

Penalaan Mandiri *Full State Feedback* LQR dengan JST Tiruan Pada Kendali *Quadrotor*

Faisal Fajri Rahani*¹, Tri Kuntoro Priyambodo ²

¹Program Pascasarjana Ilmu Komputer, DIKE, FMIPA UGM, Yogyakarta

²Departemen Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: *¹faisal.fajri.r@gmail.com, ²mastri@ugm.ac.id

Abstrak

Quadrotor adalah salah satu jenis pesawat tanpa awak yang memiliki kemampuan terbang landas vertikal. Pada penelitian ini dirancang dan dibuat sistem yang mampu menstabilkan *quadrotor* saat kondisi terbang dengan mempertahankan sudut roll, pitch, yaw, serta posisi sumbu x , y , dan z menggunakan metode *full state feedback* LQR dengan jaringan syaraf tiruan (JST).

Metode *full state feedback* LQR menggunakan 12 state dengan masing-masing konstanta K ditala dengan JST. Penelitian ini mengimplementasikan metode JST untuk mengubah konstanta feedback pada sudut roll, pitch, dan yaw serta sumbu x , y , dan z . Metode jaringan syaraf tiruan menggunakan 12 input layer, 48 hidden layer, dan 1 output layer.

Pengujian dengan JST memperbaiki *rise time* hingga $\pm 2,18$ detik pada sudut roll, $\pm 1,23$ detik pada sudut pitch, dan $\pm 0,31$ detik pada sudut yaw. Perbaikan nilai *settling time* hingga $\pm 2,41$ detik pada sudut roll, $\pm 1,23$ detik pada sudut pitch, dan $\pm 1,07$ detik pada sudut yaw. Perbaikan nilai *steady state error* sebesar $\pm 0,61\%$ pada sudut roll, $\pm 4,88\%$ pada sudut pitch, dan $\pm 0,82\%$ pada sudut yaw.

Kata kunci—pesawat tanpa awak, JST, UAV

Abstract

Quadrotor is one type of unmanned aerial vehicle that has the ability to vertical takeoff and landing. In this research, a system designed to stabilize *quadrotor* during flight condition by maintaining at angle of roll, pitch, yaw, and x , y , and z axis position using LQR full state feedback with artificial neural network (ANN).

The LQR full state feedback method uses 12 states with each K constant being tuned with ANN. This research implements ANN method to change feedback constant at angle of roll, pitch, and yaw and x , y , and z axis. The artificial neural network method uses 12 input layers, 12 hidden layers, and 1 output layer.

Testing with ANN improved the *rise time* to ± 2.18 seconds at the roll angle, ± 1.23 seconds at the pitch angle, and ± 0.31 seconds at the yaw angle. Improved *settling time* value up to ± 2.41 seconds at roll angle, ± 1.23 seconds at pitch angle, and ± 1.07 seconds at yaw angle. Improved *steady state error* value of $\pm 0.61\%$ at roll angle, $\pm 4.88\%$ at pitch angle, and $\pm 0.82\%$ at the yaw angle.

Keywords—unmanned aerial vehicle, neural networks, UAV

1. PENDAHULUAN

Unmanned Aerial Vehicle (UAV) didefinisikan sebagai wahana terbang tanpa awak yang dapat terbang secara otonom (mandiri) dengan sistem auto pilot atau dapat dioperasikan dengan menggunakan kendali radio dari jarak jauh. Salah satu dari jenis UAV yang sering digunakan saat ini adalah quadrotor. Popularitas quadrotor telah meningkat kalangan penggemar, pemerhati militer serta peneliti karena memiliki fleksibilitas yang baik dan kemampuan manuver yang handal[1].

Quadrotor adalah salah satu jenis dari UAV yang dapat lepas landas dalam landasan yang terbatas, hal ini dikarenakan quadrotor dapat terbang dan mendarat secara vertikal [2]. Quadrotor menggunakan empat buah rotor dan empat buah bilah baling yang digunakan untuk terbang [3]. Agar dapat mencapai keadaan stabil maka quadrotor membutuhkan sistem kendali kesetabilan yang sesuai dengan peruntukannya.

Ada beberapa metode kendali yang telah banyak diimplementasikan pada sistem kendali quadrotor. Diantara metode kendali tersebut adalah metode PID (Proportional Integral Derivatif), Fuzzy, Ant Colony Optimization [4], *Full State Feedback* dengan LQR (Linier Quadratic Regulator), dan lain sebagainya. Salah satu metode kendali yang banyak digunakan di dalam dunia penerbangan yaitu metode full state feedback LQR.

Metode *full state feedback* LQR merupakan pengembangan dari metode *pole placement* dan HJB (Hamiltonian Jacobi Bellman) dengan cara mengatur *state* sistem pada sistem bernilai nol dengan memasukkan seoptimal mungkin [5]. Hal tersebut bertujuan untuk meminimalkan nilai *overshoot* dan mendapatkan tanggap waktu yang sesuai dengan kriteria sistem tersebut. *Full state feedback* LQR merupakan metode kendali yang baru dan dapat menerapkan sistem banyak masukan dan banyak keluaran atau *Multiple Input Multiple Output* (MIMO).

Metode kendali *full state feedback* LQR menggunakan sistem *state space* pada sistem kendalinya [6]. Sistem tersebut dapat diterapkan pada *quadrotor* dengan menentukan jumlah *state* yang digunakan. Jumlah *state* yang digunakan pada kendali *full state feedback* LQR bergantung pada jumlah masukan dan orde dari sistem tersebut.

