

Altitude Estimation and Control of an Insect-Scale Robot with an Onboard Proximity Sensor

E. Farrell Helbling, Sawyer B. Fuller, and Robert J. Wood

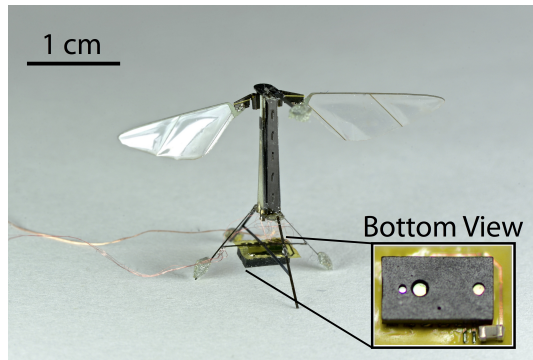
Abstract Insect-scale micro-air vehicles (MAVs) require careful consideration of the size, weight and power for each component. The inherent instability of the system, exacerbated by the faster dynamics that result from increasing angular accelerations with decreasing scale, requires high bandwidth sensing to maintain stable flight. The Harvard RoboBee is the first MAV under 100 mg to demonstrate controlled flight using external motion capture cameras to measure the position and orientation of the vehicle during flight. Prior research into onboard sensing has demonstrated several sensors that provide sufficiently high-bandwidth and low-latency feedback to stabilize the attitude of the robot. To achieve autonomous flight, the vehicle needs to sense its attitude, altitude, and either lateral position or velocity. Here we build on previous work by incorporating a sensor that is size- and power-compatible with insect-scale flight, capable of estimating distance by measuring the time-of-flight of an infrared laser pulse. This sensor has sufficiently low latency to allow the robot to maintain constant altitude over multiple flight experiments. This work on onboard altitude control represents the latest results in achieving autonomous control and visually-guided flight.

1 Introduction

Insect-scale MAVs (50-500 mg) are envisioned for many applications, including hazardous environment exploration and assisted agriculture. However, flight at this scale poses a control challenge for stability, as rotational acceleration increases with a decrease in the vehicle's characteristic length, scaling as l^{-1} [15]. The vehicle's flight dynamics are also unstable, requiring the flight controller to perform contin-

E. Farrell Helbling, Sawyer B. Fuller, and Robert J. Wood
Harvard University John A. Paulson School of Engineering and Applied Sciences and the Wyss Institute for Biologically-Inspired Engineering, 60 Oxford Street, Cambridge, MA 02138 USA,
e-mail: ehelbling@seas.harvard.edu

Fig. 1 The Harvard RoboBee is an 80 mg flapping-wing MAV that has demonstrated controlled flight using external motion capture cameras and reflective markers on the base and nose of the robot. Here we incorporate an infrared (IR) time-of-flight (ToF) sensor mounted on the base of the robot and use it to detect and control distance from the ground during flight. Inset: VL6180x (ST Microelectronics) mounted on a custom flex circuit. The sensor is 5 mm in length.



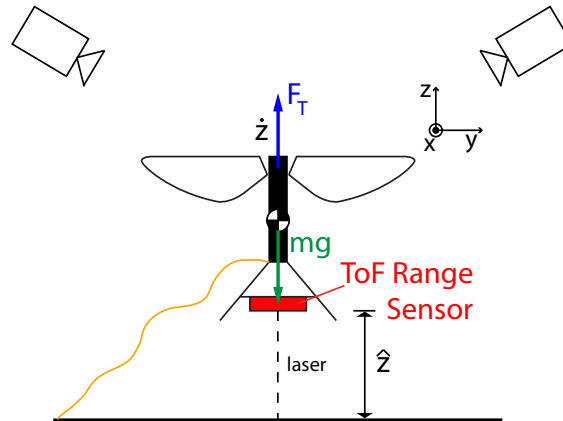
uous corrective maneuvers [1]. Therefore, in addition to meeting the stringent mass and power requirements for vehicles at this scale, onboard sensors must also provide low latency, high bandwidth information to the active controller.

The Harvard RoboBee (see Fig. 1) is the first MAV under 100 mg to lift its own weight and demonstrate controlled flight under external power [16]; however, an external, camera-based motion capture system (Vicon T040 System, Oxford UK) was used to estimate the position and orientation of the robot. For the RoboBee to operate in unstructured environments, it must be equipped with sensors that can estimate the vehicle's state to stabilize the dynamics and sense its external environment. Insects are a key source of inspiration for flight control, they rely on a multitude of sensory organs that provide estimates of relevant state variables to use in combination with reflexive or high-level control architectures to adjust flight accordingly [21].

