

A UAV for Destructive Surveys of Mosquito Population

Mary Burbage, An Nguyen, Kyle Walker, Nhan Phung, Vinh Truong, Erik Van Aller, and Aaron T. Becker

Abstract—This paper introduces techniques for mosquito population surveys in the wild using electrified screens (bug zappers) mounted to a UAV. Instrumentation on the UAV logs the UAV path and the GPS location, altitude, and time of each mosquito elimination. Changing the path of the UAV changes the number of mosquitoes encountered. We pose this as a new problem in robotic coverage and provide a simulator for mosquitoes in flight and a UAV with a screen. We compare four baseline algorithms to establish a benchmark. Hardware experiments with a UAV equipped with an electrified screen provide real-time measurements of (former) mosquito locations and mosquito-free volumes.

I. INTRODUCTION

Mosquito-borne diseases kill millions of humans each year [1]. Because of this threat governments worldwide track mosquito populations. Tracking individual mosquitoes is difficult because of their small size, wide-ranging flight, and preference for low-light. Tracking studies have chosen to use small ($1.2\text{ m} \times 2.4\text{ m}$) indoor regions [2] or mating swarms outdoors [3].

The dominant tools for tracking mosquito populations are stationary traps that are checked at weekly intervals (*e.g.* Encephalitis Vector Surveillance traps and/or gravid traps [4]). Recent research has focused on making these traps smaller, cheaper, and capable of providing real-time data [5], [6]; however, they still rely on attracting mosquitoes to the trap. This paper presents an alternate solution using an electrified bug-zapping screen mounted on an unmanned aerial vehicle (UAV) as shown in Fig.1 to seek out the mosquitoes in their habitat. As the UAV follows a path, it sweeps out a volume of air, temporarily removing all the mosquitoes in this volume. By monitoring the voltage across this screen, we can track individual mosquito contacts.

This process can be modeled as sampling without replacement from a point cloud of mobile particles using a mobile agent. The point cloud particles are generated from a known or unknown distribution, and the mobile agent clears all particles in a swept-out region each time step. UAV's have strict energy budgets, which limit the flight time to a maximum denoted by T . The goal is to design a trajectory for the mosquito screen with duration less than T that, in probability, samples the most particles. We assume the agent can detect each particle collision and use it to modify a planned trajectory. This paper presents both open-loop trajectories and policies with feedback.

The paper is arranged as follows. After a review of related work in §II, we present a benchmark simulation for

M. Burbage, A. Nguyen, K. Walker, N. Phung, V. Truong, E. Van Aller, and A. Becker are with the ECE Department at the University of Houston, TX. atbecker@uh.edu.

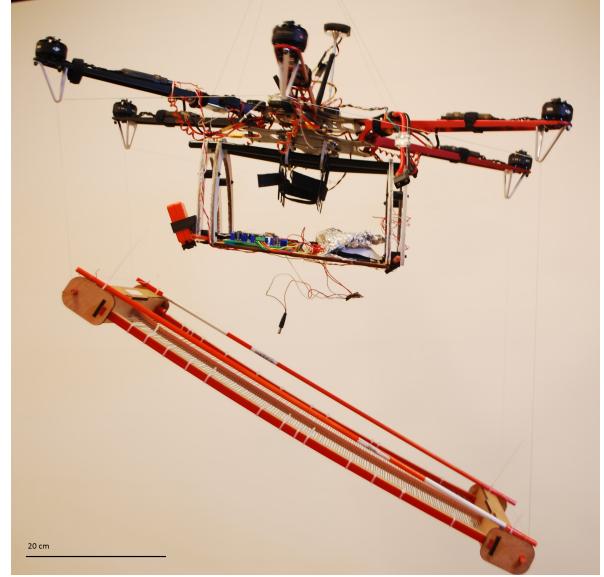


Fig. 1. A hexcopter UAV carrying a $35\text{ cm} \times 74\text{ cm}$ rectangular bug-zapping screen. An onboard microcontroller monitors the voltage across the screen and records the time, GPS location, and altitude for each mosquito strike.

comparing trajectories in §III. Simulation results comparing four baseline simulations are in §III-A and four benchmark simulations in §III-B. We present a design and rationale for a UAV with bug zapper in §IV. We then present hardware experiments with the UAV in §V and conclude with directions for future research in §VI.

II. RELATED WORK

A. Robotic coverage

Robotic coverage has a long history. The basic problem is one of designing a path for a robot that ensures the robot visits within r distance of every point on the workspace. For an overview see [7]. This work has been extended to use multiple coverage robots in a variety of ways, including using simple behaviors for the robots [8], [9]. The key difference in the mosquito coverage problem is that the mosquitoes can move, recontaminating an area previously cleared. We instead have a probability of coverage, as in [10]. This is closely related to the art gallery problem [11] but with limited range of visibility.

B. Mosquito Control Solutions

Mosquito control has a long history of efforts associated both with monitoring mosquito populations [12] and with eliminating mosquitoes. The work involves both draining

potential breeding grounds and destroying living mosquitoes [13]. An array of insecticidal compounds has been used with different application methods, concentrations, and quantities, including both larvicides and compounds directed at adult mosquitoes [14].

