Minimum Time Trajectory for Helicopter UAVs: Computation and Flight Test[1,2]

**Michael Zhu[*], Shupeng Lai[†], Randy Boucher[‡],
Ben Chen[†], Xiang Cheng[†], and Wei Kang[‡]**

[*]ONR SEAP Program, Naval Postgraduate School
Monterey, CA, USA and
Los Altos High School, Los Altos, CA, USA

[†]Department of Electrical Engineering
National University of Singapore, Singapore

[‡]Department of Applied Mathematics
Naval Postgraduate School, Monterey, CA, USA

Abstract

This paper serves as an integral part of a project in which the main objective is to develop the theory and algorithms of computational methods for optimal UAV trajectory planning in obstacle-rich environment. In this paper, we apply a Galerkin method of optimal control to the model of HeLion, a helicopter UAV developed by the UAV Team from the National University of Singapore (NUS). The goal of the project is to compute and test minimum-time trajectories for the unmanned system. We use nonlinear optimal control to formulate the problem, which is subject to the dynamical system of differential equations and state-control bounds of HeLion. The dynamical system is defined by a set of fifteen dimensional nonlinear differential equations. Different from previous papers on this model, more challenging constraints including higher order derivatives of the states are included

---

in the formulation. The problem does not have an analytic solution. We numerically solve the problem using a Galerkin method. Then the computed trajectory is verified in flight tests using HeLion.

Keywords: Minimum-time control, Helicopter UAV, Galerkin computational optimal control, flight test

1 Introduction

This paper is an integral part of a project in which the main objective is to develop the theory and algorithms of computational methods for optimal UAV trajectory planning in an obstacle-rich environment. Due to the complexity of the system and its constraints, it is a significant challenge for autonomous UAVs to fly in an optimal manner with respect to use of fuel, expenditure of time, or distance traveled. In [7, 5, 10], a pseudospectral method of dynamic optimization was applied to the problem of minimum time trajectory planning for helicopter UAVs. Due to the requirements of some onboard equipment and safety concerns, we found that for some missions the trajectories in [7, 5] are too aggressive. Their acceleration and jerk are larger than the preferred upper bounds. It raises a technical issue of how to compute optimal trajectories that satisfy not only the dynamic model of differential equations but also additional constraints involving second (acceleration) and 3rd (jerk) order derivatives of the system state variables.

In this paper, we integrate a set of additional constraints on the acceleration and jerk of UAVs into the optimal control problem. In addition, the rate of change for control inputs are also bounded. Different from [7, 5] in which a pseudospectral method was used, we decided to use Galerkin type of discretization for computational optimal control, a method that reduces the estimation error when taking high order derivatives of the state variables. This work is a continuation of the research effort presented in [7, 5, 10] where a library of useful trajectories are being developed for HeLion, a helicopter UAV built in National University of Singapore, for various scenarios including optimal trajectories with and without obstacles, sharp-turn and pointing, and optimal trajectories with jerk control.

Rather than path planning, in this paper we address the problem of trajectory planning. Therefore, we compute the time functions of the commands to the system actuators as well as the time functions of all state variables. In addition, we require that the trajectory minimizes a cost functional, in this case the minimum flying time. Existing approaches of path planning emphasizes stability and robustness. Due to the complexity of helicopter aerodynamics and terrain constraints, optimal trajectory planning cannot be solved analytically. Therefore, finding numerical solutions is the way to go.