Pengendalian terbang quadrotor menggunakan metode kendali full state feedback LQR memiliki kekurangan. Sistem kendali yang menggunakan pemodelan sistem seperti kendali *full state feedback* LQR akan menghasilkan keluaran yang kurang akurat akibat dari linierisasi kendali dengan LQR serta perhitungan parameter pemodelan kendali yang kurang tepat [7]. Kekurangan tersebut dapat juga berupa komponen konstanta *feedback* bersifat tetap sedangkan sistem ini membutuhkan kendali yang dapat menyesuaikan diri saat terkena gangguan tertentu [8]. Guna membuat komponen konstanta *feedback* berubah sesuai dengan kebutuhan sistem serta sesuai dengan kondisi saat terbang dibutuhkan algoritma pendukung yang dapat.

Beberapa metode dapat digunakan untuk menala komponen konstanta *feedback* seperti jaringan syaraf tiruan. Jaringan syaraf tiruan sebagai representasi tiruan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran seperti pada otak manusia [9]. Jaringan syaraf digunakan untuk menyesuaikan nilai konstanta *feedback* [7]. Metode jaringan syaraf tiruan digunakan sebagai sistem untuk menala konstanta *feedback* pada sistem kendali full state feedback LQR agar sesuai dengan kebutuhan sistem serta dapat menyesuaikan terhadap gangguan yang ada.

2. METODE PENELITIAN

2.1 Analisis Sistem

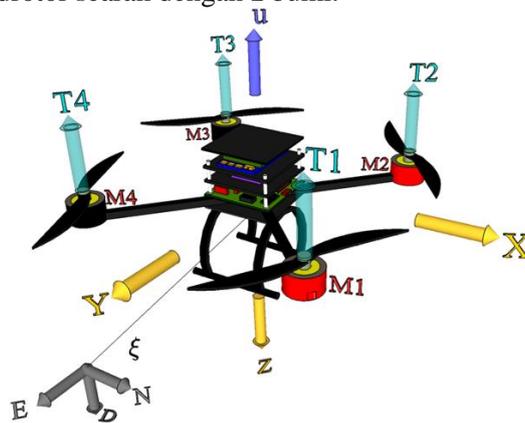
Quadrotor terdiri dari empat buah rotor yang masing- masing berada pada ujung- ujung dari kerangka quadrotor. Arah putaran masing- masing rotor diatur sedemikian rupa dengan dua buah rotor yang berputar searah jarum jam dan dua buah rotor yang lain berputar

berlainan dengan arah jarum jam. masing- masing rotor dengan putaran yang searah diletakkan bersebrangan dengan rotor yang memiliki arah putaran yang sama. Masing- masing rotor tersebut dikendalikan sedemikian rupa hingga quadrotor dapat mempertahankan posisinya.

Konfigurasi quadrotor terdiri dari dua jenis, yaitu konfigurasi terbang plus (+) dan konfigurasi terbang silang (x). Konfigurasi ini akan mempengaruhi cara kerja quadrotor dalam mengendalikan setabilan dan posisinya. Setiap rotor pada sistem quadrotor digunakan untuk menghasilkan gaya angkat dan torsi. Setiap rotor dikendalikan dengan kecepatan tertentu pada masing- masing rotor untuk menjaga kesetabilan pada quadrotor. Guna memperoleh kesetabilan pada quadrotor maka dibutuhkan sistem kendali untuk mengatur masing- masing rotor.

Quadrotor dapat bergerak secara translasi terhadap sumbu x, y, dan z bumi dengan mengatur kecepatan masing- masing rotor. Gerakan keempat rotor dengan kesetabilan yang seimbang akan menghasilkan gaya angkat untuk quadrotor. Gaya angkat tersebut dapat dimanfaatkan quadrotor untuk bergerak pada sumbu z bumi. Sistem quadrotor bergerak dengan orientasi pitch yaitu sistem quadrotor yang bergerak terhadap sumbu x bumi. Sudut orientasi roll digunakan sistem quadrotor untuk bergerak terhadap sumbu y bumi.

Posisi antara kerangka badan quadrotor dan orientasi bumi ditunjukkan pada Gambar 1. Arah sumbu positif x kerangka quadrotor adalah ke depan pada sistem quadrotor searah dengan x bumi. Arah sumbu y body frame ke kiri quadrotor searah dengan y bumi sedangkan sumbu z kearah atas kerangka quadrotor searah dengan z bumi.



Gambar 1 Sumbu x,y dan y quadrotor dan bumi[3]

Kesetabilan terbang pada quadrotor dikatakan stabil pada gerakan rotasi jika quadrotor dapat mempertahankan sudut rotasi pada nilai toleransi sebesar 5% [10]. Pada sistem ini dibutuhkan sebuah sensor orientasi. Sensor orientasi digunakan untuk mengukur sudut- sudut yang dibentuk oleh pergerakan quadrotor terhadap sumbu bumi. Nilai sudut untuk orientasi pitch dan roll pada sistem ini didapatkan dari sensor orientasi IMU. Pembacaan sensor IMU menggunakan pustaka MPU6050. Pada sensor IMU tersebut terdapat sensor akselerometer untuk mengukur percepatan translasi serta terdapat sensor giroskop sebagai pembaca data percepatan sudut. Nilai sudut orientasi yaw (ψ) pada sistem ini menggunakan sensor kompas HMC5883L.

Kendali yang dikembangkan pada penelitian ini menggunakan sistem kendali *fullstate feedback* LQR dengan jaringan syaraf tiruan. Jaringan syaraf pada sistem ini digunakan untuk menala komponen *fullstate feedback* K agar dapat berubah sesuai dengan kebutuhan sistem. Jaringan syaraf tiruan yang digunakan pada sistem ini menggunakan 3 buah lapisan. Lapisan pada jaringan syaraf tersebut terdiri dari lapisan masukan, lapisan tersembunyi, dan lapisan keluaran. Penalaan komponen umpan balik K pada sistem ini dilakukan saat sistem quadrotor dalam keadaan terbang.