Various traps have been designed to capture and/or kill mosquitoes with increasing sophistication in imitating human bait as designers strive to achieve a trap that can rival the attraction of a live human [15]. In recent history, methods have also included genetically modifying mosquitoes so that they either cannot reproduce effectively or cannot transmit diseases successfully [16], and with the recent genomic mapping of mosquito species, new ideas for more targeted work have been formulated [17].

Popular methods to control mosquitoes such as insecticides are effective, but they have the potential to introduce long-term environmental damage and mosquitoes have demonstrated the ability to become resistant to pesticides [18]. Traditional electrified screens (bug zappers) use UV light to attract pests but have a large bycatch of non-pest insects [19].

C. Robotic Pest Management

As GPS technology has flourished and data processing has become cheaper and more readily available, researchers have explored options for implementing the new technologies in breeding ground removal [20] and more effective insecticide dispersion [21]. Low cost UAV's for residential spraying are under development [22]. Even optical solutions have been considered, including laser containment [23] or, by extension, exclusion and laser tracking and extermination [24].

III. SIMULATION

The goal is to design a trajectory with duration less than T that is likely to detect the greatest number of mosquitoes. We assume the UAV can detect each collision and use that information to modify a planned trajectory. This section introduces an open-source simulation platform for comparing the control policies. The MATLAB code used is available at [25].

Ten thousand mosquitoes are randomly placed within a square area one hundred meters on a side. A satellite image of Houston was used as the simulation environment, and each mosquito is programmed to move according to a biased random walk at a speed up to 0.4 m/s and with a direction heading biased toward the greenest of the pixels surrounding its current position. This imitates the live mosquitoes' preference for vegetative areas.

The simulation is initialized by a mosquito distribution generated by running the mosquito movement routine five thousand times, simulating 1.4 hours of flying time, before the robot begins to search. Because mosquitoes do not care about boundaries, a toroidal mapping is used to keep them in the workspace. Fig. 2 shows the mosquitoes before and after biasing.

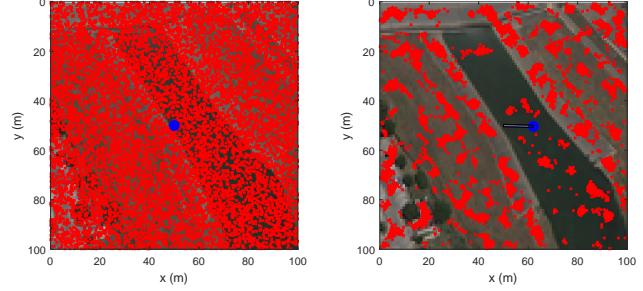


Fig. 2. A $100 \text{ m} \times 100 \text{ m}$ image with ten thousand simulated mosquitoes (red markers). The mosquitoes, initially uniformly distributed (left), have had 1.4 hours to bias themselves toward green areas of the image (right). A robot (blue circle) in the center is preparing to start a bug-zapping run.

The flying robot eliminates any modeled mosquito that intersects its path. The robot can detect the time each mosquito is eliminated and use this data as feedback for a motion policy. The robot is tested with four different policies. The first is a plain boustrophedon or lawn-mowing path [7]. The robot begins in one corner of the workspace and methodically progresses back and forth, advancing one screen width at each turn. If it covers the entire field in the allotted time, it begins covering the field again.

The second uses a random bounce algorithm. The robot begins in the center of the workspace and moves with a heading that varies randomly up to $\pm 0.2 \text{ rad}$ from its previous heading and bounces off the perimeter of the workspace with a random heading equally biased between 0 and $2\pi \text{ rad}$, excluding headings leading outside the workspace.

The third path begins with the boustrophedon path but switches to a square spiral path when the rate of mosquito kills exceeds a threshold. Once the kill rate falls below another threshold, the path returns to a boustrophedon path. The final path combines the random bounce path with the square spiral in the same way as the third path.

For the main body of the simulation, a loop runs a series of iterations in which each mosquito moves one step and the robot moves one step. In that step, the path traced by the bug-zapping screen is calculated, and any mosquitoes in that path are considered to have been counted and killed.

To keep the routines comparable, the robots use the same speed and same number of iterations in each test as well as the same image for biasing the mosquito flight. The baseline simulations used 12 m/s and fifteen minutes of flying time, which is enough time for the robot to completely cover the 100m workspace.

