Agents of Autonomy: A Systematic Study of Robotics on Modern Hardware

Mohammad Bakhshalipour
bakhshalipour@cmu.edu
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA

Phillip B. Gibbons
gibbons@cs.cmu.edu
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA

ABSTRACT

As robots increasingly permeate modern society, it is crucial for the system and hardware research community to bridge its long-standing gap with robotics. This divide has persisted due to the lack of *(i)* a systematic performance evaluation of robotics on different computing platforms and *(ii)* a comprehensive, open-source, cross-platform benchmark suite.

To address these gaps, we present a systematic performance study of robotics on modern hardware and introduce *RoWild*, an open-source benchmark suite for robotics that is comprehensive and cross-platform. Our workloads encompass a broad range of robots, including driverless vehicles, pilotless drones, and stationary robotic arms, and we evaluate their performance on a spectrum of modern computing platforms, from low-end embedded CPUs to high-end server-grade GPUs.

Our findings reveal that current architectures experience significant inefficiencies when executing robotic workloads, highlighting the need for architectural advancements. We discuss approaches for meeting these requirements, offering insights for improving the performance of robotics.

The full version of the paper is available in [11], and the source code of the benchmark suite is available in [2].

CCS CONCEPTS

• **Computing methodologies → Robotic planning**; **Modeling methodologies**; • **General and reference → Measurement**; **Evaluation**; **Performance**; • **Computer systems organization →** *Real-time system architecture*; **Architectures**.

KEYWORDS

Robotics, Computer Architecture, Benchmarking

ACM Reference Format:
Mohammad Bakhshalipour and Phillip B. Gibbons. 2024. Agents of Autonomy: A Systematic Study of Robotics on Modern Hardware. In *Abstracts of the 2024 ACM SIGMETRICS/IFIP PERFORMANCE Joint International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS/PERFORMANCE Abstracts '24), June 10–14, 2024, Venice, Italy*. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3652963.3655043

SIGMETRICS/PERFORMANCE Abstracts '24, June 10–14, 2024, Venice, Italy

ACM ISBN 979-8-4007-0624-0/24/06
https://doi.org/10.1145/3652963.3655043

1 INTRODUCTION

The advancement of robotics technology is rapidly changing the world we live in. With predictions of 20 million robots by 2030 [1] and a market capitalization of US$210 billion by 2025 [23], it is clear that robotics will play an increasingly important role in society. To become widespread, robots need to meet the demands of real-world environments, which necessitates them being autonomous and capable of performing complex artificial intelligence (AI) tasks in real-time.

Computer hardware and architecture play a paramount role in realizing real-time robotics, evidenced by the deployment of robot-specific hardware accelerators in the architecture of the latest edge processors. Recent robotic platforms [14, 15, 17] include hardware accelerators for operations like tree-extension and ray-casting that have massive usage in robotics; Intel's multi-robot system [14] has a full-fledged "Robot SoC."

Surprisingly, the computer systems research community has under-explored robotics. This is borne out by the scant few publications in top computer systems conferences. There is a large gap between robotics and the computer systems community, depriving robotics of many improvements achievable by system-level techniques.

The gap is largely because of the lack of *(i)* a systematic performance study and *(ii)* a comprehensive, open-source, cross-platform benchmark suite. As a result of *(i)*, the robotic tasks, their performance requirements, and their system-level implications are unclear to the community. And, due to *(ii)*, the few research papers include only one [9, 19] or a couple of applications [20, 22] in their evaluations, leaving the impact on other applications unknown.

2 THE ROWILD BENCHMARK SUITE

This paper aims to bridge this gap by introducing *RoWild*, a comprehensive, open-source, cross-platform robotic benchmark suite. The challenge of benchmarking robotics lies in the vast array of applications, from self-driving cars to home-assistant robots, with more to come in the future. It is impractical to represent all of these applications in a single benchmark suite. *RoWild* overcomes this challenge by exploiting the fact that different robots, despite their different applications, perform a finite set of common "robotic tasks" such as scene understanding and pathfinding. For instance, both self-driving cars and home-assistant robots require scene understanding. Nevertheless, the algorithms and constraints in conducting such tasks can vary widely across different applications.