Variable	Physical meaning (unit)
p_x, p_y, p_z	Position vector in NED-frame (m)
u, v, w	Velocity vector in body-frame (m/s)
ϕ, θ, ψ	Roll, pitch, and yaw angles (rad)
p, q, r	Roll, pitch, and yaw angular rate in body-frame (rad/s)
a_s, b_s	Longitudinal and lateral tip-path-plane (TPP) flapping angles
$\delta_{ped,int}$	Intermediate state in yaw rate gyro dynamics

Table 1: Physical meanings of the state variables

2 Problem Formulation

The minimum-time trajectory planning can be formulated as a problem of optimal control

$$\min J = \int_{t_0}^{t_f} 1 dt \tag{1}$$

subject to

$$\dot{x} = f(x, u)$$

$$x_{\min} \leq x \leq x_{\max}, \quad u_{\min} \leq u \leq u_{\max}$$

$$\dot{u}_{\min} \leq \dot{u} \leq \dot{u}_{\max}$$

$$a_{\min} \leq a \leq a_{\max}, \quad J_{\min} \leq J \leq J_{\max}$$

$$O(x) \geq 0 \quad \text{(obstacles)}$$

$$x(t_0) = x_0, x(t_f) = x_f, t_f \text{ is unspecified}$$

The state and control constraints defined by $x_{\max}$, $x_{\min}$ $u_{\max}$, and $u_{\min}$ represent the limitations of the variables. The differential equation of $f(x, u)$ represents the helicopter model, which is a system of fifteen nonlinear differential equations. Details about the helicopter model can be found in [3]. The model is based on two coordinate frames, i.e., the body frame and the north-east-down (NED) frame. The state consists of the following variables

$$x = \begin{bmatrix} p_x & p_y & p_z & u & v & w & \cdots \\ \phi & \theta & \psi & p & q & r & a_s & b_s & \delta_{ped,int} \end{bmatrix}^T$$

The control input consists of the following variables

$$u = \begin{bmatrix} \delta_{col} & \delta_{lat} & \delta_{lon} & \delta_{ped} \end{bmatrix}^T$$

These variables are explained in Table 1 - 2.

δ_{lat}	Normalized aileron servo input (-1, 1)
δ_{lon}	Normalized elevator servo input (-1,1)
δ_{col}	Normalized collective pitch servo input (-1, 1)
δ_{ped}	Normalized rudder servo input (-1, 1)

Table 2: Physical meanings of the control variables

$-\infty \leq p_x \leq \infty$	$-\infty \leq p_y \leq \infty$	$-\infty \leq p_z \leq 0$
$-4 \leq u \leq 4$	$-4 \leq v \leq 4$	$-4 \leq w \leq 4$
$-0.18 \leq \phi \leq 0.18$	$-0.18 \leq \theta \leq 0.18$	$-0.18 \leq \psi \leq 0.18$
$-0.4 \leq p \leq 0.4$	$-0.4 \leq q \leq 0.4$	$-0.4 \leq r \leq 0.4$
$-0.28167 \leq \delta_{col} \leq -0.086355$	$-0.1 \leq \delta_{lat} \leq 0.1$	$-0.1 \leq \delta_{lon} \leq 0.1$
$-0.1 \leq \delta_{ped} \leq 0.1$		

Table 3: The lower and upper bounds for state and control variables

In the system model, the state variables and control inputs are bounded by $x_{\max}$, $x_{\min}$ $u_{\max}$, and $u_{\min}$. The details of the bounds are listed in Table 3. These bounds are relatively conservative for safety reasons.

The obstacles are not included in the examples, which was partially addressed in [10]. Required by the onboard equipment and test flight missions, we have to make sure that the acceleration and jerk of the UAV are bounded. In addition, the rate of change for control variables are also bounded. Different from the model in [7, 5, 10], we include additional constraints for the variables a, J, and $\dot{u}$, where

$$a = \frac{d}{dt}\sqrt{u^2 + v^2 + w^2}, \quad J = \frac{da}{dt}$$

3   Galerkin Method for Optimal Control

In this paper we adopt a continuous Galerkin (CG) method of computational dynamic optimization. CG methods for optimal control have been recently developed and have shown much promise in solving a wide variety of optimal control problems [2]. In this project we use a CG method based on Legedre-Gauss-Lobatto (LGL) quadrature nodes. The CG method approximates the states and controls with globally interpolating N-th order Lagrange polynomials at the LGL nodes. The LGL nodes, $t_0 = -1 < t_1 < \cdots < t_N = 1$, are defined by

