MATEC Web of Conferences **388**, 04008 (2023) https://doi.org/10.1051/matecconf/202338804008
2023 RAPDASA-RobMech-PRASA-AMI Conference

Heterogeneous computing for low-cost robotic platforms

Raynhardt van Zyl[1*] and *Callen* Fisher[2]

[1]Department of Electrical and Electronic Engineering, Stellenbosch University, South Africa, 20700857@sun.ac.za
[2]Department of Electrical and Electronic Engineering, Stellenbosch University, South Africa, cfisher@sun.ac.za

Abstract: This study presents a novel approach to the design of cost-effective autonomous rovers using a heterogeneous computing architecture. The research focuses on the development and evaluation of an autonomous rover system, priced under \$2 700 (USD), that can operate independently in various environments. The system's performance was assessed in real-world conditions, demonstrating an average speed of 0.5 m/s, an energy consumption of 0.2 kWh, and a decision-making error rate below 5 %. The results suggest that the use of heterogeneous computing architectures can lead to the development of affordable and efficient autonomous navigation systems. The paper concludes by discussing potential future research directions and the broader implications of these findings for the field of autonomous systems.

1 Introduction

The advent of autonomous rovers has sparked a revolution across a multitude of sectors, including agriculture, mining, and space exploration, leading to a significant increase in research and development initiatives [1,2]. These autonomous systems, capable of navigating and performing tasks in diverse environments, hold immense potential for enhancing operational efficiency and safety. However, one of the primary challenges that hinders the widespread adoption of these systems is their high computational requirements. Autonomous navigation, especially in unfamiliar or volatile environments, demands robust computational power, often leading to high costs that render these systems inaccessible for many potential applications.

This research aims to address this critical challenge by proposing an innovative solution: the use of heterogeneous computing architectures. Heterogeneous computing [3], a paradigm that integrates diverse processors or computational units, each optimized for specific tasks, offers a promising avenue for enhancing computational efficiency while reducing costs [4]. By distributing various computational tasks, such as sensor data processing, localization, mapping, and path planning, across different processing units based on their capabilities and

* Raynhardt van Zyl: 20700857@sun.ac.za
This research is funded by the NRF under grant number 129830

performance characteristics, one can achieve a more efficient and cost-effective autonomous rover system.

The primary objective of this research is to demonstrate the feasibility of developing an affordable autonomous rover system using a heterogeneous computing architecture without compromising on performance. The system is designed to operate with minimal human intervention, capable of autonomous path planning and dynamic obstacle avoidance. Furthermore, this research aims to contribute to the broader field of autonomous systems by providing a potential blueprint for creating cost-effective solutions for autonomous navigation and operation.

This study explores the potential of heterogeneous computing in the development of low-cost autonomous rovers, aiming to advance ones understanding of autonomous systems and open new avenues for their application in various sectors. The findings of this research could have significant implications for the future of autonomous navigation and operation, potentially making these technologies more accessible and affordable across a range of industries.

2 Subsystems

The design of autonomous rovers involves the integration of several key subsystems. In this study, the focus is placed on five primary subsystems: Assembly, Localisation, Mapping, Path Planning, and Control [5].

The assembly subsystem is responsible for the physical construction and integration of the rover's components. A four-wheel skid-steer drive mechanism was utilized with a detachable panel with dedicated voltage rails for each heterogeneous computing system [6]. This can be seen in Figure 1. The total cost for this subsystem, which includes the wheels, chassis, voltage regulators, motor drivers, and motors, was $525 (USD). The rover was able to maintain a speed of 0.5 m/s, and the voltage rails were within 2 % of their desired voltages.

