Using a multi-robot system for improved path planning

Kyle Ahlschlager[1], and *Callen* Fisher[1*]

[1]Department of Electrical and Electronic Engineering, Stellenbosch University, South Africa
We would like to acknowledge the National Research Foundation (NRF) for funding this project/research, Grant Number 129830.

Abstract. Numerous terrestrial robotic platforms immoderately use computational power for path planning. These platforms typically use a vision system to identify obstacles and perform path planning. In cases where the vision systems are unable to function due to larger obstacles in the area, the paths are chosen as random functions of the given terrain, often resulting in missteps, and moving away from the target location. In this research, we present a multi-robot system comprised of a terrestrial robot with a tethered aerial drone. By making use of the additive overhead view the target location can be identified and the exploration and path planning algorithms biased, subsequently reducing the computational cost, and creating a more efficient path planning approach. Focus has been placed on the control architecture of the system; allowing for the various systems to operate independently in the event of catastrophe due to the unpredictable nature of the environments they operate in.

1 Introduction

State-of-the-art terrestrial robots, such as Spot [1] and RHex [2], are robust platforms that are capable of traversing rough terrain. These robots are often employed in obstacle-ridden areas and are expected to locate and navigate to points of interest. However, when the terrain includes larger obstacles, the vision systems of these robots are unable to function effectively. The path planning and exploration algorithms deployed on these robots are time-consuming and computationally expensive [3]. In this research, we present a highly capable terrestrial robot, that can deploy a tethered aerial drone to influence the exploration algorithm to find the goal location and most efficient route toward it.

The use of tethered unmanned aerial vehicles (TUAV) or tethered drones have been explored in a variety of applications ranging from mapping underground mines to public safety and crowd management. The use of TUAVs in mine mapping is closely related to the type of work our research focuses on, as such the Oxpecker and Rhino research conducted in 2022 was used as a basis for the design of the drone system [4]. While a benefit of tethered drones is the ability to power them from a constant power source on the ground, prior research

* Corresponding author: cfisher@sun.ac.za

has concluded that the power loss resulting from powering the drone from the batteries of the unmanned ground vehicle (UGV) was excessive [4].

The paper is structured as follows; Section 2 describes the design methodology of the system, followed by the implementation of the various path planning algorithms, detailed in Section 3. Section 4 describes the experimental setup and their results. Concluding in Section 5 with a discussion of the results and future work for the project described in Section 6.

2 Design methodology

A RHex-styled robot was developed to traverse the desired rough-terrains and a five-inch quadcopter has been tethered to the RHex platform via a four-point winch to provide a system for capturing the additive overhead view, this platform is shown in Figure 1. The design of the physical system is detailed in Section 2.1 and the controller architecture in Section 2.2.

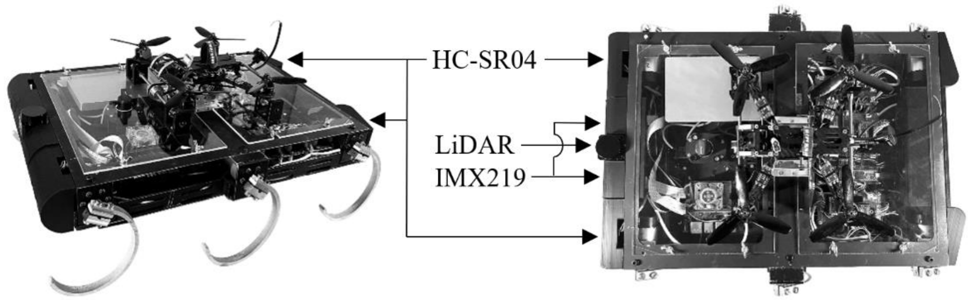

Fig. 1. The developed RHex platform with tethered drone with attached sensors.

2.1 Mechanical design

The platform's mechanical design is described by four main areas, the RHex frame, the C-shaped compliant legs, the winch, and the drone. The frame is the structural backbone of the project, and the legs provide the basis for the RHex platform's unique motion. The drone captures the additive overhead view used to improve the path planning and exploration algorithms. The winch tethers the drone to the RHex and controls the drone's altitude.

