Miniaturized robotics and wireless for environmental sensing

Nishant Elkunchwar

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science in Mechanical Engineering

University of Washington

2021

Committee:
Sawyer B. Fuller
Vikram Iyer
Jaehyun Chung

Program Authorized to Offer Degree:
Mechanical Engineering

University of Washington

Abstract

Miniaturized robotics and wireless for environmental sensing

Nishant Elkunchwar

Chair of the Supervisory Committee:
Sawyer B. Fuller
Department of Mechanical Engineering

Commercially available sub-gram (20-30 mg), low cost sensors can enable sensing for a wide variety of applications. Recently developed insect scale robots like the University of Washington Robofly make for an ideal candidate to make use of these sub-gram sensors. Moreover, these sensors can also be attached to birds or insects to study their behaviors, like foraging patterns, body temperature etc. However, a number of challenges need to be addressed before such systems can be deployed on insects or robots at this scale. This thesis focuses on solutions to three particular problems in this area: implementation and demonstration of a biology-inspired source seeking algorithm, persistent operation of aerial robots by harvesting solar power, and development of a sensing platform for insect behavior studies. The results show that miniaturizing hardware and designing robot behavior inspired from biology can help solve these challenges. As advances in this domain lead to stable flight with aerial insect scale robots, the results shown in this work can help deploy teams of such robots to search for any number of hazardous sources over a long period of time without the need to set up a power infrastructure. This can be particularly useful in disaster-struck areas. Not only does this work show solution to these problems by implementing biology-inspired algorithms and enabling persistent operation, but it also develops a platform that can help study the behavior of other species, leading to more biology-driven robot development.

Chapter 1 – The small form factor of palm sized unmanned aerial vehicles (UAVs) combined with their ability to freely maneuver in 3D space with holonomic trajectories and carry custom sen-

sors makes them an ideal platform for autonomous source seeking in challenging environments. Equipped with the appropriate sensor, a small UAV could autonomously navigate towards light or heat sources such as forest fires or locate a radio-frequency (RF) transmitter attached to anything from a package in a warehouse to an animal tagged with a radio tracker. Leveraging small UAVs for this task however requires addressing their size weight, power, and computational constraints. While prior source seeking robots have used search strategies that require extensive training, such as reinforcement learning, we instead look to biology and employ a simple 'run and tumble' gradient following algorithm inspired by bacterial chemotaxis. The result is a computationally inexpensive approach requiring as little as 30 instructions/second, allowing this strategy to scale down to millimeter scale robots with small microcontrollers. Using insights from simulation, we report a success rate of 91% in real-time demonstrations of our UAV navigating towards a fire or light source while avoiding obstacles. Measurements from a small Bluetooth transmitter indicate it also produces a compatible gradient at ranges of 50–100 m. We conclude by discussing how this technique could scale down to sub-cm microrobots seeking RF power sources.

Chapter 2 – Constrained battery life on current Unmanned Aerial Vehicles (drones) limits the time they can operate and distance they can travel. We address this challenge by harvesting solar power to enable duty-cycled operation on a palm-sized drone. We present a scaling analysis that suggests that more solar power can be collected per unit mass of the drone as scale reduces, favoring small drones. By charging from the sun, the drone can operate for more than a single charging cycle, enabling extended mission time, and long-distance travel. To realize this, we design a high efficiency charging circuit and introduce two innovations. The first is a photovoltaic array that passively folds down while in flight to reduce air drag and automatically opens during landing due to the ground effect. The second is a sensor system and controller that autonomously finds suitable charging sites that are flat and well-lit. The drone can be fully charged in 3 hrs using the solar array and charging circuit with an average efficiency of 90.84%. Each charge enables a 4.7 min flight, allowing the drone to travel up to 1.2 km in a day. We also discuss how this platform could be used

to take periodic measurements for smart agriculture or wildlife tracking, rapidly deploy wireless networks, or deploy microrobots in the future.

Chapter 3 – Lightweight sensors coupled with wireless chips can help understand insect behavior. We first program an existing bluetooth enabled platform attached to an invasive Asian hornet species, *Vespa mandarinia*, to transmit sensor measurements to nearby receivers and also store them on-board. We demonstrate the utility of the system by tagging a live hornet and collecting data from it. We also design and fabricate another platform with a VHF radio and a microcontroller. This platform makes it easier to integrate new components and test application programs while still weighing just 113 mg and can be used for a wide variety of environmental sensing applications.

TABLE OF CONTENTS

LIST OF FIGURES

Figure Number Page

LIST OF TABLES

GLOSSARY

UAV: Unmanned Aerial Vehicle

RF: Radio Frequency

SWAP: Size, weight and power

FSM: Finite State Machine

ADC: Analog-to-Digital Converter

I^2C: Inter-Integrated Circuit

RSSI: Received Signal Strength Index

UART: Universal Asynchronous Receiver-Transmitter

IMU: Inertial Measurement Unit

MCU: Microcontroller Unit

PV CELL: Photovoltaic cell

IOT: Internet of Things

MPPT: Maximum Power Point Tracking

LIPO BATTERY: Lithium Polymer battery

MOSFET: Metal Oxide Semiconductor Field Effect Transistor

PFET: p-type Field Effect Transistor

LIDAR: Light Detection and Ranging

SD CARD: Secure Digital card

LP-WAN: Low Power Wide Area Network

VOC SENSOR: Volatile Organic Compounds sensor

LORA CHIPSET: Long Range chipset

TWI: Two Wire Interface

VHF: Very High Frequency

DPSS LASER: Diode-Pumped Solid State laser

USB: Universal Serial Bus

SWD: Serial Wire Debug

FPC: Flexible Printed Circuit

ACKNOWLEDGMENTS

Firstly, I would like to thank Prof. Sawyer Fuller, Prof. Vikram Iyer and Prof. Jaehyun Chung for serving on my thesis committee. I am particularly grateful to prof. Fuller and prof. Iyer for their constant guidance and support during this research. Their help and advice has had a pivotal impact on this work.

I would also like to thank all collaborators and co-authors of papers from this research: Prof. Sawyer Fuller, Prof. Vikram Iyer, Suvesha Chandrasekaran, Melanie Anderson, Yash Talwekar, Krishna Balasubramanian and Jessica Noe.

Special thanks are due to all members from the Autonomous Insect Robotics lab, specially to Yogesh Chukewad, Johannes James and Daksh Dhingra for insightful discussions and assistance with experiments. I also want to thank the Air Force Office of Scientific Research (AFOSR), grant no. FA9550-14-1-0398 by The Air Force Center of Excellence on Nature-Inspired Flight Technologies and Ideas (NIFTI) for providing the drone hardware used in this research and Chris Looney from Washington State Department of Agriculture for help with capturing hornets for tagging.

Finally I want to express gratitude to my friends and family for being a constant source of positivity and support, which is responsible for not just motivating but also directly enabling me to become the person I am today.

Chapter 1

BIOLOGY-INSPIRED SOURCE SEEKING AND OBSTACLE AVOIDANCE ON A PALM SIZED DRONE

Authors: Nishant Elkunchwar[1], Vikram Iyer[2], Melanie Anderson[1], Krishna Balasubramanian[1], Jessica Noe[1], Yash Talwekar[1], Sawyer B. Fuller[1]

1.1 Introduction

Unmanned Aerial Vehicles (UAVs) are versatile, with applications ranging from rescue operations [5] to toxic algal bloom detection [6]. In addition to surveying an area by systematically traversing it, UAVs augmented with sensing capabilities could intelligently and autonomously navigate towards points of interest to locate a signal source. Consider the concrete problem of using a UAV to find a forest fire: fires, especially large ones, produce significant amounts of heat and light, therefore creating a gradient of these quantities that can be measured at a distance. A drone with a temperature or light sensor could follow this gradient to autonomously locate the fire. Similarly, radiofrequency (RF) emissions can be detected at long distances. Recent advancements in miniaturization of radio tags [7] present the potential to turn any object into a signal source that could be measured at a distance. This includes locating trackers attached to keys, packages in warehouses, avalanche beacons [8], or even radio tracking of invasive species like giant hornets in a forest [9, 10].

Realizing these applications, however, requires addressing a number of practical and technical challenges. First, many of these application scenarios such as those inside a building or in a forest

[1]Department of Mechanical Engineering, University of Washington

[2]Paul G. Allen School of Computer Science and Engineering, University of Washington

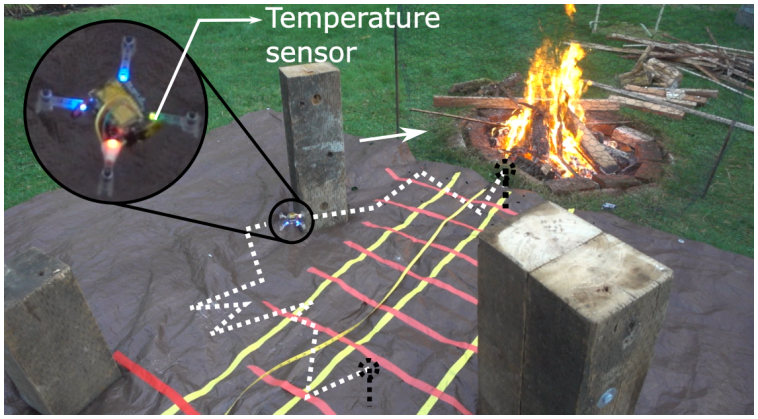


Figure 1.1: Our palm-sized drone uses a biology-inspired algorithm while avoiding obstacles to reach a source such as a fire, RF emission, or light source.

require operating in a cluttered airspace full of obstacles. This becomes difficult for large UAVs as their size prevents them from maneuvering in these environments. Recently developed palm-sized drones [11] are ideal for these scenarios as they are small enough to navigate around obstacles, or fly over them when needed by leveraging their ability to maneuver in 3D space which overcomes a significant limitation of ground-based vehicles.

Scaling down to smaller drones, however, introduces a number of technical challenges as they pose significant size, weight, and power (SWaP) constraints. In addition to limits on the sensors, and flight time, the SWaP constraints require many of these platforms to use small microcontrollers with limited computational capabilities. Therefore, it is imperative to find a computationally efficient and robust means to search for sources and avoid obstacles in complex environments.

If we look to nature however, we observe that even the simplest organisms can perform source seeking behaviors. For example light source attraction through positive phototaxis is a well-known phenomenon found in a broad spectrum of nocturnal organisms [12]; similarly chemotaxis, or the biasing of movement towards environments that contain higher concentrations of beneficial chemicals, is observed in even unicellular organisms such as bacteria [13]. In this work we leverage these insights to utilize a simple, bio-inspired algorithm and design an end-to-end system that guides a palm-sized drone towards a signal source while avoiding obstacles.

While recent work [14] has demonstrated successful light source seeking using reinforcement learning, this approach has a number of shortcomings for our target applications in complex and unknown environments. This approach cannot be generalized because the control policy generated is limited by the training data; the robot cannot seek a source in the center of a room as it never experienced that scenario. Similarly, scenarios without obstacles produced roundabout trajectories due to the robot's tendency to avoid the center of the room and it was not able to navigate around closely spaced obstacles. Moreover, this algorithm has a high computational overhead, making it difficult to scale to SWaP limited platforms.

We instead examine less complex stochastic methods. Prior work has explored the idea of stochastic source seeking [15, 16, 17, 18]; while these works provide mathematical guarantees for convergence, they have not been implemented on real hardware and do not incorporate obstacle avoidance which is essential in cluttered environments. Other works have also analyzed bio-inspired source seeking algorithms [19, 20, 21], but are also limited to simulation. In contrast, we seek to implement a source seeking system that can run in real-time with a palm-sized UAV in different environments.

While others have attempted tasks like RF source localization[22], the use of ground robots limits it to a nonholonomic action space. Additionally, source seeking using multiple robots has been performed in [23, 24], but to optimize for SWaP constraints we focus on a single robot design which does not require other infrastructure or communication.

To develop our bio-inspired search algorithm we look to chemical sensing efforts using mobile robots which has been actively researched since the 1990s, with a focus on gas source localization. A number of bio-inspired algorithms have been developed, mimicking the search procedures of animals like the silkworm moth *Bombyx mori*; however, these algorithms are designed for gas sensing in turbulent airflow[25]. For non-dynamic sources such as light, temperature, and radio signals, we can explore an even simpler class of strategies that rely solely on concentration gradients, such as iterative chemo-tropotaxis and Braitenberg vehicle strategies. These two approaches are feasible from a computational complexity perspective, but they require at least two sensors. To achieve a truly minimal approach which can scale to even smaller robots, we adopt the E. Coli

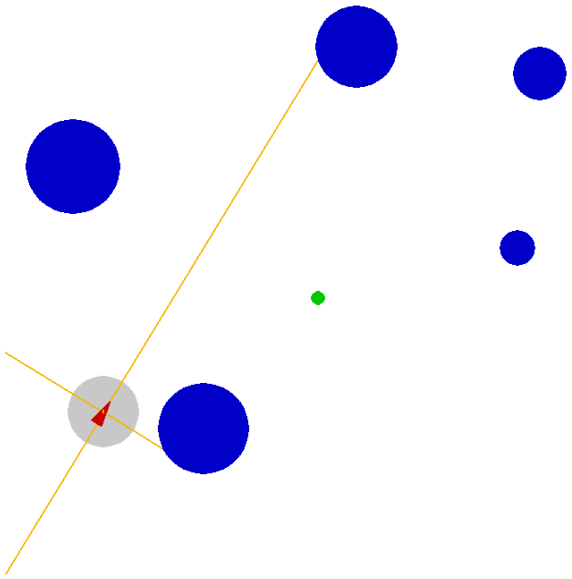


Figure 1.2: Screenshot of the simulation environment. The blue circles are randomly placed obstacles.

run-and-tumble algorithm which can be implemented with a single sensor [25].