Sistem kendali memiliki beberapa parameter yang digunakan untuk menggambarkan

spesifikasi dari sistem kendali tersebut. Batas nilai toleransi steady-state error untuk sistem linier orde 1 adalah 2% dari rentang nilai yang dikendalikan [10]. Sedangkan sistem yang dibuat pada penelitian ini adalah sistem nonlinier orde 2, maka batas toleransinya adalah 5%. Rentang nilai yang dikendalikan pada sudut pitch dan roll adalah sebesar 90° dikarenakan sudut yang lebih besar dari 90° merupakan kondisi dimana quadrotor dapat terbalik. Rentang nilai sudut yaw sebesar 360° dikarenakan nilai satu rotasi penuh adalah 360° .

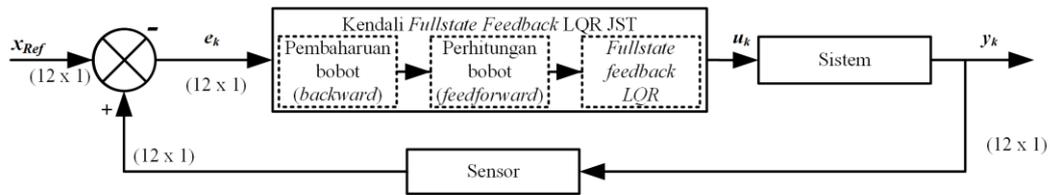
Penelitian quadrotor parameter nilai yang dapat dijadikan acuan kestabilan terbang quadrotor pada sudut pitch, roll, dan yaw. Acuan diambil dari nilai risetime, maximum overshoot, dan settling time. Guna mengetahui tingkat keberhasilan dari penelitian ini, maka diperlukan adanya parameter spesifikasi dari sistem. Spesifikasi sistem digunakan digunakan sebagai acuan keberhasilan dari kendali quadrotor. Spesifikasi sistem kendali yang digunakan pada penelitian ini yaitu:

- Nilai steady-state error dibawah batas toleransi 5% dari nilai maksimal yaitu sebesar 90° untuk sudut pitch dan roll yang dikendalikan dan 360° untuk sudut yaw, sehingga batas toleransi maksimal sudut adalah $\pm 4,5^{\circ}$ untuk sudut pitch dan roll sedangkan sebesar $\pm 18^{\circ}$ sudut yaw [10].
- Nilai steady-state error di bawah batas toleransi ± 2 meter untuk gerak translasi posisi x dan y dalam mempertahankan posisi quadrotor [11].
- Sistem dapat menjaga ketinggian terhadap setpoint posisi z vertikal yang ditentukan pengguna dengan toleransi 5% dari ketinggian setpoint terhadap permukaan bumi . Ketinggian yang dapat ditentukan oleh pengguna sebesar 1m [11].
- Risetime yang dihasilkan lebih cepat dari 1 detik untuk sudut pitch dan roll dan untuk sudut yaw lebih cepat dari 4 detik, settling time yang dihasilkan lebih cepat dari 3 detik untuk sudut pitch dan roll dan untuk sudut yaw lebih cepat dari 6 detik, serta maximum overshoot untuk sudut pitch dan roll yang dihasilkan lebih kecil dari 20% dan untuk yaw yang dihasilkan lebih kecil dari 25% [11].

2. 2 Rancangan Sistem Kendali

Sistem kendali yang diusulkan pada penelitian ini adalah kendali full state feedback LQR dengan jaringan syaraf tiruan untuk mempertahankan posisi pada sistem terbang quadrotor. Sistem kendali ini dibuat agar quadrotor dapat mempertahankan posisinya saat terbang. Kendali ini akan ditanam pada sistem mikrokontroler yang terpasang pada quadrotor.

Rancangan sistem kendali pada quadrotor secara umum ditunjukkan oleh Gambar 2. Pada sistem ini terdapat nilai masukan sistem yang dapat berupa nilai yang telah ditentukan maupun nilai- nilai variabel yang dapat diubah melalui pengendalian dengan pengendali jarak jauh. Nilai masukan referensi tersebut akan dibandingkan dengan nilai bacaan sensor. Hasil selisih dari bacaan sensor dengan nilai set point disebut nilai eror. Nilai eror inilah yang digunakan sebagai acuan kontroler untuk mengendalikan sistem agar sesuai dengan nilai referensi atau set point yang diinginkan. Kontroler akan memberikan perintah kepada final control element sebagai alat untuk mengendalikan sistem agar sesuai dengan yang diinginkan. Keluaran dari final control element biasanya juga terdapat gangguan dari luar sistem. Keluaran pada final control element yang diselisihkan dengan masukan gangguan akan dianggap sebagai sistem proses yang sedang berjalan. Keluaran dari proses tersebut akan dianalisa oleh sistem dengan pembacaan melalui sensor yang digunakan. Hal tersebut dilakukan untuk memperbaiki kendali sistem apabila kendali belum sesuai dengan yang diinginkan.

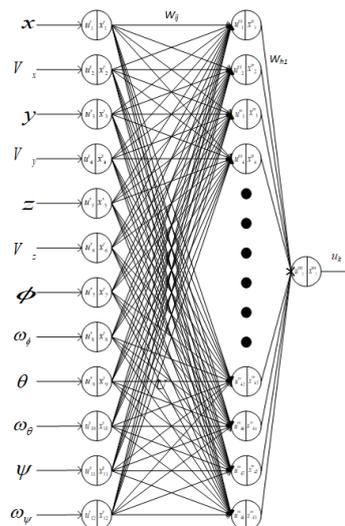


Gambar 2 Sistem Kendali

Sistem kendali pada quadrotor untuk kendali sikap menggunakan kendali full state feedback dengan 12 state dari masukkan sensor. Hasil dari masukkan tersebut kemudian dimasukkan pada sistem *full state feedback* dengan mengalikan hasil masukkan dengan konstanta K berupa matriks (12×4) . Nilai konstanta K pada sistem ini sebelumnya diolah menggunakan sistem jaringan syaraf buatan. Sistem jaringan syaraf tiruan ini akan mengubah nilai K yang telah di tala dengan sistem LQR dengan nilai konstanta Q dan R pada setiap sumbu lalu dioperasikan dengan perhitungan Riccati. Masukkan jaringan syaraf tiruan ini sama dengan masukkan sistem pada quadrotor.

