RobotPerf: An Open-Source, Vendor-Agnostic, Benchmarking Suite for Evaluating Robotics Computing System Performance

Víctor Mayoral-Vilches[1,2], Jason Jabbour[3], Yu-Shun Hsiao[3], Zishen Wan[4], Alejandra Martínez-Fariña[1], Martiño Crespo-Álvarez[1], Matthew Stewart[3], Juan Manuel Reina-Muñoz[1], Prateek Nagras[1], Gaurav Vikhe[1], Mohammad Bakhshalipour[5], Martin Pinzger[2], Stefan Rass[6,2], Smruti Panigrahi[7], Giulio Corradi[8], Niladri Roy[9], Phillip B. Gibbons[5], Sabrina M. Neuman[10], Brian Plancher[11], Vijay Janapa Reddi[3]

I. Introduction and Overview

In order for robotic systems to operate safely and effective in dynamic real-world environments, their computations must run at real-time rates while meeting power constraints. Accelerating robotic kernels on heterogeneous hardware, such as GPUs and FPGAs, is emerging as a crucial tool for enabling such performance [1], [2], [3], [4], [5], [6], [7]. Such computational improvements, combined with the growing dependency on ROS 2 [8], [9] across the robotics community, accentuates the community's demand for a standardized, industry-grade benchmark to evaluate varied hardware solutions.

Recently, there has been a plethora of workshops and tutorials focusing on benchmarking robotics applications [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], and while benchmarks for specific robotics algorithms [21], [22] and certain end-to-end robotic applications, such as drones [23], [24], [25], [26], do exist, the nuances of analyzing general ROS 2 computational graphs on heterogeneous hardware is yet to be fully understood.

In this paper, we introduce *RobotPerf*, an open-source and community-driven benchmarking tool designed to assess the performance of robotic computing systems in a standardized, architecture-neutral, and reproducible way, accommodating the various combinations of hardware and software in different robotic platforms (see Figure 1). RobotPerf focuses on evaluating robotic workloads in the form of ROS 2 computational graphs on a wide array of hardware setups, encompassing a complete robotics pipeline, emphasizing real-time critical metrics, and incorporating two distinct benchmarking methodologies. These approaches are: black-box testing, which measures performance by eliminating upper layers and replacing them with a test application, and grey-box testing, an application-specific measure that observes internal system states with minimal interference. The framework is open-source, user-friendly, easily extendable for evaluating custom ROS 2 computational graphs, and collaborates with major hardware acceleration vendors for a standardized benchmarking approach. We validate our approach through heterogeneous hardware benchmarks.

[1]Acceleration Robotics, Spain. [2]Alpen-Adria-Universität Klagenfurt, Austria. [3]Harvard University, USA. [4]Georgia Institute of Technology, USA. [5]Carnegie Mellon University, USA. [6]Johannes Kepler University Linz, Austria. [7]Ford Motor Company, USA. [8]AMD, USA. [9]Intel, USA. [10]Boston University, USA. [11]Barnard College, Columbia University, USA.

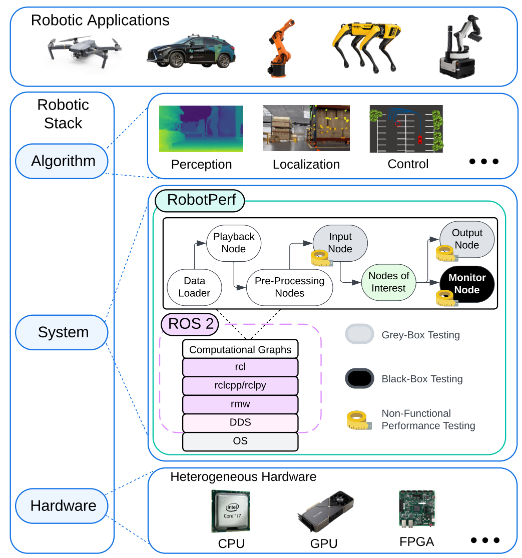

Fig. 1: A high level overview of RobotPerf.