A. Baseline Results

One hundred trials were performed with each coverage path and the results evaluated. The boustrophedon successfully covered the entire field in every trial ($\mu = 100\%$, $\sigma = 0\%$), while the random bounce covered only 64.7% of the field ($\mu = 64.7\%$, $\sigma = 1.1\%$) on average. The boustrophedon with spiral covered 91.5% of the field ($\mu = 91.5\%$, $\sigma = 0.6\%$), and the random bounce with spiral

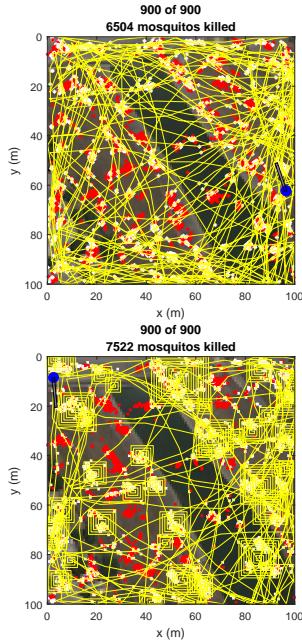


Fig. 3. Four sample simulations. The robot (blue circle) and the area covered by the bug-zapping net in an iteration (blue rectangle) are shown along with a population of ten thousand mosquitoes. Those killed during the simulation are white, and those that survived are red. Left column: random bounce, right column: boustrophedon. Bottom row adds a subroutine to perform spiral movements when the mosquito density is high.

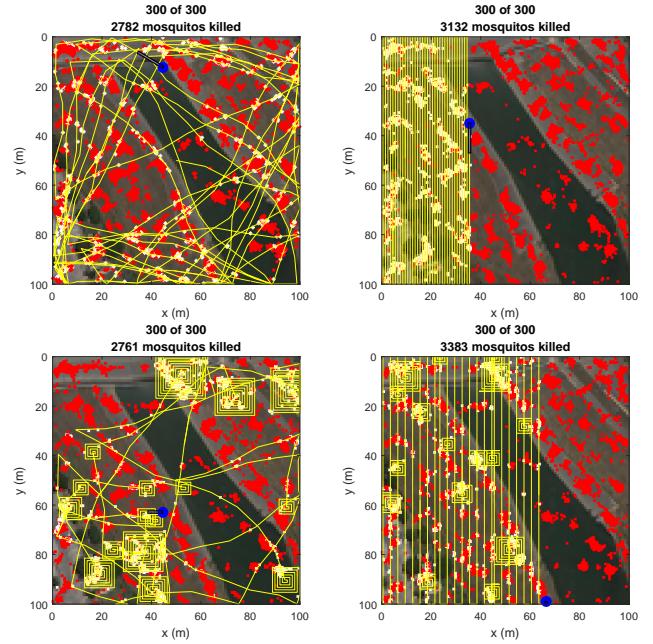


Fig. 5. Left column: random bounce, right column: boustrophedon. Bottom row adds a subroutine to perform spiral movements when the mosquito density is high.

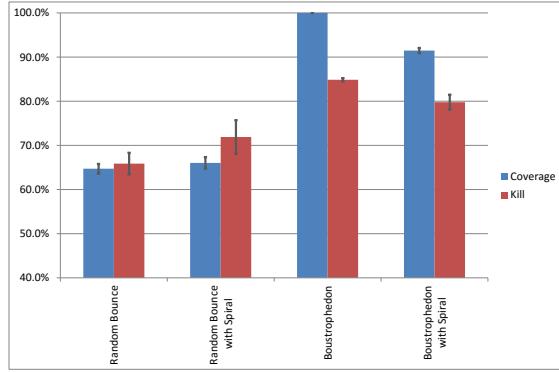


Fig. 4. Comparison of percentage of area covered and percentage of mosquitoes killed in fifteen minutes for the four coverage patterns. Plots show aggregate results of 100 trials, using the workspace shown in Fig. 2.

covered 66.0% ($\mu = 66.0\%$, $\sigma = 1.3\%$). Due to the higher coverage rates, the boustrophedon killed significantly more mosquitoes ($\mu = 84.9\%$, $\sigma = 0.4\%$) than the random bounce ($\mu = 65.9\%$, $\sigma = 2.5\%$), though the addition of the spiral to the random bounce showed an improvement over the plain random bounce ($\mu = 71.9\%$, $\sigma = 3.8\%$). Including the spiral with the boustrophedon degraded its performance ($\mu = 79.8\%$, $\sigma = 1.7\%$). Fig. 3 and Fig. 4 show the paths for the baseline trials and their success rates, respectively.

B. Benchmark Results

Having set a baseline in which full coverage was clearly the best method, we next considered the problem of a field which is too large for full coverage within the allotted time. To simulate this, we used the same environment but set the flight time to five minutes and repeated the hundred iterations of each method.

With reduced flying time, the coverage and kill rates were much lower than in the baseline tests, but the spiral improved the kill rate for both the boustrophedon and the random bounce paths. Sample paths are displayed in Fig.5. Results are illustrated in Fig.6, which shows the best kill rate of 35.0% ($\mu = 35.0\%$, $\sigma = 3.0\%$) for the boustrophedon with spiral path, followed by the random bounce with spiral path at 32.5% ($\mu = 32.5\%$, $\sigma = 3.5\%$). The boustrophedon and random bounce paths had equal kill rates of 31.2%, though the boustrophedon had a narrower standard distribution ($\mu = 31.2\%$, boustrophedon: $\sigma = 0.5\%$, random bounce: $\sigma = 2.8\%$) and covered a greater percentage of the workspace than the random bounce, 35.7% as opposed to 30.6%. We consider this a benchmark against which testing of new methods may be measured in the future.