In this paper we present an end-to-end implementation and demonstrate that this simple 'run and tumble' algorithm works robustly. The system successfully locates 91% of the tested heat and light sources including a fire in real time while avoiding obstacles as shown in Fig. 1.1 and video provided in the supplemental material. Additionally, we develop a simulation framework and take measurements of an RF source versus distance to demonstrate the potential for further applications.

The rest of the paper is organized as follows: Section 1.2 presents the algorithm for the source seeking along with detailed analysis of computational complexity. Section 1.3 presents the system components required for the study, followed by hardware experiments on the drone in Section 1.4. Section 1.5 concludes the results and discusses scalability.

1.2 Algorithm testing in simulation

We began designing and testing the algorithm in simulation using the Python library PyGame [26]. This allowed for refining of the algorithm while avoiding damage and wear on the hardware. The simulated environment consisted of five randomly placed circular obstacles of random size between 40 and 120 pixels in an arena measuring 700×700 pixels. We use a standard inverse square law model to model light propagation from a point source at the center. The inverse square law model was qualitatively verified in the real world (section 1.4). Simulated robot motion commands were designed to correspond to commands in the drone's control library. This facilitated the direct transfer of the algorithm from simulation to robot.

The laser distance sensors were also simulated to give distances to obstacles in the front, right, back and left directions of the robot. To avoid the robot leaving the arena, the screen boundaries were also considered obstacles for the simulated distance sensors. We implemented a behavior-based algorithm for source seeking. The robot performed three behaviors:

- **run**: move forward with a constant velocity,
- **tumble**: turn left or right to a random angle ranging from $0°$ to $180°$, and
- **avoid-obstacle**: move directly away from the closest detected obstacle along the direction of the distance sensor, and change heading slightly.

A simple finite state machine described in Algorithm 1 was designed [27] to switch between these behaviors. We implemented a tumble instead of strafing in a random direction for source seeking because the sensor needs to be pointing towards the source in applications like temperature seeking. In the case of obstacle avoidance, we simply strafed away from the obstacle, bypassing the need for a tumble. The robot declared reaching the light source when the intensity exceeded a threshold value set empirically based on the light source. The robot also successfully navigated narrow passages between closely placed obstacles. The simulation assumes:

- both sensor readings are noiseless

Algorithm 1: Source seeking algorithm with obstacle avoidance. Parameter values are as given in table 1.1

Data: Sensor readings (obtained asynchronously in experiment and synchronously in simulation)
Result: Control commands for source seeking

```
Function avoid-obstacle():
    move away from closest obstacle for ao_time;
    turn away from closest obstacle by ao_angle;
    start moving forward with fwd_velocity;
Function run():
    move forward with fwd_velocity for run_time;
Function tumble():
    if random(0,1) ≤ 0.5 then
        turn right by random(0, 180) degrees;
    else
        turn left by random(0, 180) degrees;
    start moving forward with fwd_velocity;
while signal_strength < stop_threshold do
    if min(distance_sensors) < obst_threshold then
        avoid-obstacle();
    else
        if signal_strength > prev_strength then
            run();
            if source is temperature then
                stop for 3 sec
        else
            tumble();
declare source and land;
```

Table 1.1: Parameter values for algorithm 1

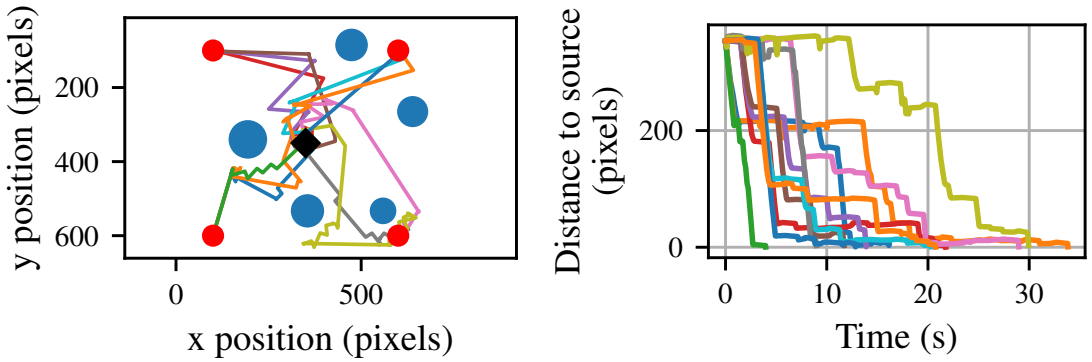
Parameter	Simulation	Light seeking	Heat seeking
`stop_threshold`	10^4	800 lux	13°C
`fwd_velocity`	90 pixels/sec	0.1 m/s	0.2 m/s
`ao_time`	100 ms	2 s	0.5 s
`ao_angle`	0.1°	20°	20°
`run_time`	10 ms	1 s	1 s
`obst_threshold`	40 pixels	0.5 m	0.35 m

- the light intensity follows an inverse squared model
- the simulated laser distance sensors had full range for the entire arena
- the obstacles did not cast any shadows

Despite the assumptions made during development of the algorithm in simulation, physical experiments demonstrated that the algorithm can perform robustly with real sources in the presence of noise. It was also shown that for light, heat, and bluetooth sources, the signal strength distribution was qualitatively a monotonically decreasing function with some noise.

Fig. 1.3 shows trajectories of the algorithm executing the search in a given environment from varying initial conditions. The distance to the obstacle on average decreased with time in all trajectories we simulated. Intermittent increases in distance arise when a random tumble results in a trajectory away from the source, or when an obstacle avoidance maneuver interrupts a path with decreasing distance. However, these intermittent increases in distance can be decreased by choosing a small time interval between successive iterations. In practice, this is limited by the robot's dynamics and an appropriate time was chosen in experiments. These results suggest that this algorithm is able to robustly perform source seeking.

Table 1.2 shows that increasing the threshold distance increases the seek time. However, in practice, a very small threshold may lead to collisions with obstacles. The chosen value depends



(a) (b)

Figure 1.3: (a) Multiple trajectories (different colors) with different starting points (red dots) at the same distance from the source (black diamond), (b) Variation of distance of the robot from the light source for the trajectories in fig. 1.3a with time.

`obst_threshold`	average seek time for 10 runs
20 pixels	6.54 s
40 pixels	11.10 s
80 pixels	13.74 s
100 pixels	22.88 s

Table 1.2: Seek times in simulation with different threshold distances for triggering obstacle avoidance behavior as per algorithm 1

on the robot size, velocity and latency of the algorithm loop. In simulation, the latency could be made very small to yield a very efficient algorithm. In experiments, the light and temperature sensors provided a measurement every 120 ms and 200 ms respectively, and the robot's dynamics preclude immediate changes in direction. Accordingly, we chose a threshold of 50 cm for light and 35 cm for temperature seeking to ensure that the robot was able to consistently avoid obstacles. We note the values in table 1.1 were chosen such that the algorithm works in a variety of scenarios and do not require tuning using a-priori knowledge about the signal distribution or environment. A turn command was also added to the `avoid-obstacle()` behavior since without this command, the robot could get stuck on obstacles in the front.

Next we analyze the required computation. At every time increment of 500 ms, first the moving average filter is updated with 6 instructions: subtract the oldest value in the queue from the running total, increment a memory pointer, add n to queue pointer (for an n-length moving average window), store new reading at the top of the queue, add new reading to the running total, and divide by the number of samples. Then the algorithm performs a comparison for source declaration, as well as four distance threshold comparisons and another field strength comparison for run and tumble. A single control command is issued if the action determined by the mentioned comparisons is `run()`. Actions corresponding to `tumble()` and `avoid-obstacle()` require two and three control commands, respectively. The `run()` and `avoid-obstacle()` routines also need one timer comparison each. This gives a total of 14 to 16 instructions per cycle, or 28 to 32 per second.

1.3 System components

1.3.1 Sensors

A BH1750FVI light intensity sensor (3 mm x 1.6 mm) was used to measure light intensity values. It consists of a photoresistor and an on-board 16-bit ADC integrated in a single chip that outputs digital intensity values. The sensor has a range of 1-65535 lux and was used in 1 lux resolution mode at 8 Hz. This sensor was mounted on the top of the drone facing up assuming a downward shining light source. For temperature sensing, a TI HDC2010 sensor (1.49mm x 1.49 mm) was used

which can measure temperatures ranging from $-40°C$ to $125°C$ with a 14 bit resolution at 5 Hz. The sensor was positioned facing the front of the drone. Both sensors use a standard I^2C interface. To measure Bluetooth signal strength, an evaluation module for the nRF52840 was attached to the drone and programmed to output its signal strength information (RSSI) over a UART link.

The Bitcraze multiranger deck was used to detect obstacles. It consists of five VL53L1X time of flight laser range finders on board with a range of 4 meters and a 50 Hz frequency. The sensors provide distances to lateral obstacles in four perpendicular directions as well as to overhead obstacles.

The Bitcraze optic flow deck was also used for navigation which consists of a PMW3901 optic flow sensor with a VL53L1X time of flight sensor. It provides the horizontal velocities and vertical position of the drone which can be fused with the on-board IMU to obtain 3D position.

1.3.2 Crazyflie quadcopter

A commercially available palm-sized drone platform called Crazyflie 2.1 by Bitcraze was used as the platform for implementing this algorithm. The robot weighs 27 g and measures 112 mm across. It is equipped with a Cortex-M4 microcontroller for onboard processing. The default firmware [28] was modified to include custom drivers for the light and temperature sensors to obtain sensor measurements.

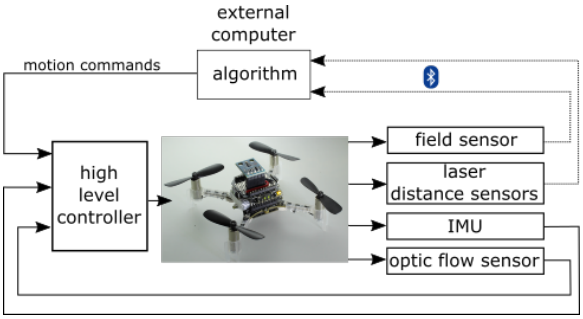


Figure 1.4: Block diagram showing interaction of system components. The external computer sent control commands as per algorithm 1

The drone is configured to send light intensity or temperature measurements, multiranger dis-

tances and its 3D position to an external computer using its onboard Bluetooth radio. Based on these sensor values, a Python program computed control commands to be sent back to the drone to be executed on-board (fig. 1.4).

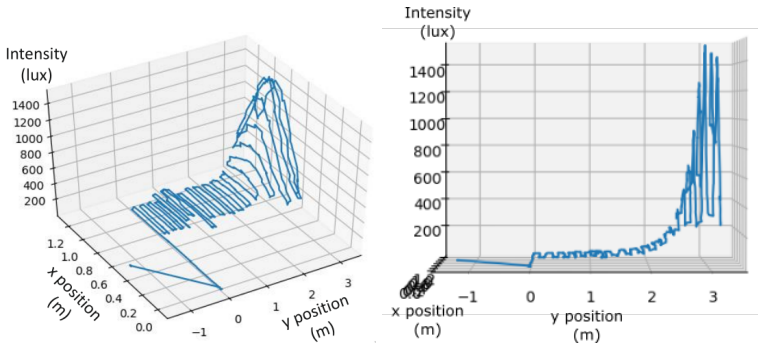
1.4 Experiments

(a) (b)

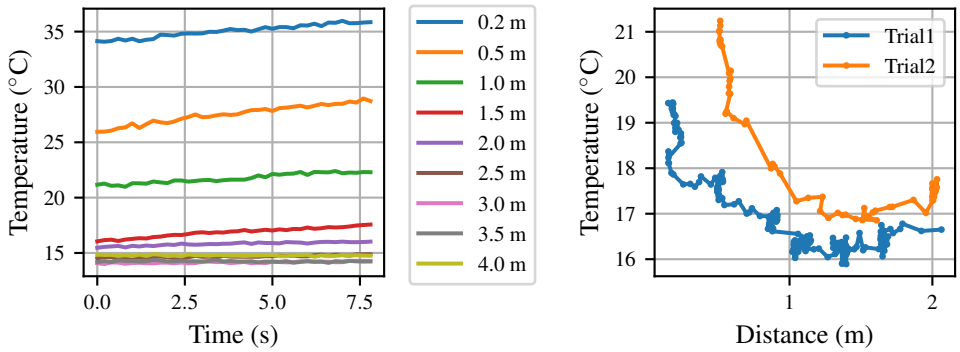
Figure 1.5: Experiment setup: (a) Arena for light source seeking, (b) Arena for temperature source seeking

Fig. 1.5 shows the arenas used for physical experiments. It measured approximately 8 m^2 and consisted of a single source and 2 to 3 obstacles in the robot's path.