The RHex-styled robot's frame was based on the designs of previous versions, such as the X-RHex [5]. It comprises two runners with three cross beams attached, making the frame both rigid and lightweight. The frame was slotted to allow for easier cable routing and mounting points. Two thin aluminium plates were used to create a 'sandwich' structure to increase the frame's overall rigidity. Figure 2.a shows the designed frame. A C-shaped leg design was adopted due to its simplicity and versatility over various terrains [6]. Figure 2.b shows the offset applied to the leg to allow for the alignment of the motor shaft and leg centre, this improves the stability of the robot when standing. The standing period coincides with the point of maximum force on the leg.

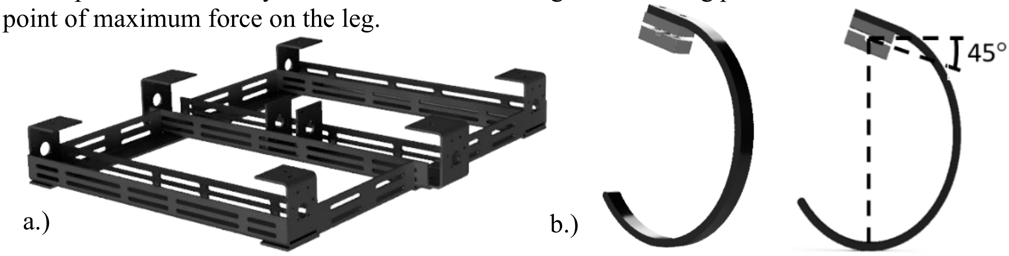

Fig. 2. Image (a) showing the designed RHex frame without the top and bottom plates. Image (b) showing the C-Shaped compliant leg with motor coupler.

The winch that tethers the drone to the RHex was designed to be simplistic, and it consists of a single drive motor that connects two rods with a 1:1 gear ratio. These rods afford the additional function of acting as a spool for winding the winch cable, as represented in Figure 3.a. This design allows four contact points on the drone to reduce the complexity required to maintain flight stability. The drone used is the Eachine Wizard X220 V3. Special mounts were designed and 3D-printed to attach to the frame near each motor and provide points for the attachment of the tether cables. Figure 3.b shows the chosen drone. The camera (RunCam Eagle Two) used to capture the additive aerial view is mounted to a servo-controlled pan-tilt gimbal on the underside of the drone, this allows for capturing a full 360-degree view of the area without requiring the rotation of either the Drone or RHex. The drone has a fixed landing area, and a limit switch is used to detect the state of the drone.

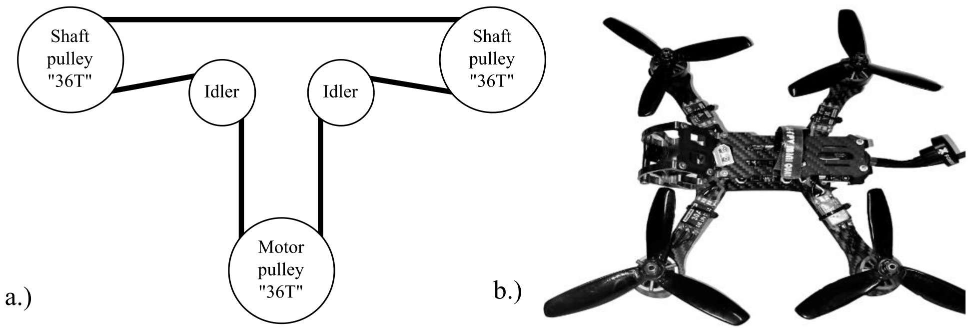

Fig. 3. Image (a) showing a representation of the winch system used to tether the drone. Image (b) showing the drone used to capture the additive overhead view.

The properties of the RHex platform are summarised in Table 1. The drive motor for the tether was chosen to provide enough torque to reel the drone in at full thrust and provide the additional ability to winch the RHex when needed. The tethered drone in flight is shown in Figure 4.

Table 1. Properties of the designed RHex platform.

