

Battery State-Of-Charge Based Altitude Controller for Small, Low Cost Multirotor Unmanned Aerial Vehicles

Michal Podhradský · Calvin Coopmans ·
Austin Jensen

Received: 1 September 2013 / Accepted: 11 September 2013 / Published online: 2 October 2013
© Springer Science+Business Media Dordrecht 2013

Abstract Small unmanned aerial vehicles (UAV) are typically driven by Lithium polymer batteries. The batteries have their own dynamics, which changes during discharge. Classical approaches to altitude control assume a time-invariant system and therefore fail. Adaptive controllers require an identified system model which is often unavailable. Battery dynamics can be characterized and used for a battery model-based controller. This controller is useful in situations when no feedback from actuators (such as RPM or thrust) is available. After measuring the battery dynamics for two distinct types of batteries, a controller is designed and experimentally verified, showing a consistent performance during entire discharge test and a consequent flight verification.

Keywords Unmanned aerial vehicles · UAV · Vertical take-off and landing · VTOL · Hexarotor · Multirotor · Altitude control · Battery monitoring and modeling

1 Introduction

Robust altitude control of a multirotor Unmanned Aerial Vehicle (UAV) is one of the most difficult control problems of vertical take-off and landing (VTOL) UAVs, of which the most common are quadrotors and hexarotors.

Small low cost UAVs are typically driven by Lithium-ion polymer (LiPo) batteries, because of their high energy density, high charge and discharge rates, long lifetime, lack of memory effect [18] and affordable cost. Inherent dynamics of Lipo batteries changes during discharge and degrades the flight and control performance.

This change in dynamics affects the UAV's ability to maintain desired altitude. The goal of this paper is to:

- Analyze a battery and an actuator dynamics
- Describe its effect on the flight performance
- Briefly summarize existing control strategies for altitude control
- Propose a novel altitude controller and a battery monitoring system for small UAVs
- Experimentally verify the controller performance

First, an overview of UAV platform architecture can be found in Section 1.1. A necessary background about UAV altitude control is given in Section 1.2. Battery and actuator models are described in Section 2. A summary of existing solutions is presented in Section 3. The proposed

M. Podhradský (✉) · C. Coopmans · A. Jensen
Utah Water Research Laboratory,
Utah State University, Logan, UT, USA
e-mail: michal.podhradsky@aggiemail.usu.edu
URL: <http://aggiemail.usu.edu>

controller is described in Section 4. The experimental set-up is described in Section 5.1 and the laboratory experimental results which prove efficiency are shown in Section 5.4. Implementation of the controller on an existing UAV and its flight verification is presented in Section 6.

1.1 AggieAir Platform

AggieAir [12, 13] is a small, low-cost, autonomous, multi spectral remote sensing platform [17], which has been developed over the last few years at the Center for Self Organizing and Intelligent Systems (CSOIS) at Utah State University (USU).

AggieAir has reached a stable and robust level in development and has begun work on various applications through a service center at the Utah Water Research Laboratory (UWRL [3]) at USU. AggieAir utilizes both VTOL and Fixed wing platforms. Example UAVs, a current hexarotor platform is shown in Fig. 1, and a current fixed wing platform is shown in Fig. 2. An open-source Paparazzi autopilot [2] is used for flight control, with consumer grade electronics and sensors, while maintaining excellent flight characteristics and reliability [14].

Avionics of such a small low cost UAV consist of inertial measurement unit (IMU), which measures acceleration, angular rate and magnetic field in three axis, attitude heading and reference



Fig. 1 AggieAir multirotor platform—hexarotor, ready for an indoor flight



Fig. 2 AggieAir fixedwing platform—minion, during landing maneuver

system (AHRS) which combines IMU measurements and provide attitude estimation, and a GPS sensor providing an absolute position altimeter (altitude above mean sea level), pressure sensors for precise altitude estimation relative to a certain set point. Optionally an inertial navigation system (INS) which combines measurements from all mentioned sensors are fused together to estimate attitude and position can be used instead of AHRS [6].

In addition a radio transmitter/receiver is necessary for telemetry and remote control. The autopilot unit runs control loops on-board and controls the actuators to keep the desired the attitude and altitude. An overview of the AggieAir system is shown in Fig. 3. Control loops of a small VTOL UAV are described in [7].

AggieAir platform was chosen as an example of a typical small low-cost UAV. However, the control system presented in this paper is directly applicable to any multirotor UAV with similar architecture.

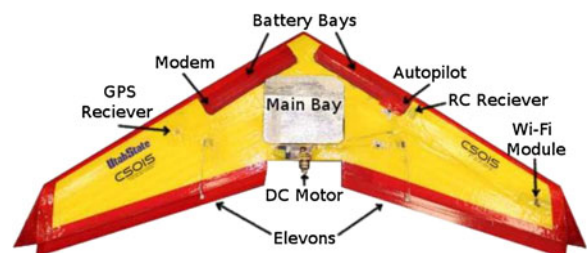


Fig. 3 AggieAir platform overview—Deltawing

1.2 Altitude Control Obstacles

Precision agricultural applications, 3D mapping and other civilian applications of UAVs require a precise altitude control (i.e. within 1 m [17]).

Relative altitude estimation in multirotor UAVs is typically based on measurements from an accelerometer and a pressure sensor. GPS provides absolute position, but its vertical accuracy is rarely better than few meters in perfect conditions so its not commonly used. Pressure sensors have resolution of ± 10 cm, but they drift in time, thus introducing error into the relative estimate. Acceleration data are burdened with noise from motor induced vibrations, thus they have to be corrected with pressure readings. Additional sensors (such as ultrasonic or laser altimeters) can be used, however their use is limited to close proximity of the ground. Note that in altitudes above 20 m, the UAV is usually subject to strong wind gusts (≥ 10 m/s) and pressure changes [17], which further affect the controller performance.

A traditional approach combining accelerometer and pressure sensor data and using a classical PID control system can provide satisfactory altitude hold despite the changing battery dynamics. However, its performance strongly depends on properly tuned altitude estimator, size of the platform (heavier UAVs with higher inertia are more difficult to control [23]), and the operational altitude.

The proposed controller offers an alternative approach, which can be used when altitude estimation cannot be tuned, electronic speed controllers (ESC) do not provide feedback loop around Revolutions Per Minute (RPM) of the motors, or when classical PID control fails (Fig. 4).