IV. HARDWARE DESIGN

This section examines the components of the mosquito UAV system, shown in Fig. 1. This includes the UAV, electrified screen, and surveying electronics as well as discussion of the energy budget. The design for the mosquito UAV system been assigned U.S. Provisional Patent Application [26].

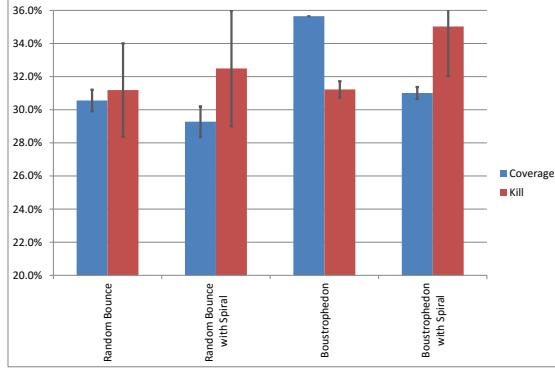


Fig. 6. Comparison of percentage of area covered and percentage of mosquitoes killed in five minutes for the four coverage patterns. Plots show aggregate results of 100 trials, using the workspace shown in Fig. 2.

A. UAV

The UAV is a custom-built, 177 cm wingspan hexacopter, controlled by a Pixhawk flight controller running ArduPilot Mega flight software. The UAV has a 3DR GPS module using the UBX NEO-7 chipset.

B. Screen

The screen is constructed from laser-cut acrylic and plywood. The frame is 35 cm × 74 cm with a 30.5 cm × 61 cm opening. This opening is spanned by a row of spring-steel 1.1 mm diameter wires. These wires are spaced 3 mm apart. They are divided into two sets of alternate wires, one of which is held at the reference voltage and the other of which is held at 1.4 kV above the reference.

C. Location of screen

The UAV carries the bug-zapping screen, which is suspended by fishing line at each corner. The location of this screen determines the efficacy of the mosquito UAV, measured in mosquitoes detected per second of flight time.

For manufacturing ease, the electrified screen is a rectangle with a width of d_s . The screen is suspended a distance h_s beneath the UAV flying at height h_d . We chose to suspend the screen beneath the UAV to avoid the weight of the rigid frame that would be required if the screen were above the UAV. This screen can be suspended at any desired angle θ in comparison to horizontal, as shown in Fig. 7. Two key parameters are the distance h_s and the optimal angle θ . The goal is to clear the greatest volume of mosquitoes per second, a volume defined by the UAV forward velocity v_f and the cross sectional area $h_m \times d_s$ cleared by the screen, as shown in Fig. 8.

To hover, the UAV must push sufficient air down with velocity v_d to apply a force that cancels the pull of gravity. The UAV and screen combined have mass m_d and its cross section can be approximated as a square with a side length of d_d . The mass flow of air through the UAV's propellers is

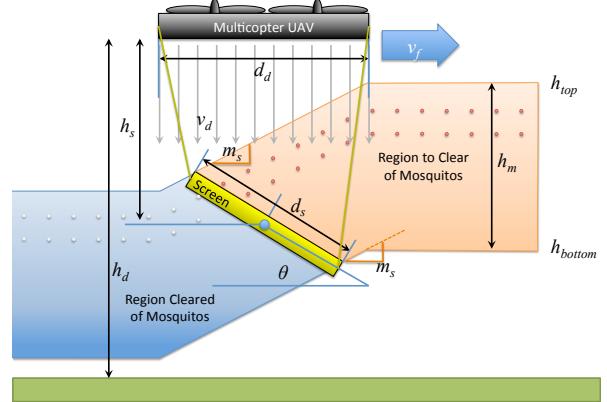


Fig. 7. The UAV suspends a rectangular bug-zapping screen beneath it. Propwash pushes incoming mosquitoes downwards, and the UAV clears a volume $h_m \times d_s \times v_f$ each second. Circles show two mosquitoes at equal time intervals relative to the UAV.

equal to the product of the change in velocity of the air, the density of the air ρ_a , and the cross sectional area.

We assume that air above the UAV is quiescent, so the change in velocity of the air is v_d m/s.

$$\text{Force gravity} = (\text{mass flow}) \cdot \text{air velocity}$$

$$m_d \cdot g = (v_d \cdot \rho_a \cdot d_d^2) \cdot v_d \quad (1)$$

Then the required propwash, the velocity of air beneath the UAV, for hovering is

$$v_d = \sqrt{\frac{m_d g}{\rho_a d_d^2}} \quad (2)$$

The flight testing site in Houston, Texas is 15m above sea level. At sea level the density of air ρ_a is 1.225 kg/m³. The UAV and instrumentation combined weigh 5.1 kg with a width of 0.75 m. The acceleration due to gravity is 9.871 m/s². Substituting these values into (2) gives $v_d = 8.5$ m/s.

