

# Comparison of Gradient Methods for Gain Tuning of a PD Controller Applied on a Quadrotor System

Jinho Kim<sup>a</sup>, Stephen A. Wilkerson<sup>b</sup> and S. Andrew Gadsden<sup>a</sup>

<sup>a</sup>University of Maryland at Baltimore County, 1000 Hilltop Ct., Baltimore, MD, 21250, USA;

<sup>b</sup>U.S. Army Research Laboratory at Aberdeen, MD, 21001, USA

## ABSTRACT

Many mechanical and electrical systems have utilized the proportional-integral-derivative (PID) control strategy. The concept of PID control is a classical approach but it is easy to implement and yields a very good tracking performance. Unmanned aerial vehicles (UAVs) are currently experiencing a significant growth in popularity. Due to the advantages of PID controllers, UAVs are implementing PID controllers for improved stability and performance. An important consideration for the system is the selection of PID gain values in order to achieve a safe flight and successful mission. There are a number of different algorithms that can be used for real-time tuning of gains. This paper presents two algorithms for gain tuning, and are based on the method of steepest descent and Newton's minimization of an objective function. This paper compares the results of applying these two gain tuning algorithms in conjunction with a PD controller on a quadrotor system.

**Keywords:** Optimization, Gain Tuning, Gradient Method, PI Control, Quadrotor

## 1. INTRODUCTION

Over the past decade, many researchers and institutions have focused on unmanned aerial vehicles (UAVs) research. One of the most popular types of UAV systems is referred to as a quadrotor or quad-copter (Fig. 1). This system is a four rotor vertical take-off and landing (VTOL) UAV. The quadrotor has advantages over both helicopters and fixed-wing aircraft because, not only can the quadrotor lift heavier payloads, but it can also take-off and land vertically. Moreover, a small-sized quadrotor is agile, highly maneuverable, and has a safer flight due to the four rotor design. Due to these advantages, quadrotors have been used in a wide-variety of missions, including: reconnaissance, search and rescue, exploration of disaster areas, and so on [1-4].

As shown in Fig. 1, a quadrotor consists of four rotors fixed to a rigid cross frame. This system has a smaller number of control inputs than number of degrees of freedom. In this case, in order to control the under-actuated system, four inputs related to each rotor's angular velocity may be implemented. Many researcher studies on quadrotors have been conducted, and there is an increasing body of control knowledge in this area. Various quadrotor modeling approaches have been presented in [5-7] and many nonlinear control techniques were proposed in [8-10].

One of the most popular control methods applied to UAVs is the PID controller. This is mainly due to the fact that PID controllers are relatively easy to implement and program, fine-tune, and are conceptually simple. Generally speaking, the proportional gain is used to improve the system rise time and response, the integral gain is used to modify steady-state errors, and the derivative gain is used to modify any system overshoot. To start the tuning process, one typically sets the proportional gain first. Secondly, the derivative term can be changed to help reduce the percent overshoot in the system. Finally, if there is significant steady-state error (between the desired state values and the actual state values), then the integral term is modified. The PID tuning is often iterative in nature. In [11] and [12], a quadrotor with a PD and a PID controller was introduced for stabilizing the orientation angles and altitude. A vision-based control of the quadrotor on top of a PD controller was presented in [13]. The authors designed a PD controller and then demonstrated the application of a visual servo control of a quadrotor in [13].

---

Jinho Kim: E-mail: [umbcjhkim@umbc.edu](mailto:umbcjhkim@umbc.edu), Telephone: 1 410 455 3397

Stephen A. Wilkerson: E-mail: [stephen.a.wilkerson.civ@mail.mil](mailto:stephen.a.wilkerson.civ@mail.mil), Telephone: 1 410 278 3966

S. A. Gadsden: E-mail: [gadsden@umbc.edu](mailto:gadsden@umbc.edu), Telephone: 1 410 455 3307



Figure 1. A quadrotor made by DJI (F450) with propeller guards.

The PID method has the following design procedure which involves gain scheduling. A number of papers found in the literature deal have considered this topic [14] and [15]. In [14], PID gain scheduling is developed using fuzzy logic. This fuzzy gain scheduling scheme is demonstrated with the physical model for control of temperature process. In [15], PID controller and gain scheduling control strategy are used for fault tolerant control. This strategy is generally applied in the aerospace and industrial field.