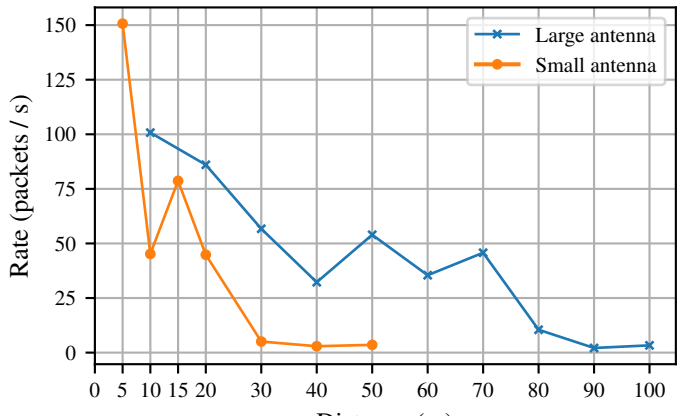
The light intensity distribution was characterized by manually flying the drone in a closely-spaced trajectory at a 0.3 m height and recording the intensity values (Fig. 1.6a (a)). The light intensity qualitatively follows an inverse square distribution with distance from the source. Fig. 1.6 (b) shows that the the temperature around a fire, measured using the same procedure, follows a similar distribution, with more noise. This field characterisation was used to determine the threshold value used to declare the source had been found. The top plot in Fig. 1.6b shows that the temperature sensor takes a few seconds to stabilise, so a stop command was added after every `run()` command during fire-seeking trials. Fig. 1.6b(c) shows measurements from two Bluetooth transmitters programmed to send packets periodically. The large antenna has a smartphone form factor; the small antenna has a light-weight insect-scale form factor [7]. We performed measurements in an open field with the transmitter and receiver each at a height of 1 m. We observed a large variance in Received Signal Strength Indicator (RSSI) caused by multipath fading. While a strategy of



(a) Light intensity distribution of an indoor light source using drone's position estimate. (left) with 2D position (right) along a straight line path towards the source.



(b) Temperature distribution for a fire using ground truth distance (left), and drone position estimate (right).



(c) Bluetooth source packet detection rate distribution with ground truth distance.

Figure 1.6: Signal distributions for light, temperature and radio packets

averaging combined with circular or side-to-side motion patterns could be used to mitigate this, we found that packet detection rate produced a gradient similar to those we observed for temperature and light.

The same algorithm as designed in simulation was implemented on the drone with parameter values given in table 1.1. The sensor measurements were obtained asynchronously at an external computer. All communication between the drone and the external computer was done via the on-board Bluetooth link. Fig. 1.4 shows the system components. To reduce sensor noise, field intensity readings were smoothed using a 10-element moving average filter for the light source. For the temperature source, the top plot in fig. 1.6b shows that the temperature sensor takes some time to stabilise. To address this, we added a stop command for 3 s after every run command and increased the moving average window to 30 samples. The robot moved in a 2D plane 0.3 m above ground.

The robot successfully localised the light source in 4 out of 4 trials and localised the fire in 6 out of 7 trials (one crash with an obstacle), giving a success rate of 100% and 85.71% respectively. Fig. 1.7 and Fig. 1.8 show the trajectories of the drone during the successful trials. The mean time to localize the light and temperature sources was 1 min 58 s and 1 min 13 s respectively. A video of these trials is available in the supplemental material.

1.5 Conclusion and scaling discussion

In this work we demonstrated that a biology-inspired search strategy can be used to locate a signal source using an aerial robot as small as the palm of a human hand. The system only utilizes feedback from sensors carried on-board the drone to navigate towards the source and avoid obstacles. This work is an important step toward practical application of such SWaP constrained robots toward source seeking operations such as fire localization.

Next we discuss how these results are applicable to more severely SWaP-constrained robots that are significantly smaller, measuring a few centimeters or less. The simulation software allows us to explore other signal propagation models. Such distributions could incorporate noise models or empirical measurements from physical environments of interest. This allows for rapid experimentation to inform the design of future source following robots and tuning of parameters such as

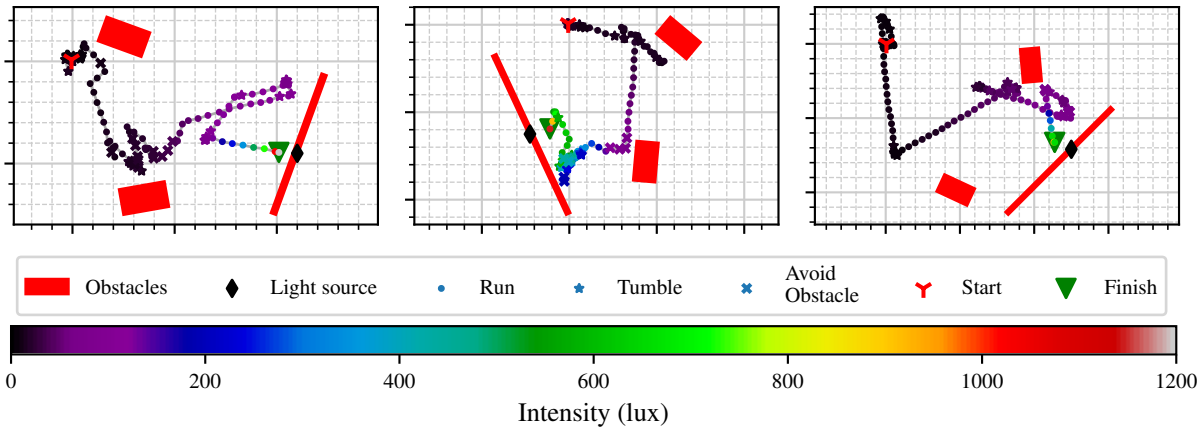


Figure 1.7: Drone path for light source seeking trials. The flight path was estimated using the drone's internal state estimator. Obstacle and source positions are approximate. Major gridlines are every 1 m and minor gridlines are every 20 cm.

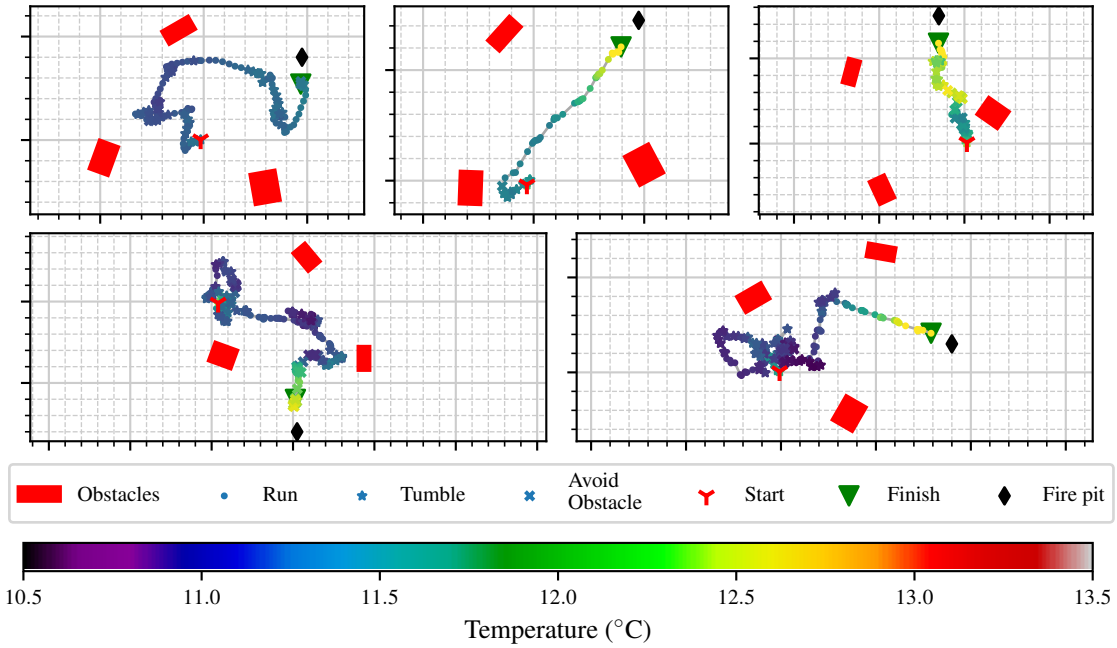


Figure 1.8: Drone path for heat source seeking trials (the source is a fire). The flight path was estimated using the drone's internal state estimator. Obstacle and source positions are approximate. Major gridlines are every 1 m and minor gridlines are every 20 cm.

the velocity, run time, or tumble angle.

More importantly, our bio-inspired algorithm has two key advantages that allow it to scale down to insect-scale robots. First, it only requires a single sensor, so the robot can be physically very small. A single phototransistor ($\approx$ 1 mg, 1 mm) can measure light [29]. Our fire seeking experiments used a small (3 mg, 1.49x1.49 mm) temperature sensor that has been used for insect-scale applications [30, 7]. Alternative sensors such as such as biological antennae for gas sensing could be used as well [31].

Second, the algorithm requires minimal computation, approximately 30 instructions per second. This constitutes a negligible fraction of the flight control system on our robot platform. Moreover, multiple insect-scale robot platforms have recently taken steps toward power autonomy [3, 32, 33]. As these efforts continue to progress, we anticipate a growing need for high level control algorithms that can allow these robots to perform useful tasks. Our bio-inspired run and tumble algorithm is an excellent candidate for this task as it can easily be implemented on the small microcontrollers used in these robots.

We emphasize that while building a further miniaturized robot is beyond the scope of this paper, the exact same sensor hardware (temperature/light sensors) and algorithm could be directly deployed on smaller robots. This technique could also help address the problem of powering small robots. For example, passive diode-detector circuits as shown in [30] can be used to measure the amplitude of an RF signal. If this were combined with the source seeking technique described in this work, it could allow a robot to come into close proximity to a RF power source such as a WiFi router and harvest emissions to recharge itself [34].

Future work could also improve on the algorithm itself. The current implementation restricts the run motion to 2D, but this could be expanded into 3D space as well. The algorithm can be made more efficient by obtaining directional information by fusing information from other sensors. While we focus here on a minimalist implementation, robots with cameras or multiple sensors placed apart could de-randomize the tumble direction and improve source seeking times.

Chapter 2

TOWARD BATTERY-FREE FLIGHT: DUTY CYCLED RECHARGING OF SMALL DRONES

Authors: Nishant Elkunchwar[1], Suvesha Chandrasekaran[1], Vikram Iyer[2], Sawyer B. Fuller[1]

(Paper accepted at IEEE IROS 2021)

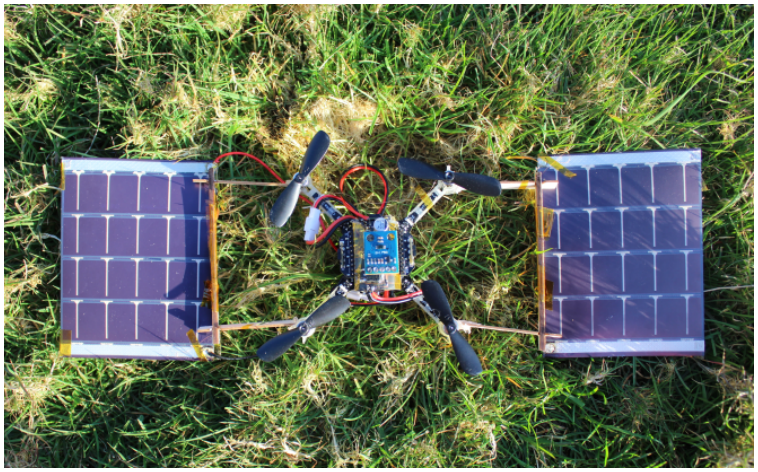


Figure 2.1: The palm-sized drone carries two lightweight solar panels, a maximum-power-point tracking charging circuit, and a sensor package that allows it to fly autonomously and detect suitable sites for charging.

2.1 Introduction

Unmanned Aerial Vehicles, also known as UAVs or drones, have begun to transform a variety of fields. They can enable rapid surveys of large areas for smart agriculture [35, 36, 37], wildlife conservation [38], and serve as a platform for carrying novel sensors [39], to name a few examples.

[1]Department of Mechanical Engineering, University of Washington

[2]Paul G. Allen School of Computer Science and Engineering, University of Washington

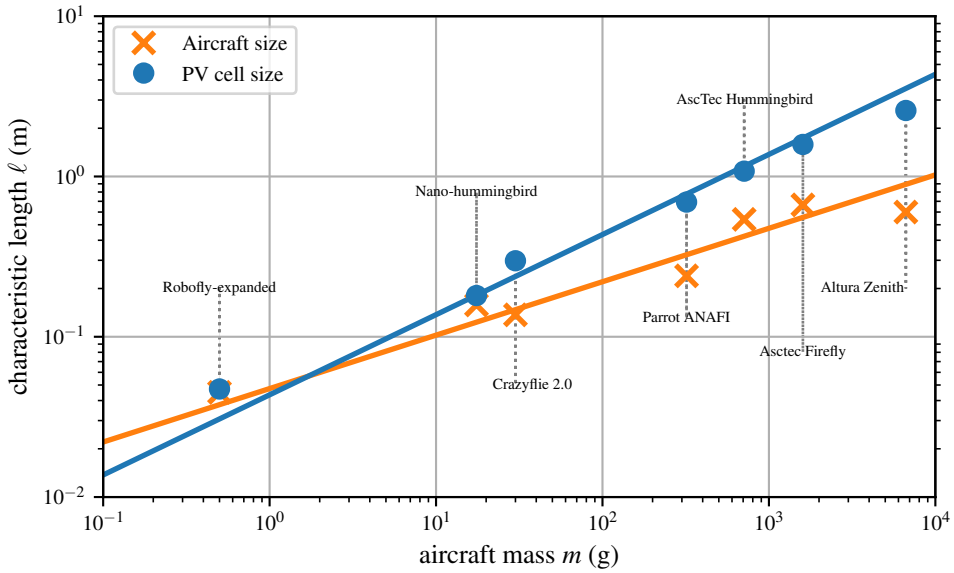


Figure 2.2: Photovoltaic (PV) powered hovering favors small aircraft because of physical scaling effects. PV size needed for continuous hovering flight in $\frac{1}{3}$ sun falls below robot size below about 1 g.