Due to propwash, an initially hovering mosquito will fall when under the UAV at a rate of v_d . Relative to the UAV, the mosquito moves horizontally at a rate of $-v_f$. As shown in Fig. 7, we can extend lines with slope $-v_d/v_f$ from the screen's trailing edge to h_{top} and from the leading edge to h_{bottom}

$$\begin{aligned} h_{top} &= h_d - h_s + \frac{d_s}{2} \sin(\theta) + \frac{d_d + d_s \cos(\theta)}{2} \frac{v_d}{v_f} \\ h_{bottom} &= h_d - h_s - \frac{d_s}{2} \sin(\theta) + \frac{d_d - d_s \cos(\theta)}{2} \frac{v_d}{v_f} \\ h_m &= h_{top} - h_{bottom} = d_s \left(\frac{v_d}{v_f} \cos(\theta) + \sin(\theta) \right) \end{aligned} \quad (3)$$

The optimal angle is therefore a function of forward and propwash velocity:

$$\theta = \text{ArcTan} \left(\frac{v_f}{v_d} \right) \quad (4)$$

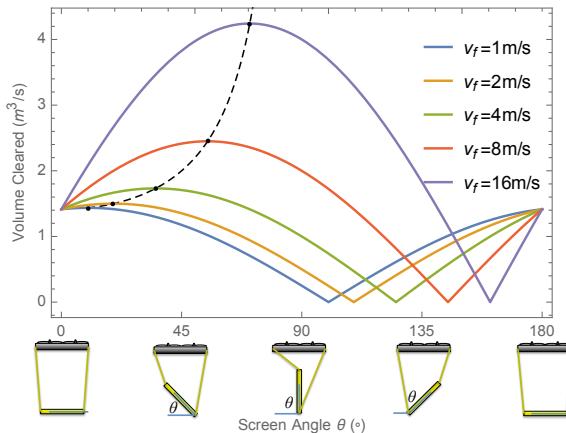


Fig. 8. The volume cleared by a UAV is a function of screen angle θ and forward velocity v_f . Dotted line shows the optimal angle given in (4).

To ensure the maximum number of mosquitoes are collected, the screen must be sufficiently far below the UAV $h_s > \frac{d_s}{2} \sin(\theta) + \frac{d_d + d_s \cos(\theta)}{2} \frac{v_d}{v_f}$ and the bottom of the screen must not touch the ground, $h_d > h_s + \frac{d_s}{2} \sin(\theta)$.

There are practical limits to h_s as well. Tests with $h_s > 2\text{ m}$ were abandoned because the long length caused the screen to act as a pendulum, introducing dynamics that made the system difficult to fly.

Changing the flying height h_d of the UAV will target different mosquito populations because mosquitoes are not distributed uniformly vertically. Gillies and Wilkes have demonstrated that different species of mosquitoes prefer to fly at different heights [27].

D. Electronics

The electrical detection and logging system is powered by a 9 V lithium ion battery applied directly to the controller and two AA 3 V lithium ion batteries applied to the power circuit for the screen. The power circuit outputs a high DC voltage across the screen. A protection circuit, shown in Fig. 9, steps this voltage down to a suitable level for monitoring by the ADC of the controller. The controller uses a GPS shield for monitoring the location and altitude as well as a real time clock to timestamp each data point collected from the system.

The power circuit uses a BJT and center tap transformer to invert a DC input voltage to AC and apply it to the primary winding of a step-up transformer. The voltage at the secondary winding of the transformer is boosted and rectified to two high voltage output capacitors. The protection circuit utilizes a voltage divider to reduce the voltage to a level suitable for the controller; this divider utilizes a potentiometer to adjust the ratio of screen output voltage and the voltage seen by the controller. A capacitor is used at the input of this circuit to smooth the unstable DC voltage at the screen output. A small series resistor is also used to limit residual low frequency current. A bidirectional transient voltage suppression (TVS) diode is then used to restrict both positive and negative high voltage transients that

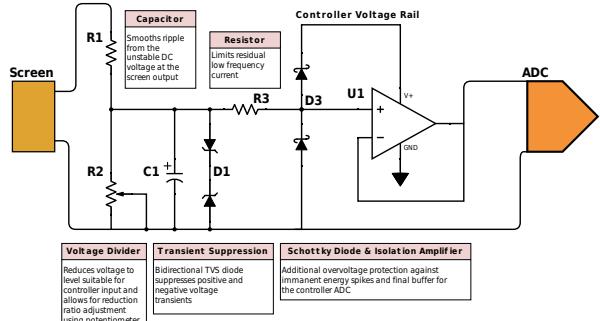


Fig. 9. Circuit diagram of the probe circuit for the electrified screen.

might otherwise propagate through the divider and capacitor. Buffering the controller are a Schottky diode and isolation amplifier. Both are powered by the controller supply and are used to protect against voltage spikes that would destroy the TVS diode.