The significance of the battery dynamics for larger (>1 kg) platforms can be seen in Fig. 5. It shows a radio controlled indoor flight of a hexarotor at constant altitude. A 4-cell LiPo battery is fully charged at the beginning, and discharges during this 19 min flight. The pilot increases throttle command to keep constant power output from the battery and thus constant altitude. During the flight, the pilot had to increase the throttle by around 10 % (comparing the beginning and end of flight). After the 18th minute of flight, the

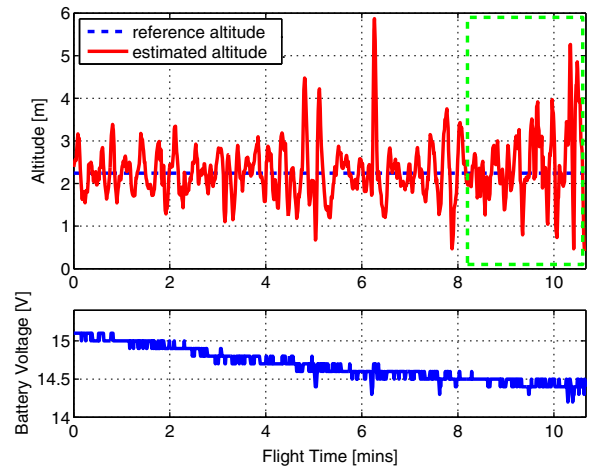


Fig. 4 *Top*: altitude tracking during autonomous outdoor hexarotor flight—the green box shows increasing oscillations. *Bottom*: closed loop battery voltage

battery voltage suddenly drops as the battery is almost completely depleted and the voltage begins to collapse. Before the hexarotor lands the voltage dropped even below the minimal recommended limit for LiPo batteries, 12 V (3 V for each cell) which could damage the battery.

Clearly it is important to know the battery state of charge for safety reasons, as well as to adjust the control according to the battery dynamics.

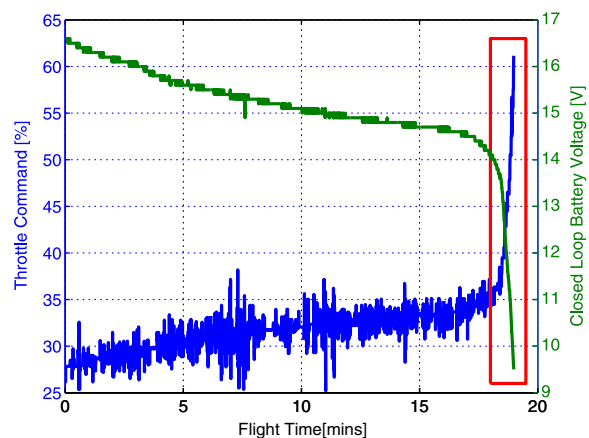


Fig. 5 Change in throttle command during indoor hexarotor flight—the red box shows where the battery is starting to collapse

2 Battery and Actuator Models

An actuator of a multirotor UAV typically consists of a LiPo battery, a brushless direct current (BLDC) motor and an ESC which controls the RPM of the motor.

2.1 Battery Model

An equivalent circuit representation of the Chen and Mora's battery model [9] is shown in Fig. 6. Although the model was originally validated on NiMH and Li-ion batteries, it can still be successfully applied to LiPo batteries, since LiPo and Li-ion batteries have very similar characteristics [21].

The left half of the model describes the variation of the state of charge (SOC) of the battery ($x_1 = \text{SOC}$, $x_1 \in [0, 1]$), the right half models the variation of battery output voltage y as a function of the charge/discharge current [21]. Note that SOC is the percentage of the maximum possible charge that is present inside a rechargeable battery [4]. All the circuit components C_{ts} , C_{tl} , R_s , R_{ts} , R_{tl} and E_0 are functions of x_1 :

$$C_{ts}[\text{F}] = -k_4 e^{-k_1 x_1} + k_3 \quad (1)$$

$$C_{tl}[\text{F}] = -k_6 e^{-k_2 x_1} + k_5 \quad (2)$$

$$R_s[\Omega] = k_7 e^{-k_8 x_1} + k_9 \quad (3)$$

$$R_{ts}[\Omega] = k_{10} e^{-k_{11} x_1} + k_{12} \quad (4)$$

$$R_{tl}[\Omega] = k_{13} e^{-k_{14} x_1} + k_{15} \quad (5)$$

$$E_0[\text{V}] = -k_{16} e^{-k_{17} x_1} + k_{18} + k_{19} x_1 - k_{20} x_1^2 + k_{21} x_1^3 \quad (6)$$

$$C_c[\text{C}] = 3600 C_f f_2 \quad (7)$$

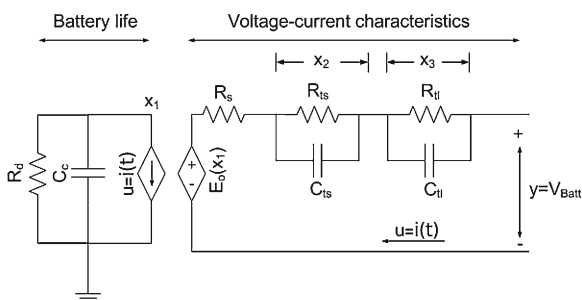


Fig. 6 Chen and Mora's battery model

where $k_i > 0$ for $i = 1 \dots 21$ and $k_1 < k_2 < k_3 < k_4 < k_5 < k_6$. E_0 is the open-circuit voltage (OCV) of the battery. In Eq. 7 $f_1, f_2 \in [0, 1]$ are factors taking into accounts the effects of temperature and charge-discharge cycles [21]. $C[\text{A}]$ stands for maximal battery capacity. For a new battery at a room temperature $f_1 = f_2 = 1$, but f_2 will decrease after each charge-discharge cycle (in fact representing state of health—SOH of the battery). This allows the model to account for ageing of batteries, which is characterized by loss of maximal charge the battery can store [22]. The various resistances, capacitances, and constants ($k_1 \dots k_{21}$) are independent of $i(t)$. Note that with decreasing SOC resistance of the battery (R_s, R_{ts}, R_{tl}) increases and capacitance (C_{tl}, C_{ts}) decreases [9].

State space realization of the battery model is provided by [19] as:

$$\dot{x}_1 = -\frac{1}{C_c} i \quad (8)$$

$$\dot{x}_2 = -\frac{x_2}{R_{ts} C_{ts}} + \frac{i}{C_{ts}} \quad (9)$$

$$\dot{x}_3 = -\frac{x_3}{R_{tl} C_{tl}} + \frac{i}{C_{tl}} \quad (10)$$

$$y = E_0 - x_2 - x_3 - i R_s \quad (11)$$

y represents the voltage output from the battery (if $i = 0$, $y = \text{OCV}$), x_2 represents the voltage drop across $R_{ts} \parallel C_{ts}$ and x_3 stands for the voltage drop across $R_{tl} \parallel C_{tl}$; $x_2, x_3 \in \mathbb{R}$ and $\mathbf{x}_0 = [1, 0, 0]^T$

The OCV(E_0) is a function of SOC(x_1), which itself depends on temperature and SOH. Assuming a constant ambient temperature, the dependence of OCV on SOC has the following properties:

1. The OCV increases monotonically with SOC
2. The OCV drops in an exponential fashion as the SOC approaches 0 %
3. The OCV rises in an exponential fashion as the SOC approaches 100 %
4. In the large region between 20 % and 85 % of SOC the relation is nearly linear

which are captured in the battery model. Figure 7 shows an experimental verification of the OCV vs. SOC dependency, which has the predicted properties. The SOC was measured using Coulomb

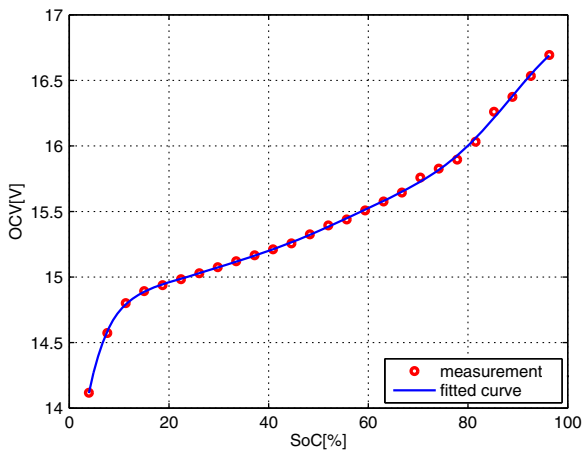


Fig. 7 Dependency of OCV on SOC

counting technique (i.e. current integration), more details can be found in [22].

