

## SISTEM KONTROL *SWARM* UNTUK *FLOCKING* WAHANA NIR-AWAK QUADROTOR DENGAN OPTIMASI ALGORITMA GENETIK

Endra Joelianto<sup>\*1</sup>, Winarendra Satya Rajasa<sup>2</sup>, Agus Samsi<sup>3</sup>

<sup>1,2,3</sup>Institut Teknologi Bandung, Bandung

Email: <sup>1</sup>ejoel@tf.itb.ac.id, <sup>2</sup>winarendra.satya@gmail.com, <sup>3</sup>samsi@tf.itb.ac.id

\*Penulis Korespondensi

(Naskah masuk: 16 April 2020, diterima untuk diterbitkan: 15 November 2021)

### Abstrak

Quadrotor merupakan wahana udara nir-awak jenis lepas landas atau pendaratan vertikal berbentuk silang dan memiliki sebuah rotor pada setiap ujung lengannya dengan kemampuan manuver yang tinggi. *Swarm* quadrotor yang terdiri dari sekumpulan quadrotor akan menjadi suatu *swarm* yang baik, sesuai dengan kriteria *swarm* oleh Reynold yaitu dapat menghindari tumbukan, menyamakan kecepatan, dan pemusatan *swarm*. Pengontrolan *swarm* quadrotor memiliki tingkat kerumitan yang tinggi karena melibatkan banyak agen. Riset pengembangan *swarm* quadrotor masih belum banyak dilakukan dan masih membuka peluang untuk meneliti dengan metoda lain yang lebih baik dalam menghasilkan *swarm*. Makalah ini mengusulkan pengontrolan *swarm* quadrotor yang terdiri dari dua tingkat lup kontrol. Lup pertama adalah pengontrol sistem model *swarm* untuk membangkitkan lintasan *swarm* dan lup kedua merupakan pengontrol pada quadrotor untuk melakukan penjejakan lintasan *swarm*. Pengontrol pertama menggunakan pengontrol proporsional derivatif (PD), sedangkan pengontrol kedua menggunakan regulator linier kuadrat (RLK). Pengontrol yang dirancang memiliki parameter yang banyak, sehingga pemilihan parameter yang optimal sangat sulit. Pencarian parameter optimal pada pengontrol model *swarm* quadrotor membutuhkan teknik optimasi seperti algoritma genetik (AG) untuk mengarahkan pencarian menuju solusi yang menghasilkan kinerja terbaik. Pada makalah ini, penalaan dengan optimasi AG hanya dilakukan pada pengontrol PD untuk menghasilkan lintasan *swarm* terbaik, sedangkan matrik bobot RLK dilakukan secara uji coba. Hasil simulasi *swarm* pada model quadrotor menunjukkan parameter  $k_a$ ,  $k_b$ ,  $k_v$ , dan  $k_f$  yang diperoleh menggunakan AG menghasilkan pergerakan *swarm* yang baik dengan kesalahan RMS pelacakan 0,0094 m terhadap fungsi obyektif. Sedangkan ketika parameter  $a$ ,  $b$  dan  $c$  dicari menggunakan AG, tidak berpengaruh banyak dalam memperbaiki hasil simulasi *swarm* quadrotor.

**Kata kunci:** quadrotor, *swarm* cerdas, tarikan / tolakan, algoritma genetik, penjejakan lintasan

## *SWARM CONTROL SYSTEM FOR UNMANNED QUADROTOR FLOCKING WITH GENETIC ALGORITHM OPTIMIZATION*

### Abstract

The quadrotor is a type of take-off or vertical landing unmanned aerial vehicles with a cross shape and has one rotor at each end of its arm with high maneuverability. A quadrotor swarm consisting of a group of quadrotors leads to a good swarm, according to Reynold's swarm criteria, which accomplishes collision avoidance, velocity matching, and flock centering. Quadrotor swarm control has a high level of complexity because it involves many agents. Research on the development of quadrotor swarm has received insignificant attention and it still opens opportunities to research other methods that are better at producing swarm. The paper proposes the control of a quadrotor swarm consisted of two levels of control loops. The first loop controls the swarm model system to generate the swarm trajectory and the second loop is the controller on the quadrotor to track the swarm path. The first controller uses a proportional derivative controller (PD), while the second controller uses the linear quadratic regulator (LQR). The controller that is designed has many parameters, so the optimal parameter selection is very difficult. The search for optimal parameters in the swarm model controller requires optimization techniques such as the genetic algorithm (GA) to direct the search for solutions that produce the best performance. In this paper, tuning with the optimization of GA is only done for the PD controller in order to produce the best swarm trajectory, while the weight matrices of the LQR are done on a trial error basis. Swarm simulation results of a quadrotor model system show the parameters  $k_a$ ,  $k_b$ ,  $k_v$ , and  $k_f$  obtained using GA produce a good swarm movement with RMS error 0.0094 m of the objective function. Whereas when parameters  $a$ ,  $b$  and  $c$  are searched using GA, it does not have much effect in improving the quadrotor swarm simulation results.