E. Energy Budget

Tests with an oscilloscope show that in the steady state, a $30.5\text{ cm} \times 61\text{ cm}$ screen and electronics have a power consumption of 3.6 W. During a zap, the screen voltage monitoring circuit shorts briefly when the mosquito contacts the screen. Fig. 10 shows the time sequences for battery and screen voltages, current, and power during five mosquito zaps. The initial current spike recovery can be modeled with an exponential fit.

$$i = 69.1e^{-2.7 \times 10^4 t} \text{ A} \quad (5)$$

The fit in (5) gives us a time constant of $2.7 \times 10^4 / \text{s}$ for the short and a recovery time of $80 \mu\text{s}$. Multiplying voltage by current to find the instantaneous power ($p = iv$) and integrating the area under the power curve show a total energy consumption of 4.2 mJ for each zap. Recharging the screen requires more power and is represented in the latter part of the curves. The overall recovery time is about 160 ms. Most of the energy is consumed charging and maintaining the charge on the screen rather than in zapping the mosquitoes.

V. EXPERIMENTS

The results for two trial flights are described below. A boustrophedon path with 2 m spacing was generated to cover a region $90\text{ m} \times 50\text{ m}$. The path was generated using Mission Planner software from ardupilot.org [28]. For the first trial, at 21:10 on September 13, the flight was 11 minutes in duration with an average velocity of 3 m/s at a nominal altitude of 7 m. Wind was 3.5 m/s from the east. The net recorded 5 mosquito kills. For the second trial, at 19:42 on September 14, the flight was 4 minutes in duration with an average velocity of 4 m/s at a nominal altitude of 2.5 m. Wind was 3.2 m/s from the southeast. The net recorded 58 mosquito kills.

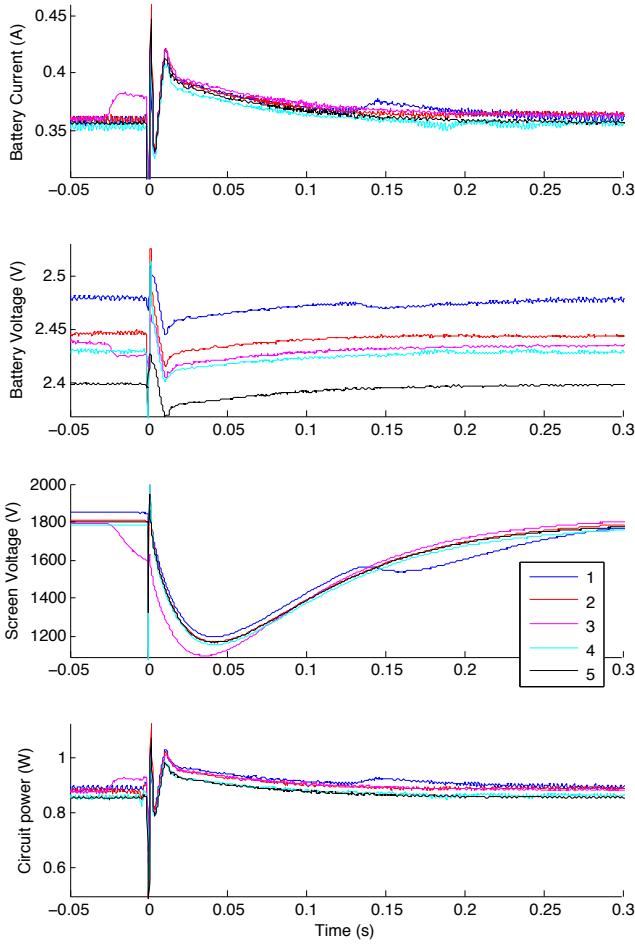


Fig. 10. Current, voltage, and power traces for five *Culex quinquefasciatus* mosquitoes as each contacts the bug-zapping screen. Contact causes a brief short that recovers in 160 ms.

This corresponds with the pilot's estimation, receiving 1 mosquito bite (3 mosquito landings) the first night and over ten bites the second night. The flight paths in an orthographic and side view are shown in Fig. 11.

For each trial the UAV took off from a resting position on top of the net which in turn was atop a plywood landing pad. The micro-controllers were started up, the net was energized, and the UAV was powered on. After establishing a stable hover at 7 m, flight control was switched to the autonomous mission.

These results not only tell where mosquitos were but also show where mosquitos were not. This is a key difference from stationary traps such as [5], [6].

VI. CONCLUSION AND FUTURE WORK

This paper presented a new problem in robotic coverage that attempts to maximize the number of moving particles detected when they are sampled without replacement. We provided a benchmark simulation and showed that techniques exist that outperform a simple boustrrophedon search.