Current UAV platforms however have short operational lifetimes which limit their utility. This is due to the low energy density of available battery technologies and the high energy cost for flight. In this work we seek to address this challenge by using miniaturization to reduce the latter and using solar energy harvesting to recharge an on-board battery (Fig. 2.1), significantly increasing the drone's airtime [40].

To understand why solar power may be advantageous for small robots, consider the physics of scale as robot size diminishes. For a robot measuring ℓ from propeller tip to propeller tip, its "scale" is ℓ. A cube of length ℓ has surface area $6\ell^2$, volume ℓ^3, and mass $m = \rho\ell^3$ where ρ is its density. Neglecting constants to emphasize scale effects, we can say that area scales with ℓ^2 or area $\sim \ell^2$, and volume and mass $\sim \ell^3$. Solar power favors small scale because of its high surface-area-to-volume ratio: array area $\sim \ell^2$ while power needed varies with mass $\sim \ell^3$.[3] Therefore, the

[3]For small scale, power reduces slightly faster, making this conservative. Assuming inviscid flow, helicopter disc-loading theory [41] dictates that power $p = (mg)^{\frac{3}{2}}/\sqrt{2\rho A} \sim \ell^{3\frac{1}{2}}$, where g is the gravitational constant, ρ is air density, and $A \sim \ell^2$ is the area swept by the rotors.

Aircraft	Mass (g)	Power (W)	PV size (m)
Aerialtronics Zenith [43]	6650	666	2.58
AscTec Firefly [44, 45]	1600	251.03	1.58
AscTec Hummingbird [46]	710	116.55	1.08
Parrot ANAFI [47]	320	47.95	0.69
Crazyflie 2.0 [48]	30	8.88	0.30
Nano-hummingbird [49]	17.5	3.27	0.18
Robofly-expanded [2]	0.5	0.22	0.05

Table 2.1: PV cell size needed for continuous solar-powered hovering for various UAVs. Power requirements were estimated using battery capacity and flight time except for robofly-expanded and nano-hummingbird where the power requirements were provided in the respective papers. The Robofly mass is estimated for a configuration consisting of a 143 mg mechanism [2], 100 mg boost converter [3], a 100 mg battery [4], and a 150 mg sensor package

relative amount of power available from solar photovoltaics scales as ℓ^{-1}, favoring reduced scale. This trend can be observed in physical drones: If drone mass, size, and photovoltaic size required to power continuous hovering flight are plotted, they follow equivalent power laws (Fig. 2.2). To create this plot, we considered drones capable of sensor-autonomous hovering, assumed $\frac{1}{3}$ sun and 30% efficient cells.

Trend lines associated with these two scaling laws are given by $m = \alpha_2 \ell^2$ and $m = \alpha_3 \ell^3$. We fit free parameters α_1 and α_2 to mass m, characteristic length ℓ, and required photovoltaic array size ℓ_a (where ℓ_a is the length on one side of a square array) for all data points k according to $\alpha_2 = \text{ argmin } \Sigma_k \left(\log(\alpha_2 \ell_{ak}^2) - \log m_k\right)^2$ and $\alpha_3 = \text{ argmin } \Sigma_k \left(\log(\alpha_3 \ell_k^3) - \log m_k\right)^2$. The trend lines suggest that array size needed for continuous flight falls below vehicle size at around 1 g. For this analysis, we neglected photovoltaic mass because we anticipate that drones will be powered by ultra-thin photovoltaics that have been realized in the laboratory as thin as 1–10 μm [42].

We observe that while there still exists a gap between currently available drone platforms and the light-weight high-efficiency solar panels required for continuous flight, this is a similar challenge seen in battery-free internet of things (IoT) sensors. Radio communication often dominates the power budget of these devices, however many IoT radios support ultra low power sleep modes

that allow for duty-cycled operation at very low average power. In this work we take inspiration from this approach to design a system that uses solar recharging to enable duty-cycled operation of a palm-sized drone. We present an end-to-end system design for duty-cycled recharging including a solar power harvester and battery charging circuit, a passive hinge mechanism to allow the panels to fold up upon take-off and re-extend for charging upon landing, and a computationally efficient method for safe landing site detection. This approach allows for extended flights, periodic operation, and enables robust operation in a variety of light conditions.

Designing this system requires addressing a number of challenges. While miniaturization reduces the power required to fly, small drones such as the CrazyFlie also have a limited payload of 15 g. To operate within these constraints, we designed a lightweight solar cell array and harvesting circuit with maximum power point tracking (MPPT) that weighs 1.83 g. The charging circuit has an 90.84% average charging efficiency across varying light levels. We performed tests in real-world outdoor lighting conditions with varying sunlight intensity due to intermittent clouds and show that the battery can be fully recharged in 3 hrs.

In addition to minimizing weight, we must also consider the physical attachment of the solar cell to make sure the drone can still take off and perform flight control. Attaching large, flat solar arrays can affect airflow of the propellers and the robot's stability. To enable takeoff and controlled, stable flight, we design a light-weight mounting structure that enables maximal airflow and a passive hinge mechanism that allows the solar array to fold inwards on takeoff and deploy again upon landing.

To show end-to-end operation, we also consider the landing sequence required for recharging. We develop a computationally efficient means of surveying a landing site to enable safe automated landing to recharge upon reaching a low battery threshold. All code and circuit layout files developed in this work is openly accessible [50].

2.2 Related work

While tethered powering [51, 52] of UAVs can help mitigate the low battery capacity problem for surveillance in a small area, the drone's movement is severely restricted by the tether. Wireless power transfer [53] using stationary [54] or mobile [55, 56] base stations, as well as networks of base

stations [57], requires the drone to search for and align with charging pads, which necessitates bulky sensors and computational capabilities for autonomous drones or pre-computed fixed locations, in addition to planning for the mobile base stations [56]. Similar requirements arise for autonomously docking robots to electrical outlets for charging ground robots [58, 59] or hot-swapping UAV batteries [60]. Solar-assisted flight has been explored for larger drone platforms [61, 40] however their size (> 1.6 m) significantly increases their cost, and their size limits utility for many applications. Additionally these designs are fragile as they use large arrays of unpackaged monocrystalline solar cells, and are cannot perform agile maneuvers due to their large flat form factor.

While fixed-wing aircraft possess the appropriate flight power, size and weight to be continuously powered by existing solar panels [62], they cannot be used for applications that require hovering flight. A prototype for a fixed wing aircraft which can be reconfigured as a drone was developed in [63]; however, experimental validation of the system's transition between states was not demonstrated in-flight.

2.3 System design

Our end-to-end system to enable duty-cycled recharging of a drone consists of three main components: a solar array and charging circuit to harvest energy, an attachment mechanism and hinge to hold the solar cells, and a computationally efficient method for safe landing site detection that can run onboard the drone. We explain the design of each of these components in detail below.

2.3.1 Components

We design our platform around a 21 g palm-sized drone platform (Crazyflie 2.1 [11]). While this small drone lies slightly above the 1 g threshold for continuous solar powered flight (Section 2.1), augmenting it with solar power harvesting can enable periodic flights for sensing tasks, longer flight distances by pausing to recharge, and operation in the absence of power infrastructure.

To design this system we require a thin, lightweight photovoltaic array and charging circuit. To build this we use thin film, flexible. amorphous silicon cells (Powerfilm Inc.). The cells are

Component	Weight (g)
Drone	21.12
Two solar Panels [MPT4.8-75]	3.96
Mounting mechanism	0.83
Charging circuit	1.83
Battery	8.16
Optic flow deck	1.73
Multiranger deck	2.30
BH1750 light sensor	0.80
Total	40.73

Table 2.2: Weights of the components used in this work

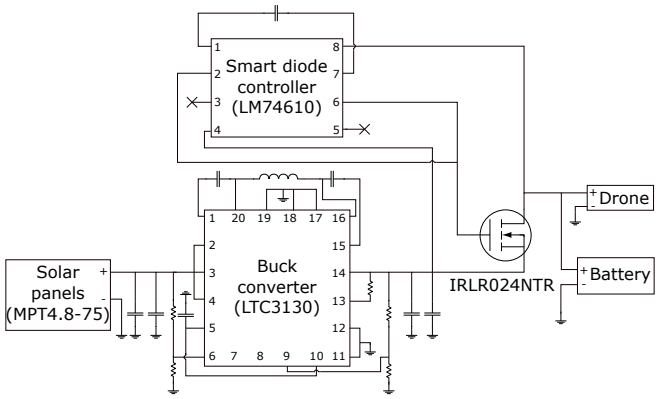


Figure 2.3: Charging circuit schematic

200 μm thick which makes them lightweight. Additionally, their flexibility makes them robust to damage. Each cell measures 94 mm $\times$ 73 mm and supplies power at up to 0.13 W/g in 1 sun. Each panel is rated for an operating current of 50 mA, an operating voltage of 4.8 V, and an open-circuit voltage of 7.4 V. Combining two of these cells would give a maximum power of 0.48 W and allow for recharging the drone in 3 hrs.

The output of solar cells however varies with amount of sunlight available, and non-ideal power sources like solar panels have characteristic voltage-current curves [64]. Extracting maximum power from a solar cell requires controlling its voltage by actively varying the load, in this case

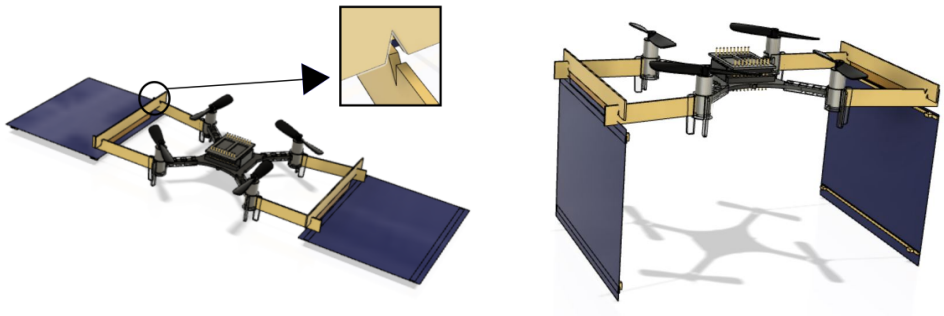
the battery charging rate. Additionally, the solar cells output 4.8 V while a fully charged LiPo battery has a maximum voltage of 4.2 V. To optimize battery charging, we design a harvesting circuit that achieves both voltage conversion and load variation for maximum power point tracking (MPPT).

We design this circuit using an Analog Devices LTC3130 buck converter which can tolerate input voltages up to 25 V which covers the full output range of the solar cell, and it can be configured to perform MPPT. The high voltage allows for a number of potential solar cell configurations of different sizes in series or parallel. Because our design uses two cells which will be placed flat on either side of the drone and will likely operate in similar light conditions, we connect them in parallel to allow the MPPT controller to regulate them to the same voltage.

While the buck converter chip produces an output capable of charging the battery, in low lighting conditions when the solar cell output falls too low, we need a solution to prevent battery discharge. To protect against this undervoltage condition, we added an N-channel MOSFET (IRLR024NTR, International Rectifier) with a smart diode controller (LM74610, Texas instruments) to function as an ideal diode. This overcomes the disadvantage of forward conduction voltage drop in case of Schottky diodes and also the addressees the inefficiency of PFETs at handling high load current at low input voltages. No overcurrent protection was required since the maximum possible current from our solar array is 200 mA, which is significantly lower than the maximum 2 C charging rate allowed for the 240 mA battery on the drone. Fig. 2.3 shows the schematic of the full charging circuit. The complete fabricated circuit measures 39 mm $\times$ 21 mm and weighs 1.83 g. Table 2.2 shows the all of the components used in this work and their corresponding weights.

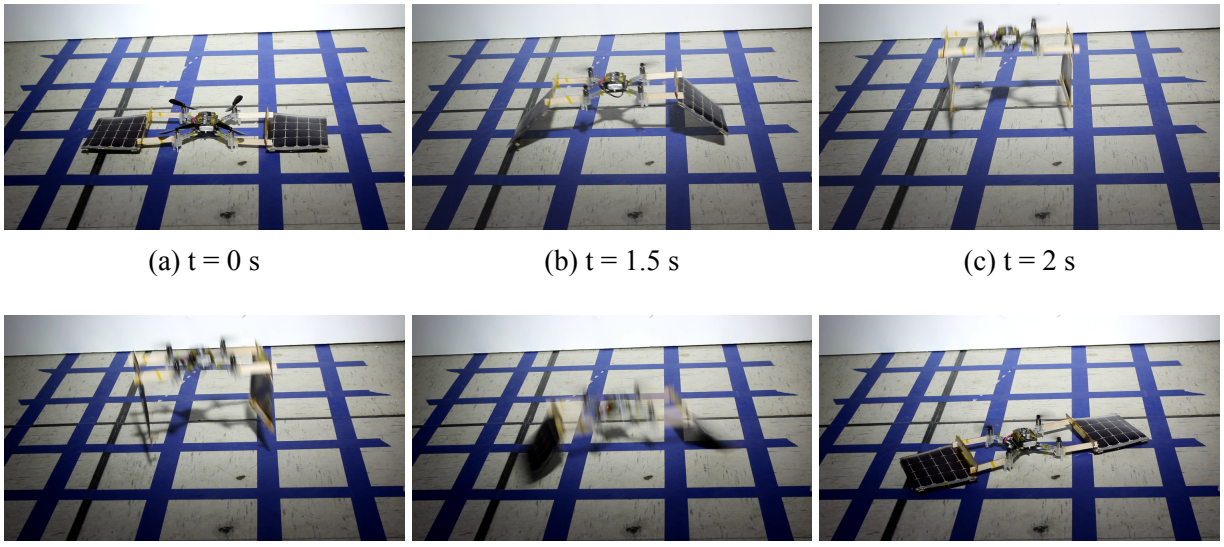
2.3.2 Folding solar panel mounts

In addition to designing a charging circuit to harvest power, our solar array and charging circuit need to be mounted to the drone as well. This introduces a number of challenges. We observed that if the cells are placed above or too close to the propellers, they significantly reduce airflow and therefore lift. One solution is to create a frame that holds the panels further away from the propellers. We observed however that mounting a mass this far from the center of the robot causes



(a) (b)

Figure 2.4: Solar panel support structure and mechanism by which they fold downward during flight, mounted on the Crazyflie model [1]. (a) On the ground, the panels lay flat to collect the sun. Inset shows press-fit mechanism used to interlock balsa support structure. (b) Upon takeoff, flexure-based hinges allow the panels to fold downward, facilitating stable flight by reducing the drone's moment of inertia.