The properties of LiPo battery dynamics can be summarized as:

1. Open Circuit Voltage decreases with SOC (discharge)
2. Closed Loop Voltage decreases with increased load and with SOC
3. Internal battery resistance increases with lower SOC
4. Battery capacity decreases as the battery ages (SOH is becoming lower)

For the purpose of this work, only new batteries are assumed, and no SOH is estimated.

2.2 Actuator Model

For actuation, BLDC motors are typically used, since they provide higher efficiency and more torque per Watt than brushed DC motors [25]. BLDC motors for UAVs are almost always sensor less trapezoidal motors with stator coils in the core of the motor (outrunners). It is because of a combination of lower cost (no position sensor needed), and higher efficiency over sinusoidal/inrunner motors [25]. Motors are characterized by their number of magnetic poles and a motor constant [11]. An example of the actuator is shown in Fig. 8.

To drive a BLDC motor an electronic speed controller is used. ESC consists of a controller,

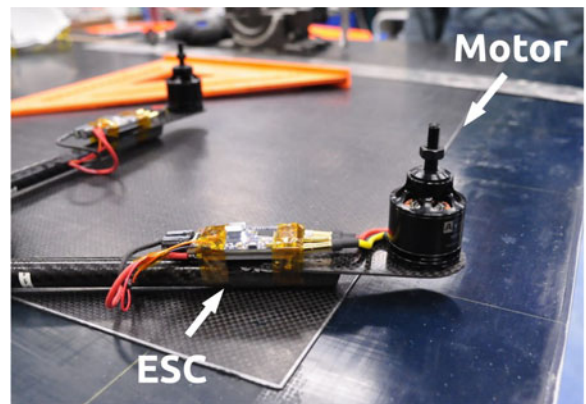


Fig. 8 An example of a multirotor actuator—Av-Roto BLDC motor (without a propeller) and Mystery 40A ESC

MOSFET driver and at least six MOSFET transistors. To sense the position of the motor, back electromotive force (back-emf) is fed to the micro controller. An illustrative diagram of a typical ESC is shown in Fig. 9. The driver used pulse width modulation (PWM, typically 16 kHz) to open/close individual MOSFETs, according to commands from the micro controller. MOSFETs then feed the phase with current.

Such actuator connected to a power supply, can be described as a first order system with delay [11]:

$$G(s) = \frac{K_p}{1 + T_p} e^{-T_d s} \quad (12)$$

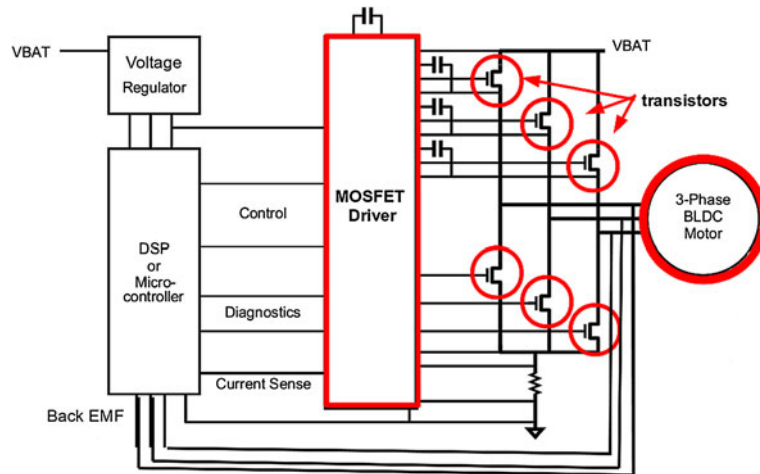
where T_d is a communication delay between the autopilot and the ECS. If the control rate is 512 Hz, $T_d = 1.95$ ms. T_p is typically between 16 and 100 ms. K_p is strongly dependent on the combination of a motor and a propeller [11]. The produced thrust directly depends on RPM.

However, battery dynamics affects the actuator and the real system is no longer time invariant (LTI). To understand this relationship, the ESC functionality has to be described in more detail. BLDC motor is rotated by induced voltage on its armature. For trapezoidal motors, a square current wave is used and the relation between current and induced voltage is as follows [25]:

$$V_k(t) = R_k i_k(t) + L \frac{di_k(t)}{dt} \quad (13)$$

where $k = 1 \dots 3$ is a number of phase, R is resistance of the wiring and L is inductance of the

Fig. 9 A typical ESC and motor connection scheme. Micro controller (left), MOSFET driver (middle), MOSFETs and BLDC motor on the right [1]



coil. The motor speed depends on the duty cycle of the MOSFET driver, and on energy delivered to it during one cycle. Energy accumulated in the inductor during one cycle:

$$W[J] = L \int_0^T i dt \quad (14)$$

T is constant as the duty cycle is the same. From Eqs. 3, 4 and 5 can be seen that the resistance of the battery increases during discharge. Since the battery is not an ideal current source, the rate of change of the supplied current is limited. With increased resistance, the maximal rate of change of the current become lower. Energy delivered to the motor during one cycle will decrease, so does RPM of the motor.

3 Comparison of Existing Solutions

In this section the most common multirotor altitude control methods are compared. They can be divided into classical control (PID regulator with feedforward) which assumes linear time invariant (LTI) system and Adaptive Control which tackles the problem of time-varying system.

There is a number of vision-based [26, 27] and visual servoing altitude control techniques [10, 15] which can be implemented. However, they are not investigated in this work, because the typical mission is assumed to be outdoors, in higher altitudes (up to hundreds of meters) and in rural

area or wilderness. In such environments it is hard to guarantee sufficient amount of distinct features in the camera image during whole mission, which would affect the control performance.

3.1 Classical Control

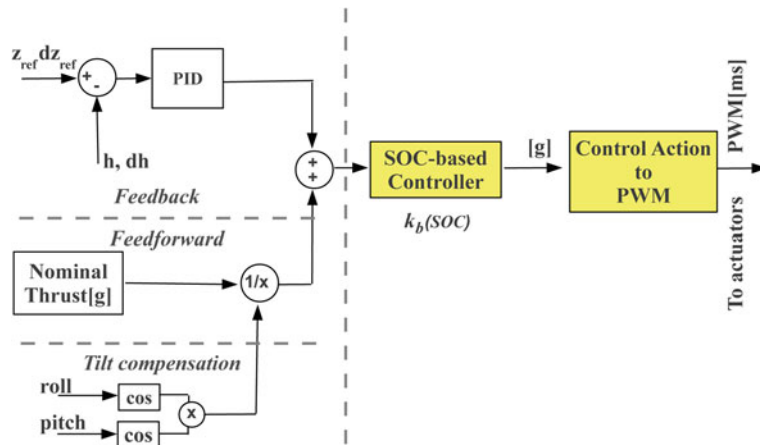
The most common altitude control system used in multirotor UAVs is a PID regulator with feedforward terms. Feedforward is set manually and gives a baseline of thrust to be applied to keep a UAV in constant altitude because the PID feedback control input is small in comparison with nominal thrust. Although very simple to implement, this technique often does not provide acceptable performance because the real system is time-variant. In other words, if the controller is tuned for a full battery pack, performance degrade as the battery is depleted, as can be seen in Fig. 4.

