accelerators. To this end, we utilized Docker containers for isolating the application and reducing external noise, and then we correlated the performance of the simulator with system metrics.

An important observation, as illustrated in Fig. 4, reveals a strong positive correlation between the Function Call Rate (FCR) and the Frames Per Second (FPS) generated through computer vision functions. This correlation holds true across both AWS and cloudFPGA. Specifically, the Cumulative Distribution Function (CDF) from 100 runs indicates that higher FCR values, as gauged by Ray's profiler, correspond to a greater number of processed frames. This practically translates to the simulator being able to make the virtual world more closely mirror real-world dynamics and states.

When correlating FCR with system metrics, AWS instances showed a high positive correlation with CPU and Cache (L3) utilization but a neutral correlation with network utilization. In contrast, cloudFPGA exhibited less dependence on CPU involvement. These findings are important as they reveal resource utilization patterns that need to be considered in complex cloud-based accelerated computing platforms.

Building on these insights, we developed a novel empirical metric — composability, that offers a measure of the efficacy

[1]https://github.com/cloudFPGA/cFp_Zoo

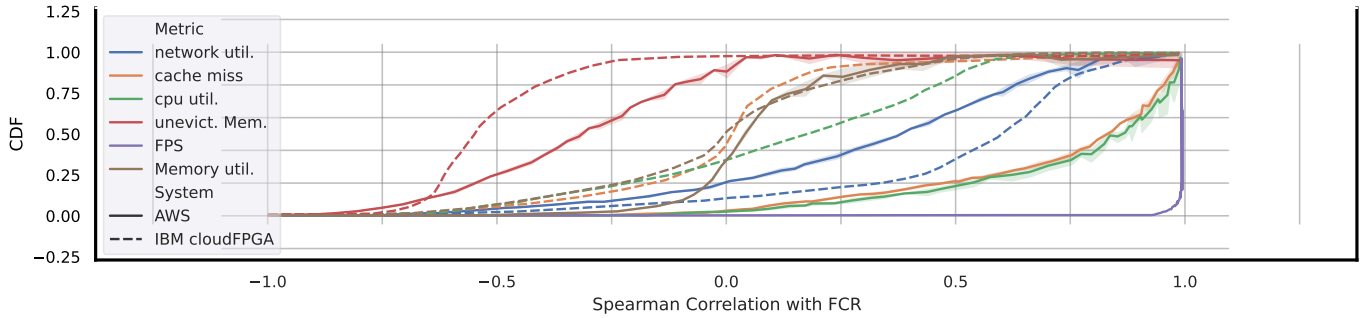

Fig. 4: The Function Call Rate (FCR) of the virtual world simulator in relation to the CDF of performance metrics from 100 runs. The AWS metrics were derived from AWS CPU (c6a.4xlarge), GPU (g4dn.4xlarge), and FPGA (f1.4xlarge) instances.

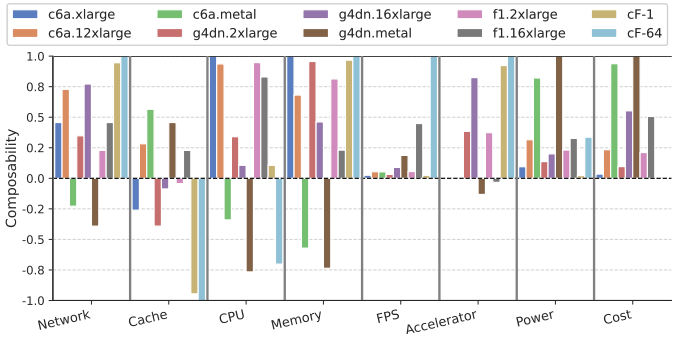

Fig. 5: Composability per resource across different instances.

of specific resource selection within an instance for a particular workload. Composability goes beyond generic metrics, such as VM performance and energy efficiency, that often overlook the nuanced dynamics of resource utilization. Mathematically, it's expressed as an average of the Spearman correlation between each resource utilization and the performance of the workload, as defined in Equation 1:

$$C_i = \frac{1}{n}\sum_{k=1}^{n}\frac{1}{m}\sum_{j=1}^{m}\frac{1}{s}\sum_{l=1}^{s} corr(R_{i,k,j,l}, P_{i,k,j,l}) \qquad (1)$$

In this equation, C_i represents the composability for an instance type i, and n, m, and s denote the number of application scenarios (CV kernels), resources, and image sizes respectively. $R_{i,k,j,l}$ and $P_{i,k,j,l}$ are the resource utilization and performance of instance i under scenario (CV) k for image size l. The performance is measured in Frames Per Second (FPS), while the resource utilization derives from system/OS/Docker performance counters for every resource type.