Vision has been shown to be particularly important for navigation at small scales; GPS is too imprecise, with an accuracy of 1-10 m, and currently available emissive sensors such as radar, scanning laser rangefinders, and sonar do not meet the weight requirements (~ 40 mg) of the vehicle [4]. MAVs have adopted vision sensors for navigation at scales one to two orders of magnitude larger than the RoboBee, including [7] and [3]. Other researchers have demonstrated high bandwidth, low latency, and lightweight biomimetic optical sensors with the goal of stabilizing many aspects of the RoboBee's flight, including altitude on fixed guide wires [9], and orientation after takeoff [11].

In addition to these vision-based sensors, inertial measurement units (IMUs) – a system consisting of gyroscopes, accelerometers, and magnetometers – have long been used for flight control and stabilization of larger aircraft. Demand for miniaturization by the consumer electronics industry has resulted in sensors that now meet the mass (~ 10 -100 mg) and power (~ 10 mW) requirements of the RoboBee and have been integrated onto the vehicle. These sensors demonstrated low-latency feedback and sufficient noise rejection, controlling motion about the pitch and yaw

Fig. 2 Model of the altitude dynamics of the RoboBee and diagram of the flight apparatus. The altitude controller assumes upright orientation, reducing the model to a single dimension, z . The thrust force, F_T , generated by the flapping wings acts along the body z -axis, with the gravitational force of the body acting at the center of mass. The robot moves with velocity, \dot{z} , along the z -axis. The sensor (mounted below the robot) estimates the distance of the robot from the ground, \hat{z} , with Vicon cameras estimating the lateral position and attitude during flight. The robot is tethered for power and control signals, as well as sensor communications.



axes [12], as well as providing flight stabilization at takeoff and hovering [10]. In addition to stabilizing the attitude of the vehicle, the hovering controller from [6] requires estimates of rotation rate about the body axes, the absolute orientation of the vehicle, the lateral velocity and position, as well as the altitude to hover with asymptotic stability.

In addition to facilitating altitude regulation, an altitude estimate, when combined with other sensors, can be used to estimate other aspects of the vehicle's motion. Future research into autonomous, visually-guided flight at this scale will require accurate altitude estimation. An onboard, downward-facing optic flow sensor coupled with an absolute altitude estimate can provide the vehicle's forward velocity, as in [13] and demonstrated by landing honeybees in nature [21].

Here we consider a millimeter-scale infrared (IR) time-of-flight (ToF) sensor to measure the absolute position of the vehicle from the ground. We demonstrate that the sensor meets the mass and power requirements of the vehicle while providing feedback at a sufficient rate (~ 50 Hz) to control the altitude of the vehicle during hovering flight.

This paper describes the altitude dynamics of the RoboBee and estimates the minimum feedback rate necessary to stabilize this degree of freedom (Sect. 2); discusses the sensors that meet the mass, power, and frequency requirements as well as describes the integration and calibration of the ToF sensor onboard the vehicle (Sect. 3); and demonstrates controlled flight of the vehicle with onboard altitude estimates from the sensor at multiple setpoints (Sect. 4).

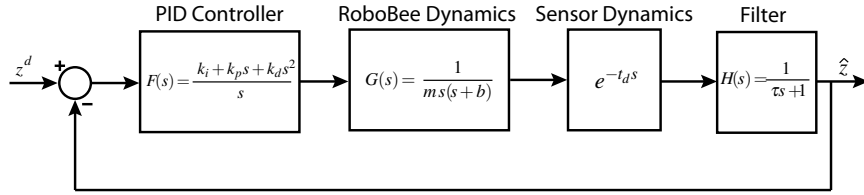


Fig. 3 The closed-loop dynamic model of the robot and sensor for perturbations away from the hovering setpoint. The PID controller (gains k_p , k_i , k_d) computes a thrust force F_T that minimizes error between the desired setpoint z^d and the measured altitude \hat{z} . This thrust force acts as an input to the dynamics of the robot (Eq. 1), forcing the robot to a new altitude. The sensor reads this new altitude with some time delay t_d and this response is filtered with a low pass filter with cutoff frequency $f_c = 20$ Hz.

2 Altitude Dynamics

The RoboBee used in these experiments was first presented in [17]. Piezoelectric actuators drive each of the wings, and the system has been shown to produce lift and body torques to take off, land, hover and perform aggressive maneuvers [6]. The hovering controller is composed of three sub-controllers – the lateral controller, the attitude controller, and the altitude controller. In this work, we eliminate the need for an altitude estimate from Vicon by providing it instead from an onboard sensor (see Fig. 2).