The main advantage of classical control is simplicity (no system model required, can be tuned experimentally). However it can be used only for application with weak requirements on altitude control.

3.2 Adaptive Control

Unlike simple PID control, adaptive controllers require a kind of system model. Although a model of multirotor dynamics is known [7, 24], identifying the model is a tedious process. The actuator, consisting of an ESC and a BLDC motor

Fig. 10 Battery based altitude control diagram—with the battery compensation block



(described in Section 2.2), can be identified separately from the rest of the system [11] if necessary.

Adaptive control approaches can be divided as follows:

3.2.1 PID + Adaptive Feedforward

The aforementioned PID controller can be augmented with adaptive feedforward. Adaptively estimated is nominal thrust, required for hover. Full model and more details are given in [2]. A Kalman filter with kinematic model is used, so no knowledge about the multirotor model is necessary. However, the controller still has to be tuned for a specific airframe.

3.2.2 Model Predictive Control

Model predictive control (MPC) is another option for altitude control. Although promising better performance, it requires full model of the UAV, which can be difficult to obtain [5].

3.2.3 Sliding Mode Control

Another popular control solution is sliding mode control [8, 20]. Again, an identified model of the UAV is required.

4 Battery State-Of-Charge-Based Controller

The State-Of-Charge-based controller extends the classical PID controller with constant feedforward (see Section 3.1) with gain scheduling compensat-

ing for battery dynamics. The scheduling is based on the actual SOC of the battery. The advantage is that no state-space model or a transfer function of the system is needed, only the battery dynamics has to be measured.

Mathematically it can be expressed as (assuming a PID controller with feedforward):

$$u(t) = \left(k_p e + k_I \int_0^t e(\tau) d\tau + k_D \frac{de(t)}{dt} + k_{ff} \right) \times k_b(SOC) \quad (15)$$

Figure 10 shows the block diagram of the proposed controller. The feedback section and tilt compensation is unchanged, and the battery compensation block (function $k_b(SOC)$) is added to the feedforward line. The battery dynamics measurements are described in next section. The main advantage is that the controller compensates for changes in the time-varying system and its performance is consistent. The actual function $k_b(SOC)$ is to be characterized in the next section.

5 Battery Dynamics Measurement

In order to design the SOC-based controller, the battery dynamics must be measured. For this reason an experimental test bench was built and batteries were characterized, as described in this section.

5.1 Instrumentation

In order to measure thrust of the actuators and SOC of the battery, a test bench based on [11] was

developed. The data acquisition and interface to sensors is done by an Arduino MEGA 2560 with a custom expansion board. The testbed solid model is shown in Fig. 11.

Force (Measurement Specialties FC2231) and current (Allegro MicroSystems ACS756SCA-050B) analog sensors are filtered with resistor-capacitor filters to prevent excessive noise. The force sensor error is $\pm 3.25\%$, the current sensor error is $\pm 5\%$ according to the data sheets. The whole system captures data at 12 Hz and sends them to the computer via USB, with post processing done in MATLAB. The actuator consists of Mystery 40A ESC, T-motor MT2814 KV770 motor and $12 \times 3.8''$ propeller, which is a suitable combination for quad or hexarotor. The ESC is controlled from an Arduino PWM port at 50 Hz rate. The complete test bed prepared for the measurement is shown in Fig. 12.

Two different 4-cell LiPo batteries were used: Zippy 5000 mAh 40 C and MaxAmps 11000 mAh 40 C. Both batteries are shown in Fig. 13.

5.2 Experimental Set-Up

In order to measure the battery dynamics, the following experiment was conducted. The actuator was set to a constant throttle of 55 % ($PWM = 1.54$ ms), which produces around 1000 grams of thrust for a fully charged battery. The change in throttle was measured as well as current and battery voltage. The experiment ran until the battery was depleted, which was clearly marked by a sudden decrease in battery voltage, right before the collapse (i.e. until the closed loop voltage dropped below 12 V).

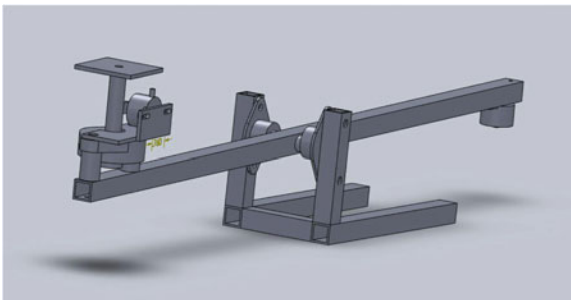


Fig. 11 3D Model of the test bench

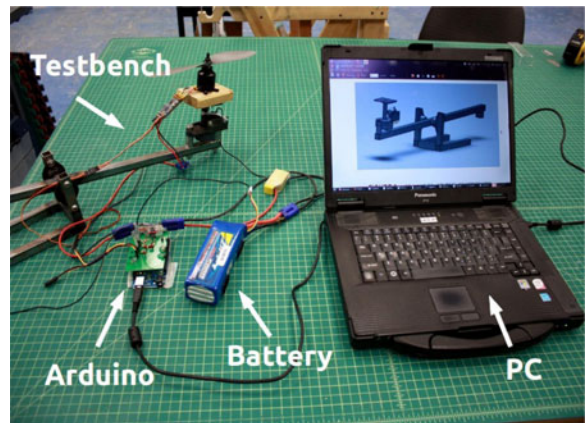


Fig. 12 Experimental setup: test bench, Arduino board and computer

Measured battery discharge current is shown in Fig. 14. The current was integrated as:

$$Q_i(t) = \int_0^t i_b(\tau) d\tau \quad (16)$$

The percentage of remaining SOC is defined as [16]:

$$SOC(t) = 100 \left(\frac{Q_c - Q_i(t)}{Q_c} \right) \quad (17)$$

where Q_c is the maximal current capacity present when $SOC = 100\%$. Note that for 11 Ah battery (MaxAmps): $Q_c = 11[A] \times 3600[s] = 39600[C=A \times s]$. For 5 Ah battery (Zippy): $Q_c = 18000[C]$.