RHex Dimensions (WxLxH)	40x56x18 cm
Leg motors (Gear ratio, Torque)	50:1, 21 kg·cm
Winch motors (Gear ratio, Torque)	150:1, 49 kg·cm
Weight	6.8 kg

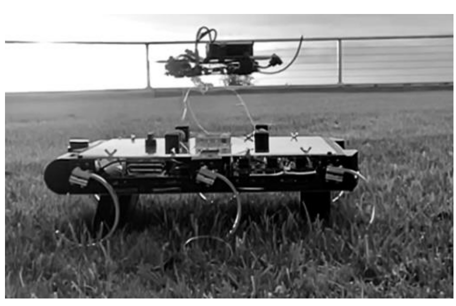

Fig. 4. The tethered drone deployed on the RHex platform.

2.2 Controller design

The control system of the RHex robot was designed to be split over three microcontrollers, two Teensy 4.0s and a Teensy 4.1. The Teensy 4.0s are hereafter referred to as the motion controllers. The system was designed with the two motion controllers operating in open loop with respect to the environment. Sensor feedback from ultrasonic distance sensors and current sensors are connected to the Teensy 4.1 which provides instructions for the motion controllers to follow. The motion controllers are only aware of whether they must be stationary or moving at any given time. This design has proven effective in controlling the RHex platform since the minimum required six PID loops is computationally expensive when considering the necessary speed and position measurements for each leg. Additionally, the required GPIO connections for each tripod is more than twenty pins, in contrast with the twenty-four easily accessible pins of each motion controller, this in-part inspired the distributed controller design implemented in this research.

Each leg is considered an object and has a unique clock-based trajectory generator, with a corresponding trajectory follower contained in this object. This approach allows for independent control of each leg to implement the standard Tripod gait and more complex gaits such as stair-climbing [6]. The control architecture used per leg on the RHex platform is shown in Figure 5. The control approach uses separate controllers for position and speed, this was done to allow the motion controllers to disable one of the controller loops when not in use and subsequently increase the loop speed of each controller.

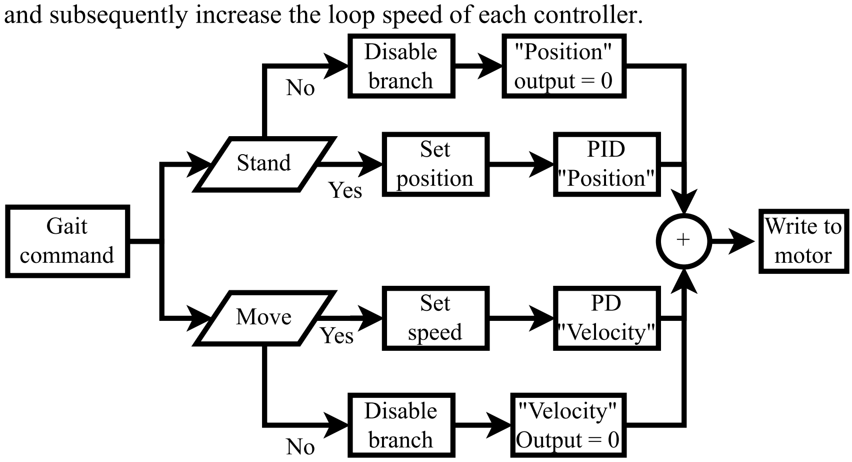

Fig. 5. Control architecture for each leg of the RHex platform.

The RHex utilises an Nvidia Jetson Nano single-board computer to perform all simultaneous localisation and mapping (SLAM), exploration, and path planning algorithms [7]. The actuators (brushed DC gear motors) are controlled by the motion controllers that utilise separate control loops for position and speed control of the actuators while obtaining feedback from the motor encoders. The Teensy 4.1 receives path commands from the Jetson Nano and converts these to gait instructions. The motion controllers then implement these instructions. The Jetson Nano is connected to two front-mounted IMX219 cameras with a 77-degree field-of-view (FOV), these cameras are used for object detection and depth estimation in the open loop path following discussed in Section 3. A 360-degree LiDAR is mounted to the front of the RHex robot at the same height as the top plate, this mounting was chosen as any objects detected by the LiDAR are inherently impassable by the RHex, while objects below this point are considered manageable to the RHex and will be further examined by the cameras as to whether they should be avoided. Mounting the LiDAR on the drone was considered, but given the drone's higher potential for failure, it was more sensible to mount most sensors on the RHex platform.