II. Summary of Results

We conduct comprehensive benchmarking using RobotPerf to provide case studies of its uses and insights. RobotPerf's source code and documentation are available at `https://github.com/robotperf/benchmarks` and its methodologies are currently being used in industry.

First, given our ability to benchmark 18 platforms (bottom of Figure 2), RobotPerf is capable of benchmarking heterogeneous hardware platforms and workloads, paving the way for community-driven co-design and optimization of hardware and software. Second, we show how the defualt "one-size-fits-all" hardware selection strategy fails to capitalize on the nuanced differences in workload demands. For example, the latency radar plot for control (Figure 2 col 3, row 1), shows that the i7-12700H (I7H) outperforms the NVIDIA AGX Orin Dev. Kit (NO) on benchmarks `C1`, `C3`, `C4`, and `C5`, but is 6.5× slower on benchmark `C2`. Finally, we show how hardware acceleration can improve performance. For example, in the perception benchmarks (Figure 2 col 1), we include AMD's Kria KR260 hardware solution with and without a domain-specific hardware accelerator (ROBOTCORE Perception, a soft-core running in the FPGA for accelerating perception computations). We find that hardware acceleration can enable performance gains of as much as 11.5× (from 173 ms down to 15 ms for benchmark `a5`).

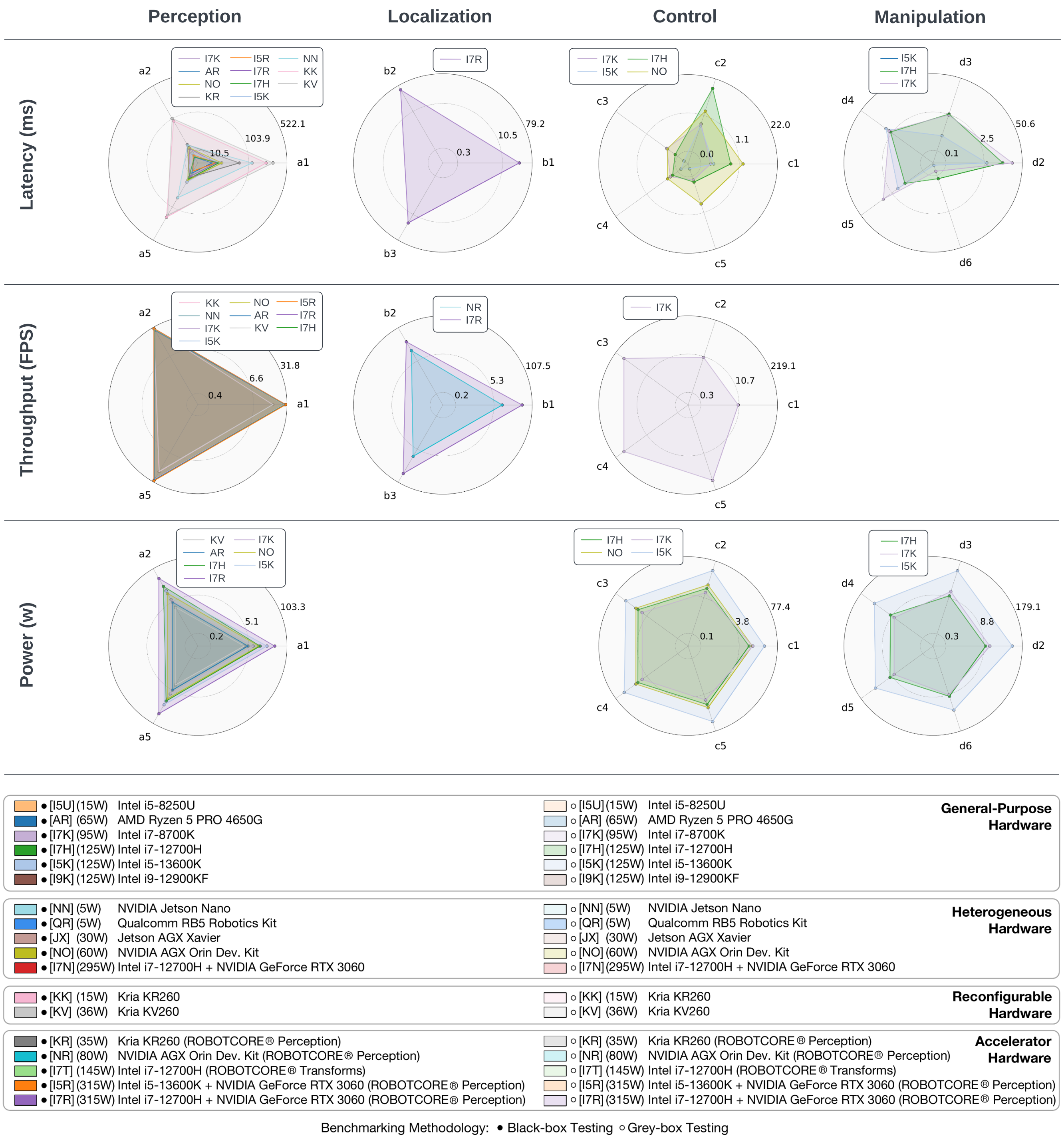

Fig. 2: Benchmarking results on diverse hardware platforms across perception, localization, control, and manipulation workloads defined in RobotPerf `beta` Benchmarks. Radar plots illustrate the latency, throughput, and power consumption for each hardware solution and workload, with reported values representing the maximum across a series of runs. The labels of vertices represent the workloads defined in our open-source repository at https://github.com/robotperf/benchmarks. Each hardware platform and performance testing procedure is delineated by a separate color, with darker colors representing Black-box testing and lighter colors Grey-box testing. In the figure's key, the hardware platforms are categorized into four specific types: general-purpose hardware, heterogeneous hardware, reconfigurable