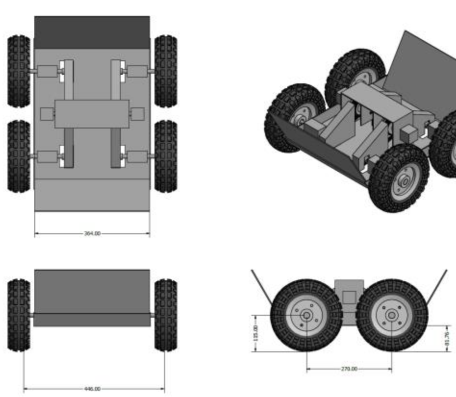

Fig. 1: Mechanical design

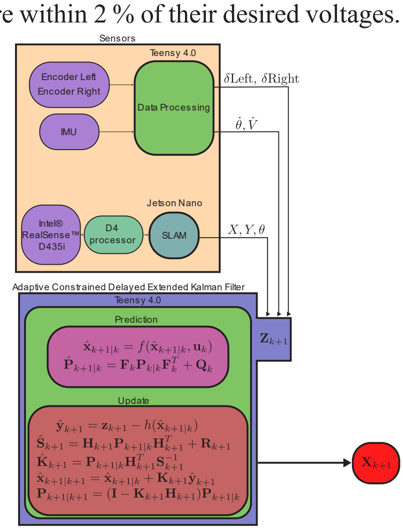

Fig. 2: Localisation subsystem

The localisation subsystem uses an Adaptive Constrained Delayed Extended Kalman Filter (ACDEKF) to determine the rover's spatial positioning [7]. This system, whose sensors cost $540 (USD) (IMU, Encoders and Intel D435i), fuses data from wheel encoders, an Inertial Measurement Unit (IMU), and a modified Simultaneous Localization and Mapping (SLAM) algorithm. The system overview can be seen in Figure 2. The system operates at 60 Hz and achieves an RMS of 0.002 m and 0.0008 °.

The mapping subsystem employs a dynamic mapping algorithm that uses a visibility mask from a LiDAR sensor to filter noisy data from the SLAM algorithm [8]. This approach reduces network bandwidth usage by filtering the data at the source (Nvidia Jetson Nano), where the data is then fused on an external, wirelessly connected PC (intel NUC). The system overview can be seen in Figure 3 and a mapping example can be seen in Figure 4. The total cost of this subsystem's sensors was \$80 (USD) (2D LiDAR), and it operates at 12 Hz on average.

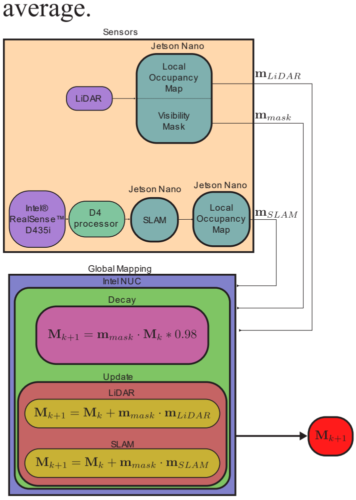

Fig. 3: Mapping subsystem

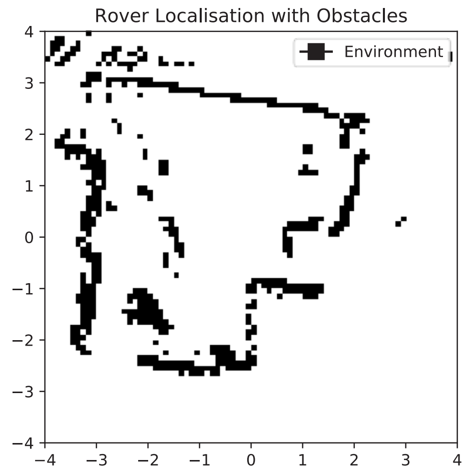

Fig. 4 : Mapping example

The path planning subsystem uses three main algorithms: RRT* [9], B-Spline path smoothing [10], and the nearest neighbour algorithm. The system overview can be seen to in Figure 5. These algorithms, which contain no sensor hardware, are used to outline an optimal trajectory for the rover, factoring in environmental and robotic constraints. The subsystem can regenerate a path at approximately 10Hz on average.