**Keywords:** quadrotor, swarm intelligence, attraction / repulsion, genetic algorithm, track tracking

## 1. PENDAHULUAN

*Swarming* merupakan agregasi individu dengan ukuran yang relatif sama untuk membentuk gerombolan atau bergerak secara massal (Blum & Merkle, 2008). Perilaku ini dilakukan oleh makhluk hidup di alam untuk merespon lingkungan sekitarnya dengan lebih cepat, contohnya untuk mencari bahan makanan atau nutrisi pada suatu daerah. Perilaku *swarming* ini muncul karena keuntungan yang didapatkan dari kebersamaan, yaitu dapat melakukan pergerakan secara terkoordinasi (Reynolds, 1987). Model *swarm* menginspirasi para peneliti untuk menerapkan model *swarm* untuk aplikasi rekayasa, sehingga berbagai keunggulan yang dihasilkan model *swarm* dapat diterapkan pada agen artifisial seperti wahana nir-awak. Wahana nir-awak dapat bergerak membentuk formasi sesuai yang diinginkan untuk melakukan suatu tugas bersama-sama (Gazi & Passino, 2003)

Manuver suatu *swarm* secara koheren seakan-akan *swarm* tersebut merupakan satu organisme, sering diamati pada kawanan hewan seperti pada burung atau ikan. Berkelompok (*Flocking*) secara khusus mengacu pada burung (Blum & Merkle, 2008). Pergerakan hewan-hewan ini sebagai suatu kelompok yang terkoordinasi menginspirasi para peneliti untuk meniru pergerakan ini untuk aplikasi sistem artifisial (Turgut, Çelikkanat, Gökçe, & Şahin, 2008). *Flocking* pertama kali diterapkan pada *swarm* artifisial oleh Reynold (Reynolds, 1987). Reynold membuat simulasi *flocking* menggunakan sekumpulan individu yang dia namakan "*boi*d", yang diasumsikan dapat mendeteksi jarak, arah dan orientasi individu lain didekatnya. Reynold menyimpulkan *flocking* dapat dicapai dengan memanfaatkan tiga perilaku berikut; *collision avoidance* (menghindari tabrakan), *velocity matching* (menyamakan kecepatan) dan *flock centering* (pemusatan *swarm*).

Model *swarm* menginspirasi para peneliti untuk menerapkan pada aplikasi rekayasa, sehingga keuntungan dan keunggulan yang dihasilkan oleh model *swarm* dapat diimplementasikan pada agen artifisial seperti wahana nir-awak. Wahana nir-awak dapat bergerak membentuk formasi sesuai yang ditentukan, untuk melakukan suatu pekerjaan bersama-sama (Burohman, Widyotriatmo, & Joeliyanto, 2016) (Ferrante, Turgut, Huepe, Stranier, Pinciroli, & Dorigo, 2008) (Ferrante, Turgut, Huepe, Stranier, Pinciroli, & Dorigo, 2012) (Joeliyanto & Qurthobi, 2011) (Widyotriatmo, Joeliyanto, Prasdianto, Bahtiar, & Nazaruddin, 2017). Salah satu konsep wahana udara nir-awak (WUNA) yang sedang mendapat banyak perhatian adalah quadrotor.

Quadrotor merupakan kendaraan udara nir-awak jenis *vertical take-off or landing (VTOL)* dengan bentuk silang dan memiliki sebuah rotor pada setiap ujung lengannya. Pergerakan quadrotor dilakukan dengan melakukan variasi kecepatan keempat rotornya. Quadrotor memiliki kemampuan

manuver yang tinggi, sehingga pada penelitian ini diharapkan *swarm* yang terdiri dari sekumpulan quadrotor akan menjadi suatu *swarm* yang baik, sesuai dengan kriteria *swarm* yang dipaparkan oleh Reynold yaitu dapat menghindari tumbukan, menyamakan kecepatan, dan pemusatan *swarm* (Reynolds, 1987).