In this paper, two methods of gain tuning are presented and compared. The gain tuning is used by a PD controller for improved quadrotor performance. The algorithms generate optimal gain values to minimize the objective function by using two types of gradient methods: the method of steepest descent, and Newton's method. The first-order gradient method is applied first in an effort to obtain optimal gain values. The first-order gradient method uses only first derivatives in determining a proper gradient direction, and is therefore also referred to as the steepest descent method. Secondly, the second-order gradient strategy will also be implemented. This strategy, referred to as Newton's method, uses higher derivatives to select a suitable search direction [16]. The purpose of this paper is to identify which gain tuning technique is ideal for the PD controller applied on the quadrotor system.

This paper is organized as follows. Section 2 introduces the quadrotor model and the PD controller which is used in this paper. Section 3 describes the gain tuning algorithms; the steepest descent method, and Newton's method for optimization of the objective function. In Section 4, the application of these algorithms and the simulation results are presented and compared. Finally, Section 5 presents concluding remarks and future work.

## 2. QUADROTOR MODEL AND PD CONTROLLER

In this paper, the quadrotor consists of four fixed rotors, has fixed-pitch-angle blades, and a rigid cross frame. The quadrotor is controlled by varying thrust forces generated by each rotor. The model of the quadrotor is shown in Fig. 2.

This system is considered an under-actuated system, has four inputs  $\mathbf{u} = [u_1, u_2, u_3, u_4]$ , and six state variables  $\mathbf{x} = [x, y, z, \phi, \theta, \psi]$ . Two state variables cannot be controlled directly.

This topic is covered in more detail in Section 2. B. Along the  $x$ -axis, the quadrotor has to create a pitch angle,  $\theta$ , in order to increase the speed of rotors 1 and 3 while attempting to maintain the speeds of rotor 2 and 4. This is due to the fact that the  $x$ -translational motion is related to the tilt of the body frame with respect to the  $y$ -axis. Similarly, the quadrotor is able to fly along the  $y$ -axis due to the rotation of the frame  $x$ -axis, which generates a roll angle,  $\phi$ . In order to generate a yaw angle (in a counter clock-wise direction),  $\psi$ , the speeds of rotors 1 and 3 have to be increased while the speeds of rotor 2 and 4 are proportionally decreased.

It is assumed that the quadrotor is symmetric with respect to the  $x$  and  $y$  axes. Thereby, the center of gravity aligns with the geometric center of the quadrotor. The length between each rotor and the geometric center of the quadrotor is defined by  $l$ . The thrust forces, perpendicular to the  $x$ - $y$  plane, are generated by the four rotors and are defined by  $T_i$  ( $i = 1, 2, 3, 4$ ).

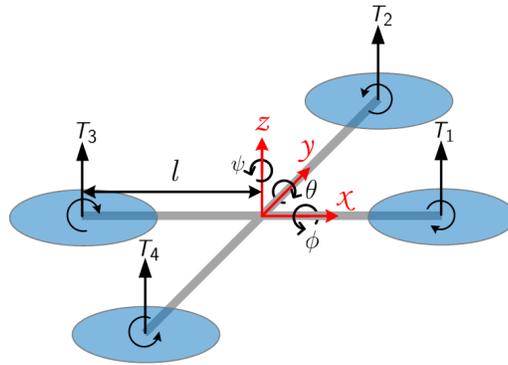


Figure 2. A configuration of a quadrotor model with Euler angles.

## 2.1 Quadrotor Model

The dynamic equations of the quadrotor model may be derived from a Lagrange approach, and is simplified as follows:

$$\ddot{x} = u_1 (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi), \quad (1)$$

$$\ddot{y} = u_1 (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi), \quad (2)$$

$$\ddot{z} = u_1 (\cos \phi \cos \theta) - g, \quad (3)$$

$$\ddot{\phi} = u_2 l, \quad (4)$$

$$\ddot{\theta} = u_3 l, \quad (5)$$

$$\ddot{\psi} = u_4, \quad (6)$$

where  $[x, y, z]$  are positions of the quadrotor in the inertial frame;  $[\phi, \theta, \psi]$  Euler angles represent roll, pitch, and yaw angles, respectively; and  $g$  the acceleration of gravity.