(a) t = 0 s (b) t = 1.5 s (c) t = 2 s

(d) t = 2 s (e) t = 2.4 s (f) t = 2.6 s

Figure 2.5: A sequence of frames taken from a video of a single flight in which the drone took off and then subsequently landed after performing a short automated maneuver in the air. (a–c) During takeoff, the photovoltaic panels fold down passively. (d–f) During landing, the panels fold out automatically as well due to aerodynamic pressure under the drone due to the ground effect. Note that the landing sequence occurs over a much shorter period of time.

instability during takeoff. Although the mass is within the payload specs of the drone, in a series of 12 tests it was only able to successfully lift off 25% of the time (see results). Additionally, we observed that the drone was unstable and quickly crashes.

To address this we designed a passive hinge mechanism that allowed the solar panels to fold downward as soon as the drone lifts off. Folding in this way reduces the horizontal surface area and hence the total drag. In addition to folding inward, the panels passively fold back out when the robot lands. We designed the mechanism so the robot could take off at 70% thrust. This was calculated after accounting for the weight of the circuitry, solar array, and the recommended payload capacity of 15 g of the drone.

We mounted two solar panels to the sides of the drone, using lightweight beams to hold them 4.5 cm away from the drone to avoid blocking the the downwash of the propellers. The beams were constructed from 2.3 mm balsa and machined using a CO_2 laser. They were bonded to the drone's motor sleeves using hot-melt glue (ethyl vinyl acetate). We created a flexure joint between the balsa structure and the PV array using adhesive-backed polyimide film (Kapton). This hinge was placed as high as possible to minimize the vertical distance between the center of pressure of the folded PV panels and the drone's center of mass. This minimizes the torque that occurs due to drag as the drone moves laterally.

The complete panel mounting solution is shown in Figure 2.4. The beams were interlocked using a press fit into a small V-notch inspired by techniques in wood-working that leverage the pliability of the wood to create press-fit joints without the need for fasteners. We then reinforced the joints with a thin layer of cyanoacrylate glue. We added small feet-like projections to the base of the distal ends of the panels to ensure they were level when the drone is on the ground.

To open the panels when the drone lands, we take advantage of the ground effect, which is an increase in air pressure that occurs below the drone when it is near the ground. This causes the panels to push outward as the drone descends, so that they lie flat upon landing.

2.3.3 Safe Landing Procedure

The section above describes a mechanical design that enables safe liftoff and flight with the solar array, however because the solar cells are folded inward and the flight time is shorter than the charging time, our system also requires a safe landing strategy. Before landing the drone must be able to detect obstacles at the landing site.

A flat terrain check for drone landing has previously been demonstrated using a camera and LIDAR [65, 66], however these systems required components that are far too heavy for a palm-sized drone. Moreover, these methods are computationally intensive.

Our solution instead consists of a simple strategy of collecting distance measurements using a downward facing VL53L1X time-of-flight rangefinding sensor already present on the drone. The drone performs a small maneuver over the ground and decides whether the terrain is safe by analyzing the standard deviation of those measurements. [67] uses a similar method but requires human operators to select the threshold. We chose a fixed threshold of 15 mm based on experiments performed on various surfaces.

Additionally to maximize the charging rate we would like to land the drone in a location with a minimum threshold light intensity, so that it avoids shaded areas. The threshold was chosen to be 1000 lux for indoor experiments and 100,000 lux (1 sun) for outdoor experiments. To do this, we use a light intensity sensor (BH1750, ROHM Semiconductor) along with a driver developed in [68]. We choose a final landing site that has a light intensity above the desired threshold as well as one that is determined to be flat as per the standard deviation of the downward-facing time of flight sensor measurements.

2.4 Results

The video accompanying the paper shows the flight performance and landing site selection algorithm. We evaluate each component of our system below.

Solar charging. To evaluate the solar cells and charging circuit described in section 2.3 to charge the drone's LiPo battery, we placed the solar array and charging circuit flat on the ground outdoors

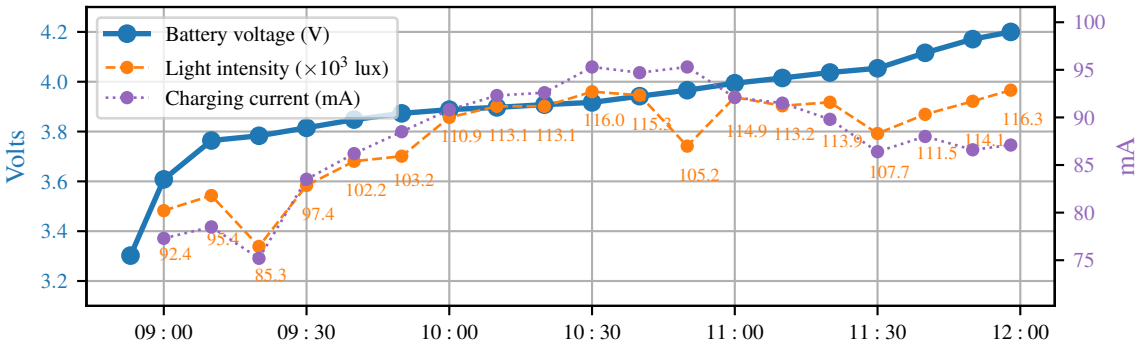


Figure 2.6: Battery voltage and charging current variation with light intensity for a complete charge cycle. The first datapoint is open-circuit voltage, rest are while battery is connected and charging

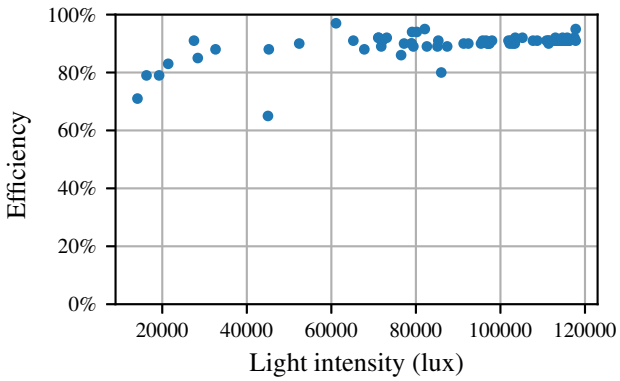


Figure 2.7: Efficiency of the charging circuit varies with received sunlight intensity. The circuit is more efficient for higher amount of light

under sunlight with light intensity varying with the time of day and presence of clouds to measure performance in real world light conditions. We measured the current and voltage at the input and output of the charging circuit using digital multimeters (Fluke 287) and measured light intensity using a digital lux meter (Fluke 941) placed next to the panels, every 5 min. While data was collected manually for these experiments future versions could integrate voltage, current and light sensing with the drone telemetry. Fig. 2.6 shows the battery voltage, charging current output to the battery and sunlight intensity variation for one complete charge. The total charging time for this test was 3 hrs 3 min. Using the input and output power measurements, we calculate the charging circuit's efficiency shown in Fig. 2.7. While the efficiency varies with light intensity, it is consistently above 60%, with an average 90.84% for the experiment. We note that solar cell output scales non-linearly with light intensity, and expect the charging time to be even lower for bright, sunny days due to more light as well as increased charging efficiency at higher light levels (fig. 2.7.) The maximum recorded light intensity over the experiment was 117800 lux and the average intensity was 107162 lux (just over 1 sun).

We also evaluated different solar array sizes to provide empirical measurements of the scaling trends described in Fig. 2.2 and demonstrate potential solutions for drones with higher payload capacity. We performed these experiments in real outdoor lighting conditions, tabulating the average light intensity along with these results in Table 2.3. The results show the charging time could be decreased to less than 1.5 hrs. This suggests a higher duty cycle is possible on clear days. The scaling when doubling the array size by using four MPT4.8-75 cells is approximately linear as expected. We note, however, that charging with the larger MPT6-75 cells was measured under lower light due to weather constraints resulting in only a small decrease in charging time.

Flight performance. We next evaluated the drone's ability to take off while carrying the PV cells. As shown in Fig. 2.5, the drone is able to lift off and the cells fold downward. We performed 12 flights on a drone equipped with the hinge-based panel folding mechanism. We observed consistent, stable flight and autonomous maneuvers in all trials at speeds of up to 1.4 m/s. To determine if folding is needed, we performed 12 additional flights on a drone in which the cells held rigidly outward. The drone was only able to successfully take off in 3 out of 12 trials (25% success rate).

Panels	Mass (g)	Avg intensity ($\times 10^3$ lux)	Avg efficiency	Charge time (min)
two MPT4.8-75	3.96	107.16	90.84%	183
four MPT4.8-75	7.92	71.41	89.61%	84
four MPT6-75	9.20	65.74	86.94%	80

Table 2.3: Time taken to recharge a battery from 3.3 V to 4.2 V for different cell configurations and lighting conditions

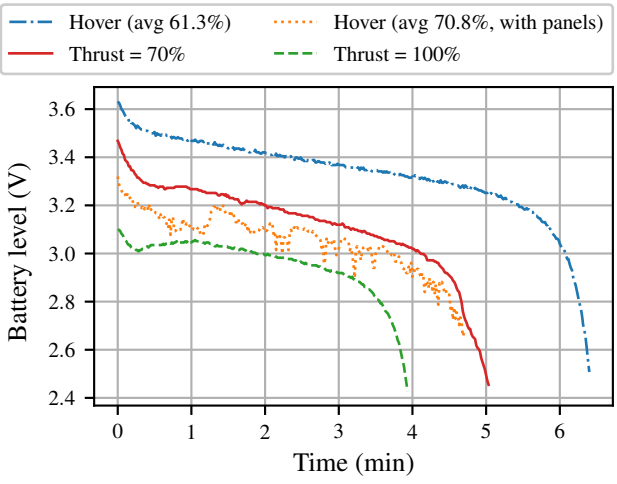


Figure 2.8: Plot of battery voltage with thrust levels

This is due to disruption of airflow and higher drag when panels are oriented laterally under the rotors as well as the large torques caused by mounting panels away from the center of mass which disrupt the controller and cause instability.

To estimate total flight time of the drone, we recorded the battery voltage starting from full charge for two fixed thrust levels as well as the thrust required to hover with and without the solar array. Data was recorded using the python library provided by the drone manufacturer [69] which also contains logging functionality. Fig. 2.8 shows the resulting discharge curves. At the hover thrust required when lifting, the drone can fly for about 4.7 min. When flying at maximum velocity this enables flight distances of approximately 400 m.

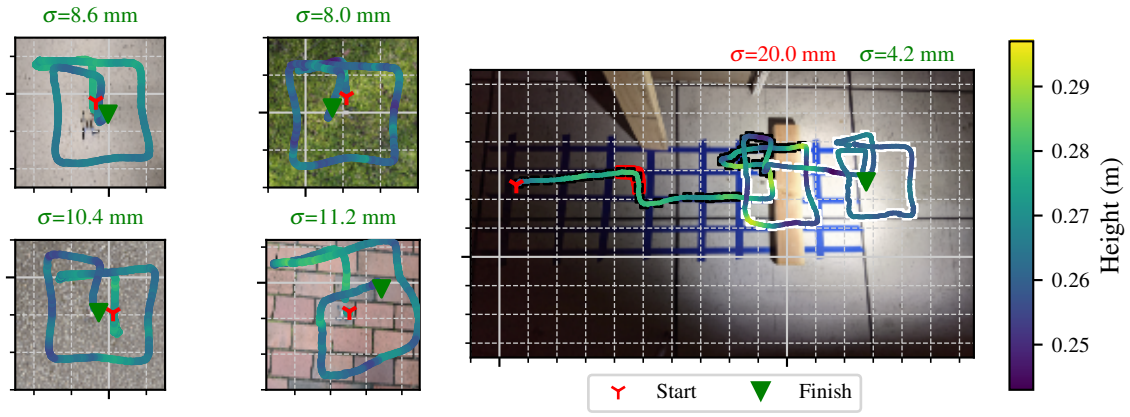


Figure 2.9: (left) Measured flatness over various surfaces. Moving clockwise from the top the surfaces are: terrazzo tiles, short grass, concrete paver tiles, gravel. (right) Automated landing site suitability detection. The drone searches for a site with enough light (a black background under the trajectory indicates insufficient light, < 1000 lux) that is flat. It also avoids nearby obstacles by detecting them with laterally-oriented laser rangefinders (red background under the trajectory). Once the light intensity is greater than the chosen threshold (white background), the drone moves in a square trajectory around the area to be tested for flatness. If the standard deviation of the time of flight measurements over the square is greater than a heuristic threshold, it keeps searching for another landing area. Finally, the drone lands when it has found a suitable site.