Nilai u yang merupakan masukkan proses dengan kalkulasi perhitungan hasil perkalian antara *gain feedback* dengan *state* sistem yang akan dikendalikan. Sinyal masukkan kendali ini selanjutnya akan digunakan untuk pengendalian kecepatan tiap tiap rotor. Setiap rotor didefinisikan sebagai final control element pada sistem quadrotor.

2. 3 Sistem Jaringan Syaraf Tiruan



Gambar 3 Arsitektur Jaringan Syaraf Tiruan

Input Layer

Input layer merupakan masukkan pada sistem jaringan syaraf. Masukkan pada sistem ini menggunakan 12 buah masukkan. Masukkan- masukkan jaringan syaraf tersebut menggunakan masukkan seperti Gambar 3 dimaksudkan agar masukkan jaringan syaraf sama dengan masukkan dalam sistem kendali *quadrotor* itu sendiri.

Hidden Layer

Pada *hidden layer* jaringan syaraf tiruan digunakan enam buah *node*. Masukkan dari *hidden layer* pada jaringan syaraf tiruan adalah hasil jumlahan dari perkalian bobot W_{ij} dengan nilai hasil aktivasi *node* tersebut. Dimana i merupakan urutan *node* masukkan dan j merupakan urutan *node hidden*.

Output Layer

Pada *layer* keluaran terdapat satu *node* yang memperoleh nilai dari hasil jumlahan dari perkalian bobot \mathbf{W}_{hi} dan keluaran dari *node hidden* x_h^2 . Aktivasi pada setiap *node* pada ketiga lapisan menggunakan fungsi aktivasi *sigmoid*. Fungsi aktivasi tersebut bertujuan untuk menjaga nilai dari masing-masing *node* pada setiap lapisan tidak berlebih dan masih saling berkolerasi pada setiap lapisan.

Bobot Jaringan Syaraf

Sistem jaringan syaraf yang digunakan pada sistem ini ditanamkan pada perangkat keras secara langsung dengan pemrograman tertanam. Sistem jaringan syaraf ini menggunakan inisialisasi bobot awal W_{ij} dengan persamaan (1).

$$W_{ij} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \quad (1)$$

Sedangkan untuk nilai bobot jaringan syaraf dari *hidden layer* ke *output layer* menggunakan nilai awal konstanta \mathbf{K} hasil dari penalaan menggunakan metode LQR. Bentuk dari bobot W_{hi} menggunakan persamaan (2).

$$W_{hi} = [K_{11} \quad K_{21} \quad K_{31} \quad K_{41} \quad K_{12} \quad K_{22} \quad K_{32} \quad K_{42} \quad \dots \quad K_{212} \quad K_{312} \quad K_{412}] \quad (2)$$

Nilai bobot W_{hi} pada sistem ini merupakan representasi nilai konstanta *feedback* yang digunakan pada sistem kendali *quadrotor*. Hubungan jaringan syaraf tiruan dan kendali *full state feedback* disini yaitu pada penalaan nilai \mathbf{K} sebelumnya yang telah ditentukan oleh metode LQR. Nilai \mathbf{K} tersebut dimasukkan sebagai nilai bobot awal pada bobot W_{hi} .

Algoritma back-propagation

Sistem jaringan syaraf tiruan pada rancangan kendali ini menggunakan nilai kesalahan sistem yang berasal dari keluaran sistem secara *real time*. Respon sistem dianalisa dengan menghitung nilai kesalahan keluaran sistem yang berupa 12 state yang direpresentasikan dengan nilai tertentu. Nilai kesalahan pada sistem ini berupa nilai bacaan sensor pada sistem yang kemudian diselisihkan dengan nilai state referensi sistem.

Tujuan utama dari jaringan syaraf tiruan pada sistem ini adalah untuk meminimalkan nilai rerata kesalahan sistem yang dikuadratkan (J). Pada sistem ini menggunakan fungsi eror pada persamaan (3). Nilai merupakan keluaran sistem berupa bacaan sensor. Nilai X_{ref} merupakan nilai referensi sistem berupa masukkan sistem maupun nilai yang telah ditetapkan. Nilai $y_{(n)}$ merupakan nilai dari keluaran sistem pada setiap state sistem. Selisih dari kedua nilai tersebut merupakan sebuah nilai kesalahan sistem. Nilai n merupakan banyaknya state yang

diamati pada sistem ini yaitu sebanyak 12 state sistem.

$$j = E_p = \sum_n \frac{1}{2} (y_{(n)} - X_{ref(n)})^2 \quad (3)$$

Guna mengatur perubahan nilai bobot ini maka digunakan fungsi *learning rate* η dengan fungsi pembaharuan bobot sebesar 0,01. Nilai bobot \mathbf{W}_{h1} akan selalu berubah bergantung dengan masukkan sistem jaringan syaraf tiruan. Nilai bobot tersebut selanjutnya digunakan untuk menentukan keluaran pada sistem kendali *full state feedback*.