The control inputs  $u_1, u_2, u_3, u_4$  are defined as follows:

$$u_1 = \frac{1}{m} (T_1 + T_2 + T_3 + T_4), \quad (7)$$

$$u_2 = \frac{1}{J_1} (T_2 - T_4), \quad (8)$$

$$u_3 = \frac{1}{J_2} (-T_1 + T_3), \quad (9)$$

$$u_4 = \frac{C}{J_3} (T_1 - T_2 + T_3 - T_4), \quad (10)$$

where  $u_1$  is the total thrust;  $u_2, u_3$ , and  $u_4$  are the pitch, roll, and yaw moments, respectively;  $J_i$  ( $i = 1, 2, 3$ ) is the moments of inertia with respect to the axes; and  $C$  is the force-to-moment scaling factor.

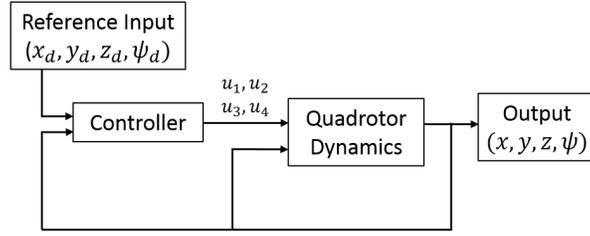


Figure 3. Structure of the quadrotor system with controller.

## 2.2 PD Controller

In this subsection, the PD controller for the quadrotor model is presented. The entire quadrotor control system is illustrated in Fig. 3. Equations (1)-(6) can be divided into a fully-actuated subsystem (11) and an under-actuated subsystem (12)-(13) as follows:

$$\begin{bmatrix} \ddot{z} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} u_1 (\cos \phi \cos \theta) - g \\ u_4 \end{bmatrix}, \quad (11)$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} u_1 \cos \psi & u_1 \sin \psi \\ u_1 \sin \psi & -u_1 \cos \psi \end{bmatrix} \begin{bmatrix} \cos \phi \sin \theta \\ \sin \phi \end{bmatrix}, \quad (12)$$

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} u_2 l \\ u_3 l \end{bmatrix}. \quad (13)$$

In order to control the under-actuated subsystem, the  $x$  and  $y$  positions will be controlled by using  $\phi$  and  $\theta$  indirectly. These variables are closely related to each other. Hence,  $\phi$  and  $\theta$  may be used to control  $x$  and  $y$  inputs, and are found using (14) and (15).

$$\phi_d = \sin \psi (\alpha \dot{e}_x - \beta e_x) - \cos \psi (\alpha \dot{e}_y + \beta e_y), \quad (14)$$

$$\theta_d = \cos \psi (\alpha \dot{e}_x + \beta e_x) + \sin \psi (\alpha \dot{e}_y + \beta e_y), \quad (15)$$

where  $\alpha$  and  $\beta$  are constant values,  $\dot{e}_x := \dot{x}_d - \dot{x}$ ,  $e_x := x_d - x$ ,  $\dot{e}_y := \dot{y}_d - \dot{y}$  and  $e_y := y_d - y$ . Then we can control desired  $x$ - $y$  plane motion by using  $u_2$  and  $u_3$  with PD controller. Also, we can define  $u_1$  and  $u_4$  to control the  $z$  and  $\psi$  states directly. Therefore, the PD controller of the quadrotor can be written as per (16)-(19).

$$u_1 = k_{p,z}(z_d - z) + k_{d,z}(\dot{z}_d - \dot{z}) + g, \quad (16)$$

$$u_2 = k_{p,\phi}(\phi_d - \phi) + k_{d,\phi}(\dot{\phi}_d - \dot{\phi}), \quad (17)$$

$$u_3 = k_{p,\theta}(\theta_d - \theta) + k_{d,\theta}(\dot{\theta}_d - \dot{\theta}), \quad (18)$$

$$u_4 = k_{p,\psi}(\psi_d - \psi) + k_{d,\psi}(\dot{\psi}_d - \dot{\psi}), \quad (19)$$

where  $k_p$  and  $k_d$  are proportional and derivative gains, respectively.