$$t_0 = -1, \, t_N = 1, \text{ and}$$

for $k = 1, 2, \ldots, N - 1$, t_k are the roots of $\dot{L}_N(t)$

where $\dot{L}_N(t)$ is the derivative of the N-th order Legendre polynomial $L_N(t)$. The discretization works in the interval of $[-1, 1]$. It is proved in approximation theory that the polynomial interpolation at the LGL nodes converges to the solution under L^2 norm at the rate of $1/N^m$, where m is the smoothness of the solution [4]. If the solution is C^∞, then the polynomial interpolation at the LGL nodes converges at a spectral rate, i.e. it is faster than any given polynomial rate. In a CG method, the state trajectory, $x(t)$, is approximated by the vector

$$\bar{x}^{Nk} \approx x(t_k) \in \mathbb{R}^n, \quad k = 1, 2, \ldots, N$$

Similarly, $\bar{u}^{Nk}$ is the approximation of $u(t_k)$. The CG method for solving optimal control problems is a good all-around method for the approximation of smooth functions, integrations, and differentiations, all critical to accurately solving problems of this form. Discretization of the problem's dynamics is an extremely important part of the method. A solution to the differential equation $\dot{x} = f(x, u)$ may be approximated by the CG method at the LGL nodes with the following formulation

$$\sum_{j=0}^{N} D_{ij} \bar{x}^{Nj} = c_i, \quad i = 0, \ldots, N \tag{2}$$

where the $(N + 1) \times (N + 1)$ differentiation matrix D is defined by

$$D_{ij} = \int_{-1}^{1} \phi_i \frac{d\phi_j}{dt} dt, \quad i, j = 0, \ldots, N$$

and the $(N + 1) \times 1$ right-hand-side (RHS) vector c is defined as

$$c_i = \int_{-1}^{1} \phi_i f(x(t), u(t)) dt, \quad i = 0, \ldots, N$$

The Lagrange polynomial ϕ is obtained from the general definition

$$\phi_i(t) = \prod_{\substack{j=0 \\ j \neq i}}^{N} \frac{(t - t_j)}{(t_i - t_j)}, \quad i = 0, \ldots, N$$

and differentiating the equation for the Lagrange polynomial yields

$$\frac{d\phi_i}{dx}(t) = \sum_{\substack{k=0 \\ k \neq i}}^{N} \left(\frac{1}{t_i - t_j} \right) \prod_{\substack{j=o \\ j \neq i \\ j \neq k}}^{N} \frac{(t - t_j)}{(t_i - t_j)}, \quad i = 0, \ldots, N$$

If LGL quadrature rule is used, the differentiation matrix and RHS vector can be calculated with the relationships

$$D_{ij} = \sum_{k=0}^{N} \phi_i(t_k) \frac{d\phi_j}{d\xi}(t_k) w_k, \quad i,j = 0,\ldots,N$$

and

$$c_i \approx \bar{c}_i = \sum_{k=0}^{N} \phi_i(t_k) f(\bar{x}^{Nk}, \bar{u}^{Nk}) w_k, \quad i = 0,\ldots,N$$

respectively, where the LGL weight w is defined by

$$w_k = \frac{2}{N(N+1)} \frac{1}{[L_N(t_k)]^2}, \quad k = 0,\ldots,N$$

Note that LGL quadrature rule is exact for polynomial integrands of degree less than or equal to $2N - 1$. The discretization of the state and control constraints is simple,

$$x_{\min} \leq \bar{x}^{Nk} \leq x_{\max}, \quad u_{\min} \leq \bar{u}^{Nk} \leq u_{\max} \tag{3}$$

for $0 \leq k \leq N$. The discretization of the acceleration and jerk requires discrete differentiation. The derivative at LGL nodes can be approximated by matrix multiplication,