RoWild comprises a wide range of robotic tasks, encompassing the software pipeline of practically all autonomous robots. With versatility in mind, *RoWild* implements each task with various algorithms and parameters. This flexible approach enables the configuration and pipelining of tasks to model the end-to-end computation of diverse robotic applications. By including a broad set of tasks and algorithms, *RoWild* is capable of modeling numerous robotic applications, thus providing a comprehensive benchmark suite.

Our choice of implemented tasks and algorithms stems from an analysis of 29 industrial robots, encompassing a diverse range from arm manipulators and home-cleaning robots to self-driving vehicles. This analysis was undertaken to ensure a broad yet relevant selection. Additionally, we incorporate state-of-the-art research algorithms (e.g., deep learning-based pathfinding) into *RoWild*, positioning it as a suitable suite for future robotics. We develop *RoWild* with essential considerations: specifically, *RoWild* is high-performance, simulator-friendly, versatile, and modular. Table 1 compares key features of *RoWild* and the related work.

Table 1: Comparison of related work features with RoWild. More ✔ is better. ✣ shows variable outcomes (✘ or ✔) based on specifics like kernel or simulator.

Paper/Repository	Scope	End-to-End	High-Perf.	Simulator-Friendly	Multi-Platform	Versatile & Modular	System-Level Analysis
Lin et al. [18] Yu et al. [24]	Self-Driving Cars	✔	✔✔	Unknown (private)	✔✔	Unknown (private)	✔
MAVBench [12]	Drones	✔	✔	✣	✔	✘	✔
One-off [8, 13, 16]	Single Task	✘	✣	✣	✘	✔	✘
ROS [7]	Broad	✘	✣	✘	✘	✔	✘
Educational [3, 6]	Broad	✘	✘	✣	✘	✘	✘
RTRBench [10]	Broad	✘	✔	✔	✘	✔	✘
RoWild	Broad	✔	✔✔	✔	✔✔	✔	✔✔

3 COMPREHENSIVE PERFORMANCE STUDY

This paper's second contribution is to investigate the system-level implications of robotics. Using *RoWild*'s tasks, we model six different end-to-end robotic applications and evaluate them on a spectrum of platforms, ranging from low-end embedded CPUs to high-end server-grade GPUs. Table 2 summarizes the modeled applications, and Table 3 presents the evaluated platforms.

Table 2: The modeled end-to-end applications.

Name	Mission	Environment	Name	Mission	Environment
DeliBot	Delivery	Our Campus	PatrolBot	Patrolling	Our Campus
MoveBot	Manipulation	Synthetic	HomeBot	Cleaning	Hypersim [21]
FlyBot	Photography	FR Campus [4]	CarriBot	Transportation	Intel Lab [5]

Table 3: The evaluated compute platforms.

Platform	Cores	Freq. (GHz)	TDP (W)	Memory (GB)
ARM Cortex A57 CPU	4	1.43	10	4
Nvidia Maxwell GPU	128	0.92	10	4
Intel Xeon Gold CPU	20 (×2)	2.10	125	384
Nvidia Titan X GPU	3584	1.41	250	12

While previous studies [10, 12] have conducted some system-level analysis (e.g., CPU vs. GPU) specific to their applications, these analyses remained at a high level. In contrast, our study delves deeper to investigate low-level implications, including the efficacy of caching, prefetching, and vectorization.

4 EVALUATION HIGHLIGHTS

Our system-level investigations reveal *significant inefficiencies in the architecture* of today's prevailing compute platforms when executing robotic workloads. Specifically, we find:

- **Vectorization Is Ineffective.** Despite the large silicon real-estate it occupies, CPU vectorization does little for robotic tasks: in the common case of an axis-unaligned orientation, the robot's memory layout is not vectorization-friendly.
- **On-Edge Parallelism Hits The Memory Wall.** Massive parallelism on edge platforms (e.g., Nvidia's Jetson Nano) is bottlenecked by memory: the memory wall is hit well before Amdahl's law.
- **Simple Prefetchers Are Inadequate.** While simple data prefetchers can help with robotics workloads, complex AI algorithms used in robotics often use irregular data structures, producing memory patterns that defeat commercial prefetchers.
- **Caches Perform Significant Unnecessary Data Movements.** Hardware caches are unaware of the robots' software semantics; in many cases, caches work *in opposition* to them. As a consequence, caches perform excessive data movements, utilizing the memory hierarchy inefficiently.