Fig. 13 Top: Zippy 5000 mAh 40 C 4-cell, Bottom: Max-Amps 11000 mAh 40 C 4-cell Lipo batteries

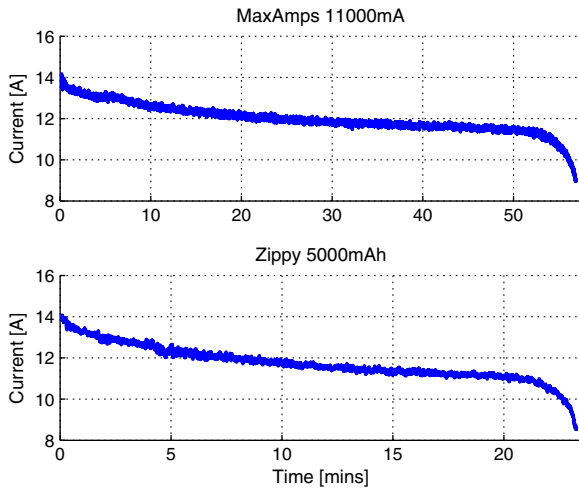


Fig. 14 Battery discharge experiment—measured current during discharge

Note that as the batteries were new, their nominal maximal capacity was used. The discharge experiment is shown in Fig. 15. The thrust is proportional to the battery power output $F(\text{grams}) \propto P(\text{Watts}) = I(\text{Amps}) \times U(\text{Volts})$.

To obtain conversion from grams of thrust to PWM command, the actuator must be characterized [11]. Such conversion is necessary for the experimental verification of the controller,

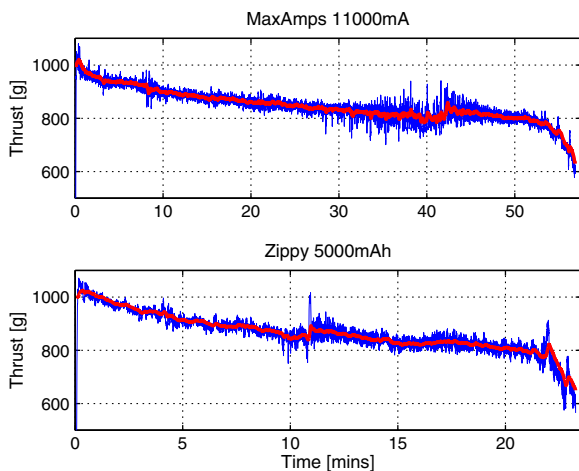


Fig. 15 Battery discharge experiment—thrust variations during discharge (blue: raw data, red: filtered data). Filtered with exponential moving average (EMA) filter, $\alpha = 0.01$

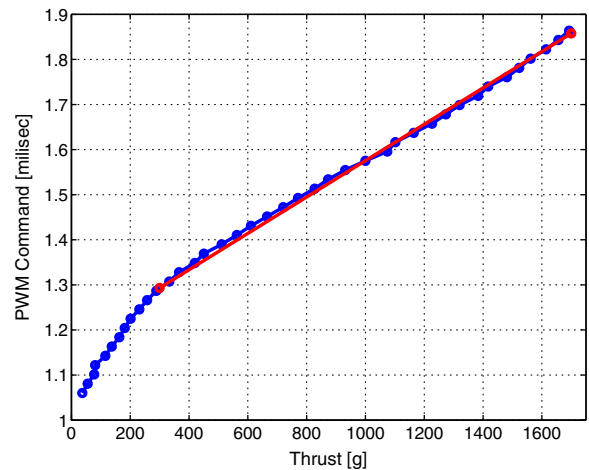


Fig. 16 Dependency of thrust on PWM command width (Mystery 40A ESC, T-motor MT2814 KV770 motor and 12×3.8 propeller), blue: measured data, red: linear approximation

when thrust set point (instead of an altitude set point) is used (see Section 5.4). The actuator was connected to a power supply, simulating a fully charged battery. PWM command was changed to cover whole admissible range of the ESC (1.1–1.9 ms) and produced thrust was measured. The data for each command step was averaged to obtain the resulting plot in Fig. 16. The measured data was approximated with a linear

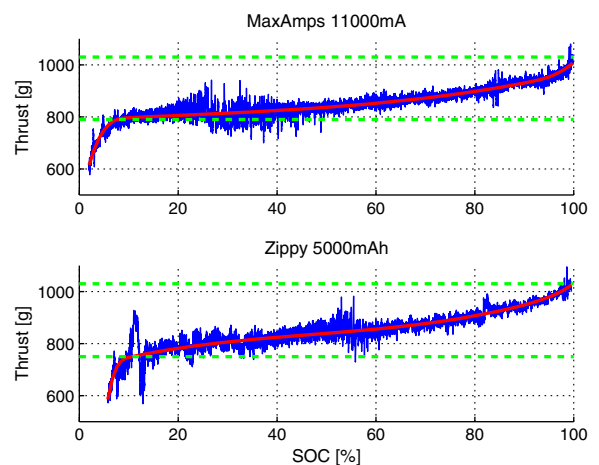


Fig. 17 Thrust dependency on SOC (blue: raw data, red: least-square spline approximation, green: Thrust at 10 % and 90 % SOC)

Table 1 Least-squares spline approximation for thrust measurements

Order	# Knots	Knot 1		Knot 2		
4	2	10% SOC		90% SOC		
Battery	c_1	c_2	c_3	c_4	c_5	c_6
MaxAmps	615.45	791.79	819.87	836.18	955.19	1,006.53
Zippy	585.81	735.03	859.25	811.60	983.17	1,025.00

function ($y = ax + k$, $a = 4.0323 \times 10^{-4}$, $k = 1.1722$, $x \in (300, 1700)$) to avoid non-linearity.

5.3 Controller Identification

The noisy force and current measurements were interpolated using least-squares approximation to obtain dependency of thrust on SOC. The end of the battery pack is identified as when the closed-loop-voltage (CLV) drops below 12 V. Knowing the 5 % measurement error of the current sensor, the estimated SOC aligns well with the battery capacity. Due to the inherent error in measurements, the flight should be terminated at 10 % SOC, so the observed voltage drop does not occur.

The dependency of produced thrust on battery SOC is shown in Fig. 17. The overall change in thrust (100 %–10 % SOC) is about 20 % for MaxAmps battery and about 25 % for Zippy battery. The thrust curve is almost linear on this range of SOC, except for an exponential drop

from fully charged battery to 90 %, and then another drop before the battery collapses (below 10 % SOC). The measured thrust was interpolated using least-squares spline approximation with coefficients from Table 1, the two knots were chosen to separate the almost linear piece and two highly non-linear parts.

Assuming that the change in thrust over the SOC is identical for whole range of throttle, it can be normalized. The normalized spline approximation is shown in Fig. 18. To obtain function $k_b(SOC)$ (“Thrust-Bonus”) of the battery compensator, the normalized throttle curve must be inverted.

To avoid computing a non-linear curve, the inverted thrust bonus is approximated with a piecewise linear function ($y = ax + k$), divided into four segments. The original and linearized curve is shown in Fig. 19, the parameters of piecewise linear function, including root-mean-square (RMS) error of the approximation, are in Table 2.