$$\bar{a}^{Nk} = \sum_{j=0}^{N} d_{kj} \sqrt{\bar{u}^{Nj} + \bar{v}^{Nj} + \bar{w}^{Nj}} \tag{4}$$

$$\bar{J}^{Nk} = \sum_{j=0}^{N} d_{kj} \bar{a}^{Nj}$$

where d_{ij} is the differentiation matrix in pseudospectral method [4]. This approximation of derivatives has a high order convergence, which is important for the convergence of computational optimal control. The acceleration and jerk constraints are discretized as follows

$$a_{\min} \leq \bar{a}^{Nk} \leq a_{\max}, \quad J_{\min} \leq \bar{J}^{Nk} \leq J_{\max} \tag{5}$$

The bounds on the rate of change for control variables are treated in a similar way. Lastly, the cost functional $J[x(\cdot), u(\cdot)]$ is approximated by the LGL quadrature rule,

$$J[x(\cdot), u(\cdot)] \approx \sum_{k=0}^{N} g(\bar{x}^{Nk}, \bar{u}^{Nk}) w_k, \quad k = 0,\ldots,N \tag{6}$$

(u, v, w)	$(0, 0, 0)$
(ϕ, θ, ψ)	$(0.038944, 0.0008856, 0)$
(p, q, r)	$(0, 0, 0)$
$(a_s, b_s, \delta_{ped,int})$	$(0.0049121, -0.00087278, 0)$
$(\delta_{lat}, \delta_{lon}, \delta_{col}, \delta_{pet})$	$(-0.17461, 0.0061268, -0.0025549, 0)$

Table 4: Howering trim value for the states and controls

To summarize, a finite dimensional approximation of the state and control trajectories exist at the LGL nodes. The system of differential equations can be approximated using the discrete differentiation (2). The state and control constraints are approximated using (3). The acceleration and jerk constraints are approximated using (4)-(5). The cost function can be approximated by the LGL quadrature rule (6). Integrating these discretization elements together yields a finite dimensional nonlinear programming, which can be solved numerically. In our computations, sequential quadratic programming is used.

4   Optimal Trajectories

We use two examples to test the algorithm. As a starting point, the algorithm generates an time-optimal trajectory for a 70 meter flight with only the physical constraints of the helicopter model. At the end points A and B, the UAV hovers with a zero velocity and hovering trim values given in Table 4. The details of the numerical optimal trajectory for this first example are shown in Figures 1- 4. A polynomial order of 80 was used. In order to validate the trajectories of state and control in continuous-time, a Runge-Kutta solver is run, using the system model based on a spline interpolation of the optimal control at the nodes. The dots in the figures represent the results from the optimization program, while the lines represent the trajectory from Runge-Kutta method. For all fifteen states and four controls, there is almost no visible disparity between the Galerkin method results and those of the Runge-Kutta solver. In particular, the paths generated are within 1 centimeter of each other throughout the 70 meter flight. Although the angular velocities and flapping angles are highly nonlinear, the optimal trajectory is highly consistent with the Runge-Kutta method.

In the second example, additional constraints are added in order to ensure stability during flight-testing. Constraints are put on the derivatives of the control inputs; and the norms of the velocity, acceleration, and jerk.