Initial experiments with the UAV and electrified screen track the location of a mosquito-killing UAV as it patrols a

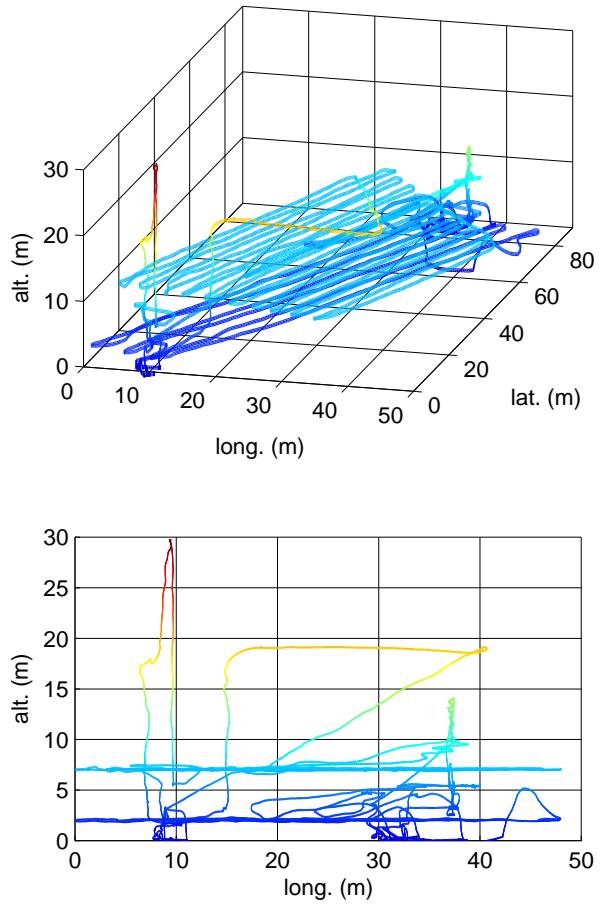


Fig. 11. Flight data from two runs over the same area, the first at 7 m elevation and the second at 2 m. Both trials used 2 m spacing between rows and covered a region 90 m \times 50 m. The landing pad is located at [0, 10, 0]. The UAV flies to 30 m when autonomous flight finishes, and the pilot resumes command.

field and maps mosquito kills.

There are a number of refinements to the simulation algorithm that could be pursued in future work, including changes to both the mosquito-biasing algorithm and the robot flight simulation. The model may be expanded to three dimensions. Additional search paths may be compared to the existing algorithms. These and other considerations will make a more realistic model for future work.

Further testing of the multi-copter UAV is indicated and will allow more extensive testing of the robustness and accuracy of the hardware design. This includes field tests of the simulation algorithms. New sensors that can identify and detect flying insects [5] may be added to the UAV and enable it to proactively steer toward insect swarms and identify insects in realtime.

The concept may be extended to a non-destructive population survey in which the screen could be replaced with a net and, with appropriate lighting, the camera used to record capture events. Teams of UAV's could work together to map areas more quickly and measure gradients of the distribution

to find large mosquito populations quickly.

ACKNOWLEDGMENT

The authors acknowledge the helpful advice and feedback from Martin Reyna Nava, MS, Medical Entomologist and Technical Operations Manager and Mustapha Debboun, Ph.D, BCE, Director Mosquito Control Division, of the Harris County Public Health & Environmental Services, Mosquito Control Division.

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1553063. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