## 3. GRADIENT METHODS

In this section, an optimization problem is considered. The best or optimal values of PD controller gains need to be defined in an effort to improve state trajectory following (i.e., difference between the actual and desired state values). This section describes two gradient algorithms, so that the optimized gain values for minimizing the defined objective function can be calculated. In this paper, the objective function  $J$  defined in (20) is considered. This  $J$  function is the sum of the square of each difference between the reference input and the actual output.

$$J(\chi(\mathbf{x}_d, \mathbf{k})) = \int_0^{t_f} (\mathbf{x}_d - \mathbf{x})^2 dt \quad (20)$$

where function  $\chi$  is the system of the quadrotor related with gain  $\mathbf{k}$ ;  $\mathbf{x} = [x, y, z, \psi]$ ,  $\mathbf{x}_d = [x_d, y_d, z_d, \psi_d]$ ,  $\mathbf{k} = [k_{p,z}, k_{d,z}, k_{p,\phi}, k_{d,\phi}, k_{p,\theta}, k_{d,\theta}, k_{p,\psi}, k_{d,\psi}]$  and  $t_f$  is a final time.

### 3.1 The Method of Steepest Descent

One of the most popular gradient optimization algorithms is the method of steepest descent, whereby the gradient direction is chosen to make the objective function fully-minimized at each step. This is an iterative algorithm which generates a sequence of points. Each new point corresponds to decreasing the value of the objective function [16].

The algorithm, based on this steepest descent method, can be summarized as per Fig. 4. In this procedure, a small change of gain values are first applied, and values of  $J$  are computed with previous gain values  $\underline{k}$  and the changed gain values  $\Delta\underline{k}$ . Note that  $J_i$  denotes  $J(\chi(\underline{x}_d, \underline{k}_i))$ . Secondly, a gradient of  $J$  is derived,  $\partial J/\partial\underline{k}$ , from (21); neglecting terms of higher order. Next, gain values are obtained for the next step, thereby making  $J$  minimized as per the computed  $\partial J/\partial\underline{k}$  and step size  $\lambda$ . This process is repeated iteratively, and obtains increasingly optimized gain values. Hence, this algorithm is repeated until the difference between the previous  $J$  and current  $J$  is smaller than some desired  $\varepsilon$ . Finally, the optimized gain values may be determined for the quadrotor system by minimizing the function  $J$ .

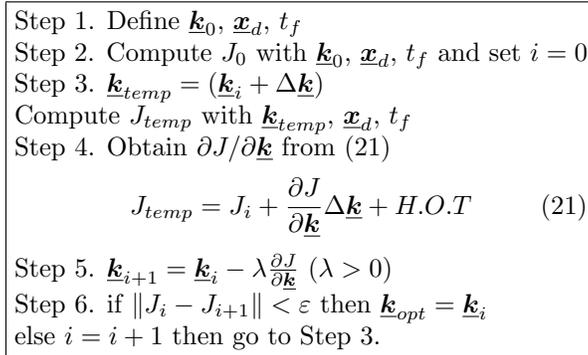


Figure 4. Algorithm based on the Steepest Descent Method.

In order to find the most suitable gradient, selecting a small value of  $\Delta\underline{k}$  is required. Also, setting a small step size,  $\lambda$ , can avoid gain value divergence. This method do not give a global solution, so that appropriate initial gain values are required.

### 3.2 Newton's method

This subsection described Newton's method. It is similar to the algorithm summarized in Fig. 4. Newton's method, however, uses first and second derivatives, whereas the steepest descent method neglects higher order terms above the first order. Due to higher derivatives used in Newton's method, the result of this algorithm generally performs better than the method of steepest descent.