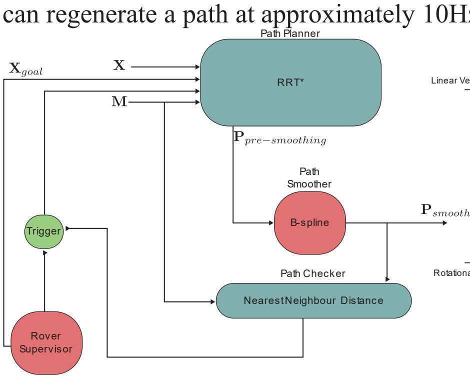

Fig. 5: Path planning subsystem

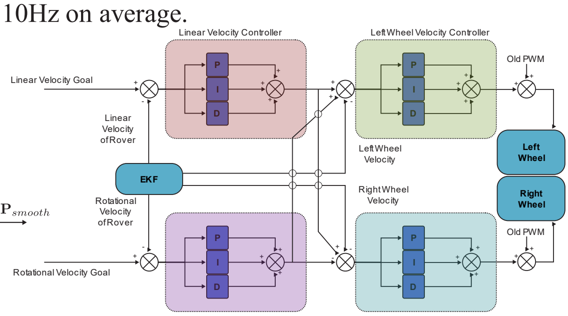

Fig. 6 : Control subsystem

The control subsystem uses a modified cascade Proportional Integral Derivative (PID) controller [11]. This controller updates the Pulse Width Modulation (PWM) signal for each wheel motor, regulating the rover's motion. The system overview can be seen in Figure 6. The system operates at 100 Hz on average. The cost of this subsystem was covered by the other subsystems mentioned above.

Each subsystem plays a crucial role in the overall operation of the autonomous rover, contributing to its ability to navigate autonomously with minimal human intervention.

3 Heterogeneous Computing Architecture

The core of any autonomous robotic system is its computational architecture, with the chosen architecture significantly influencing overall system performance [12]. In this research, a heterogeneous computing architecture was utilized, which is an innovative model that combines various specialized processing units to perform specific tasks [13]. This architecture, costing a total of $1 780 (USD), incorporates a Teensy 4.0 microcontroller, a Jetson Nano, an Intel NUC-i7, and an Intel D4-ASIC part of the Intel RealSense D435i.

Heterogeneous computing architecture is a method that leverages the unique capabilities of different processing units, allowing for the execution of complex tasks with enhanced efficiency and lower power consumption. This approach has gained traction in robotics due to its potential for high-performance computing [14]. The system layout is shown in Figure 7, shows how the hardware elements are connected.

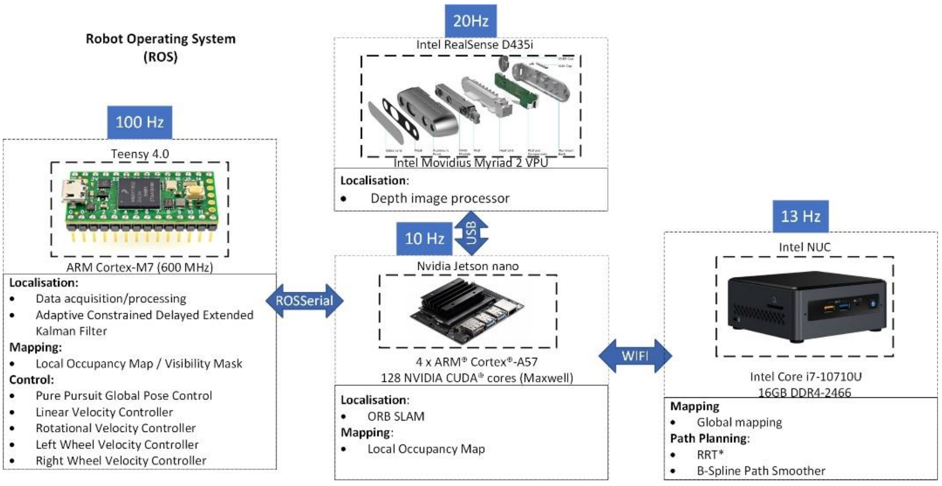

Fig. 7 : Heterogeneous computing layout

The Jetson Nano, developed by NVIDIA, is a high-performance, low-power System on a Chip (SoC) designed to accelerate machine learning applications. It integrates a quad-core ARM (Advanced RISC Machine) Cortex-A57 CPU and a 128-core NVIDIA Maxwell GPU onto a single chip, facilitating efficient execution of complex computations. The Jetson Nano was responsible for running the ORB SLAM algorithm and managing the Local Occupancy Map.