$$-0.1 \leq \dot{\delta}_{col} \leq 0.1 \qquad 0.12 \leq \dot{\delta}_{lat} \leq 0.12$$
$$-0.12 \leq \dot{\delta}_{lon} \leq 0.12 \qquad -0.12 \leq \dot{\delta}_{ped} \leq 0.12$$
$$\sqrt{u^2 + v^2 + w^2} \leq 3.8 \quad |a| < 0.8, \; |J| < 0.85$$
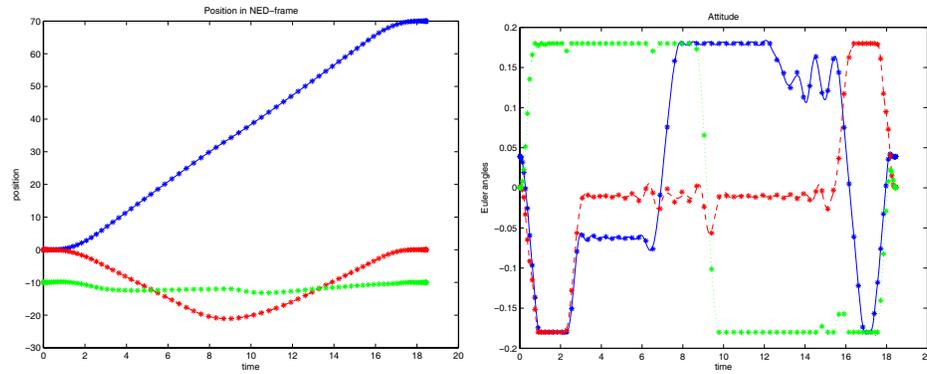
Figure 1: x, ϕ -solid; y, θ - dash; z, ψ - dot.
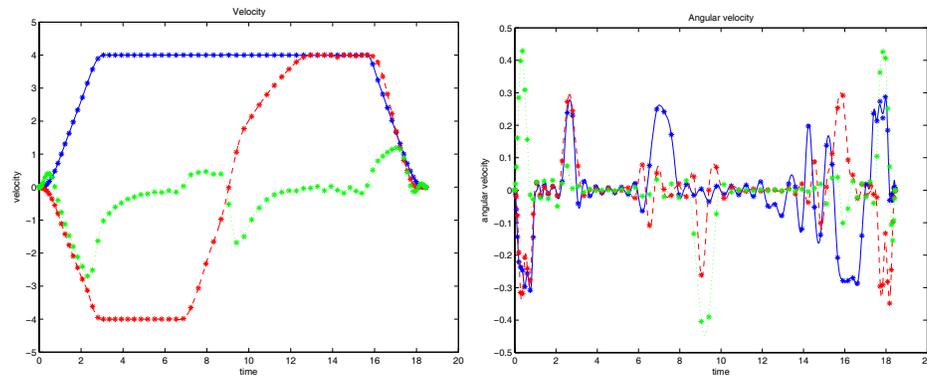
Figure 2: u, p -solid; v, q - dash; w, r - dot.
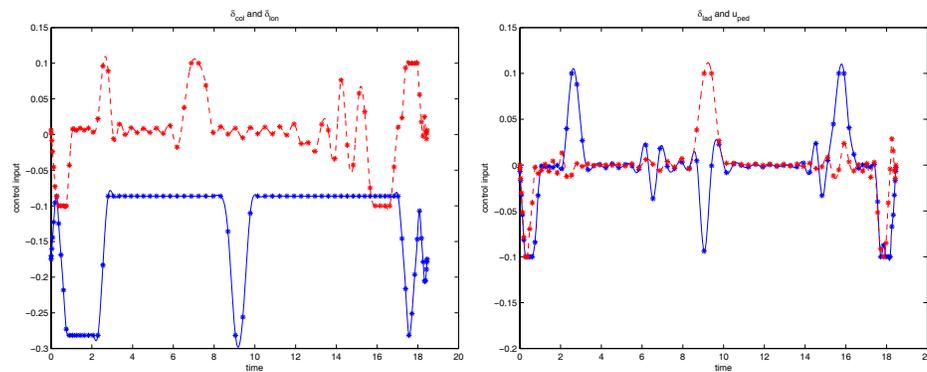
Figure 3: δ_{col}, δ_{lon} -solid; δ_{lad}, u_{ped} - dash.