A differential equation of (21), with the necessary condition  $\partial J/\partial\underline{k} = 0$ , is used to determine the second-order gradient. Then (21) can be rewritten as follows:

$$\frac{\partial J_{temp}}{\partial\underline{k}} = \frac{\partial J}{\partial\underline{k}} + \left[ \frac{\partial^2 J}{\partial\underline{k}^2} \Delta\underline{k} \right]^T + H.O.T \cong 0 \quad (22)$$

Next,  $\underline{k}_{i+1}$  is calculated neglecting terms above second order minimizing  $J$  as per (23).

$$\underline{k}_{i+1} = \underline{k}_i - \lambda \left[ \left[ \frac{\partial^2 J}{\partial\underline{k}^2} \right]^{-1} \left[ \frac{\partial J}{\partial\underline{k}} \right] \right]^T \quad (\lambda > 0) \quad (23)$$

To handle the second-order gradient in this challenge, Newton's method for nonlinear least-squares is applied.

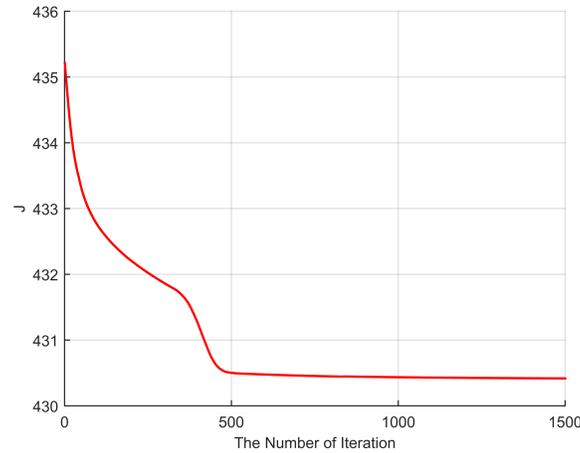


Figure 5.  $J$  through the 1st order gradient method.

First, consider the following problem:

$$\min. \sum_{i=1}^n (r_i(\chi))^2 \quad (24)$$

where  $r_i : \mathbb{R}^n \rightarrow \mathbb{R}$  are given functions and  $\chi$  is the function of quadrotor dynamics related with  $x_d$  and gain  $\underline{k}$ . In this paper,  $r_i(\chi)$  can be defined as  $x_d - x_i$ . Then, the objective function is written as  $J = \underline{\mathbf{r}}^T \underline{\mathbf{r}}$  defining  $\underline{\mathbf{r}} = [r_1, \dots, r_n]^T$ .

Second, compute the  $j$ -th component of  $\nabla J(\chi)$ ,

$$\nabla J(\chi) = \frac{\partial J}{\partial k_j} = 2 \sum_{i=1}^n r_i(\chi) \frac{\partial r_i}{\partial k_j}. \quad (25)$$

Denote the first-order gradient  $G_1$  by  $\frac{\partial r_i}{\partial k_j}$ , such that the gradient of  $J$  can be written as

$$\nabla J(\chi) = 2G_1(\chi)^T \underline{\mathbf{r}}(\chi). \quad (26)$$

Third, compute the second-order gradient of  $J$  as follows.

$$\frac{\partial^2 J(\chi)}{\partial x_k \partial x_j} = 2 \sum_{i=1}^n \left( \frac{\partial r_i(\chi)}{\partial x_k} \frac{\partial r_i(\chi)}{\partial x_j} + r_i(\chi) \left( \frac{\partial^2 r_i(\chi)}{\partial x_k \partial x_j} \right) \right) \quad (27)$$

In this application, the last term of (27) involves the second derivatives of the function  $\underline{\mathbf{r}}$ , such that the term is considered negligible. Therefore, the next step,  $\underline{\mathbf{k}}_{i+1}$ , can be computed with the following equation, commonly known as the *Gauss-Newton method*.

$$\underline{\mathbf{k}}_{i+1} = \underline{\mathbf{k}}_i - (G_1(\chi)^T G_1(\chi))^{-1} G_1(\chi)^T \underline{\mathbf{r}}(\chi) \quad (28)$$

Finally, with this method, optimized gain values may generally be obtained through recursive iterations.