Menentukan nilai bobot dari lapisan *hidden* ke lapisan keluaran

Untuk mencari nilai bobot dari *node* ke \mathbf{W}_{h1} lapisan *hidden* ke lapisan keluaran menggunakan persamaan (4) sampai (9).

$$w_{h1}(k+1) = w_{h1}(k) - \eta \frac{\partial j}{\partial w_{h1}} \quad (4)$$

dimana n adalah *learning rate* dengan nilai 0,01. Nilai delta J didapatkan dengan persamaan (5).

$$\frac{\partial j}{\partial w_{h1}} = \frac{\partial j}{\partial y} \cdot \frac{\partial y}{\partial u} \cdot \frac{\partial u}{\partial w_{h1}} \quad (5)$$

Dimana y merupakan keluaran respon dari sistem kendali *quadrotor*. Nilai u_k adalah keluaran hasil perhitungan jaringan syaraf tiruan. Nilai \mathbf{W}_{h1} merupakan nilai bobot dari lapisan *hidden* ke lapisan keluaran jaringan syaraf tiruan. Dari persamaan (5) maka dijabarkan persamaan (6) hingga persamaan (8).

$$\frac{\partial j}{\partial y} = x_h^2(k) \quad (6)$$

$$\frac{\partial y}{\partial u} = \frac{\partial y}{\partial t} / \frac{\partial u}{\partial t} = \frac{\dot{y}}{\dot{u}} \approx \frac{\Delta y}{\Delta u} = \frac{y(k) - y(k-1)}{u(k) - u(k-1)} \quad (7)$$

$$\frac{\partial u}{\partial h_1} = \sum_n \frac{1}{2} (y_{(n)} - X_{ref(n)})^2 \quad (8)$$

Nilai X_{h1} merupakan nilai hasil aktivasi dari lapisan *hidden*, nilai y merupakan keluaran sistem kendali *full state feedback*. Hasil dari penjabaran persamaan (4) hingga persamaan (8) maka dapat diperoleh persamaan (9) untuk mendapatkan nilai bobot X_{h1} yang baru.

$$\frac{\partial j}{\partial w_{h1}} = x_h^2(k) \cdot \frac{y(k) - y(k-1)}{u(k) - u(k-1)} \cdot \sum_n \frac{1}{2} (y_{(n)} - X_{ref(n)})^2 \quad (9)$$

Menentukan nilai bobot dari lapisan masukan ke lapisan *hidden*

Untuk mencari nilai bobot dari *node* i dari lapisan masukan ke *node* j lapisan *hidden* menggunakan persamaan (10).

$$w_{ij}(k+1) = w_{ij}(k) - \eta \frac{\partial j}{\partial w_{ij}} \quad (10)$$

dimana n adalah *learning rate* dengan nilai 0,01. Nilai dari delta J didapatkan dengan persamaan (11).

$$\frac{\partial j}{\partial w_{ij}} = \frac{\partial j}{\partial y} \cdot \frac{\partial y}{\partial u} \cdot \frac{\partial u}{\partial w_{ij}} \quad (11)$$

Nilai W_{ij} merupakan nilai bobot dari lapisan *hidden* ke lapisan keluaran jaringan syaraf tiruan. Dari persamaan (11) maka dijabarkan persamaan (12) hingga persamaan (14).

$$\frac{\partial j}{\partial y} = w_{hi} x_i^1(k) \quad (12)$$

$$\frac{\partial y}{\partial u} = \frac{\partial y}{\partial t} / \frac{\partial u}{\partial t} = \frac{\dot{y}}{\dot{u}} \approx \frac{\Delta y}{\Delta u} = \frac{y(k) - y(k-1)}{u(k) - u(k-1)} \quad (13)$$

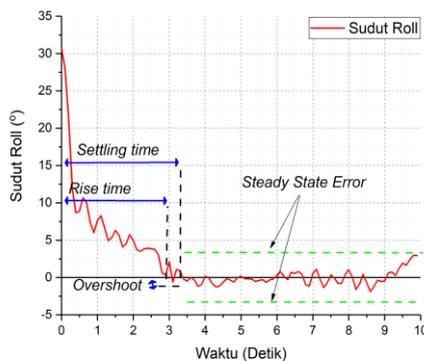
$$\frac{\partial u}{\partial w_{ij}} = \sum_n \frac{1}{2} (y_{(n)} - X_{ref(n)})^2 \quad (14)$$

Nilai x merupakan nilai hasil aktivasi dari lapisan *hidden*. Hasil dari penjabaran persamaan (12) hingga persamaan (14) maka dapat diperoleh persamaan (15) untuk mendapatkan nilai bobot W_{ij} yang baru.

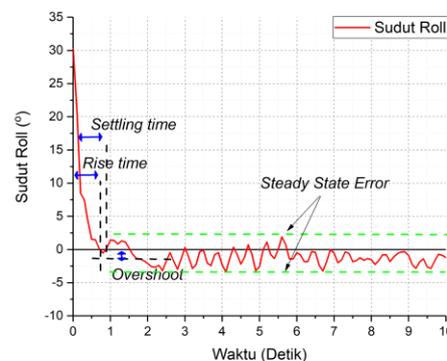
$$\frac{\partial j}{\partial w_{ij}} = w_{hi} x_i^1(k) \cdot \frac{y(k) - y(k-1)}{u(k) - u(k-1)} \cdot \sum_n \frac{1}{2} (y_{(n)} - X_{ref(n)})^2 \quad (15)$$