3 Path planning and exploration algorithms

To navigate an area, the platform is required to use an exploration algorithm to locate a goal object or point of interest and perform path planning to navigate to this location with the most efficient route. For this, three commonly applied path planning methods were tested, A*, RRT, and RRT* [8 - 11].

Each method was tested on the Jetson Nano using data obtained from the front-mounted LiDAR and stereo cameras. This data provided the basis for a sectional map in which a minimum of two paths were available, in typical applications the robot would follow the path with the lowest cost [7]. This approach, although effective in many environments, is far from perfect and often results in following a path leading away from the physical goal's location or even leading toward an impassable area. Consequently, more data is required to increase the effectiveness of these path planning algorithms. To address this, we provide an additive overhead view to add a bias to the algorithms. This bias can be in the form of positive weighting for paths leading toward the goal direction or applying a goal-biased sampling approach where the random sampling process is replaced with a probability distribution with a higher weighting for locations closer to the goal location.

By applying the biasing methods to each algorithm, the overall effectiveness of the additive overhead view can be estimated, and a suitable algorithm chosen for physical testing. Using a simulated environment with an arbitrary goal location, the biasing techniques are evaluated. The average time taken to find an optimal path is measured, as well as the total number of incorrect decisions or failures to find an optimal path. The results of this testing over ten simulations are shown in Table 2. As can be seen, RRT* displayed the most improvement to the biasing techniques, but A* has been chosen as the primary path planning algorithm for testing due to its impressive speed and robustness. Additionally, A* is a grid-based search algorithm and this aligns with the methods employed to process the LiDAR data.

Table 2. Comparison of the chosen path planning algorithms.

Algorithm	Baseline		With biasing		Improvement	
	Time (ms)	Failures	Time (ms)	Failures	Time (ms)	Failures
A*	23.976	2	15.358	1	8.618	1
RRT	4.221	1	3.098	0	1.122	1
RRT*	78.537	0	46.982	0	31.555	0

In practice, the overhead view provided by the tethered drone is used to either locate the goal and bias the algorithm toward the goal or to apply an appropriate weighting to each potential path based on the condition of the terrain corresponding to each path. This will allow the robot to reach the desired location without risking damage to the robot and conserving as much energy as possible. Figure 6 shows the decision tree used to determine when to deploy and when to recall the drone. The deployment of the drone relies on the feedback from the RHex platform's onboard cameras to determine whether it has reached the

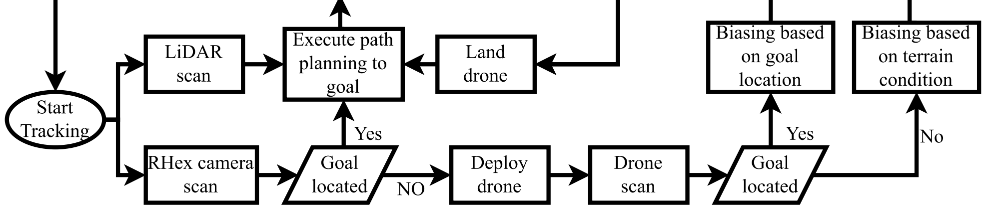

Fig. 6. Decision tree used to determine the deployment of the drone.

desired location or is unable to locate the next point in its path. By applying the RRT* algorithm with the additive overhead view, biasing can be achieved by both modifying the heuristic function for cost estimation and modifying the random sampling process. The effectiveness of this dual biasing is evaluated in Section 4 on the physical platform.

4 Experimentation and results

To test the functionality of the platform a simple test procedure was established. A goal object was selected, specifically an IR light, and this was placed in a room with multiple obstacles that are tall enough to block the view of the ground-based cameras, subsequently requiring the deployment of the drone. Each obstacle in the test was at minimum double the height of the RHex.