Koordinasi gerakan antara dua atau lebih quadrotor diperlukan untuk mengoptimalkan pekerjaan yang akan dilakukan oleh quadrotor. Oleh karena itu, studi koordinasi kawanan quadrotor adalah topik yang sangat menarik. Penelitian tentang kawanan quadrotor telah dilakukan oleh (Kushleyev, Mellinger, Powers, & Kumar, 2013) (Leonard, Al Savvaris, & Tsourdos, 2012) (Faelden, Vicerra, Lim, Sybingco, Dadios, & Bandala, 2017) (Halci, Gazi, & Cihan, 2019) (Trizuljak, Duchoň, Rodina, Babinec, Dekan, & Mykhailyshyn, 2019) (Hönig, Preiss, Kumar, Sukhatme, & Ayanian, 2018) (Choutri, Lagha, Dala, & Lipatov, 2018) (Lazim, Husain, Mohd Subha, Mohamed, & Mohd Basri, 2017). Pada (Choutri, Lagha, Dala, & Lipatov, 2018), kawanan quadrotor diteliti dengan menggunakan pendekatan pembentukan pemimpin-pengikut (*leader-follower*). Pada (Lazim, Husain, Mohd Subha, Mohamed, & Mohd Basri, 2017), pendekatan optimasi kawanan partikel (*Particle Swarm Optimization (PSO)*) diterapkan untuk membuat kontrol formasi yang optimal atas sekelompok quadrotor. Penelitian-penelitian tersebut di atas belum mempertimbangkan pembangkitan trajektori *swarm* yang optimal oleh pengontrol dengan algoritma genetik (AG). Sehingga kawanan quadrotor akan lebih cepat konvergen dan memiliki kesalahan pelacakan yang minimum.

Pemodelan dan simulasi *swarm* akan melibatkan banyak agen dan memiliki tingkat kerumitan yang tinggi. Permasalahan utama adalah mencari parameter optimal yang berjumlah banyak pada pengontrol model *swarm*. Oleh karena itu, teknik optimasi seperti algoritma genetik (AG) akan membantu dalam mencapai kinerja obyektif yang diinginkan. AG adalah algoritma pencarian adaptif heuristik yang terinspirasi oleh konsep seleksi alam dan genetik. AG merepresentasikan pencarian acak secara cerdas untuk menyelesaikan permasalahan optimasi, dan memanfaatkan informasi historis untuk mengarahkan pencarian menuju solusi yang menghasilkan kinerja terbaik (Chakraborty, 2010) (Kramer, 2017) (Mirjalili, 2019). Penerapan AG pada quadrotor masih terbatas pada pengontrolan gerak quadrotor dengan pengontrol proporsional integral derivatif (PID) (Hakiim, Dharmawan, & Faizah, 2017).

Pada makalah ini, pengontrol yang digunakan terdiri dari dua tingkat lup kontrol (Joeliyanto & Sagala, 2012). Lup kontrol pertama digunakan untuk menghasilkan lintasan *swarm* dan lup kontrol kedua menggunakan regulator linier kuadratik (RLK) yang biasa dikenal sebagai linear quadratic regulator (LQR). RLK sampai sekarang masih diterapkan

sebagai pengontrol pada berbagai aplikasi (Marada, Matousek, & Zuth, 2017) (Possieri, Sassano, Galeani, & Teel, 2020) (Mohammed & Abdulla, 2018) (Das, Gurralla, & Shenoy, 2016) (Ge, Wang, & Lv, 2018) (Cao, Tang, & Zhang, 2017). Penggunaan RLK pada quadrotor hanya digunakan untuk mengontrol gerakan manuver, belum pada gerakan kawanan (İçen, Ateş, & Yeroğlu, 2017) (Okoyere, Bousbaine, Poyi, Joseph, & Andrade, 2019).

Fokus bahasan makalah ini adalah perancangan pengontrol lup pertama dengan pengontrol proporsional derivatif (PD) yang diharapkan dapat menghasilkan lintasan *swarm* yang optimal dengan agen berupa model quadrotor. Lintasan yang optimal akan menghasilkan karakteristik *swarming* yang ideal melalui parameter pengontrol PD yang dihasilkan oleh AG. Sistem kontrol lintasan *swarm* yang dihasilkan akan mampu membuat agen-agen melacak dan menuju lokasi target yang diharapkan sambil menjaga *flocking* dengan menggunakan RLK. Selain itu, pergerakan *flocking* mampu meminimumkan fungsi obyektif kesalahan pelacakan.