#### 4. SIMULATION

In this section, the aforementioned algorithms are applied and simulated on a quadrotor PD controller. The simulation results show the change of the objective function, the variation of the gain values per each iteration, and the final trajectory of the quadrotor with optimized gain values. This data is used for comparison purposes. The simulation parameter settings were defined as follows:

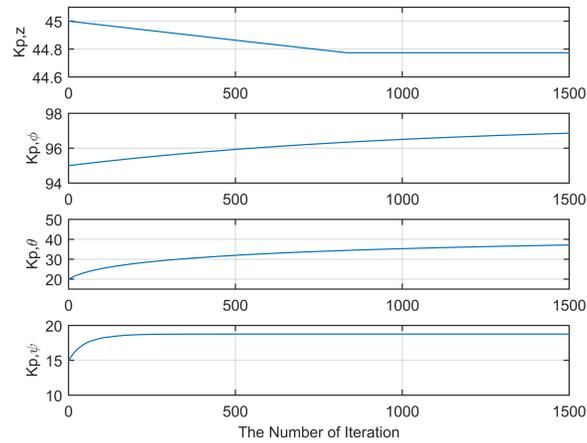


Figure 6.  $K_p$  through the 1st order gradient method.

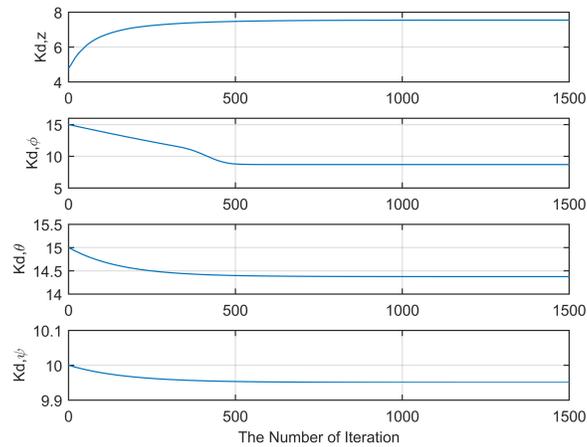


Figure 7.  $K_d$  through the 1st order gradient method.

$$\begin{aligned}
 \underline{k}_0 &= [45, 4.8, 95, 15, 20, 15, 15, 10], \\
 m &= 1 \text{ kg}, \\
 l &= 0.5 \text{ m}, \\
 g &= 9.81 \text{ m/s}^2, \\
 \underline{x}_{init} &= [0, 0, 0, 0] \\
 \underline{x}_d &= [2, -3, 4, \pi/2 \text{ rad}] \\
 t_f &= 25 \text{ sec}
 \end{aligned} \tag{29}$$

In this simulation, the quadrotor is tasked with flying from the origin (0,0,0) to the desired point (2, -3, 4). The objective function is calculated and used to tune the controller gains. It is assumed that measurement and system noise exists with a mean 0 and standard deviation 0.05.

The results of the first-order gradient algorithm are depicted in Figs. 5-7. As illustrated in Fig. 5, the variation of the objective function converges to the minimized point after about 500 iterations. Also, the convergence of proportional and derivative gain values to the optimized values are shown in Figs. 6 and 7, respectively. The

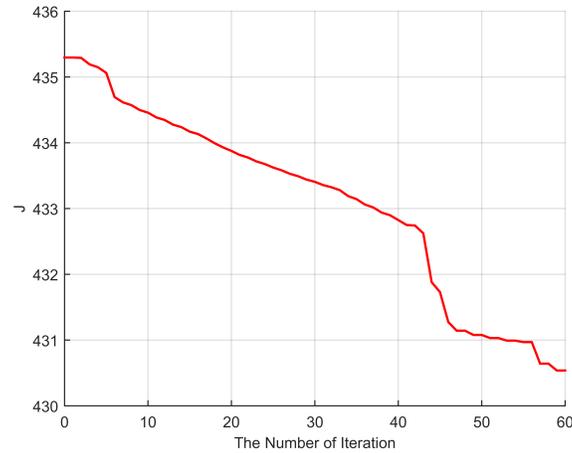


Figure 8. Change of  $J$  with the 2nd order gradient method.