Composability, being an average of Spearman correlations, ranges between -1 to 1, with higher values indicating higher suitability of an instance for a specific workload. This metric allows us to expand our analysis to include instance-specific metrics such as cost and power consumption, which indirectly correlate with performance. Figure 5 corroborates the widely anticipated observation that cost and power are inherently intertwined - typically, more powerful instances are expectedly more expensive and power-consuming. This reflects the absence of negative composability for these particular metrics. All the other resources have different composability values, due to their varying degrees of interdependence and potential for interference in a particular VM setup.

Composability, therefore, holds significant potential as a metric to inform instance selection and workload scheduling strategies, as well as shed light on potential over-provisioning of resources. Its consideration of various factors and intricate interdependencies within a VM setup can help optimize trade-offs between performance, cost, and resource utilization.

IV. Discussion & Conclusions

APEIRON is a composability library extending Ray's application layer to efficiently handle VWS computations on diverse infrastructure, including disaggregated systems like IBM's cloudFPGA. A key focus of ongoing research is the comprehensive exploration of various accelerator types. Built on the dynamic computation model of Ray, APEIRON is well-suited for fine-grained simulation workloads. One notable innovation of this work-in-progress is the introduction of composability as an empirical metric. This offers nuanced insights into the relationships between resource utilization and application performance, crucial for optimized resource allocation. However, large-scale analysis and experimentation with this metric remain ongoing. As part of future work, we anticipate the application of composability to CDI architectures, which could lead to substantial improvements in performance, along with energy and cost efficiency.

Notices

Acknowledgment: This work is partially funded by the EU Horizon 2020 Programme under grant agreement No 957269 (EVEREST)

References

[1] J. Chai *et al.*, "Deep learning in computer vision: A critical review of emerging techniques and application scenarios," *Machine Learning with Applications*, vol. 6, p. 100 134, 2021.
[2] J. Leng *et al.*, "Digital twins-based smart manufacturing system design in industry 4.0: A review," *Journal of manufacturing systems*, 2021.
[3] J. Xiong *et al.*, "Augmented reality and virtual reality displays: Emerging technologies and future perspectives," *Light: Science & Applications*, 2021.
[4] L.-H. Lee *et al.*, *All one needs to know about metaverse: A complete survey on technological singularity, virtual ecosystem, and research agenda*, 2021.
[5] P. Moritz *et al.*, "Ray: A distributed framework for emerging AI applications," in *OSDI*, Carlsbad, CA: USENIX Association, 2018.
[6] R. Cheng *et al.*, "Are we ready for metaverse? a measurement study of social virtual reality platforms," ser. IMC, Nice, France: ACM, 2022.
[7] H. Yu *et al.*, "Automatic virtualization of accelerators," in *Proceedings of the Workshop on Hot Topics in Operating Systems*, ser. HotOS '19, Bertinoro, Italy: Association for Computing Machinery, 2019.
[8] Y. Shan *et al.*, "Towards a fully disaggregated and programmable data center," ser. APSys '22, New York, NY, USA: ACM, 2022.
[9] Q. Zhang *et al.*, "Optimizing data-intensive systems in disaggregated data centers with teleport," ser. SIGMOD '22, Philadelphia, PA, USA: Association for Computing Machinery, 2022.
[10] E. S. Ashish Nadkarni, "Worldwide composable infrastructure forecast, 2021–2025," IDC, Tech. Rep. US47689621, 2021.
[11] E. Ong. "Scale-out data centers: The best is yet to come." (2022).
[12] "Open-source field programmable gate arrays for the cloud." (2022), [Online]. Available: https://github.com/cloudfpga.
[13] M. Rocklin, "Dask: Parallel computation with blocked algorithms and task scheduling," in *14th python in science conference*, Citeseer, 2015.
[14] A. Dosovitskiy *et al.*, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017.