The altitude controller assumes that the robot is always in the upright orientation – an assumption that is maintained by the attitude controller. This requires that the thrust vector is always aligned with the z -axis, along the same axis as the gravitational force. Assuming that the robot maintains an attitude that is nearly upright, the controller can be designed around a linearization of its dynamics at hover, neglecting second-order effects arising from perturbations from zero attitude.

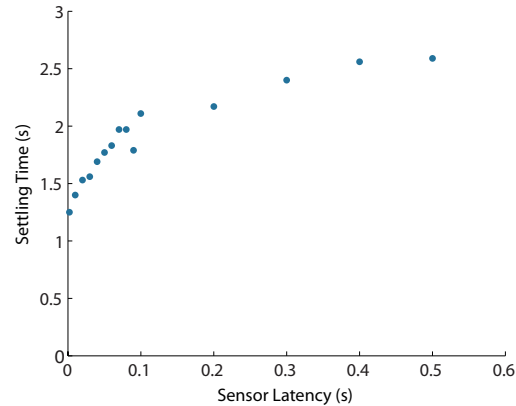
An understanding of the vehicle dynamics along the body z -axis is required to determine the maximum latency (sensor time delay) permissible for altitude control. The altitude dynamics can be described by a linear, second-order system:

$$\ddot{z} = \frac{F_T}{m} - b\dot{z} - g, \quad (1)$$

where F_T is the thrust force, m is the mass of the robot ($m = 110 \times 10^{-6}$ kg), b is the damping constant ($b = 1.2 \text{ s}^{-1}$ [5]), and g is the gravitational acceleration.

To determine the maximum latency in sensor estimates, we computed the closed loop poles of the system and determined the maximum settling time. The altitude controller is a proportional-integral-derivative (PID) controller that reduces error between a desired setpoint and the measured height of the robot. A feedforward term in the controller balances the gravitational force, and thus the input is the thrust force. The sensor latency is modeled using a second-order Padé approximation of pure time delay, and the raw sensor data is processed with a low-pass filter ($f_c = 20$ Hz). We simulated multiple time delays in the range $t_d = 0.01 - 0.5$ s (see Fig. 3).

Fig. 4 The system settling time with respect to sensor latency. Modeling the closed-loop system in Fig. 3, we calculated the closed loop poles of the system to determine settling time with various sensor dynamics. The controller gains at each trial were computed with the Ziegler-Nichols method and the phase margin is $\phi_m = 60^\circ$ for all trials.



As we are focused on determining the relationship between sensor latency and the settling time of the system, we used the Ziegler-Nichols method [2] to determine the controller gains of the system in order to respond to the changes in latency in a consistent manner.

Fig. 4 displays the settling time of the system plotted against various time delays of sensor estimates for the feedback system shown in Fig. 3. As expected, as the time delay of the sensor increases, the settling time also increases. It is interesting to note that the rate of increase in settling time decreases with sensor latency, suggesting that sensor latency will decrease system performance but does not lead to instability with proper tuning of the controller. With this relationship, we are able to determine the minimum sensor time delay for maneuvers that require specific settling times and analyze the tradeoffs between system performance and power and computation costs. For our current applications, a 2.5 s settling time is adequate and therefore sensor latency less than 0.5 s is reasonable.

3 Sensor Selection and Integration

The current vehicle has severe restrictions on payload mass and onboard power consumption. While [17] measured lift force sufficient to carry an additional 70 mg payload, the maximum payload carried by the robot during hovering flight was 40 mg, to allow for additional control authority in [10].

While the vehicle does not currently carry an onboard power supply, minimizing the power consumed by sensing is a major consideration for future applications. The vehicle consumes 19 mW of power during flight [16], consistent with similarly sized insects [8]. We are allotting 10 mW towards sensing and computation, based on power requirements of current high performance sensors.