Landing site selection. We demonstrated that it is possible for the drone to use its on-board sensor suite, plus a light sensor, to detect suitable landing sites. Fig. 2.9 shows an example of our algorithm in operation. Our system detects whether there is light present and whether the terrain is flat enough

to ensure that the drone can take off again and the panels will lie flat. Fig. 2.9 shows example flatness test trajectories on different terrain such as grass and rocks, including an example trial in which the drone automatically rejected two sites because they were too dark or had uneven terrain (wooden block placed as an obstacle) . To initiate the autonomous landing sequence, a conservative threshold of 2.8 V was used based on the discharge curve at 100% thrust (fig. 2.8). The drone records battery voltage on-board. Since any non-zero thrust induces a voltage sag, choosing a threshold based on the 100% thrust curve ensures that transient voltage sags due to thrust changes during hovering do not cause an accidental early landing decision. Another strategy to minimize transients can be to calculate a moving average of the last few voltage measurements.

The flat site selection process described in this work was designed considering the limited payload, power and computational capabilities of the platform and hence has some limitations. Firstly, since flatness is measured with a downward facing time-of-flight sensor, any factors that might cause changes in the drone's height (for example payload near maximum payload limit, low battery, etc) will be registered as unflat terrain. Secondly, while executing a reference trajectory like a square used in this work, the localization estimate is obtained from the drone's estimator based on measurements from an inertial measurement unit, an optic flow sensor and a downward-facing range finder. This position estimate performs poorly on plain, featureless surfaces, or surfaces with repetition as can be seen in the case of concrete paver tiles in fig. 2.9. For better location estimation, ultra-wide bandwidth based location estimation [70], SteamVR based localization [71] and MotionCapture localization [72] are supported by the drone manufacturer with the platform firmware [28].

2.5 Conclusion

This paper presents a drone recharging system using thin, lightweight solar panels, enabling duty-cycled operation without any human intervention. This approach avoids the need for a complex autonomous battery replacement system and associated planning and docking. We demonstrate an end-to-end design including a 1.83 g, 90.84% efficient power harvesting circuit, passive folding mechanism to enable stable flight with the solar array, and landing site selection algorithm. The

solar panels and mounting mechanism constitute about 11.7% of the total mass of the platform. We show this system can fully charge the drone within 3 hrs 3 min in real world light conditions to enable a 4.7 min flight.

Our results support two main conclusions. The first is that solar power favors small size. And the second is that it is possible, though appropriate design of a deployable photovoltaic array, charging circuit, and landing site selection, for small robots to operate for much longer than is possible on a single battery charge. This serves as an important milestone toward battery-free, indefinite-time flight for aerial robots.

A number of advancements will facilitate drones with indefinite flight time. The first is to integrate more advanced photovoltaic cells. For example, 2.5% duty cycle demonstrated in this work could be dramatically increased by using technologies such as triple-junction Gallium Arsenide cells (e.g. MicroLink devices) which can exceed 30% efficiency and are only 40 μ m thick. Such cells provide a power density exceeding 1 W/g, which is greater than 7$\times$ that of the cells used in this work. This and future advances in photovoltaics allow us to move along and adjust the PV cell size line shown in Fig. 2.2. Second, there is room to improve the efficiency of the vehicle such as by using brushless motors and a lighter airframe. Lastly, our results suggest that a move to still smaller drones such as the insect-sized 150 mg UW Robofly [2] will allow operation at the left side of the intersection of the two lines in Fig. 2.2. In that regime, it may be possible to fly indefinitely with a PV array that is smaller than the physical dimension of the robot.

The system we demonstrated here could be used for a variety of applications in its current state. Specifically, solar power harvesting can enable long distance flights and remote operation with zero power infrastructure. We highlight three potential applications below.

Smart agriculture and aerial surveys. Emerging smart agriculture technologies such as Microsoft FarmBeats [35] requires capturing aerial images of a farm. This system would allow a single, low-cost drone to periodically fly and survey a field without any infrastructure, enabling use in a variety of environments including in developing countries with unreliable power. Additionally, the drone could either log data to an SD card [73], or transmit it using low-power wide-area network (LP-WAN) technologies. These radios can transmit up to 17 km, weigh 613 mg, and can consume tens

of milliwatts on average, which is significantly lower than flight [74]. This system would allow a drone to travel 6 times the distance with two 3-hour recharges (typical in one day). Flying at its default maximum velocity of 1.4 m/s the drone could travel about 1.2 km per day, which would allow it to traverse large farms or perform persistent aerial surveys needed for wildlife tracking [38]. Between flights the system can also double as a traditional IoT sensor node. Many environmental sensors are lightweight (< 1.5 g), and low power (< 1 mA) allowing them to run with minimal impact on charging: a temperature / humidity / pressure / VOC sensor (Bosch BME680 [75], $\approx$17 mg, 3.7 µA), light sensors (On Semi LV0104CS [76], 1.4 mg, 70 µA, [77]), a CO_2 gas sensor (Sensirion SCD40 [78], 0.55 g, 0.5 mA), electrochemical gas sensors (SPEC Sensors 100-102 [79], 1.4 g, 10-50 µW), a camera [80] (24 mg, <2 mW), a pressure sensor (Bosch BMP388 [81], 6.3 mg, 3.4µA).

Wireless network deployment. This system could also be used for rapid deployment of a wireless network in an emergency scenario. For example, if drones were equipped with a LoRa chipset that can weigh as little as 613 mg [74], they could be programmed fly to different locations. Upon landing, the solar power could be used to power the radios.

Microrobot deployment. This platform could also be used to enable deployment of microrobots in the future. For example, the drone could fly to a remote location and deploy one or a number of terrestrial or aerial microrobots to explore the environment. The large drone could act as a power source to recharge the small robots using its comparatively large area solar panels and serve as a communication base station to relay messages.

Chapter 3

WIRELESS SENSOR TAGS FOR INSECT STUDIES AND ENVIRONMENTAL SENSING

Collaborators: Nishant Elkunchwar[1], Vikram Iyer[2], Chris Looney[3], Sawyer Fuller[1]

3.1 Introduction

Tagging and tracking various species of animals has been of interest in recent years. Experiments have been conducted to track radio-tagged fish [82], birds [83], hornets [10, 9], bats [84] or VHF collars suitable for use on a wide variety of animals [85].

Recent efforts have been made to track individual hornets from an invasive Asian hornet species, *Vespa mandarinia* [10], pictured in figure 3.1a, to their nests. This species of hornets is native to temperate and tropical Asia and is the largest hornet species in the world. These hornets pose a major threat to bee species native to the US. The Washington State Department of Agriculture published a call for the general public to report sightings of these insects to remove their nests [86]. The experiments conducted in [10] have led to the detection of one such nest [87]. This tracking was done by attaching a bluetooth chip and using large, directional antennas 3.1b to follow the hornet as it flies back to the nest based on signal strength of this attached tag. It was determined that an individual hornet can carry about 300-350 mg of payload.

The tag also carried sensors like temperature, light intensity and an accelerometer. Real time monitoring of measurements from these sensors can help in understanding whether the hornet is inside a nest. Being able to store these measurements for analysis at a later stage can be useful to

[1]Department of Mechanical Engineering, University of Washington

[2]Paul G. Allen School of Computer Science and Engineering, University of Washington

[3]Washington State Department of Agriculture

(a) (b)

Figure 3.1: (a) Asian giant hornet *Vespa mandarinia*, (b) Searching for hornet tagged with bluetooth chip (photo: Karla Salp/WSDA)

understand behaviors like foraging that individuals of this species display. Section 3.2 describes how this existing tag and a corresponding receiver was reprogrammed in this work to achieve this and how a successful experiment to collect such data was performed (section 3.2.1). To improve the range of transmission, ease of component integration and modularization of the platform, a newer version of this tag was fabricated as a part of this work, described in section 3.3.

3.2 Sensor tag with bluetooth radio

Figure 3.2 shows the bluetooth tag that was re-programmed in this work. The circuit board was a 25 micron sheet comprising of 12.5 micron layer of copper on top of a 12.5 micron layer of polyimide. The nRF52840 collected measurements from the three sensors over a Two Wire Interface (TWI) and transmitted them in a single advertising packet every 500 ms. A slow rate of transmission was chosen to conserve on-board battery power by setting the radio into sleep mode between transmissions. The 3-axis accelerometer (BMA400, Bosch Sensortec) provided (12 bits $\times$ 3) 36 bits which were bit-shifted and concatenated to a 5-byte word. Similarly, 2 bytes were obtained from the light sensor (LV0104CS, Onsemi Semiconductors) and 4 bytes from the temperature, pressure, humidity sensor (BME280, Bosch Sensortec). These were encoded into a 15 byte bluetooth packet in the manufacturer-specific advertising data packet type defined in the bluetooth

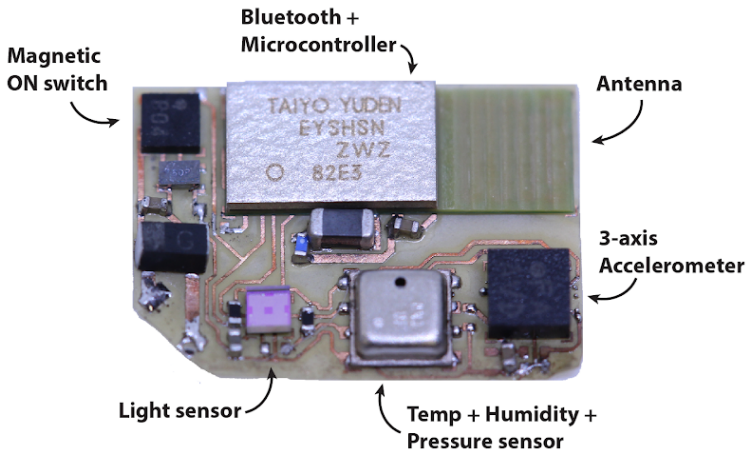


Figure 3.2: Bluetooth tag attached to the hornet with three sensors (weighs 83 mg)

specification. Each tag was identified with a custom MAC address. A lithium ion battery provided power to the circuit that lasted for about 12 hrs.

A corresponding receiver, LoPy4 (Pycom) was programmed to listen to bluetooth advertising packets from the list of custom MAC addresses of the tags. The received packets were stored in a comma-separated-value file format (csv) on an attached micro SD card along with Received Signal Strength Index (RSSI), transmitter, receiver identifiers, and a timestamp. The receiver was connected to a large lithium ion battery pack that enabled operation for about 24 hrs.

To ensure that sensor measurements collected when the hornet flies outside the receiver range were not lost, the bluetooth chip on the tag was also programmed to store these measurements to its on-board flash memory. With a bootloader and application code in the flash, 41 kB of memory was available for use. This amounts to about 2700 15-byte packets, or about 22 min worth of measurements at a 2 Hz packet broadcast rate. To record measurements over a longer period of time, only every 10th packet was stored to on-board flash memory. This amounts to about 3.6 hrs of sensor data. This information can be decoded once a tag is recovered from the hornet.

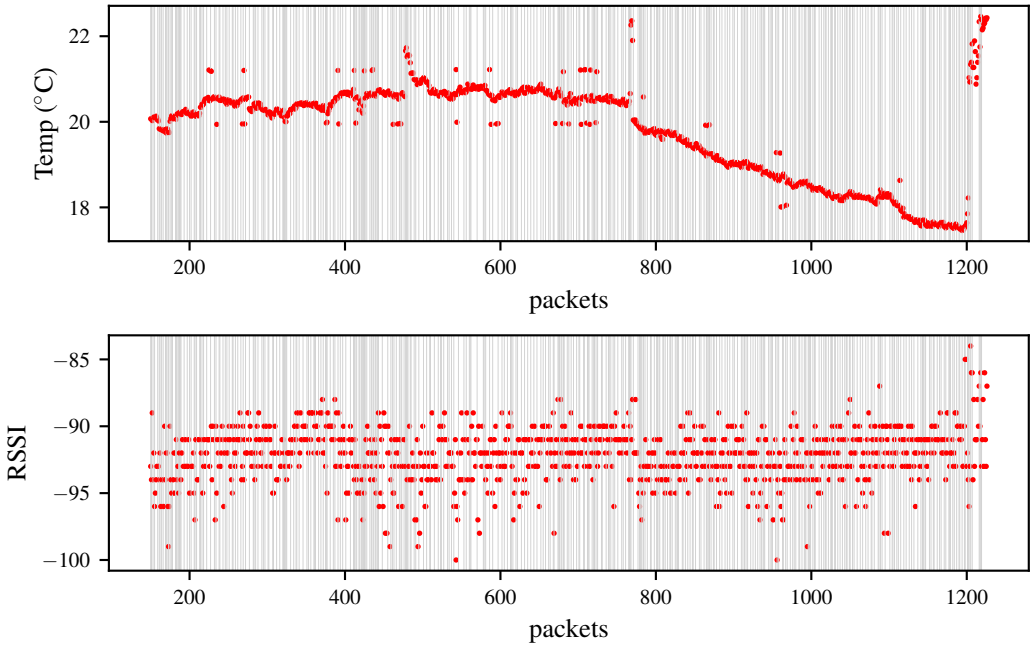


Figure 3.3: Temperature, signal strength measurements collected by a receiver. Vertical grey lines show time instants when the receiver restarted due to a power loss.