In this example, a shorter path of 50 meters is used in order to reduce er-
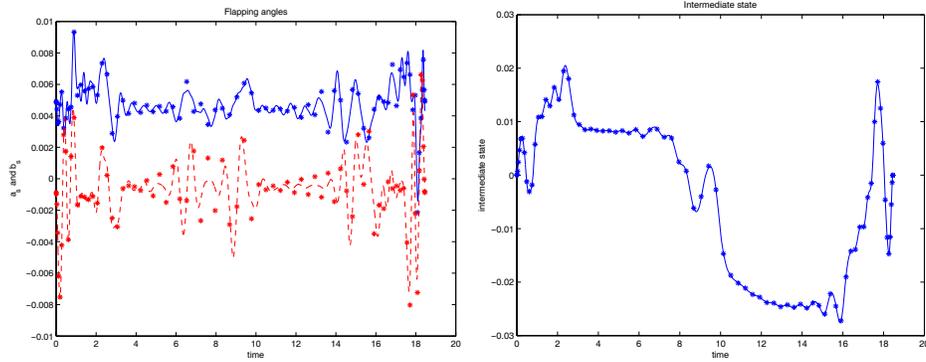
Figure 4: a_s -solid; b_s - dash; $\delta_{ped,int}$ - solid.

ror in the Runge-Kutta trajectory. The minimum-time for this trajectory is $t = 18.8269$s. With 60 nodes, the algorithm yields a path that is within 20 centimeters of the Runge-Kutta result. As shown in Figures 5 to 8, all state variables agree with their validating curves, with some error in the x-position.
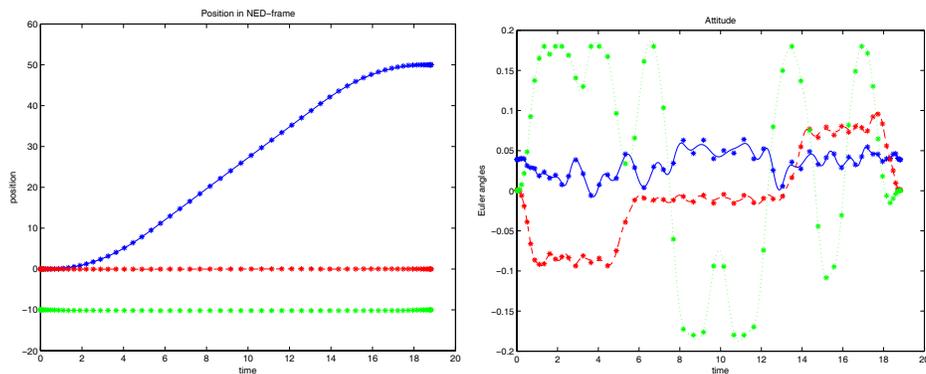
Figure 5: x, ϕ -solid; y, θ - dash; z, ψ - dot.