- [1] C. J. Murray, L. C. Rosenfeld, S. S. Lim, K. G. Andrews, K. J. Foreman, D. Haring, N. Fullman, M. Naghavi, R. Lozano, and A. D. Lopez, "Global malaria mortality between 1980 and 2010: a systematic analysis," *The Lancet*, vol. 379, no. 9814, pp. 413–431, 2012.
- [2] J. E. Parker, N. Angarita-Jaimes, M. Abe, C. E. Towers, D. Towers, and P. J. McCall, "Infrared video tracking of anophelles gambiae at insecticide-treated bed nets reveals rapid decisive impact after brief localised net contact," *Scientific reports*, vol. 5, 2015.
- [3] S. Butail, N. Manoukis, M. Diallo, A. S. Yaro, A. Dao, S. F. Traoré, J. M. Ribeiro, T. Lehmann, and D. A. Paley, "3d tracking of mating events in wild swarms of the malaria mosquito anopheles gambiae," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2011, pp. 720–723.
- [4] G. M. Williams and J. B. Gingrich, "Comparison of light traps, gravid traps, and resting boxes for west nile virus surveillance," *Journal of Vector Ecology*, vol. 32, no. 2, pp. 285–291, 2007.
- [5] Y. Chen, A. Why, G. Batista, A. Mafra-Neto, and E. Keogh, "Flying insect classification with inexpensive sensors," *Journal of insect behavior*, vol. 27, no. 5, pp. 657–677, 2014.
- [6] A. Linn, "Building a better mosquito trap," *International Pest Control*, vol. 58, no. 4, p. 213, 2016.
- [7] H. Choset, "Coverage for robotics - a survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1-4, pp. 113–126, October 2001.
- [8] D. Spears, W. Kerr, and W. Spears, "Physics-based robot swarms for coverage problems," *The international journal of intelligent control and systems*, vol. 11, no. 3, 2006.
- [9] S. Koenig, B. Szymanski, and Y. Liu, "Efficient and inefficient ant coverage methods," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1, pp. 41–76, Oct. 2001.
- [10] C. Das, A. Becker, and T. Bretl, "Probably approximately correct coverage for robots with uncertainty," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, San Francisco, CA, USA, Oct. 2011, pp. 1160–1166.
- [11] D.-T. Lee and A. K. Lin, "Computational complexity of art gallery problems," *Information Theory, IEEE Transactions on*, vol. 32, no. 2, pp. 276–282, 1986.
- [12] J. A. Dennett, A. Bala, T. Wuithiranyagool, Y. Randle, C. B. Sargent, H. Guzman, M. SHIRIN, H. K. Hassan, M. Reyna-Nava, T. R. Unnasch et al., "Associations between two mosquito populations and west nile virus in harris county, texas, 2003–06," *Journal of the American Mosquito Control Association*, vol. 23, no. 3, p. 264, 2007.
- [13] R. Peter, P. Van den Bossche, B. L. Penzhorn, and B. Sharp, "Tick, fly, and mosquito control—lessons from the past, solutions for the future," *Veterinary parasitology*, vol. 132, no. 3, pp. 205–215, 2005.
- [14] W. H. Organization, "Guidelines for laboratory and field testing of mosquito larvicides," *WORLD HEALTH ORGANIZATION COMMUNICABLE DISEASE CONTROL, PREVENTION AND ERADICATION WHO PESTICIDE EVALUATION SCHEME*, 2005.
- [15] D. V. Maliti, N. J. Govella, G. F. Killeen, N. Mirzai, P. C. Johnson, K. Kreppel, and H. M. Ferguson, "Development and evaluation of mosquito-electrocuting traps as alternatives to the human landing catch technique for sampling host-seeking malaria vectors," *Malaria journal*, vol. 14, no. 1, p. 1, 2015.
- [16] J. M. Marshall and C. E. Taylor, "Malaria control with transgenic mosquitoes," *PLoS Med*, vol. 6, no. 2, p. e1000020, 2009.
- [17] C. A. Hill, F. C. Kafatos, S. K. Stansfield, and F. H. Collins, "Arthropod-borne diseases: vector control in the genomics era," *Nature Reviews Microbiology*, vol. 3, no. 3, pp. 262–268, 2005.
- [18] M. O. Ndiath, S. Sougoufara, A. Gaye, C. Mazenot, L. Konate, O. Faye, C. Sokhna, and J.-F. Trape, "Resistance to ddt and pyrethroids and increased kdr mutation frequency in an. gambiae after the implementation of permethrin-treated nets in senegal," *PLoS One*, vol. 7, no. 2, p. e31943, 2012.
- [19] sciencedaily.com. (1997, Jul.) Snap! crackle! pop! electric bug zappers are useless for controlling mosquitoes, says uf/ifa's pest expert. [Online]. Available: <http://www.sciencedaily.com/releases/1997/07/970730060806.htm>
- [20] P. Anupa Elizabeth, M. Saravana Mohan, P. Philip Samuel, S. Pandian, and B. Tyagi, "Identification and eradication of mosquito breeding sites using wireless networking and electromechanical technologies," in *Recent Trends in Information Technology (ICRTIT), 2014 International Conference on*. IEEE, 2014, pp. 1–6.
- [21] B. Hur and W. Eisenstadt, "Low-power wireless climate monitoring system with rfid security access feature for mosquito and pathogen research," in *Mobile and Secure Services (MOBISECSERV), 2015 First Conference on*. IEEE, 2015, pp. 1–5.
- [22] J.-T. Amenyo, D. Phelps, O. Oladipo, F. Sewovoe-Ekuoe, S. Jadoonanan, S. Jadoonanan, T. Tabassum, S. Gnabode, T. D. Sherpa, M. Falzone et al., "Medizdroids project: Ultra-low cost, low-altitude, affordable and sustainable uav multicopter drones for mosquito vector control in malaria disease management," in *Global Humanitarian Technology Conference (GHTC), 2014 IEEE*. IEEE, 2014, pp. 590–596.
- [23] C. Boonsri, S. Sumriddetchkajorn, and P. Buranasiri, "Laser-based mosquito repelling module," in *Photonics Global Conference (PGC), 2012*. IEEE, 2012, pp. 1–4.
- [24] J. Kare and J. Buffum, "Build your own photonic fence to zap mosquitoes midflight [backwards star wars]," *IEEE Spectrum*, vol. 5, no. 47, pp. 28–33, 2010. [Online]. Available: <http://spectrum.ieee.org/consumer-electronics/gadgets/backyard-star-wars>
- [25] M. Burbage, S. Manzoor, and A. T. Becker. (2016, Sep.) Mosquito Drone Coverage and Elimination, MATLAB Central File Exchange. [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/59142>
- [26] A. T. Becker, "Unmanned mosquito elimination vehicles," US Patent 62/324,625, 04 19, 2016.
- [27] M. Gillies and T. Wilkes, "The vertical distribution of some West African mosquitoes (Diptera, Culicidae) over open farmland in a freshwater area of The Gambia," *Bulletin of entomological research*, vol. 66, no. 01, pp. 5–15, 1976.
- [28] ardupilot.org. (2016) Motion planner overview. [Online]. Available: <http://ardupilot.org/planner/docs/mission-planner-overview.html>