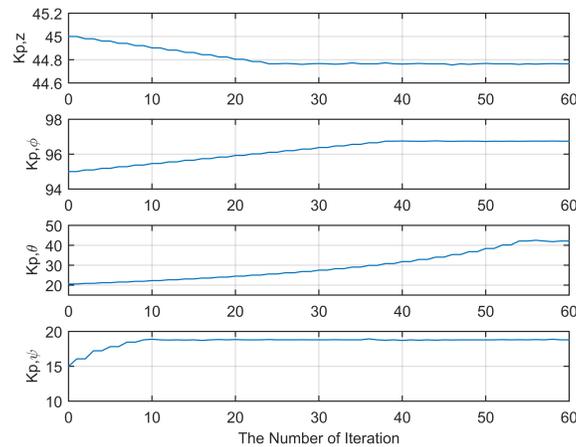


Figure 9.  $K_p$  through the 2nd order gradient method.

optimized gain values generated by the first-order gradient method,  $\mathbf{k}_{opt}$ , were found to be [44.765 7.238 96.7445 8.670 42.141 16.445 18.795 9.941].

The results of the second-order gradient algorithm are illustrated in Figs. 8-10. In Fig. 8, the value of  $J$  converges to the same minimized point as the first-order gradient algorithm after about 60 iterations; it is significantly faster. Also, the proportional and derivative gain values converged to the optimized values for a minimal  $J$ , as shown in Figs. 9 and 10.

From these results, it is demonstrated that the convergence rate of the second-order gradient method is faster than the first-order gradient algorithm. In other words, the second-order gradient method has superior properties of convergence. However, the drawback for the second-order gradient method is a high computational load required to evaluate the gradient.

It is important to note that, in some cases, both of the optimization methods may not find a global solution. Therefore, it is sometimes required to carefully set initial values; these values can be determined during a flight test or through system identification. This will help ensure that the gradient methods will perform correctly, and helps to avoid local minimums, which can sometimes yield faux-minimal gain and function values.

Finally, Fig. 11 shows the trajectory state values of the quadrotor and the corresponding optimized gain values. It is demonstrated that the desired path was followed correctly, based on a minimal  $J$ .

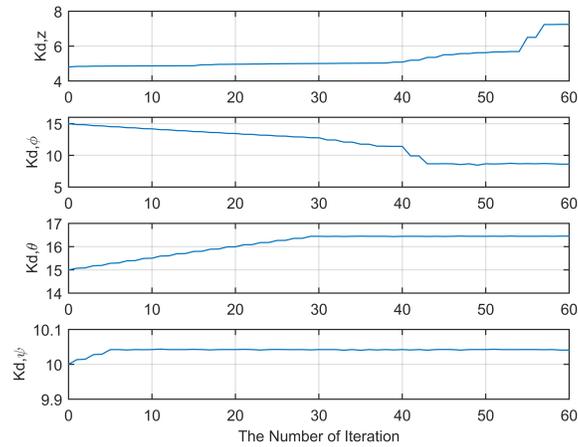


Figure 10.  $K_d$  through the 2nd order gradient method.

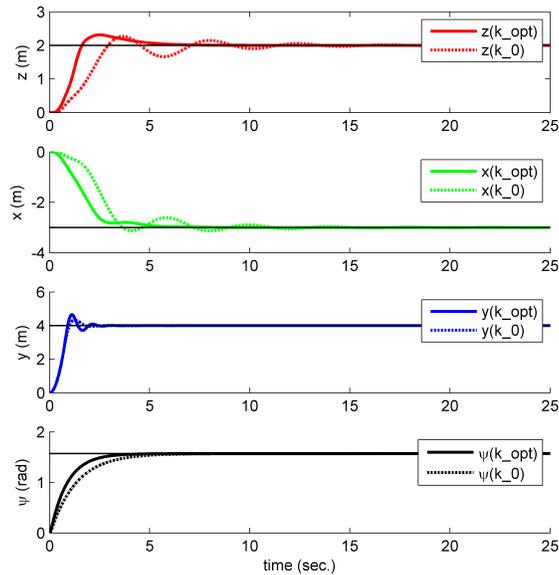


Figure 11. Trajectory of Quadrotor with Initial Gain ( $K_0$ ) and Optimized Gain ( $K_{opt}$ ).