5 Flight Testing

The platform utilized for the experiment is a fully customized quadrotor developed by NUS UAV Team. The platform is composed of carbon fiber plates and rods with a durable Acrlonitrile Butadiene Styrene. The overall dimensions are 35cm in height and 86 cm from tip-to-tip. The motors used for the platform are 740kV T-Motors with Turnigy Plush - 25A Bulletproof speed controller electronic speed controllers (ESCs). The propellers used are APC
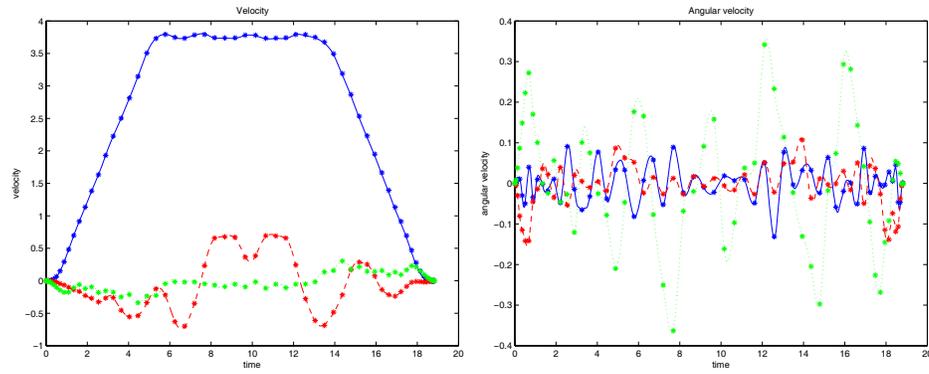
Figure 6: u, p -solid; v, q - dash; w, r - dot.
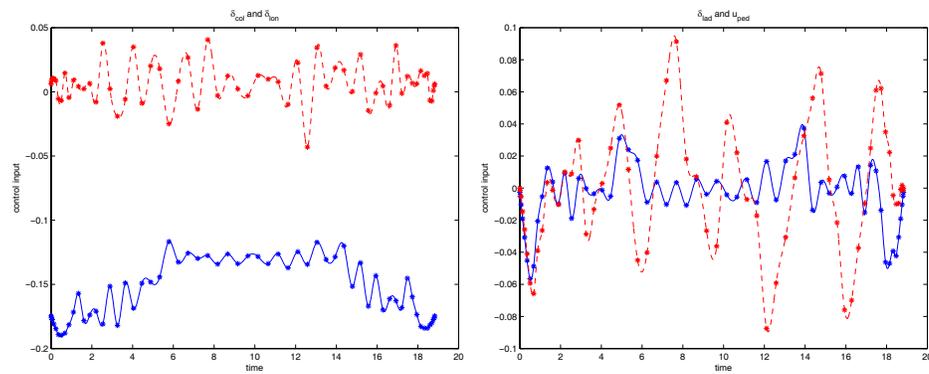
Figure 7: δ_{col}, δ_{lon} -solid; δ_{lad}, u_{ped} - dash.
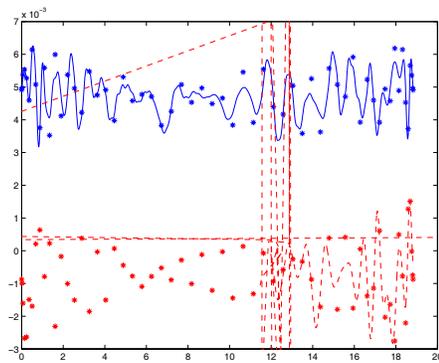
propeller setup could generate 15 kN static thrust. The final bare platform's main body weighs 1 kg. Its maximum total take-off weight reaches 3.3 kg with a 4 cell 4300mAh lithium polymer battery. Our current platform including the necessary sensor suite weighs 2.7 kg. We have tested that the platform was able to fly at 8m/s for a period of 10 to 15 minutes depending on the environmental factors.

Figure 9: The quadrotor utilized in the experiment - a platform developed by NUS UAV Team

The inner-outer loop control capability is build for the system to track a given feasible trajectory, such as the minimum-time trajectories found in the previous section. The inner attitude loop is controlled by a DJI Naza multi-rotor computer and the outer loop is controlled by a robust perfect tracking (RPT) based controller. The quadrotor is capable of performing auto GPS way point tracking mission. The flight test is performed at blackmall, Singapore, 28 Aug 2013. During the flight test, the wind gust is up to 2m/s and the GPS condition is quite good, receiving up to 11 satellites signals. The trajectories of position and velocities are shown in Figure 10. The autonomous trajectory starts at t=207.2s and ends at t=226s.The quadrotor tracks the position and velocities very smoothly. The performance of acceleration and deceleration is like human control without causing extra shaking and jerking and still results relatively high velocity which is hard to achieve by normal point to point heuristic trajectory planning.