In Table 1, we have tabulated a set of candidate sensors that meet the requirements listed above – accelerometers, IR ToF, optic flow, pressure, sonar, and IR

Table 1 Altitude sensors for Centimeter-Scale Flapping-Wing MAVs

Sensors	Mass	Power	Frequency	Remarks
Accelerometer [10]	24 mg	1 mW	1 kHz	Integration drift (2 m in 1 s)
Time-of-Flight [18]	20 mg	6 mW	50 Hz	VCSEL light source requires regulator
Optical Flow [9]	15 mg	15 mW	40 Hz	Computationally expensive
Pressure [20]	40 mg	3 mW	125 Hz	Insufficient resolution (20 cm)
Sonar [19]	80 mg	10 mW	40 Hz	Does not meet mass constraints
IR Range [14]	16 mg	6 mW	10 Hz	Dependent on reflective material

range detectors – that are currently available. Our evaluation is as follows: accelerometers lose accuracy over time due to the drift in sensor estimate after integration; additional velocity or distance sensors would need to be integrated onboard the vehicle to compensate, increasing the sensor payload of the vehicle. Optic flow sensors have demonstrated relative altitude control on the RoboBee [9]. Because these sensors only provide the angular velocity, which depends on both distance and velocity relative to an object, an additional sensor measuring absolute altitude or lateral velocity would need to be added to eliminate steady-state error [13]. Similarly scaled vehicles such as the Delfly [7] have demonstrated controlled altitude with absolute pressure sensors. The small vertical motion available (30 cm) in the flight arena is smaller than the precision of available pressure sensors which have a resolution of approximately 20 cm [20]. Sonar shows promise for altitude estimation, but does not yet meet the mass requirements for this robot. The IR range detector measures the amount of reflected light in the receiver. This sensor meets the mass and power requirements, but has a low feedback rate and is highly dependent on the material off of which the IR light is reflecting. We also found that the sensor signal can saturate in the presence of ambient light. The ToF sensor, while needing an external voltage regulator, has low mass, sufficient bandwidth, and low latency for altitude control in these experiments.

3.1 Time-of-Flight Sensor

As a first step in altitude control, we consider an IR ToF range detector (VL6180x, STMicroelectronics). Time-of-Flight sensors compute distance by measuring the time between the transmission and reception of an IR signal generated by the sensor. This distinguishes them from the more common IR range detectors that measure the amount of reflected light in the receiver. This was an important consideration given that the vehicle’s other degrees of freedom are currently sensed using Vicon motion capture cameras that emit IR light.

3.1.1 Integration

To minimize component weight, we made a custom flex circuit using a copper-clad polyimide film (18 μm copper, 12.7 μm kapton) with a capacitor on the power line to help regulate the charge close to the sensor (see Fig. 1 inset). Five 51-gauge copper wires (approximately 0.5 m long) provided power to the sensor and connect the data and clock lines for I2C communication. During operation, the sensor drew approximately 20 mA of current at regular intervals to pulse the IR VCSEL (Vertical-Cavity Surface-Emitting Laser) light source [18]. This amount of current traveling through the wire (approximately 80 Ω/m) caused the voltage at the sensor to drop below operating conditions. We connected a fifth wire to the input voltage line which served as a feedback line to an off board linear voltage regulator (NCP3337, ON Semiconductor). This provided a smooth input voltage to the sensor. The weight of the entire structure was 27 mg. We mounted the sensor on the base of the robot, with the transmitter and receiver directed at the ground (see Fig. 1). This position is close to the center of mass, thus preventing significant moments about any of the body axes. Because the sensor is light-based, there are no interactions between the robot's wing beat frequency and the sensor readings.

An ArduinoMega (ATMega256, Atmel Corporation) communicates with the sensor over I2C at approximately 60 Hz and sends the range response to a computer running xPC Target (Mathworks) through a serial (RS232) connection at 115 kbps. This communication scheme introduces about 20 ms of latency, 15 ms of which are dedicated to the sensor's ranging computation and 5 ms to communication. This produces a feedback rate of approximately 50 Hz, a tenth of the speed of the Vicon system that is currently being used to control the vehicle's altitude, but nominally sufficient according to the calculations in Sect. 2.

3.1.2 Calibration

We calibrated the sensor measurements by manually adjusting the height of the sensor while attached to the robot in the flight arena. We tracked the vehicle's altitude using the Vicon system while simultaneously recording the sensor's output. We made the decision to calibrate the sensor against the current Vicon estimates, as this Vicon system has shown sufficient accuracy for altitude control in previous experiments [17] [6]. Fig. 5 displays sensor output (scaled to meters) recorded against the reference height for a number of these trials. A line of best fit was calculated for this collection of measurements, $\hat{z} = Az + B$, where $A = 0.96$ and $B = 0.030$ mm. The slope of the line indicates that the sensor is accurately reading changes in altitude at the millimeter scale. The offset compensates for both the sensor's internal offset (the sensor is unable to detect distances less than 10 mm), as well as the calibration of the Vicon system. The estimates also become unreliable close to the surface as well as above 14 cm, providing an operating range for accurate altitude estimation. The error in the sensor estimates increases with height at a roughly constant 3% rate.