3. HASIL DAN PEMBAHASAN

3.1 Hasil Sudut Roll



Gambar 4 Grafik Uji Simulasi Sudut Roll



Gambar 5 Grafik Uji JST Sudut Roll

Pengujian pada sudut *roll* dilakukan sebanyak tiga kali dengan gangguan sebesar 30° menunjukkan hasil respon didapatkan *rise time* rata-rata sebesar pada sistem sebesar 2,85 detik untuk menuju setpoint 0° , nilai rata-rata *settling time* selama 3,1 detik, dan rata-rata *overshoot* sekitar 1,1 % hasil pengujian terlihat pada Gambar 4. Pengujian dengan menggunakan JST juga dilakukan sebanyak tiga kali dengan gangguan sebesar 30° menunjukkan hasil respon didapatkan *rise time* rata-rata sebesar 0,67 detik untuk menuju setpoint 0° , nilai rata-rata *settling time* selama 0,93 detik, dan rata-rata *overshoot* sekitar 1,27% hasil pengujian terlihat pada Gambar 5. Pengujian untuk mendapatkan nilai nilai *steady state error* dilakukan dengan mengamati respon sistem tanpa dikenai gangguan. Hasil pengujian didapatkan nilai *steady state error* pada sudut *roll* tanpa jaringan syaraf sebesar $\pm 3,45^\circ$ sedangkan nilai *steady state error* pada sudut *roll* dengan jaringan syaraf sebesar $\pm 2,84^\circ$ dengan pengujian selama 20 detik.

Pada kendali dengan menggunakan jaringan syaraf tiruan juga diamati perubahan nilai konstanta K_{ϕ} dan K_{ω} . Nilai konstanta K_{ϕ} dan $K_{\omega\phi}$ berubah sesuai dengan nilai eror yang masuk pada sistem kendali. Nilai konstanta K_{ϕ} berkisar antara 58,5554 hingga 41,6251 pada pengujian diatas, sedangkan nilai konstanta $K_{\omega\phi}$ berkisar antara 1,729 hingga 3,211 saat terdapat gangguan.

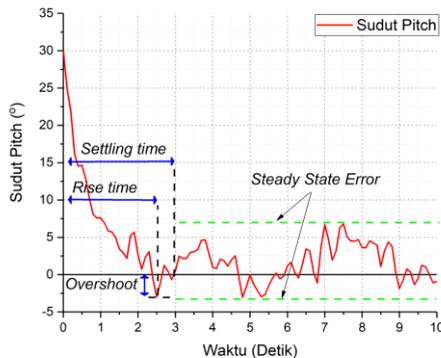
Tabel 1 Perbandingan Respon Sistem Sudut Roll

Respon transien	Tanpa JST	Dengan JST	Spesifikasi sistem
<i>Rise time</i>	2,85 detik	0,67 detik	< 1 detik
<i>Settling time</i>	3,34 detik	0,93 detik	< 3 detik
<i>Overshoot</i>	1,1 %	1,27 %	< 20 %
<i>Steady state error</i>	$\pm 3,45^{\circ}$	$\pm 2,84^{\circ}$	$\pm 4,5^{\circ}$

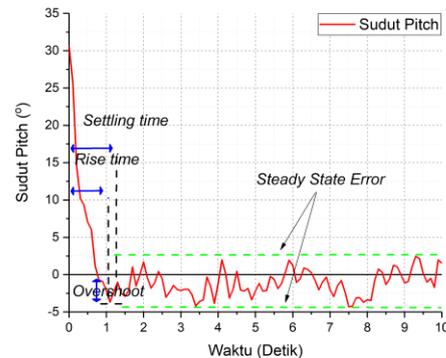
Hasil perbandingan tersebut terlihat pada Tabel 1. Hal tersebut menunjukkan bahwa respon sistem pada sudut roll dengan menggunakan jaringan syaraf tiruan lebih cepat. Nilai *rise time* yang lebih baik mengakibatkan nilai *rise time* pada kendali sudut roll sudah sesuai dengan spesifikasi sistem. Penggunaan jaringan syaraf tiruan memperbaiki nilai *settling time* sistem sehingga sesuai dengan spesifikasi sistem yang telah ditentukan.

Perbaikan nilai *rise time* dan *settling time* sistem secara tidak langsung akan mengakibatkan nilai *overshoot* menjadi lebih besar saat menggunakan kendali dengan jaringan syaraf tiruan. Nilai *overshoot* yang lebih besar tersebut merupakan konsekuensi dari cepatnya respon yang didapatkan. Nilai *steady state error* dengan menggunakan kendali jaringan syaraf tiruan menunjukkan nilai yang lebih baik.

3. 2 Hasil Sudut Pitch



Gambar 6 Grafik Uji Simulasi Sudut Pitch



Gambar 7 Grafik Uji JST Sudut pitch

Pengujian pada sudut pitch dilakukan sebanyak tiga kali dengan gangguan sebesar 30° menunjukkan hasil respon pada sudut pitch didapatkan *rise time* rata-rata sebesar pada sistem sebesar 1,9 detik untuk menuju *setpoint* 0° , nilai rata-rata *settling time* selama 2,3 detik, dan rata-rata *overshoot* sekitar 7,2 % hasil pengujian terlihat pada Gambar 6. Pengujian dengan JST dilakukan sebanyak tiga kali dengan gangguan sebesar 30° menunjukkan hasil respon pada sudut pitch didapatkan *rise time* rata-rata sebesar pada sistem sebesar 0,67 detik untuk menuju *setpoint* 0° , nilai rata-rata *settling time* selama 0,73 detik, dan rata-rata *overshoot* sekitar 8,2 % hasil pengujian dengan JST terlihat pada Gambar 7. Pengujian untuk mendapatkan nilai nilai *steady state error* dilakukan dengan mengamati respon sistem tanpa dikenai gangguan. Hasil pengujian didapatkan nilai *steady state error* pada sudut pitch tanpa jaringan syaraf sebesar $\pm 8,43^{\circ}$ sedangkan nilai *steady state error* pada sudut pitch dengan jaringan syaraf sebesar $\pm 3,55^{\circ}$ dengan pengujian selama 20 detik.