6   Conclusions

Galerkin method of optimal control is a new computational method developed in [2]. The results in this paper show that the computational algorithm is
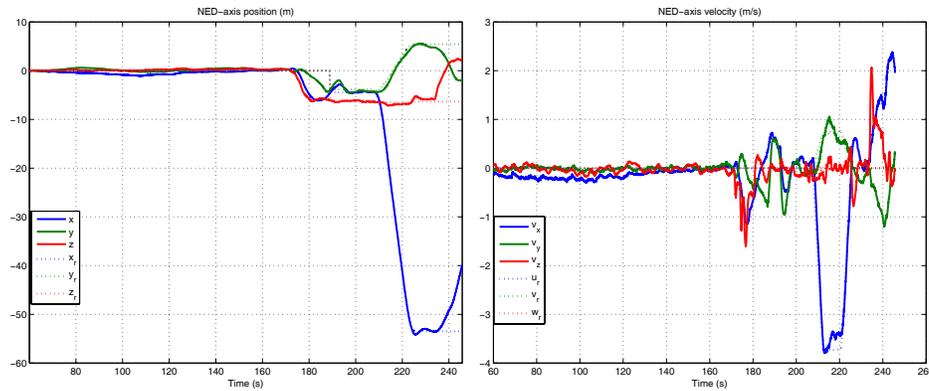
Figure 10: Tractories from the flight test

applicable to the minimum-time trajectory planning for helicopter UAVs. It is tested using two sets of different constraints based on the model of HeLion. In both cases, the program converges to feasible trajectories. A such trajectory is validated in laboratory experiments using a quadrotor unmanned helicopter developed by NUS UAV Team. This work is a part of the effort of developing a reliable computational method for nonlinear optimal control under nonlinear constraints. For future work, more aggressive trajectories is planned to be computed and tested using weaker constraints than those in this paper. In addition, theoretical foundation are being developed for the feasibility, consistency, and convergence of Galerkin optimal control method.

References

[1] N. Bedrossian, S. Bhatt, W. Kang, I. M. Ross, Zero-propellant Maneuver Guidance: Rotating the International Space Station with Computational Dynamic Optimization, Control Systems Magazine, Vol. 29 (2009), No. 5, 53-73.

[2] Randy Boucher, Galerkin Optimal Control, Ph.D. Dissertation, Naval Postgraduate School, Monterey, CA, preprint, 2013.

[3] G. Cai, B. M. Chen and T. H. Lee, Unmanned Rotorcraft Systems, Springer, New York, 2011.

[4] C. Canuto, M. Y. Hussaini, A. Quarteroni and T. A. Zang, Spectral Method in Fluid Dynamics, New York: Springer-Verlag, 1988.

[5] K. Tang, B. Wang, W. Kang, and B. Chen, Minimum Time Control of Helicopter UAVs Using Computational Dynamic Optimization, IEEE Conference on Control and Automation, Santiago, Chile, December, 2011.

[6] W. Kang, and N. Bedrossian, Pseudospectral Optimal Control Theory Makes Debut Flight - Saves NASA $1M in under 3 hrs, SIAM News, September, 2007.

[7] B. T. Gatzke, Trajectory Optimization for Helicopter Unmanned Aerial Vehicles (UAVs), NPS Thesis, 2010.

[8] Q. Gong, M. Ross, W. Kang, F. Fahroo, Connections Between the Covector Mapping Theorem and Convergence of Pseudospectral Methods for Optimal Control, J. Computational Optimization and Applications, online publication, October, 2007; Vol. 41(2008), No. 3, pp. 307-335.

[9] W. Kang, Q. Gong, and I. M. Ross. On the Convergence of Nonlinear Optimal Control Using Pseudospectral Methods for Feedback Linearizable Systems, International Journal of Robust and Nonlinear Control, 17(2007), 1251 – 1277.

[10] 71. N. Xu, G. Cai, W. Kang, and B.M. Chen, Minimum-time trajectory planning for helicopter UAVs using dynamic optimization, IEEE International Conference on Systems, Man, and Cybernetics, Seoul, South Korea, October, 2012.