The Intel NUC-i7 is a compact yet powerful computing device. It handles the computationally intensive tasks of global mapping and running the RRT* and B-Spline Path Smoother algorithms, allowing the Jetson Nano to focus on real-time accurate SLAM localisation.

The Teensy 4.0 microcontroller, built on the ARM Cortex-M7 processor core, is designed for real-time, high-speed signal processing and control applications. It executes various algorithms, such as the Adaptive Constrained Delayed Extended Kalman Filter and multiple control algorithms.

The Intel D4-ASIC, an integrated vision processor with extensive image processing capabilities, is optimized to run vision-based algorithms with minimal power consumption. It executes the Depth image processor algorithm.

The Robot Operating System (ROS) integrates these varied elements into a network optimized for power efficiency and robust real-time processing. This complex network involves the Teensy 4.0 encoding data as ROS messages via the rosserial_arduino library, relayed to the Jetson Nano through the rosserial_python node. Simultaneously, the Nano maintains a TCP/IP dialogue with the Intel NUC over Wi-Fi. Lastly, the Jetson Nano-Myriad 2 VPU data interchange happens through the VPU's SDK over a USB (Universal Serial Bus) interface.

By leveraging the distinct capabilities of these processing units, the heterogeneous computing architecture enables the execution of complex tasks with enhanced efficiency, all while maintaining lower power consumption. This approach has the potential to significantly improve the performance and cost-effectiveness of autonomous rovers.

4 System Performance and Evaluation

The performance of the autonomous rover system, which is based on a heterogeneous computing architecture, was thoroughly evaluated to assess its adaptability, efficiency, and decision-making accuracy in real-world scenarios [15].

The evaluation involved rigorous testing of the system's ability to navigate through various terrains, dynamically avoid obstacles, and maintain a consistent average speed of 0.5 m/s. The system's energy efficiency was also a key focus of the evaluation. The system consumed an average of only 0.2 kWh (measured externally from the battery terminals), a testament to the power-efficient design of its components [16].

In terms of decision-making accuracy, the system demonstrated a less than 5 % error rate (tests which resulted in a system software/hardware crash). This indicates that the system is capable of analysing sensor data accurately and making reliable navigation decisions, which is crucial for autonomous operation [16].

When compared to the NVIDIA Jetson AGX Xavier Developer Kit (32GB), which costs $3 700 (USD), the cost-effectiveness of the system becomes evident. Despite the significant price difference, the system does not compromise on functionality, demonstrating that it is possible to develop a cost-effective autonomous navigation system without sacrificing performance.

The mechanical drive and sensor hardware of the system also played a significant role in its overall performance. The rover was able to efficiently interpret sensor data and exhibit practical obstacle avoidance across varied terrains. The energy consumption of the mechanical drive and sensors was also low, further highlighting the efficiency of the design. An example of the complete system in operation is shown in Figure 8.

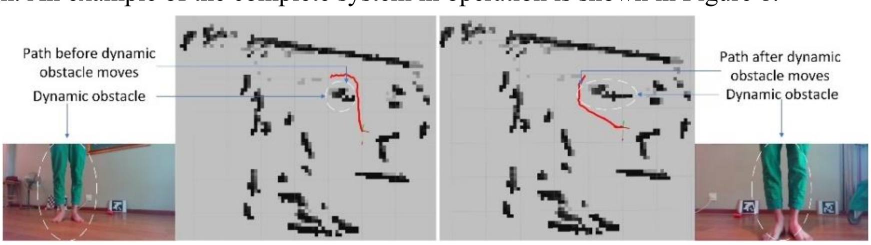

Fig. 8: Full system example

The total cost of the proposed solution was $2 501 (USD) ($1 780 (USD) for computation, $525 (USD) for the drive system and assembly, and $196 (USD) for sensors), which is less than the initial budget of $2 700 (USD) allocated for this project. This underlines the cost-effectiveness of the design.