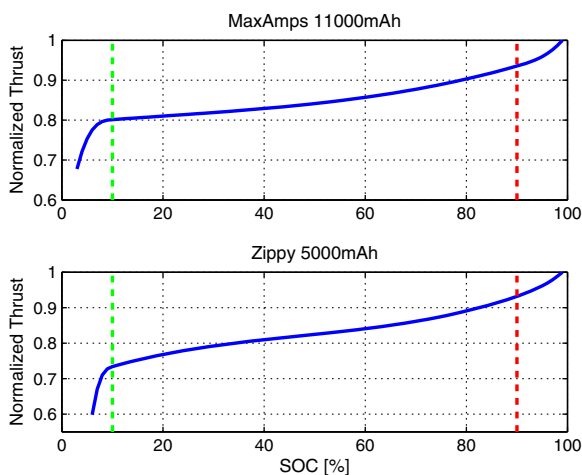


Fig. 18 Normalized spline approximation of the dependency of thrust on PWM command (green: 10 % mark, red: 90 % mark)

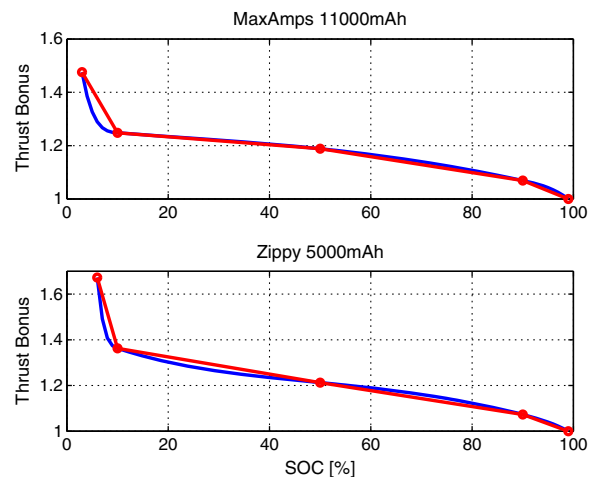


Fig. 19 Inverted nominal thrust (blue) and its piecewise-linear approximation (red)

Table 2 Piecewise linear approximation of the thrust bonus

Segment	Min.SOC	Max.SOC	α	K
MaxAmps 11000 mAh				
1	0	10	−0.0326	1.5737
2	11	50	−0.0015	1.2630
3	51	90	−0.0030	1.3367
4	91	100	−0.0069	1.6900
Zippy 5000 mAh				
1	0	10	−0.0775	2.1380
2	11	50	−0.0037	1.4005
3	51	90	−0.0035	1.3880
4	91	100	−0.0073	1.7300
Battery				
RMS error				
MaxAmps 11000 mAh	33.249			
Zippy 5000 mAh	32.222			

Having these models it is possible to implement the proposed controller.

5.4 Experimental Verification

The proposed controller was implemented on Arduino board in order to verify the controller performance on the test bench. The controller performance was measured from full battery to 10 % SOC to avoid the voltage drop. Both produced thrust and battery output power were measured. The reference thrust was arbitrarily set at the beginning of the experiment ($T_{\text{ref}} = 900[\text{g}]$ for Zippy battery and $T_{\text{ref}} = 1100[\text{g}]$ for

the MaxAmps battery). The measured thrust was smoothed with EMA filter ($\alpha = 0.001$).

The results for Zippy 5 Ah battery is shown in Fig. 20, for MaxAmps 11 Ah battery in Fig. 21. The average error of the controller is shown in Table 3. The error is calculated as the RMS error of EMA smoothed thrust from the reference thrust divided by the reference thrust ($e = \text{RMS}_T^{\text{EMA}} / T_{\text{ref}} * 100[\%]$).

The controller performed well in both cases (error under 6 %), however MaxAmps battery controller provided dramatically better results (error under 3 %). This is because the thrust gain curve for this battery (see that Fig. 19 is more linear and follows more closely the piecewise linear

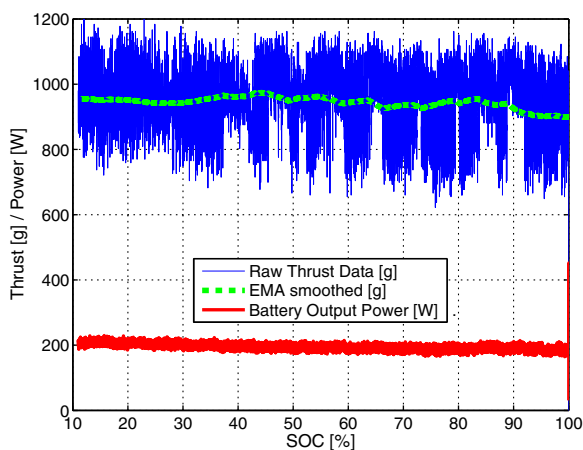
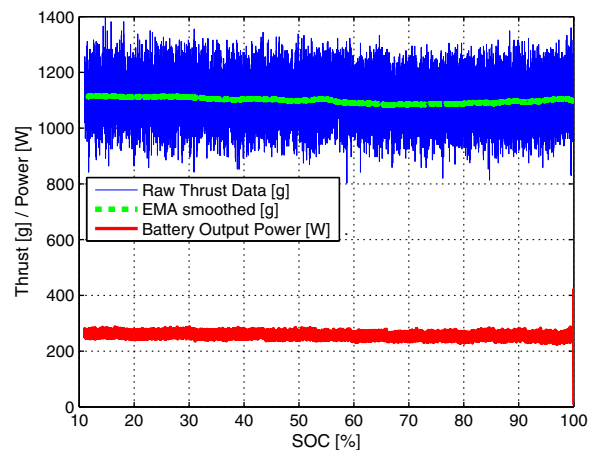
**Fig. 20** Laboratory test of the controller with Zippy 5 Ah battery**Fig. 21** Laboratory test of the controller with MaxAmps 11 Ah battery

Table 3 Laboratory experiment error

Battery	Nominal thrust (g)	Error (%)
Zippy	900	10
MaxAmps	1,100	3

approximation i.e. the battery dynamics are more linear between 100 % and 10 % SOC).

6 Flight Verification

The controller was tested on an AggieAir hexarotor platform (shown in Fig. 1). The platform uses same actuators as were used in the laboratory experiment in Section 5, and flies with a pair of MaxAmps 11Ah batteries.

To implement and verify the controller in flight, two obstacles have to be overcome: (1) initial SOC of the battery has to be known (it cannot be assumed that a fully charged battery pack is used); (2) current has to be measured during flight. To solve both issues, a battery monitoring system was designed.

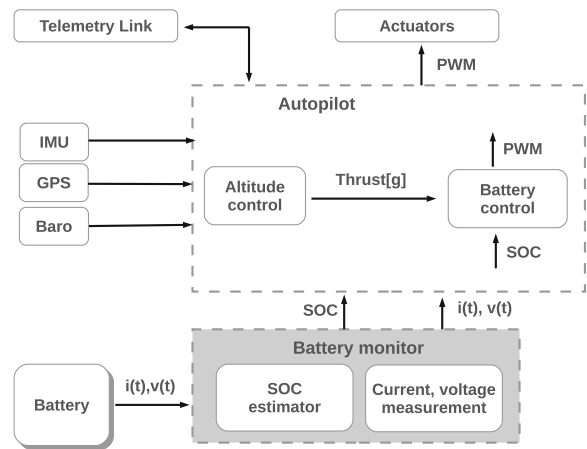
6.1 Battery Monitoring Subsystem

The battery monitoring system serves as an interface between the autopilot and the battery. It measures battery bus current, voltage and calculates SOC, which is required for proper gain scheduling of the altitude controller. A diagram of such subsystem is shown in Fig. 22.