Pada kendali dengan menggunakan jaringan syaraf tiruan juga diamati perubahan nilai konstanta K_0 dan K_{00} . Nilai konstanta K_0 dan K_{00} berubah sesuai dengan nilai eror yang masuk pada sistem kendali. Nilai konstanta K_0 berkisar antara 58,63 hingga 42,6503 pada pengujian diatas, sedangkan nilai konstanta K_{00} berkisar antara 1,729 hingga 3,211 saat terdapat gangguan.

Tabel 2 Perbandingan Respon Sistem Sudut *Pitch*

Respon transien	Tanpa JST	Dengan JST	Spesifikasi sistem
<i>Rise time</i>	1,9 detik	0,67 detik	< 1 detik
<i>Settling time</i>	2,3 detik	0,73 detik	< 3 detik
<i>Overshoot</i>	7,2 %	8,2 %	< 20 %
<i>Steady state error</i>	$\pm 8,43^\circ$	$\pm 3,55^\circ$	$\pm 4,5^\circ$

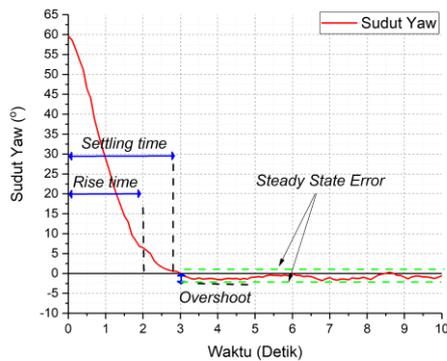
Hasil perbandingan tersebut terlihat pada

Tabel 2. Hasil kendali menggunakan jaringan syaraf tiruan pada sudut *pitch* menghasilkan nilai *rise time* yang lebih singkat. Hasil *settling time* dengan kendali tanpa jaringan syaraf tiruan juga menunjukkan nilai yang lebih dari spesifikasi sistem yang ditentukan. Hal tersebut kemudian diperbaiki dengan penggunaan jaringan syaraf tiruan. Nilai *steady state error* pada kendali tanpa menggunakan jaringan syaraf tiruan belum sesuai dengan spesifikasi sistem. Penggunaan jaringan syaraf tiruan selanjutnya memperbaiki nilai *steady state error* menjadi sesuai dengan spesifikasi sistem.

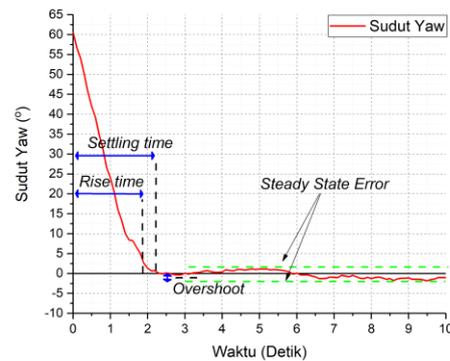
Perbaikan nilai *rise time* dan *settling time* sistem mengakibatkan nilai *overshoot* menjadi lebih besar saat menggunakan kendali dengan jaringan syaraf tiruan. Respon sistem yang lebih cepat akan menyebabkan nilai *overshoot* yang lebih besar, hal tersebut merupakan konsekuensi dari cepatnya respon yang didapatkan. Nilai *overshoot* yang lebih tinggi tersebut masih masuk dalam kriteria spesifikasi sistem.

3. 3 Hasil Sudut Yaw

Pengujian pada sudut *yaw* dilakukan sebanyak tiga kali dengan gangguan sebesar 60° menunjukkan hasil respon *rise time* rata- rata sebesar pada sistem sebesar 2,07 detik untuk menuju *setpoint* 0° , nilai rata- rata *settling time* selama 3,57 detik, dan rata- rata *overshoot* sekitar 3,57 % hasil pengujian terlihat pada Gambar 8. Pengujian dilakukan sebanyak tiga kali dengan dengan JST menunjukkan hasil respon pada sudut *yaw* didapatkan *rise time* rata- rata sebesar pada sistem sebesar 1,76 detik untuk menuju *setpoint* 0° , nilai rata- rata *settling time* selama 2,5 detik, dan rata- rata *overshoot* sekitar 1,76 % hasil pengujian terlihat pada Gambar 9. Pengujian untuk mendapatkan nilai nilai *steady state error* dilakukan dengan mengamati respon sistem tanpa dikenai gangguan. Hasil pengujian didapatkan nilai *steady state error* pada sudut *yaw* tanpa jaringan syaraf sebesar $\pm 2,3^\circ$ sedangkan nilai *steady state error* pada sudut *yaw* dengan jaringan syaraf sebesar $\pm 1,48^\circ$ dengan pengujian selama 20 detik.



Gambar 8 Grafik Uji Simulasi Sudut Yaw



Gambar 9 Grafik Uji JST Sudut yaw

Pada kendali dengan menggunakan jaringan syaraf tiruan juga diamati perubahan nilai konstanta K_{ψ} dan $K_{\omega\psi}$. Nilai konstanta K_{ψ} dan $K_{\omega\psi}$ berubah sesuai dengan nilai eror yang masuk pada sistem kendali. Nilai konstanta K_{ψ} berkisar antara 58,38 hingga 40 pada pengujian diatas, sedangkan nilai konstanta $K_{\omega\psi}$ berkisar antara 1,6 hingga 2,4 saat terdapat gangguan.

Hasil perbandingan pengujian terlihat pada

Tabel 3. Hasil respon kendali dengan nilai konstanta *feedback* hasil simulasi telah sesuai dengan nilai spesifikasi kendali. Hal tersebut ditunjukkan dengan nilai *rise time*, *settling time*, *overshoot*, dan *steady state error* yang telah kurang dari spesifikasi yang telah ditentukan.