When compared to other existing models in the market, such as the Husky A200™ UGV Robot Base priced at $10 266 (USD), and additional sensor packages like the SICK LMS-111 Lidar 2D Laser Scanner costing around $4 300 (USD), the economic advantage of this autonomous rover becomes even more apparent. Despite these significant price disparities, this system offers comparable adaptability, efficiency, and decision-making precision, making it a viable, cost-efficient model for high-performing autonomous rovers in the field.

5 Implications

The successful development and demonstration of an affordable autonomous rover system, underpinned by a heterogeneous computing architecture, lays a solid foundation for future exploration in this domain [17]. This research has the potential to revolutionize the field of autonomous systems by providing a cost-effective and efficient solution for autonomous navigation and operation.

One of the intriguing prospects for extending this work lies in the potential integration of AI-specific chips into the existing computing hardware. Each chip could be dedicated to running an independent neural network, thereby creating a mesh of neural networks, each responsible for a distinct task such as localisation, mapping, path planning, and control [18]. This modification could significantly enhance the system's performance by leveraging the inherent parallelism and efficient computation of AI-specific chips, thereby improving the real-time decision-making capabilities of the rover.

Moreover, transitioning from a deterministic approach to a more probabilistic model through the integration of AI could lead to more resilient and adaptive navigation strategies [19]. This transition could better equip the rover to handle unforeseen obstacles and real-time environmental variations, thereby enhancing its adaptability and robustness. Such advancements could pave the way for the development of more sophisticated, robust, and autonomous exploration rovers, further revolutionizing this approach to unmanned exploration in challenging landscapes.

6 Conclusion and Future Work

In conclusion, this study has successfully demonstrated the potential of heterogeneous computing architectures in developing cost-effective autonomous rover systems. By integrating a diverse set of computational units, namely a Teensy 4.0 microcontroller, Jetson Nano, NUC-i7, and Intel D4-ASIC, the designed system delivers impressive performance metrics, including an average speed of 0.5 m/s, an energy consumption of 0.2 kWh, and a decision-making error rate below 5 %. These experiments were validated on a low-cost rover (under $2 700 (USD)).

This platform could perform accurate mapping of an environment and localize itself inside the environment. Algorithms enabled it to perform exploration of an unknown environment, as well as perform path planning. These algorithms were adapted to handle dynamic objects as well.

These results underscore the viability of this system as a cost-efficient alternative to existing models, without compromising on functionality. The economic advantage is particularly evident when compared to the NVIDIA Jetson AGX Xavier Developer Kit (32 GB) and the Husky A200™ UGV Robot Base, which are significantly more expensive.

Looking forward, the insights gained from this research can serve as a foundation for further exploration and development in the field of autonomous rovers. The potential for integrating AI-specific chips and advanced algorithms into the system opens up exciting avenues for future research.

In summary, this research underscores the potential of heterogeneous computing in the development of affordable, high-performing autonomous rovers. These findings will inspire further innovation in this field, paving the way for more accessible and affordable autonomous systems across diverse applications.

References

1. L. Liebenberg, E. Boje, and G. Bright. 2018. Low-Cost Autonomous Navigation and Control of a Mechanically Steered Vehicle in a Farm Environment, Robotics, 7(3), 42.

2. M. R. Endsley. 2017. Autonomous Horizons: System Autonomy in the Air Force - A Path to the Future. Volume I-VI: Human-Autonomy Teaming. AFRL-RI-TR-2015-017. Air Force Research Laboratory, Rome, NY.

3. A. Brodtkorb, C. Dyken, T. Hagen, J. Hjelmervik, and O. Storaasli. 2010. State-of-the-art in heterogeneous computing, Scientific Programming, 18, pp 1–33.