REFERENCES

[1] 2019. How Robots Change the World. https://resources.oxfordeconomics.com/how-robots-change-the-world/.

[2] 2023. RoWild: A Comprehensive, Cross-Platform Benchmark Suite for Real-Time Robotics. https://cmu-roboarch.github.io/rowild/.

[3] n.d.. CppRobotics. https://github.com/onlytailei/CppRobotics/.

[4] n.d.. Freiburg Campus 360 Degree 3D Scans. http://ais.informatik.uni-freiburg.de/projects/datasets/fr360/.

[5] n.d.. Intel Research Lab. http://www2.informatik.uni-freiburg.de/%7Estachnis/datasets/datasets/intel-lab/intel.gfs.png.

[6] n.d.. PythonRobotics. https://github.com/AtsushiSakai/PythonRobotics/.

[7] n.d.. ROS - Robot Operating System. https://www.ros.org/.

[8] n.d.. The Open Motion Planning Library. http://ompl.kavrakilab.org/.

[9] Mohammad Bakhshalipour et al. 2022. RACOD: Algorithm/Hardware Co-Design for Mobile Robot Path Planning. In *ISCA*.

[10] Mohammad Bakhshalipour et al. 2022. RTRBench: A Benchmark Suite for Real-Time Robotics. In *ISPASS*. https://cmu-roboarch.github.io/rtrbench/.

[11] Mohammad Bakhshalipour and Phillip B. Gibbons. 2023. Agents of Autonomy: A Systematic Study of Robotics on Modern Hardware. *Proc. ACM Meas. Anal. Comput. Syst.* (2023).

[12] Behzad Boroujerdian et al. 2018. MAVBench: Micro Aerial Vehicle Benchmarking. In *MICRO*.

[13] Mihai Bujanca et al. 2019. SLAMBench 3.0: Systematic Automated Reproducible Evaluation of SLAM Systems for Robot Vision Challenges and Scene Understanding. In *ICRA*.

[14] Chieh Chung and Chia-Hsiang Yang. 2019. A Distributed Autonomous and Collaborative Multi-Robot System Featuring a Low-Power Robot SoC in 22nm CMOS for Integrated Battery-Powered Minibots. In *ISSCC*.

[15] Chieh Chung and Chia-Hsiang Yang. 2020. A 1.5-μJ/Task Path-Planning Processor for 2-D/3-D Autonomous Navigation of Microrobots. *JSSC* (2020).

[16] Stephen James et al. 2020. RLBench: The Robot Learning Benchmark & Learning Environment. *IEEE Robotics and Automation Letters* (2020).

[17] Youchang Kim et al. 2016. A 0.55V 1.1mW Artificial-Intelligence Processor with PVT Compensation for Micro Robots. In *ISSCC*.

[18] Shih-Chieh Lin et al. 2018. The Architectural Implications of Autonomous Driving: Constraints and Acceleration. In *ASPLOS*.

[19] Weizhuang Liu et al. 2021. Archytas: A Framework for Synthesizing and Dynamically Optimizing Accelerators for Robotic Localization. In *MICRO*.

[20] Sabrina M Neuman et al. 2021. Robomorphic Computing: A Design Methodology for Domain-Specific Accelerators Parameterized by Robot Morphology. In *ASPLOS*.

[21] Mike Roberts et al. 2021. Hypersim: A Photorealistic Synthetic Dataset for Holistic Indoor Scene Understanding. In *ICCV*.

[22] Jacob Sacks et al. 2018. RoboX: An End-To-End Solution to Accelerate Autonomous Control in Robotics. In *ISCA*.

[23] Statista Research Department. 2021. Global Robotics Market Revenue 2018–2025. https://www.statista.com/statistics/760190/worldwide-robotics-market-revenue/.

[24] Bo Yu et al. 2020. Building the Computing System for Autonomous Micromobility Vehicles: Design Constraints and Architectural Optimizations. In *MICRO*.