3.2.1 Data collection experiment

The tag was attached with a string tied between the abdomen and thorax of a previously captured hornet and it was allowed to fly back to a known nest. A receiver was placed near the nest and another receiver was placed a few meters away to store packets from hornets entering or leaving the nest. The setup was left for about 24 hrs before the nest was destroyed and hornets were captured. Figure 3.3 shows the measurements collected by the receiver closer to the nest over the course of the experiment. Due to power supply issues, the receiver restarted many times, resetting the timestamps at those instants.

Since determining time interval between two packets is only possible for packets arriving between two power resets of the receiver, it is difficult to draw inferences about the hornet's behavior from this dataset. However, subsequent experiments with a reliable power source might reveal in-

sights about an individual hornet's foraging time based on the combination of the amount of time spent outside of receiver range and temperature measurements which indicate whether the hornet is inside the nest. Other insights can be inferred from light intensity and acceleration measurements.

Measurements stored in the tag's on-board flash memory have not yet been decoded. These can also be useful assuming no resets of the transmitter.

3.3 General purpose and modular environmental sensing platform

A newer version of this tag was designed and fabricated to address a few shortcomings with the bluetooth tag. Firstly, to program or debug the application code on the microcontroller, or to read sensor measurements from a tag recovered from a re-captured hornet, it was necessary to solder / de-solder wires to the circuit which can be time consuming and error-prone. Although the on-board magnetic switch provided a convenient way to enable or disable battery discharge, it was still necessary to solder wires to be able to charge the battery. Moreover, bluetooth has a high path loss with distance in densely wooded areas and also has a very limited communication range.

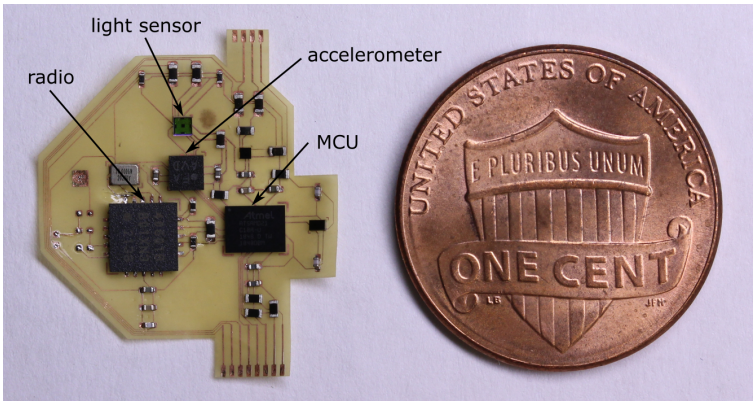


Figure 3.4: The fabricated circuit for an environmental sensing platform, weighs 86 mg. US penny for scale.

This new tag, shown in figure 3.4, uses a 433 MHz radio chip (SI4060, Silicon Labs) to enable a longer range transmission and a lower path loss in wooded environments. Similar to the 2.4 GHz frequency band used for bluetooth communication, the 433 MHz band is also unlicensed.

Moreover, Very High Frequency (VHF) radios are already being used for wildlife tracking purposes [9, 83, 84]. An SAMD microprocessor (ATSAMD21G18A, Microchip Techonology) was used since availability of UF2 [88], arduino [89] and MicroPython [90] bootloaders make it possible to rapidly develop and test application code on the platform. An added advantage of this microcontroller is that a real time clock can be enabled by connecting to an external crystal. This will ensure that all packets contain transmission timestamps, so a receiver restart will not impact the time reference for the packets. Moreover, this also helps with synchronizing or calculating time difference between packets from different transmitters. A light sensor (LV0104CS) and an accelerometer (BMA400) were added for sensing. The circuit weighs just 86 mg and was fabricated on a 12.5 micron copper + 12.5 micron polyimide sheet using a DPSS laser and then etching the copper.

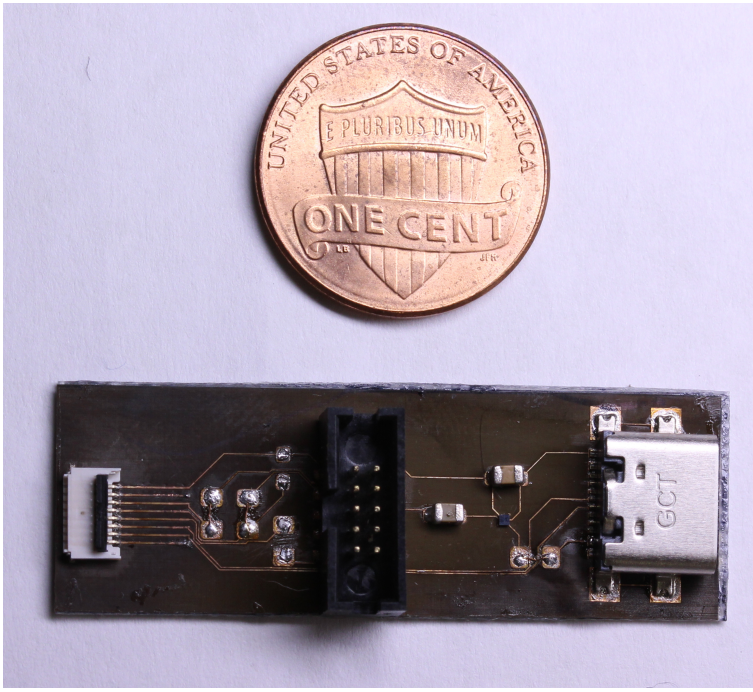


Figure 3.5: The fabricated adapter board for programming, debugging application code. US penny for scale

To program or debug application code for the microcontroller, a separate circuit, shown in fig-

ure 3.5 was made. A 60 micron FR4 sheet coated with copper on both sides was used for fabricating this circuit. A USB-C connector is used for uploading programs and providing power. A voltage regulator converts logic levels from 5 V to 3.3 V. A 10 pin Serial Wire Debug (SWD) connector is added for debugging application code. To avoid soldering wires from the tag to this circuit, a Flexible Printed Circuit (FPC) connector (Molex LLC) was used. Since the connector requires 300 microns thick circuit, two sheets of carbon fiber 90 micron thickness each and one sheet of 25 micron polyimide tape (kapton) was glued to the back of the circuit with cyanoacrylate glue. The two circuits connect as shown in figure 3.6. The total mass of the radio tag including the carbon fiber and kapton sheets is 113 mg, far below the payload capacity of 300-350 mg for *Vespa mandarinia*.

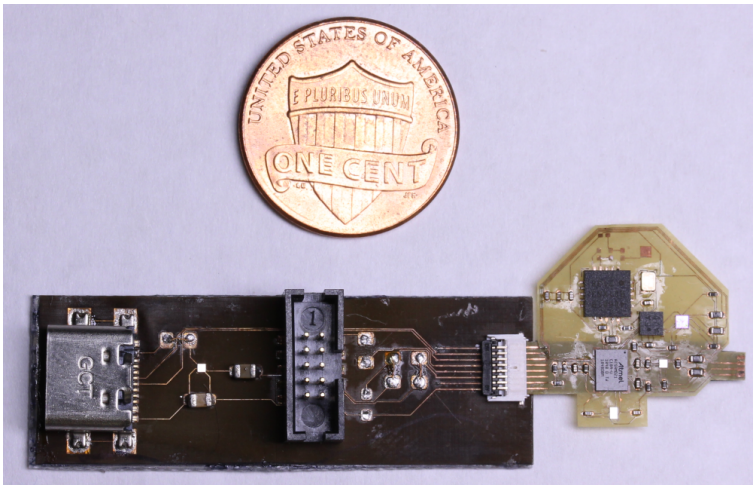


Figure 3.6: The radio tag can be connected to a programming / debugging circuit without soldering wires. US penny for scale.

Power is switched from on-board battery to the adapter board by pulling the enable pin of the voltage regulator on the tag circuit low. A similar adapter circuit can be made for charging the battery.

3.4 Conclusion and future work

Lightweight sensors coupled with wireless technology can enable studying behaviors of flying insects. In this work, an existing bluetooth platform was programmed to store sensor measurements on-board and to off-board receivers. An experiment was carried out which shows the utility of the developed system. To address a few issues with this system, a new platform was designed and fabricated. This can enable rapid testing and integration of new sensors and components and help in understanding insect behavior.

A separate adapter board needs to be made for charging the on-board battery. The tag was designed with these connections already broken out. The MCU on-board needs to be programmed to read sensor data, store it to on-board flash memory and send it using the radio chip. An antenna needs to be attached to the radio after performing impedance matching and adding the required capacitors and inductors. A separate receiver circuit also needs to be made for receiving sensor data from the tag and interface it with a storage device.

BIBLIOGRAPHY

[1] R. Shu, "Cad rendering of crazyflie." Available: https://github.com/rshum19/Crazyflie-CAD, 24 April 2017.

[2] D. Dhingra, Y. M. Chukewad, and S. B. Fuller, "A device for rapid, automated trimming of insect-sized flying robots," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1373–1380, 2020.

[3] J. James, V. Iyer, Y. Chukewad, S. Gollakota, and S. B. Fuller, "Liftoff of a 190 mg laser-powered aerial vehicle: The lightest wireless robot to fly," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–8, IEEE, 2018.

[4] M. Duduta, S. de Rivaz, D. R. Clarke, and R. J. Wood, "Ultra-lightweight, high power density lithium-ion batteries," *Batteries & Supercaps*, vol. 1, no. 4, pp. 131–134, 2018.

[5] Y. Karaca, M. Cicek, O. Tatli, A. Sahin, S. Pasli, M. F. Beser, and S. Turedi, "The potential use of unmanned aircraft systems (drones) in mountain search and rescue operations," *The American journal of emergency medicine*, vol. 36, no. 4, pp. 583–588, 2018.

[6] C. Kislik, I. Dronova, and M. Kelly, "Uavs in support of algal bloom research: a review of current applications and future opportunities," *Drones*, vol. 2, no. 4, p. 35, 2018.

[7] V. Iyer, M. Kim, S. Xue, A. Wang, and S. Gollakota, "Airdropping sensor networks from drones and insects," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, pp. 1–14, 2020.

[8] K. Grasegger, G. Strapazzon, E. Procter, H. Brugger, and I. Soteras, "Avalanche survival after rescue with the recco rescue system: a case report," *Wilderness & Environmental Medicine*, vol. 27, no. 2, pp. 282–286, 2016.

[9] P. J. Kennedy, S. M. Ford, J. Poidatz, D. Thiéry, and J. L. Osborne, "Searching for nests of the invasive asian hornet (vespa velutina) using radio-telemetry," *Communications biology*, vol. 1, no. 1, pp. 1–8, 2018.

[10] K. Schlosser, "Uw researcher put tiny tracking technology on giant hornets to help state deal with murderous pest." https://www.geekwire.com/2020/uw-researcher-put-tiny-tracking-technology-giant-hornets-help-state-deal-murderous-pest/, 2020.

[11] W. Giernacki, M. Skwierczyński, W. Witwicki, P. Wroński, and P. Kozierski, "Crazyflie 2.0 quadrotor as a platform for research and education in robotics and control engineering," in *2017 22nd International Conference on Methods and Models in Automation and Robotics (MMAR)*, pp. 37–42, IEEE, 2017.

[12] A. C. Owens and S. M. Lewis, "The impact of artificial light at night on nocturnal insects: A review and synthesis," *Ecology and evolution*, vol. 8, no. 22, pp. 11337–11358, 2018.

[13] G. H. Wadhams and J. P. Armitage, "Making sense of it all: bacterial chemotaxis," *Nature reviews Molecular cell biology*, vol. 5, no. 12, pp. 1024–1037, 2004.

[14] B. P. Duisterhof, S. Krishnan, J. J. Cruz, C. R. Banbury, W. Fu, A. Faust, G. C. de Croon, and V. Janapa Reddi, "Learning to seek: Deep reinforcement learning for phototaxis of a nano drone in an obstacle field," *arXiv*, pp. arXiv–1909, 2019.

[15] S.-i. Azuma, M. S. Sakar, and G. J. Pappas, "Stochastic source seeking by mobile robots," *IEEE Transactions on Automatic Control*, vol. 57, no. 9, pp. 2308–2321, 2012.

[16] E. Ramirez-Llanos and S. Martinez, "Stochastic source seeking for mobile robots in obstacle environments via the spsa method," *IEEE Transactions on Automatic Control*, vol. 64, no. 4, pp. 1732–1739, 2018.

[17] S.-J. Liu and M. Krstic, "Stochastic source seeking for nonholonomic unicycle," *Automatica*, vol. 46, no. 9, pp. 1443–1453, 2010.

[18] J. Cochran and M. Krstic, "Nonholonomic source seeking with tuning of angular velocity," *IEEE Transactions on Automatic Control*, vol. 54, no. 4, pp. 717–731, 2009.

[19] I. Rañó, M. Khamassi, and K. Wong-Lin, "A drift diffusion model of biological source seeking for mobile robots," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3525–3531, IEEE, 2017.

[20] R. Zou, V. Kalivarapu, E. Winer, J. Oliver, and S. Bhattacharya, "Particle swarm optimization-based source seeking," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 3, pp. 865–875, 2015.

[21] J. R. Bourne, E. R. Pardyjak, and K. K. Leang, "Coordinated bayesian-based bioinspired plume source term estimation and source seeking for mobile robots," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 967–986, 2019.