For the actual hardware realization a similar current sensor as for experimental verification was used (Allegro Microsystems ACS756SCA-150B), just with higher rating (max 150 A) as the current consumption was expected to be up to 130 A in peak. Both voltage and output of the current sensor were measured using a 10-bit ADC converted with I2C bus (Analog Devices AD7997). The software part of the subsystem (SOC calculation) was implemented on the current autopilot board. An assembled board is shown in Fig. 23.

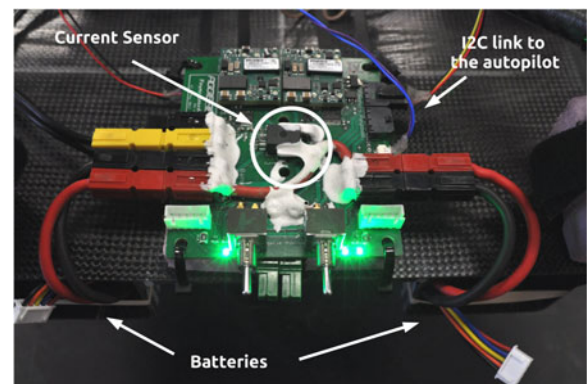
6.2 SOC Estimation

It is possible to infer the initial SOC of a battery from its OCV. If there is no load placed on the

**Fig. 22** Diagram of the developed battery monitoring subsystem

battery, the battery CLV eventually converges to OCV (details can be found in Section 2.1). A square wave discharge experiment was conducted. MaxAmps 11 Ah battery was placed on the test bench, the motor was run at 50 % throttle command for a minute, then it was switched off for one minute to allow the voltage to recover. This cycle continued until the closed loop voltage dropped below a safety threshold of 13 V. The current was integrated and the SOC was mapped to the OCV, as shown in Fig. 24. Note that for two batteries in parallel, the capacity doubles but the mapping is still valid.

To avoid computing a non-linear function representing OCV on SOC dependency (Eq. 7) in

**Fig. 23** An assembled battery monitoring system, mounted on the hexarotor and ready to fly

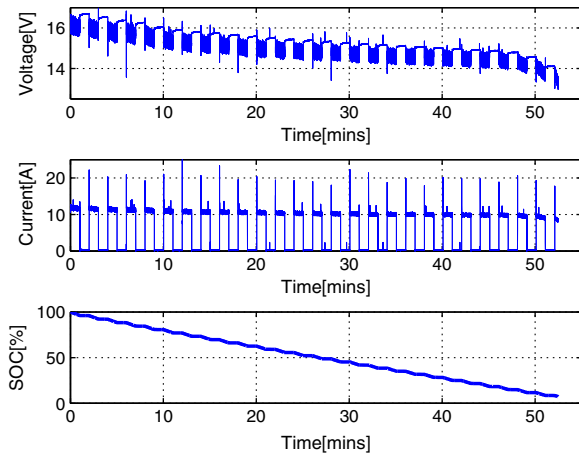


Fig. 24 Square wave discharge experiment, MaxAmps 11 Ah battery. *Top*: measured closed loop voltage, *Middle*: measured current, *Bottom*: calculated SOC

flight, a piecewise linear approximation was used instead. This approximation is shown in Fig. 25. A lookup table was calculated from the obtained values and stored in the autopilot memory.

Note that if older batteries are used, their capacity is lower and the discharge test would have to be repeated to find the actual capacity. The measured OCV on SOC dependency would however stay the same. For battery pairs, assume using a pair of equally old/new batteries, because using old and new batteries together is not recommended by the battery manufacturers.

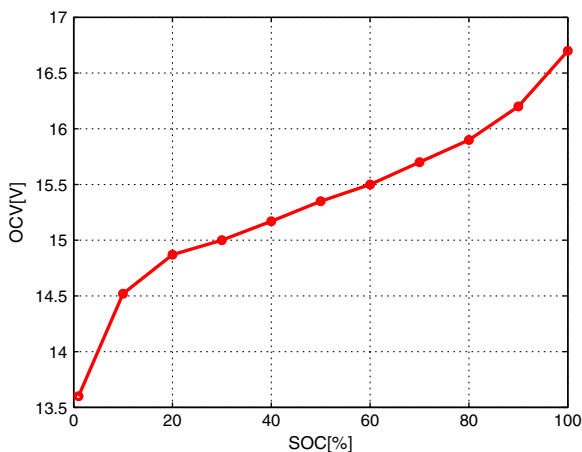


Fig. 25 Piecewise linear approximation of OCV to SOC dependency for MaxAmps 11 Ah battery

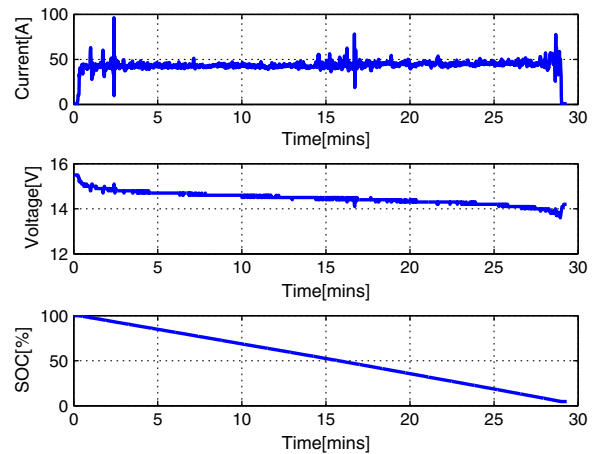


Fig. 26 Flight test with the AggieAir hexarotor. *Top*: total current consumption, *Middle*: battery voltage, *Bottom*: Estimated SOC

6.3 Flight Data

After implementing the State-Of-Charge controller and the battery monitoring system into the autopilot, and properly calibrating the SOC estimation, an actual outdoor flight test was conducted.

The flight consisted of a manual take-off and consequential switch to altitude hold mode. The altitude was held using PID control with constant feedforward, and the SOC-based controller. The

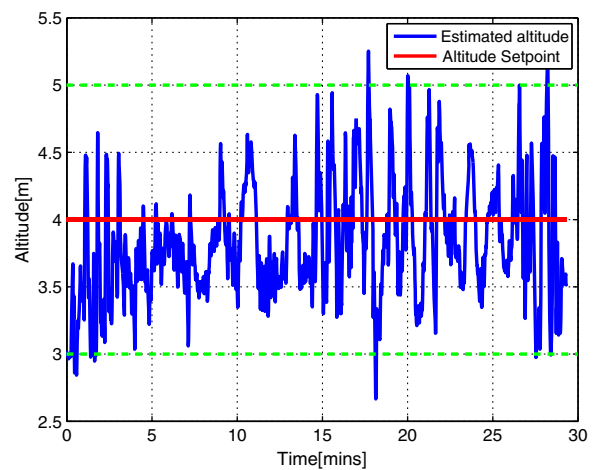


Fig. 27 Altitude hold with the AggieAir hexarotor. Vast majority of time the precision is within $\pm 1\text{m}$. *Blue*: Estimated altitude, *Red*: Altitude setpoint, *Green*: $\pm 1\text{m}$ threshold

flight time was 30 min and results are shown in Figs. 26 and 27. Overall performance is very good, and the controller is able to keep the altitude within ± 1 m from the reference.