The test was initially run with the drone disabled to establish a baseline measurement, thereafter the drone was enabled, and the robot was instructed when to deploy the drone. This was done to evaluate the effect of deploying the drone when it seemingly would not be deployed. Finally, the test was conducted with the robot operating completely autonomously and using the decision tree shown in Figure 5 to determine when to deploy the drone. Table 3 shows the average results obtained over the three testing instances. These results were obtained when running with the RRT* algorithm and for each testing instance five runs were measured. The tests were redone in a different testing environment to verify the initial results.

Table 3. Results obtained during testing of the platform.

	Baseline	User-deployment	Autonomous deployment
Time taken (s)	305	328	243
# of deployments	-	5	2
# of failures	3	0	0
Voltage drop (V)	1.6	1.2	0.7

5 Conclusion and discussion

From the results shown in Table 3 it is evident that when the drone was in use and an additive overhead view made available the platforms ability to navigate through the testing course improved significantly. This is evident from the reduction in time taken, an improvement of 62 seconds was noted and the reduction in power consumption as measured by the voltage drop over the testing, 0.7V consumed with the drone versus 1.6V without. Increasing the number of deployments per test did not yield any significant advantages and resulted in slower overall times due to the deployment time of the drone.

Additionally, using different algorithms for path-planning in the testing setup resulted in negligible performance differences and this method can consequently be assumed as suitable for use in any path planning application.This research was considered successful as the system was capable of traversing unknown terrain faster and more efficiently with the use of an aerial drone than it would be able to without it.

6 Future work

Future work will include implementing feature detection on the drone, enabling the robot to perform path planning over greater distances and not merely bias search algorithms. The project can be taken further by adding current sensors to each leg motor to accurately determine the physical touch-down and take-off points, subsequently providing the trajectory generator with more information of the physical environment and the ability to vary the controller parameters to best suit the environments.

References

1. J. Moses, G. Ford, See Spot save lives: fear, humanitarianism, and war in the development of robot quadrupeds, in Digi War, **2**, 64–76 (2021)
2. U. Saranli, M. Buehler, D. Koditschek, *RHex: A Simple and Highly Mobile Hexapod Robot*, in ESE, 20, (2001)
3. H. Wang, S. Lou, J. Jing, Y. Wang, W. Liu, T. Liu, *The EBS-A* algorithm: An improved A* algorithm for path planning*, PLoS ONE, **17**, 3 (2002)
4. B. Rocamora, R. Lima, K. Samarakoon, J. Rathjen, J. Gross, G. Pereira, *Oxpecker: A Tethered UAV for Inspection of Stone-Mine Pillars*, in Drones, **7**, 73 (2023)
5. K. Galloway, G. Haynes, B. Ilhan, A. Johnson, R. Knopf, A. Goran, B. Plotnick, M. White, D. Koditschek, *X-RHex: A Highly Mobile Hexapedal Robot for Sensorimotor Tasks* (2010)
6. E. Moore, D. Campbell, F. Grimminger, M. Buehler, *Reliable stair climbing in the simple hexapod 'RHex'*, in ICRA, **3**, 2222-2227 (2002)
7. J. Sun, J. Zhao, X. Hu, H. Gao, J. Yu, *Autonomous Navigation System of Indoor Mobile Robots*, in Mathematics, **11**, 1-21 (2023)
8. M. Korkmaz, A. Durdu, *Comparison of Optimal Path Planning Algorithms,* in TCSET, **14**, 255-258, 2018
9. C. Zammit, E. van Kampen, *Comparison between A* and RRT Algorithms for UAV Path Planning*, in Proceedings of the 2018 AIAA Guidance, Navigation, and Control Conference [AIAA 2018- 1846] American Institute of Aeronautics and Astronautics Inc. (AIAA)
10. I. Noreen, A. Khan, Z. Habib, *A Comparison of RRT, RRT* and RRT*-Smart Path Planning Algorithms,* in IJCSNS, **16,** 10 (2016)
11. P. Lajevardy, A. Mousavian, M. Oskoei, *A Comparison Between RRT* and A* Algorithms for Motion Planning in Complex Environments,* in International Congress of Electrical Engineering, Computer Science and Information Technology, **2** (2015)