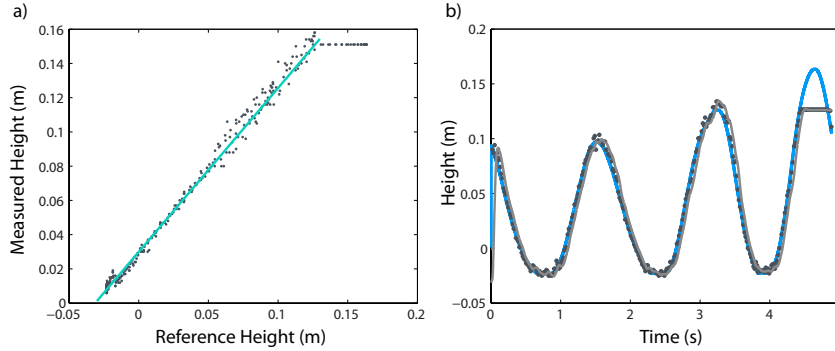


Fig. 5 Sensor Calibration. a) Sensor measurements (gray dots) were taken simultaneously with Vicon measurements (ground truth). A best fit line (teal) was computed: $\hat{z} = 0.96z + 0.030$ mm, where the offset is primarily due to the internal calibration of the sensor, which cannot detect distances less than 10 mm, and the calibration of the Vicon arena. b) Sensor measurements (gray dots) plotted on top of the Vicon measurement (blue line) with respect to time. The sensor measurements were filtered using a second-order low-pass Butterworth filter, $f_c = 16$ Hz (grey line). This filter introduced an 80 ms latency between Vicon and the sensor measurement.

Using this linear fit, the sensor measurements were plotted alongside the reference height over a period of approximately six seconds to characterize any drift the sensor may exhibit as well as to determine the effect of the latency on the estimated height. As expected, the estimated measurement aligns with the reference height over all time without any noticeable drift in the sensor. With this data, we also estimate that the sensor measurements have $t_d = 10$ ms of latency compared to the Vicon measurements. This latency increases to approximately $t_d = 80$ ms when the sensor measurements are filtered with a second-order low-pass Butterworth filter ($f_c = 16$ Hz), dictating a settling time of approximately 2 s, based on the results in Fig. 4.

4 Flight Experiments

The robot is not equipped with a power source, computational capability, or a full sensor suite, and is flown inside an controlled flight arena with a volume of approximately $0.3 \text{ m} \times 0.3 \text{ m} \times 0.3 \text{ m}$. Six motion capture cameras track the position and orientation of the robot during flight. The control computation is done on the xPC Target at a rate of 5 kHz. The xPC Target commands the power signal through a digital-to-analog converter and high voltage amplifiers. Power is supplied through a tether of four 54-gauge copper wires.

Flight testing begins with open-loop tuning to determine the torque biases about pitch and roll of the robot. To tune the robot, we begin by applying no net torque to the vehicle and observe the trajectory of the vehicle after takeoff. We then apply

trim values to oppose the observed torques. This process is repeated until the robot's takeoff is vertically upright.

For all flights mentioned in this section, the robot is attached to a three-filament kevlar thread and suspended above the surface to prevent wear on both the wings and wing hinges during crash landings. These filaments (approximately 30 cm in length) have negligible mass (0.2 mg) compared to that of the robot with the onboard sensor. In addition, we found that the filaments produce negligible torques on the robot during flight. A 10 cm thread cannot support its own weight ($20 \mu\text{g}$) when extended horizontally, indicating that the bundle cannot exert more than $0.02 \mu\text{Nm}$ of torque on the robot – small compared to the $0.35 \mu\text{Nm}$ of torque produced around the pitch and roll axes during hovering flight.

The altitude controller used during these flight experiments is described in detail in Sect. 2. The hovering flight experiments were done in multiple steps – first the vehicle was commanded to hover using Vicon estimates for altitude to tune the controller gains and provide a baseline flight performance for our results. Altitude was then controlled with estimates from the onboard sensor – four flights were commanded at an altitude of 8 cm to demonstrate repeatability and one flight at an altitude of 10 cm to test the range of the sensor. The attitude and lateral controllers used Vicon estimates of position and orientation for all flights. These are the first demonstrations of controlled altitude with onboard sensing in free flight.