Tabel 3 Perbandingan Respon Sistem Sudut yaw

Respon transien	Tanpa JST	Dengan JST	Spesifikasi sistem
<i>Rise time</i>	2,07 detik	1,76 detik	< 3 detik
<i>Settling time</i>	3,57 detik	2,5 detik	< 6 detik
<i>Overshoot</i>	3,57 %	8,2 %	< 20 %
<i>Steady state error</i>	$\pm 2,3^{\circ}$	$\pm 1,48^{\circ}$	$\pm 18^{\circ}$

Hasil respon kendali tanpa tambahan jaringan syaraf tiruan kemudian diperbaiki dengan sistem jaringan syaraf tiruan. Hasil dari penggunaan jaringan syaraf tiruan memperbaiki masing-masing parameter kendali pada sudut yaw seperti *rise time*, *settling time*, dan *overshoot*. Nilai *steady state error* dengan menggunakan kendali jaringan syaraf tiruan juga menunjukkan nilai yang lebih baik.

4. KESIMPULAN

Kesimpulan yang dapat diambil dari penelitian ini adalah telah berhasil dibuat sistem penalaan mandiri komponen full state feedback LQR dengan jaringan syaraf tiruan pada kendali quadrotor. Hasil pengujian kendali pada sudut roll, pitch, dan yaw dengan menggunakan jaringan syaraf tiruan pada kendali *full state feedback* LQR akan memperbaiki *rise time*, *settling time*, *steady state error* sesuai dengan spesifikasi sistem dibandingkan dengan kendali *full state feedback* LQR dengan konstanta hasil simulasi.

Pengujian dengan JST memperbaiki rise time hingga $\pm 2,18$ detik pada sudut roll, $\pm 1,23$ detik pada sudut pitch, dan $\pm 0,31$ detik pada sudut yaw. Perbaikan nilai settling time hingga $\pm 2,41$ detik pada sudut roll, $\pm 1,23$ detik pada sudut pitch, dan $\pm 1,07$ detik pada sudut yaw. Perbaikan nilai steady state eror sebesar $\pm 0,61\%$ pada sudut roll, $\pm 4,88\%$ pada sudut pitch, dan $\pm 0,82\%$ pada sudut yaw. Nilai masing-masing konstanta *feedback* pada kendali *full state*

feedback berubah sesuai dengan kebutuhan sistem saat dikenai gangguan. Nilai perubahan konstanta *feedback* ditala menggunakan sistem jaringan syaraf tiruan.

5. SARAN

Penelitian ini masih terdapat beberapa hal yang dapat disempurnakan. Saran- saran yang dapat digunakan untuk penelitian yang akan datang diharapkan menambahkan kendali integral untuk kendali masing- masing *state* dengan menambahkan sistem jaringan syaraf sebagai alat untuk penalaan agar sistem dapat mendekati nilai referensi dengan baik serta dilakukan penalaan komponen *Q* dan *R* secara *online* dalam pembelajaran jaringan syaraf tiruan untuk memperoleh nilai *K* serta pembelajaran secara *online* pada wahana saat terbang.

DAFTAR PUSTAKA

- [1] H. C. T. E. Fernando, a. T. a De Silva, M. D. C. De Zoysa, K. a D. C. Dilshan, and S. R. Munasinghe, "Modelling, simulation and implementation of a quadrotor UAV," *2013 IEEE 8th Int. Conf. Ind. Inf. Syst. ICIIS 2013 - Conf. Proc.*, pp. 207–212, 2013.
- [2] D. Shatat and T. A. Tutunji, "UAV quadrotor implementation: A case study," in *2014 IEEE 11th International Multi-Conference on Systems, Signals and Devices, SSD 2014*, 2014.
- [3] L. R. García Carrillo, A. E. Dzul López, R. Lozano, and C. Pégard, *Quad Rotorcraft Control*, vol. 1. London: Springer London, 2013.
- [4] T. K. Priyambodo, A. Dharmawan, O. A. Dhewa, and N. A. S. Putro, "Optimizing control based on fine tune PID using ant colony logic for vertical moving control of UAV system," *AIP Conf. Proc.*, vol. 1755, no. 2016, 2016.
- [5] E. Lavretsky and K. Wise, *Robust and Adaptive Control*. London: springer, 2013.
- [6] A. Dharmawan and I. F. Arismawan, "Sistem Kendali Penerbangan Quadrotor pada Keadaan Melayang dengan Metode LQR dan Kalman Filter," *IJEIS (Indonesian J. Electron. Instrum. Syst.*, vol. 7, no. 1, p. 49, 2017.
- [7] C. Sun, T. Lu, and K. Yuan, "Balance control of two-wheeled self-balancing robot based on Linear Quadratic Regulator and Neural Network," *2013 Fourth Int. Conf. Intell. Control Inf. Process.*, vol. 1, pp. 862–867, 2013.
- [8] P. Gautam, "Optimal control of Inverted Pendulum system using ADALINE artificial neural network with LQR," *2016 Int. Conf. Recent Adv. Innov. Eng.*, pp. 1–6, 2016.
- [9] S. Kusumadewi and S. Hartati, *Neuro-Fuzzy: Integrasi Sistem Fuzzy & Jaringan Syaraf*, 2nd ed. Yogyakarta: Graha Ilmu, 2010.
- [10] K. Ogata, *Modern Control Engineering*, 5th ed., vol. 17. pearson, 2010.
- [11] A. B. Zakaria and A. Dharmawan, "Sistem Kendali Penghindar Rintangan Pada Quadrotor Menggunakan Konsep Linear Quadratic," *Indones. J. Electron. Instrum. Syst.*, vol. 7, no. 2, pp. 219–230, 2017.