hardware, and accelerator hardware. Within each category, the platforms are ranked based on their Thermal Design Power (TDP), which indicates the maximum power they can draw under load. The throughput values for manipulation tasks and power values for localization tasks have not been incorporated into the `beta` version of RobotPerf. As RobotPerf continues to evolve, more results will be added in subsequent iterations.

REFERENCES

[1] S. M. Neuman, B. Plancher, T. Bourgeat, T. Tambe, S. Devadas, and V. J. Reddi, “Robomorphic computing: a design methodology for domain-specific accelerators parameterized by robot morphology,” in *ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2021, pp. 674–686.

[2] W. Liu, B. Yu, Y. Gan, Q. Liu, J. Tang, S. Liu, and Y. Zhu, “Archytas: A framework for synthesizing and dynamically optimizing accelerators for robotic localization,” in *MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture*, 2021, pp. 479–493.

[3] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, *et al.*, “Isaac gym: High performance gpu-based physics simulation for robot learning,” *arXiv preprint arXiv:2108.10470*, 2021.

[4] B. Plancher, S. M. Neuman, R. Ghosal, S. Kuindersma, and V. J. Reddi, “Grid: Gpu-accelerated rigid body dynamics with analytical gradients,” in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 6253–6260.

[5] V. Mayoral-Vilches, S. M. Neuman, B. Plancher, and V. J. Reddi, “Robotcore: An open architecture for hardware acceleration in ros 2,” in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 9692–9699.

[6] Z. Wan, A. Lele, B. Yu, S. Liu, Y. Wang, V. J. Reddi, C. Hao, and A. Raychowdhury, “Robotic computing on fpgas: Current progress, research challenges, and opportunities,” in *2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems (AICAS)*. IEEE, 2022, pp. 291–295.

[7] S. Liu, Z. Wan, B. Yu, and Y. Wang, *Robotic computing on fpgas*. Springer, 2021.

[8] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “Ros: an open-source robot operating system,” in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.

[9] V. Mayoral-Vilches, “ros-robotics-companies,” https://github.com/vmayoral/ros-robotics-companies, [Accessed: July 9, 2023].