4.1 Vicon Estimates

Once we determined the robot's torque biases, we had to determine the controller gains in subsequent closed loop experiments. To first ensure that the robot had sufficient control authority to hover stably about a setpoint, Vicon was used to provide the estimated altitude measurement. The additional 27 mg at the base of the robot requires larger thrust forces, and therefore larger flapping amplitudes, to lift off the ground. The black line in Fig. 6 gives an example altitude trajectory of the vehicle during a hovering flight. The vehicle reaches and maintains the altitude setpoint during the 6 s flight (RMS error between the trajectory and the setpoint is 7 mm).

4.2 Sensor Estimates

Having determined that the loaded robot has the control authority to stably hover around a set point, the next step was to determine if the sensor feedback rate was sufficient to control altitude during free-flight. In these flight experiments, the filtered sensor measurements provided input to the altitude controller, while Vicon estimates provided the lateral position and orientation of the robot to stabilize the attitude dynamics during flight. Fig. 6a provides an example of a hovering flight with feedback from the ToF sensor. The robot takes off and reaches the setpoint in

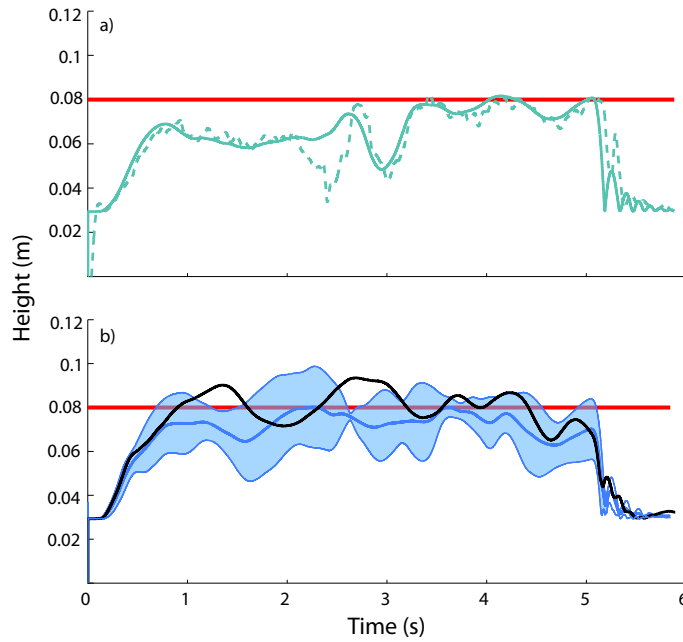


Fig. 6 Closed Loop Flight Experiments. a) The robot is commanded to reach the setpoint altitude of 8cm (red line) with the sensor providing altitude feedback. The sensor measurement (dashed) and the Vicon reference altitude (solid) are plotted. b) The mean and standard deviation of four hovering flight experiments with altitude estimation provided by the sensor are plotted (blue). For reference, the baseline hovering flight with altitude estimation from Vicon is plotted (black).

approximately three seconds. The sensor output tracks the Vicon measurement for the majority of the flight. The largest difference is the sharp downward peak between two and three seconds. This error is most likely due to one of the tethers obstructing the sensor's view of the ground. Three additional trials were performed at this desired setpoint. The mean and standard deviation of these flights can be seen in Fig. 6b alongside the Vicon baseline flight. In these trials, the robot is able to reach the setpoint and sustain hovering flight with an average RMS error of 13.75 mm from the setpoint. We then selected a second altitude setpoint to demonstrate the sensor's ability to provide accurate feedback at other altitudes (see Fig. 7). A sample flight sequence is shown in Fig. 8, where the robot reaches the desired setpoint between 1–2 s and maintains that altitude for the remainder of the flight.

Fig. 7 Sensor Feedback for Hovering Flight at 10 cm altitude. The flight is plotted in time, with the setpoint shown in red, sensor data (dashed line), and the Vicon data (solid line).