4. S. Thrun, W. Burgard, and D. Fox. 2000. A Real-Time Algorithm for Mobile Robot Mapping With Applications to Multi-Robot and 3D Mapping, Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings, 1, pp 321–328. Link

5. M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng. 2009. ROS: an open-source Robot Operating System, ICRA Workshop on Open Source Software.

6. J. Yi, H. Wang, J. Zhang, D. Song, S. Jayasuriya, and J. Liu. 2009. Kinematic Modeling and Analysis of Skid-Steered Mobile Robots With Applications to Low-Cost Inertial-Measurement-Unit-Based Motion Estimation, IEEE Transactions on Robotics, 25, pp 1087–1097.

7. A. Howard, M. J. Mataric, and G. S. Sukhatme. 2001. Relaxation on a Mesh: a Formalism for Generalised Localisation, Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the Next Millennium, 2, pp 1055–1060.

8. G. Wang, Z. Gao, Y. Zhang, B. Ma. 2018. Adaptive Maximum Correntropy Gaussian Filter Based on Variational Bayes, Sensors, 18(6). [Online]. Available: https://dx.doi.org/10.3390/s18061960

9. K. Wei, B. Ren. 2018. A Method on Dynamic Path Planning for Robotic Manipulator Autonomous Obstacle Avoidance Based on an Improved RRT Algorithm, Sensors, 18(2). [Online]. Available: https://dx.doi.org/10.3390/s18020571

10. D. Jung, P. Tsiotras. 2008. On-line Path Generation for Small Unmanned Aerial Vehicles Using B-Spline Path Templates, AIAA Guidance, Navigation and Control Conference and Exhibit. [Online]. Available: https://dx.doi.org/10.2514/6.2008-7135

11. M. Park, S. Lee, W. Han. 2015. Development of Steering Control System for Autonomous Vehicle Using Geometry-Based Path Tracking Algorithm, ETRI Journal, 37(3). [Online]. Available: https://dx.doi.org/10.4218/etrij.15.0114.0123

12. F. J. Perez-Grau, R. Ragel, F. Caballero, A. Viguria, A. Ollero. 2019. An architecture for robust UAV navigation in GPS-denied areas, Journal of Field Robotics, 36(2), pp 269-293.

13. R. Y. Zhong, X. Xu, E. Klotz, S. J. Newman. 2017. Intelligent Manufacturing in the Context of Industry 4.0: A Review, Engineering, 3(5), pp 616-630.

14. L. D. Xu, E. S. Xu, L. Li. 2018. Industry 4.0: state of the art and future trends, International Journal of Production Research, 56(8), pp 2941-2962.

15. T. J. Tanzi and J. Isnard. 2019. Autonomous system for data collection: Location and mapping issues in post-disaster environment, Comptes Rendus Physique, 20(4), pp 333-344.

16. C. L. R. McGhan, R. M. Murray, R. Serra, M. D. Ingham, M. Ono, T. Estlin, and B. G. Williams. 2015. A risk-aware architecture for resilient spacecraft operations, IEEE Aerospace Conference Proceedings.

17. L. Kunze, N. Hawes, T. Duckett, M. Hanheide, and T. Krajnik. 2018. Artificial Intelligence for Long-Term Robot Autonomy: A Survey, IEEE Robotics and Automation Letters, 3(4), pp 4023-4030.

18. L. Fridman, D. G. Brown, M. Glazer, W. W. Angell, S. Dodd, B. Jenik, J. Terwilliger, A. Patsekin, J. Kindelsberger, L. Ding, S. Seaman, A. Mehler, A. Sipperley, A. M. Pettinato, B. Seppelt, L. Angell, and B. Reimer. 2019. MIT Advanced Vehicle Technology Study: Large-Scale Naturalistic Driving Study of Driver Behavior and Interaction With Automation, IEEE Access, 7, pp 102021-102043.

19. R. L. Hewitt, A. Ellery, and A. H. J. de Ruiter. 2017. Training a terrain traversability classifier for a planetary rover through simulation, International Journal of Advanced Robotic Systems, 14(6), pp 1-14.