[10] “Icra2021 workshop cloud-based competitions and benchmarks for robotic manipulation and grasping,” June 2021. [Online]. Available: https://sites.google.com/view/icra2021-workshop/home

[11] “Icra 2022 workshop determining appropriate metrics and test methods for soft actuators in robotic systems,” May 2022. [Online]. Available: https://sites.google.com/andrew.cmu.edu/softactuatormetrics/

[12] “Icra 2022 workshop on releasing robots into the wild: Simulations, benchmarks, and deployment,” May 2022. [Online]. Available: https://www.dynsyslab.org/releasing-robots-into-the-wild-workshop/

[13] “Iros 2020 workshop on benchmarking progress in autonomous driving,” Oct. 2020. [Online]. Available: https://www.robotics.qmul.ac.uk/events/iros-2021-workshop/

[14] “Iros 2021 workshop - benchmarking of robotic grasping and manipulation: protocols, metrics and data analysis,” Sept. 2021. [Online]. Available: https://www.robotics.qmul.ac.uk/events/iros-2021-workshop/

[15] “Evaluating motion planning performance,” Oct. 2022. [Online]. Available: https://motion-planning-workshop.kavrakilab.org/

[16] “Methods for objective comparison of results in intelligent robotics research,” Oct. 2023. [Online]. Available: http://www.robot.t.u-tokyo.ac.jp/TCPEBRAS_IROS2023/index.html

[17] “Benchmarking tools for evaluating robotic assembly of small parts,” July 2020. [Online]. Available: https://www.uml.edu/research/nerve/assembly-workshop-rss-2020.aspx

[18] “2021 rss workshop on advancing artificial intelligence and manipulation for robotics: Understanding gaps, industry and academic perspectives, and community building,” July 2021. [Online]. Available: https://sites.google.com/view/rss-ai-manipulationperspective/home

[19] “Robot learning in the cloud: Remote operations and benchmarking,” July 2022. [Online]. Available: https://sites.google.com/andrew.cmu.edu/cloud-robotics-benchmarking/

[20] “Datasets and benchmarking tools for advancing and evaluating robotic manufacturing,” July 2023. [Online]. Available: https://sites.google.com/view/rss-2023-nist-moad

[21] M. Bakhshalipour, M. Likhachev, and P. B. Gibbons, “Rtrbench: A benchmark suite for real-time robotics,” in *2022 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*. IEEE, 2022, pp. 175–186.

[22] S. M. Neuman, T. Koolen, J. Drean, J. E. Miller, and S. Devadas, “Benchmarking and workload analysis of robot dynamics algorithms,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 5235–5242.

[23] B. Boroujerdian, H. Genc, S. Krishnan, W. Cui, A. Faust, and V. Reddi, “Mavbench: Micro aerial vehicle benchmarking,” in *2018 51st annual IEEE/ACM international symposium on microarchitecture (MICRO)*. IEEE, 2018, pp. 894–907.

[24] S. Krishnan, Z. Wan, K. Bhardwaj, P. Whatmough, A. Faust, S. M. Neuman, G.-Y. Wei, D. Brooks, and V. J. Reddi, “Automatic domain-specific soc design for autonomous unmanned aerial vehicles,” in *2022 55th IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE, 2022, pp. 300–317.

[25] S. Krishnan, Z. Wan, K. Bhardwaj, N. Jadhav, A. Faust, and V. J. Reddi, “Roofline model for uavs: A bottleneck analysis tool for onboard compute characterization of autonomous unmanned aerial vehicles,” in *2022 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*. IEEE, 2022, pp. 162–174.

[26] D. Nikiforov, S. C. Dong, C. L. Zhang, S. Kim, B. Nikolic, and Y. S. Shao, “Rosé: A hardware-software co-simulation infrastructure enabling pre-silicon full-stack robotics soc evaluation,” in *Proceedings of the 50th Annual International Symposium on Computer Architecture*, 2023, pp. 1–15.