[22] N. Atanasov, J. Le Ny, N. Michael, and G. J. Pappas, "Stochastic source seeking in complex environments," in *2012 IEEE International Conference on Robotics and Automation*, pp. 3013–3018, IEEE, 2012.

[23] N. A. Atanasov, J. Le Ny, and G. J. Pappas, "Distributed algorithms for stochastic source seeking with mobile robot networks," *Journal of Dynamic Systems, Measurement, and Control*, vol. 137, no. 3, 2015.

[24] S. Li, R. Kong, and Y. Guo, "Cooperative distributed source seeking by multiple robots: Algorithms and experiments," *IEEE/ASME Transactions on mechatronics*, vol. 19, no. 6, pp. 1810–1820, 2014.

[25] A. J. Lilienthal, A. Loutfi, and T. Duckett, "Airborne chemical sensing with mobile robots," *Sensors*, vol. 6, no. 11, pp. 1616–1678, 2006.

[26] P. Shinners, "Pygame." http://pygame.org/, 2011.

[27] N. Elkunchwar, K. Balasubramanian, and J. Noe, "GitHub repository for crazyflie-run-and-tumble." Available: https://github.com/thecountoftuscany/crazyflie-run-and-tumble. 2020.

[28] Bitcraze, "GitHub repository for crazyflie-firmware." Available: https://github.com/bitcraze/crazyflie-firmware/. 2020.

[29] S. B. Fuller, M. Karpelson, A. Censi, K. Y. Ma, and R. J. Wood, "Controlling free flight of a robotic fly using an onboard vision sensor inspired by insect ocelli," *Journal of The Royal Society Interface*, vol. 11, no. 97, p. 20140281, 2014.

[30] V. Iyer, R. Nandakumar, A. Wang, S. B. Fuller, and S. Gollakota, "Living iot: A flying wireless platform on live insects," in *The 25th Annual International Conference on Mobile Computing and Networking*, pp. 1–15, 2019.

[31] M. J. Anderson, J. G. Sullivan, J. L. Talley, K. M. Brink, S. B. Fuller, and T. L. Daniel, "The "smellicopter," a bio-hybrid odor localizing nano air vehicle," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6077–6082, IEEE, 2019.

[32] A. T. Baisch, C. Heimlich, M. Karpelson, and R. J. Wood, "Hamr3: An autonomous 1.7 g ambulatory robot," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5073–5079, IEEE, 2011.

[33] N. T. Jafferis, E. F. Helbling, M. Karpelson, and R. J. Wood, "Untethered flight of an insect-sized flapping-wing microscale aerial vehicle," *Nature*, vol. 570, no. 7762, pp. 491–495, 2019.

[34] V. Talla, B. Kellogg, B. Ransford, S. Naderiparizi, S. Gollakota, and J. R. Smith, "Powering the next billion devices with wi-fi," in *Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies*, pp. 1–13, 2015.

[35] D. Vasisht, Z. Kapetanovic, J. Won, X. Jin, R. Chandra, S. Sinha, A. Kapoor, M. Sudarshan, and S. Stratman, "Farmbeats: An iot platform for data-driven agriculture," in *14th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 17)*, pp. 515–529, 2017.

[36] U. R. Mogili and B. Deepak, "Review on application of drone systems in precision agriculture," *Procedia computer science*, vol. 133, pp. 502–509, 2018.

[37] Y.-H. Tu, S. Phinn, K. Johansen, A. Robson, and D. Wu, "Optimising drone flight planning for measuring horticultural tree crop structure," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 160, pp. 83–96, 2020.

[38] K. Shah, G. Ballard, A. Schmidt, and M. Schwager, "Multidrone aerial surveys of penguin colonies in antarctica," *Science Robotics*, vol. 5, no. 47, 2020.

[39] A. Fitzpatrick, A. Singhvi, and A. Arbabian, "An airborne sonar system for underwater remote sensing and imaging," *IEEE Access*, vol. 8, pp. 189945–189959, 2020.

[40] C.-F. Lin, T.-J. Lin, W.-S. Liao, H. Lan, J.-Y. Lin, C.-H. Chiu, and A. Danner, "Solar power can substantially prolong maximum achievable airtime of quadcopter drones," *Advanced Science*, vol. 7, no. 20, p. 2001497, 2020.

[41] W. Johnson, *Helicopter theory*. Courier Dover Publications, 2012.

[42] M. S. Branham, W.-C. Hsu, S. Yerci, J. Loomis, S. V. Boriskina, B. R. Hoard, S. E. Han, and G. Chen, "15.7% efficient 10-μm-thick crystalline silicon solar cells using periodic nanostructures," *Advanced materials*, vol. 27, no. 13, pp. 2182–2188, 2015.

[43] A. C. U. A. Systems, "*Aerialtronics Altura Zenith Specsheet*," 16 Jun 2017.

[44] A. Technologies, "Ascending technologies firefly," 2015.

[45] A. Technologies, "Payload options & accessories," 2015.

[46] A. Technologies, "Ascending technologies hummingbird," 2015.

[47] P. D. SAS, "Parrot anafi specsheet," 4 Jun 2018.

[48] J. Förster, "System identification of the crazyflie 2.0 nano quadrocopter," B.S. thesis, ETH Zurich, 2015.

[49] M. Keennon, K. Klingebiel, H. Won, and A. Andriukov, "Development of the nano hummingbird: A tailless flapping wing micro air vehicle," in *AIAA Aerospace Sciences Meeting*, (Reston, VA), pp. 1–24, AIAA, 9–12 January 2012.

[50] N. Elkunchwar, "GitHub repository for solar-crazyflie." Available: https://github.com/thecountoftuscany/solar-crazyflie. 2021.

[51] S. Jung, Y. Jo, and Y.-J. Kim, "Aerial surveillance with low-altitude long-endurance tethered multirotor uavs using photovoltaic power management system," *Energies*, vol. 12, no. 7, p. 1323, 2019.

[52] S. Jung, Y. Jo, and Y.-J. Kim, "Flight time estimation for continuous surveillance missions using a multirotor uav," *Energies*, vol. 12, no. 5, p. 867, 2019.

[53] M. Lu, M. Bagheri, A. P. James, and T. Phung, "Wireless charging techniques for uavs: A review, reconceptualization, and extension," *IEEE Access*, vol. 6, pp. 29865–29884, 2018.

[54] T. Campi, S. Cruciani, M. Feliziani, and F. Maradei, "High efficiency and lightweight wireless charging system for drone batteries," in *2017 AEIT International Annual Conference*, pp. 1–6, IEEE, 2017.

[55] M. R. Hayajneh and A. R. E. Badawi, "Automatic uav wireless charging over solar vehicle to enable frequent flight missions," in *Proceedings of the 2019 3rd international conference on automation, control and robots*, pp. 44–49, 2019.

[56] E. Ali, M. Fanni, and A. M. Mohamed, "Design and task management of a mobile solar station for charging flying drones," in *E3S Web of Conferences*, vol. 167, p. 05004, EDP Sciences, 2020.

[57] A. Alsharoa, H. Ghazzai, A. Kadri, and A. E. Kamal, "Spatial and temporal management of cellular hetnets with multiple solar powered drones," *IEEE Transactions on Mobile Computing*, vol. 19, no. 4, pp. 954–968, 2019.

[58] M. C. Silverman, D. Nies, B. Jung, and G. S. Sukhatme, "Staying alive: A docking station for autonomous robot recharging," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, vol. 1, pp. 1050–1055, IEEE, 2002.

[59] S. Oh, A. Zelinsky, K. Taylor, *et al.*, "Autonomous battery recharging for indoor mobile robots," in *Proceedings of the australian conference on robotics and automation*, Citeseer, 2000.

[60] B. Michini, T. Toksoz, J. Redding, M. Michini, J. How, M. Vavrina, and J. Vian, "Automated battery swap and recharge to enable persistent uav missions," in *Infotech at Aerospace 2011*, p. 1405, 2011.

[61] C. S. Goh, J. R. Kuan, J. H. Yeo, B. S. Teo, and A. Danner, "A fully solar-powered quadcopter able to achieve controlled flight out of the ground effect," *Progress in Photovoltaics: Research and Applications*, vol. 27, no. 10, pp. 869–878, 2019.

[62] H. Ross, "Fly around the world with a solar powered airplane," in *The 26th Congress of ICAS and 8th AIAA ATIO*, p. 8954, 2008.

[63] R. D'Sa, D. Jenson, and N. Papanikolopoulos, "Suav: Q-a hybrid approach to solar-powered flight," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3288–3294, IEEE, 2016.

[64] K. L. Kennerud, "Analysis of performance degradation in cds solar cells," *IEEE Transactions on aerospace and electronic systems*, no. 6, pp. 912–917, 1969.

[65] L. Chen, X. Yuan, Y. Xiao, Y. Zhang, and J. Zhu, "Robust autonomous landing of uav in non-cooperative environments based on dynamic time camera-lidar fusion," *arXiv preprint arXiv:2011.13761*, 2020.

[66] M. Mittal, A. Valada, and W. Burgard, "Vision-based autonomous landing in catastrophe-struck environments," *arXiv preprint arXiv:1809.05700*, 2018.

[67] A. Cesetti, E. Frontoni, A. Mancini, P. Zingaretti, and S. Longhi, "A vision-based guidance system for uav navigation and safe landing using natural landmarks," *Journal of intelligent and robotic systems*, vol. 57, no. 1, pp. 233–257, 2010.

[68] N. Elkunchwar, V. Iyer, M. Anderson, K. Balasubramanian, J. Noe, Y. Talwekar, and S. Fuller, "Bio-inspired source seeking and obstacle avoidance on a palm-sized drone," 2021. (in prep).

[69] Bitcraze, "crazyflie-lib-python github repository." https://github.com/bitcraze/crazyflie-lib-python, 4 Mar 2021.

[70] Bitcraze, "Loco positioning system." https://www.bitcraze.io/documentation/system/positioning/loco-positioning-system/, 2021.

[71] Bitcraze, "Lighthouse positioning system." https://www.bitcraze.io/documentation/system/positioning/ligthouse-positioning-system/, 2021.

[72] Bitcraze, "Motion capture positioning." https://www.bitcraze.io/documentation/system/positioning/mocap-positioning/, 2021.

[73] Bitcraze, "Micro sd card deck | bitcraze." https://www.bitcraze.io/products/micro-sd-card-deck/, 2021.

[74] InsightSIP, "Isp4520 smart lora and bluetooth low energy module with mcu and antennas." https://www.insightsip.com/fichiers_insightsip/pdf/ble/ISP4520/isp_lora_DS4520_R6.pdf, 2020.

[75] B. Sensortech, "Gas sensor bme680." https://www.bosch-sensortec.com/products/environmental-sensors/gas-sensors/bme680/, 2021.

[76] O. Semiconductor, "Lv0104cs ambient light sensor datasheet." https://www.onsemi.com/pdf/datasheet/lv0104cs-d.pdf, 2021.

[77] S. B. Fuller, M. Karpelson, A. Censi, K. Y. Ma, and R. J. Wood, "Controlling free flight of a robotic fly using an onboard vision sensor inspired by insect ocelli," *Journal of The Royal Society Interface*, vol. 11, no. 97, p. 20140281, 2014.

[78] Sensirion, "Co2 sensor scd4x." https://www.sensirion.com/scd40/, 2021.

[79] S. Sensors, "15x15 co sensor 1000 ppm for sensor 100-102." https://www.spec-sensors.com/wp-content/uploads/2016/04/3SP_CO_1000-C-Package-110-109.pdf, 2021.

[80] V. Iyer, A. Najafi, J. James, S. Fuller, and S. Gollakota, "Wireless steerable vision for live insects and insect-scale robots," *Science robotics*, vol. 5, no. 44, 2020.

[81] Bosch, "Bmp 388 digital pressure sensor datasheet." https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bmp388-ds001.pdf, 2021.

[82] P. Tokekar, D. Bhadauria, A. Studenski, and V. Isler, "A robotic system for monitoring carp in minnesota lakes," *Journal of Field Robotics*, vol. 27, no. 6, pp. 779–789, 2010.

[83] O. M. Cliff, D. L. Saunders, and R. Fitch, "Robotic ecology: Tracking small dynamic animals with an autonomous aerial vehicle," *Science Robotics*, vol. 3, no. 23, 2018.

[84] M. W. Shafer, G. Vega, K. Rothfus, and P. Flikkema, "Uav wildlife radiotelemetry: System and methods of localization," *Methods in Ecology and Evolution*, vol. 10, no. 10, pp. 1783–1795, 2019.

[85] H. Bayram, N. Stefas, and V. Isler, "Aerial radio-based telemetry for tracking wildlife," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4723–4728, IEEE, 2018.

[86] W. S. D. of Agriculture, "Asian giant hornets." https://agr.wa.gov/hornets.

[87] K. Schlosser, "'murder hornet' nest cut from tree in washington state; 1,500 developing insects found inside." https://www.geekwire.com/2021/murder-hornet-nest-cut-tree-washington-state-1500-developing-insects-found-inside/.

[88] Microsoft, "GitHub repository for UF2 bootloader for SAMDx1 chips." Available: https://github.com/microsoft/uf2-samdx1. 2020.

[89] Arduino, "Arduino core for SAMD21 CPU." Available: https://github.com/arduino/ArduinoCore-samd. 2020.

[90] MicroPython, "Github repository for MicroPython." Available: https://github.com/micropython/micropython. 2020.