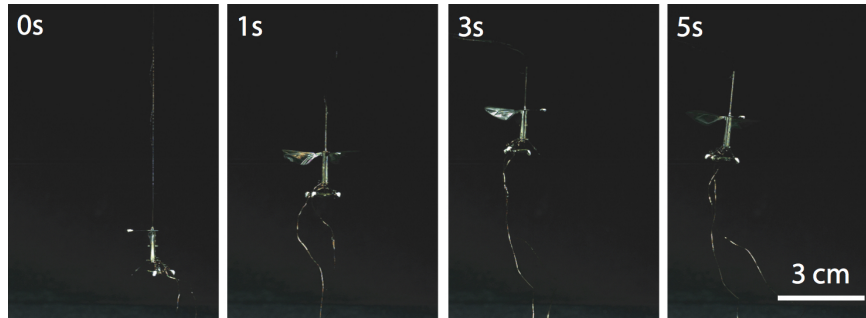
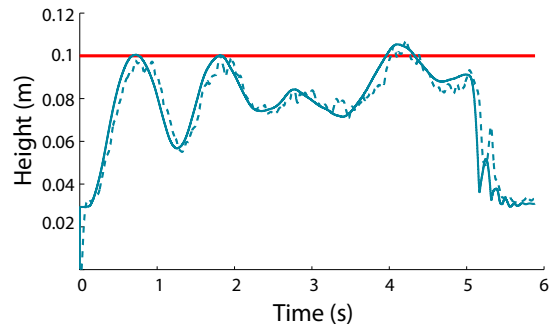
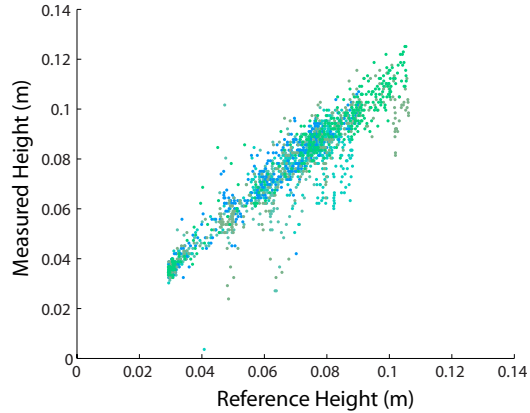


Fig. 8 Sample Flight Sequence. The robot begins at the start altitude of approximately 3 cm, begins flying to the desired altitude at 1 s, reaches the setpoint and maintains that altitude for the remainder of the flight, as demonstrated by the images at 3 s and 5 s.

4.3 Discussion

Closed loop flights with sensor estimates in the feedback loop controlled the altitude of the robot during hovering flight with altitude error of less than 1.5 cm, or within a half body length of the robot. These experiments also demonstrated that controlled hovering with sensor estimates in the feedback loop had twice the RMS error of the flights with Vicon feedback. To investigate this error, we verified the sensor measurements against the reference height from Vicon (see Fig. 9). The sensor performed as expected across all flights, with less than 5% of the measurements deviating from the linear trend. Outliers where the sensor measurement was lower than the expected value can be explained by an occluded view of the ground from the tether, and areas where the sensor measurement were higher are due to the attitude of the vehicle moving away from the vertical axis. Practically, the robot does not tilt away from the vertical axis more than 15° . At an altitude of approximately 10 cm, this second-order effect will cause deviations in sensor measurement of 4 mm larger than the true value. In the future, accurate attitude estimation may not be available and reliance on these estimates for correction may be impractical. However, given the limited number of outliers, this effect is negligible and the source of error is not sensor measurement.

Fig. 9 Sensor Calibration Verification. The sensor measurements for all flight experiments are plotted against the Vicon ground truth estimate. The linear trend with unity slope held for all flights, with outliers accounting for less than 5% of all measurements.



Previous studies [11] have shown that the power tether can produce large torques on the robot during flight, making the vehicle more difficult to control during hover. The addition of a second power tether for sensor communication only increases this effect. Additional tuning will be necessary to compensate for these effects.

5 Conclusions

Altitude sensing is a necessary component for vision-based navigation. In this work, we determined a relationship between sensor latency and settling time for the closed-loop altitude dynamics of this vehicle, elucidating the feedback requirements for various altitude maneuvers. With this information, we selected a sensor that met the mass, power, and latency constraints of the vehicle to perform altitude estimation in free flight. The closed loop flights with sensor estimates in the feedback loop are the first demonstrations of controlled altitude with onboard estimation for an insect-scale robot in free flight. These flights, coupled with the results from the calibration experiments, demonstrate that the sensor is able to accurately measure altitude, but further work is needed in tuning the controller gains to lower the average error about the desired setpoint.

The low mass of the sensor allows for additional payload, which can be used for combinations of sensors. Future work into sensory fusion, including the integration of an IMU to estimate attitude (as demonstrated in [10]) will enable the RoboBee to perform short hovering flights with only onboard sensory information. Additionally, autonomous visual navigation can be achieved with the integration of an onboard optic flow sensor in combination with an attitude estimate from the IMU and an absolute distance measure from the proximity sensor. These experiments will be the first demonstrations of sensor autonomy on an at-scale robotic insect. This work has provided an important step towards this goal.