## 5. CONCLUSIONS

In this paper, two types of optimization algorithms were presented. These strategies were used for gain tuning of a PD controller used by a quadrotor. The first-order gradient algorithm, which is based on the method of steepest descent, is a tool for finding optimal gain values that minimize the objective function. It was demonstrated that the computational load of this algorithm was not significant, however it requires a large number of iterations in order to obtain optimal gain values. As an alternative, the second-order gradient method based on Newton's method was applied on the quadrotor controller. It was demonstrated that this algorithm requires significantly fewer iterations for convergence. The results of this paper show that the second-order gradient method is suitable for application on a quadrotor system for PD gain optimization. Future research will include application of these methods on an experimental apparatus. Furthermore, uncertainties and external disturbances will be injected into the system, and the results will be studied.

## REFERENCES

1. M. Achtelik, A. Bachrach, R. Heb, S. Prentice, N. Roy, "Autonomous navigation and exploration of a quadrotor helicopter in GPS-denied indoor environments," in *Proceedings of the Robotics : Science and Systems Conference*, June 2008.
2. H. C. Yang, R. AbouSleiman, B. Sababha, E. Gjioni, D. Korff, O. Rawashdeh, "Implementation of an Autonomous Surveillance Quadrotor System," in *Proceedings of the AIAA Infotech@Aerospace Conference and AIAA Infotech@Aerospace Conference*, Seattle, WA, USA, April, 2009.
3. Q. Lindsey, D. Mellinger, and V. Kumar, "Construction with quadrotor teams," *Autonomous Robots*, vol. 33, Issue 3, pp 323-336, 2012.
4. N. Michael, S. Shen, K. Mohta, Y. Mulgaonkar, V. Kumar, K. Nagatani, Y. Okada, S. Kiribayashi, K. Otake, K. Yoshida, K. Ohno, E. Takeuchi, and S. Tadokoro, "Collaborative Mapping of an Earthquake-Damaged Building via Ground and Aerial Robots," *Journal of Field Robotics*, vol. 29, Issue 5, 2012.
5. P. Pounds, R. Mahony, P. Corke, "Modelling and control of a large quadrotor robot," *Control Engineering Practice*, vol. 18, Issue 7, pp. 691-699, 2010.
6. J. Kim, M. S. Kang, S. Park, "Accurate Modeling and Robust Hovering Control for a Quad-rotor VTOL Aircraft," *Journal of Intelligent and Robotic Systems : Theory and Applications*, vol. 57, pp. 9-26, 2010.
7. M. Bosnak, D. Matko, S. Blazic, "Quadcopter control using an on-board video system with off-board processing," *Robotics and Autonomous Systems*, vol. 60, Issue 4, 2012.
8. S. Bouabdallah, R. Siegwart, "Full Control of a Quadrotor," in *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San diago, CA, USA, Oct., 2007.
9. R. Xu, Ü. Özgüner, "Sliding Mode Control of a Quadrotor Helicopter," in *Proceedings of the 45th IEEE Conference on Decision Control*, San diago, CA, USA, Dec., 2006.
10. S. Bouabdallah, R. Siegwart, "Backstepping and Sliding-mode Techniques Applied to an Indoor Micro Quadrotor", in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain, Apr., 2005
11. S. Bouabdallah, A. Noth, R. Siegwart, "PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor," in *Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, Sep., 2004.
12. A. L. Salih, M. Moghavvemi, H. A. F. Mohamed, K. S. Gaeid, "Modelling and PID Controller Design for a Quadrotor Unmanned Air Vehicle," in *Proceedings of 2010 IEEE International Conference on Automation, Quality and Testing, Robotics*, Cluj-Napoca, Romania, 2010.
13. B. Erginer, E. Altug, "Modeling and PD Control of a Quadrotor VTOL Vehicle," in *Proceedings of the 2007 IEEE Intelligent Vehicle Symposium*, Istanbul, Turkey, Jun., 2007.
14. T. P. Blanchett, G. C. Kember, R. Dubay, "PID gain scheduling using fuzzy logic," *The International Society of Automation Transactions*, vol. 39, no. 3, pp. 317-325, 2000.
15. A. Milhim, Y. Zhang, C. A. Rabbath, "Gain Scheduling Based PID Controller for Fault Tolerant Control of a Quad-Rotor UAV," in *Proceedings of the AIAA Infotech@Aerospace Conference*, Atlanta, GA, USA, Apr., 2010.
16. E. K. P. Chong, S. H. Zak, "An Introduction to Optimization (Second Edition)," *John Wiley & Sons*, 2001.