7 Conclusion

In this paper a battery State-Of-Charge based altitude controller was developed and experimentally verified. The dynamics and model of both LiPo batteries and actuators consisting of brush less DC motors and electronic speed controller were described. The proposed controller adds a SOC-dependent gain, and provides satisfactory control in situations when feedback about actuator thrust or RPM is not available.

After characterizing battery dynamics (dependency of nominal thrust on the SOC of the battery, relation between open circuit voltage and SOC of the battery), a complete description and implementation of the controller was shown. The controller was first tested in laboratory experiment, where is showed control error of 3 % and 10 % for MaxAmps 11 Ah and Zippy 5 Ah batteries. The linearity of the battery dynamics affects the control performance, showing the higher quality batteries (MaxAmps) are more linear.

Consequently, the controller was implemented to the autopilot of AggieAir hexarotor and tested in real flight. To do this, an additional battery monitoring subsystem was developed, to correctly estimated the SOC of the batteries. The flight data prove that the controller is able to keep the airframe within 1 m of the desired set point.

Acknowledgements The authors would like to acknowledge Dr. YangQuan Chen of the University of California, Merced; Dr. Rees Fullmer of Utah State University; and Dr. Mac McKee of the Utah Water Research Laboratory.

This work is supported by Utah Water Research Laboratory MLF 2006-2013.

References

1. Allegro A4935 Datasheet: Automotive 3-phase mosfet driver. <http://www.allegromicro.com>. Accessed 28 July 2013
2. Paparazzi: The free autopilot (an opensource project). <http://paparazzi.enac.fr>. Accessed 14 July 2013
3. Utah Water Research Laboratory (web pages). <http://uwrl.usu.edu>. Accessed 20 July 2013
4. Valer, P., et al.: Battery Management Systems: Universal State-of-Charge Indication for Battery-Powered Applications. Philips Research Book. Springer Netherlands (2008)
5. Alexis, K., Nikolakopoulos, G., Tzes, A.: Model predictive quadrotor control: attitude, altitude and position experimental studies. *IET Control Theory Appl.* **6**(12), 1812–1827 (2012)
6. Barton, J.: Fundamentals of small unmanned aircraft flight. *Johns Hopkins APL Tech. Dig.* **31**(2), 142 (2012)
7. Bouabdallah, S., Siegwart, R.: Full control of a quadrotor. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007. IROS 2007*, pp. 153–158. IEEE (2007)
8. Bouadi, H., Simoes Cunha, S., Drouin, A., Mora-Camino, F.: Adaptive sliding mode control for quadrotor attitude stabilization and altitude tracking. In: *12th International Symposium on Computational Intelligence and Informatics (CINTI)*, pp. 449–455 (2011)
9. Chen, M., Rincon-Mora, G.: Accurate electrical battery model capable of predicting runtime and i-v performance. *IEEE Trans. Energy Convers.* **21**(2), 504–511 (2006)
10. Cherian, A., Andersh, J., Morellas, V., Papanikolopoulos, N., Mettler, B.: Autonomous altitude estimation of a UAV using a single onboard camera. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009. IROS 2009*, pp. 3900–3905 (2009)
11. Cheron, C., Dennis, A., Semerjyan, V., Chen, Y.: A multifunctional HIL testbed for multirotor VTOL UAV actuator. In: *Proc. of the IEEE/ASME International Conference on Mechatronics and Embedded Systems and Applications (MESA)*, 2012, pp. 44–48 (2010)
12. Coopmans, C., Di, L., Jensen, A., Dennis, A.A., Chen, Y.: Improved architecture designs for a low cost personal remote sensing platform: flight control and safety. In: *Proc. of the ASME Conference*, 2011, pp. 937–943 (2011)
13. Coopmans, C., Han, Y.: Aggieair: an integrated and effective small multi-uav command, control and data collection architecture. In: *Proc. of the 5th ASME/IEEE International Conference on Mechatronic and Embedded Systems and Applications (MESA09)*, 2009, pp. 1–7 (2009)
14. Coopmans, C., Stark, B., Coffin, C.: A payload verification and management framework for small UAV-based personal remote sensing systems. In: *Proc. of the 5th International Symposium on Resilient Control Systems (ISRCs)*, 2012, pp. 184–189 (2012)
15. Eynard, D., Vasseur, P., Demonceaux, C., Fremont, V.: UAV altitude estimation by mixed stereoscopic vision. In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 646–651 (2010)
16. Hu, Y., Yurkovich, S.: Battery state of charge estimation in automotive applications using LPV techniques. In: *Proc. of the American Control Conference (ACC)*, 2010 (2010)

17. Jensen, A., Chen, Y., McKee, M., Hardy, T., Barfuss, S.: AggieAir—a low-cost autonomous multispectral remote sensing platform: new developments and applications. In: Proc. of the Geoscience and Remote Sensing Symposium (IGARSS), 2009, vol. 4, pp. 995–998 (2009)
18. Kim, I.S.: The novel state of charge estimation method for lithium battery using sliding mode observer. *J. Power Sources* **163**(1), 584–590 (2006)
19. Knauff, M., Dafis, C., Niebur, D., Kwatny, H., Nwankpa, C.: Simulink model for hybrid power system test-bed. In: IEEE Electric Ship Technologies Symposium, 2007. ESTS '07, pp. 421–427 (2007)
20. Min, B.C., Hong, J.H., Matson, E.: Adaptive robust control (ARC) for an altitude control of a quadrotor type UAV carrying an unknown payloads. In: 2011 11th International Conference on Control, Automation and Systems (ICCAS), pp. 1147–1151 (2011)
21. Mukhopadhyay, S., Zhang, F.: Adaptive detection of terminal voltage collapses for li-ion batteries. In: Proc. of the 51st IEEE Conference on Decision and Control (CDC), 2012, pp. 4799–4804. Maui, Hawai, USA (2012)
22. Podhradsky, M., Bone, J., Jensen, A., Coopmans, C.: Small low-cost unmanned aerial vehicle lithium-polymer battery monitoring system. In: Proc. of the ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference (2013)
23. Pounds, P., Mahony, R.: Design principles of large quadrotors for practical applications. In: IEEE International Conference on Robotics and Automation, 2009. ICRA '09, pp. 3265–3270 (2009)
24. Pounds, P., Mahony, R., Corke, P.: System identification and control of an aerobot drive system. In: Information, Decision and Control, 2007. IDC '07, pp. 154–159 (2007)
25. Prasad, G., Sree Ramya, N., Prasad, P.V.N., Tulasi Ram Das, G.: Modelling and simulation analysis of the brushless DC motor by using MATLAB. *Int. J. Innov. Tech. Exp. Eng.* **1**(5), 27–31 (2012)
26. Stowers, J., Hayes, M., Bainbridge-Smith, A.: Altitude control of a quadrotor helicopter using depth map from microsoft kinect sensor. In: International Conference on Mechatronics (ICM), pp. 358–362 (2011)
27. Weiss, S., Scaramuzza, D., Siegwart, R.: Monocular slam based navigation for autonomous micro helicopters in gps denied environments. *J. Field Robot.* **28**(6), 854–874 (2011)