This work can also be applied to other MAVs, especially those with stringent payload, power or computational requirements. Given the short range of this sensor, this sensor is impractical for altitude control over a large variation in height. However, a ToF sensor could provide a precise measure of proximity and be combined with a pressure sensor to provide a coarse altitude estimate. We can envision this sensor being used for close range object detection or obstacle avoidance.

Acknowledgements The authors would like to thank Neel Doshi for his help in developing the model for the closed loop dynamics. This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. (DGE1144152), and the Wyss Institute for Biologically Inspired Engineering. 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. Malcolm J. Abzug and Eugene E. Larrabee. *Airplane stability and control, Second Edition: A history of technologies that made aviation possible*. Cambridge University Press, Cambridge, MA, 2002.
2. Karl Johan Astrom and Richard M. Murray. *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton University Press, 41 William Street, Princeton, NJ 08540, 2008.
3. Stanley S Baek, FL Garcia Bermudez, and Ronald S Fearing. Flight control for target seeking by 13 gram ornithopter. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 2674–2681. IEEE, 2011.
4. Antoine Beyeler, Jean-Christophe Zufferey, and Dario Floreano. Vision-based control of near-obstacle flight. *Autonomous Robots*, 27:201–219, 2009. 10.1007/s10514-009-9139-6.
5. Pakpong Chirarattananon. Flight control of a millimeter-scale flapping-wing robot. 2014.
6. Pakpong Chirarattananon, Kevin Y Ma, and Robert J Wood. Adaptive control of a millimeter-scale flapping-wing robot. *Bioinspiration & biomimetics*, 9(2):025004, 2014.
7. Christophe De Wagter, Sjoerd Tijmons, Bart DW Remes, and Guido CHE de Croon. Autonomous flight of a 20-gram flapping wing mav with a 4-gram onboard stereo vision system. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 4982–4987. IEEE, 2014.
8. Robert Dudley. *The biomechanics of insect flight: form, function, evolution*. Princeton University Press, 2002.
9. P.-E.J. Duhamel, N.O. Perez-Arancibia, G.L. Barrows, and R.J. Wood. Biologically inspired optical-flow sensing for altitude control of flapping-wing microrobots. *Mechatronics, IEEE/ASME Transactions on*, 18(2):556–568, 2013.
10. Sawyer B Fuller, E Farrell Helbling, Pakpong Chirarattananon, and Robert J Wood. Using a mems gyroscope to stabilize the attitude of a fly-sized hovering robot. In *IMAV 2014: International Micro Air Vehicle Conference and Competition 2014, Delft, The Netherlands, August 12-15, 2014*. Delft University of Technology, 2014.
11. Sawyer B. Fuller, Michael Karpelson, Andrea Censi, Kevin Y. Ma, and Robert J. Wood. Controlling free flight of a robotic fly using an onboard vision sensor inspired by insect ocelli. *Journal of the Royal Society Interface*, 2014. (in press).
12. E Farrell Helbling, Sawyer B Fuller, and Robert J Wood. Pitch and yaw control of a robotic insect using an onboard magnetometer. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 5516–5522. IEEE, 2014.
13. James Sean Humbert. *Bio-inspired visuomotor convergence in navigation and flight control systems*. PhD thesis, California Institute of Technology, 2005.

14. Intersil. Digital proximity sensor with interrupt function. *Data Sheet*, 2009.
15. Vijay Kumar and Nathan Michael. Opportunities and challenges with autonomous micro aerial vehicles. *The International Journal of Robotics Research*, 31(11):1279–1291, 2012.
16. Kevin Y. Ma, Pakpong Chirarattananon, Sawyer B. Fuller, and Robert Wood. Controlled flight of a biologically inspired, insect-scale robot. *Science*, May 2013.
17. Kevin Y. Ma, Samuel M. Felton, and Robert J. Wood. Design, fabrication, and modeling of the split actuator microrobotic bee. In *Proceedings of the IEEE Int Robotics and Automation Conference*, 2012.
18. ST Microelectronics. VL6180x. proximity and ambient light sensing (als) module. *Data Sheet*, 2014.
19. Murata. Ma40h1s-r. *Product Specification of Ultrasonic Transducer*, 2014.
20. Measurement Specialties. Ms5607-02ba03. barometric pressure sensor, with stainless steel cap. *Data Sheet*, 2012.
21. Mandyam V Srinivasan. Honeybees as a model for the study of visually guided flight, navigation, and biologically inspired robotics. *Physiological Reviews*, 91(2):413–460, 2011.