## 2. METODE PENELITIAN

### 2.1. Perancangan Sistem Kontrol Lintasan Swarm

Pada makalah ini, sistem kontrol *swarm* yang dirancang sistem kontrol yang terdiri dari dua tingkat lup kontrol. Lup pertama adalah kontrol model *swarm*, yang akan digunakan sebagai pembangkit referensi penjejakan lintasan. Lup kontrol kedua merupakan kontrol pada WUNA quadrotor agar setiap agen dapat melakukan penjejakan lintasan yang dihasilkan oleh model *swarm*. Sistem kontrol model *swarm* menggunakan pengontrol proporsional-derivatif (PD), dengan memperhitungkan kesalahan posisi, kecepatan, dan fungsi potensial buatan. Pada lup kontrol quadrotor, digunakan regulator linier kuadratik (RLK). Perancangan pengontrol ini mengacu pada penelitian yang dilakukan (Joelianto & Sagala, 2012) yang berdasarkan formulasi yang dilakukan (Passino, 2005).

Tujuan utama pengontrol model *swarm* ini adalah mengontrol pergerakan dinamik seluruh anggota *swarm* untuk bergerak berkelompok (*flocking*) menuju suatu target yang diinginkan. Model *swarm* yang digunakan adalah suatu model yang memiliki ciri bahwa tidak ada pemimpin di antara anggota *swarm* mengacu persamaan (1). Keadaan agen ( $i$ ) didefinisikan dengan  $x^i (i = 0, 1, 2 \dots M)$  dan diasumsikan bahwa setiap agen bergerak dalam suatu ruang *Euclidean* dimensi- $n$ , dengan persamaan gerak masing-masing agen dimodelkan dengan sebagai titik massa yang bergerak sesuai dengan hukum Newton  $m_1 a_1 = F_1$ . Persamaan gerak ini menghasilkan gerak individu sebagai berikut:

$$\begin{aligned} \dot{x}^i &= v^i \\ m_i \dot{v}^i &= u^i \end{aligned} \tag{5}$$

dengan  $x^i \in R^n$  adalah posisi dengan  $x^i \triangleq \{x_x^i, x_y^i, x_z^i\}$ ,  $v^i \in R^n$  adalah kecepatan dengan  $v^i \triangleq \{v_x^i, v_y^i, v_z^i\}$ ,  $m_i$  adalah massa, dan  $u^i \in R^n$  adalah masukan kontrol (Gazi & Passino, 2004a).

Pergerakan agen dalam *swarm* akan dipengaruhi oleh (i) jarak antar-agen, (ii) perbedaan kecepatan antar-agen, dan (iii) gaya penarik dan penolak pada profil dimana *swarm* beroperasi (Gazi & Passino, 2004a). Persamaan kontrol yang digunakan merupakan modifikasi dari persamaan kontrol (Passino, 2005) yang merupakan penurunan dari persamaan awal (1) dengan menambahkan gaya tolakan antar-agen. Kontrol *swarm* yang dimodifikasi adalah (Joelianto & Sagala 2012)

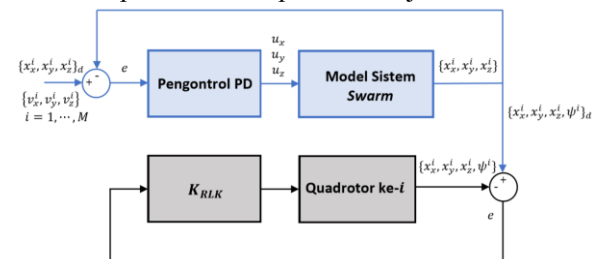
$$\begin{aligned} u^i &= -M_i k_a e_p^i - M_i k_b e_v^i - M_i k_v v^i \\ &+ M_i \sum_{i=1, i \neq j}^N (a - b) \exp\left(\frac{\|e_p^i - e_p^j\|}{c}\right) (e_p^i - e_p^j) - M_i k_f (\nabla J_p(x^i)) \end{aligned} \tag{6}$$

Pada persamaan (6) diasumsikan bahwa agen dilengkapi dengan sensor dan program sehingga massa di persamaan (5) berubah menjadi  $M_i$  berkecepatan  $v^i$ . Selanjutnya, diasumsikan nilai  $M_i = 1$  untuk menyederhanakan sistem kontrol. Persamaan yang dihasilkan (Joelianto & Sagala, 2012):

$$\begin{aligned} u^i &= -k_a e_p^i - k_b e_v^i - k_v v^i + \sum_{j=1, j \neq i}^M (a - b) \exp\left(\frac{-\|e_p^i - e_p^j\|}{c}\right) (e_p^i - e_p^j) - k_f (\nabla J_p(x^i)) \end{aligned} \tag{7}$$

Pengontrol pada persamaan (7) memiliki penguat proporsional, yaitu  $k_a$  (penguatan untuk kestabilan posisi anggota) dan  $k_b$  (penguatan konvergensi kecepatan). Sedangkan penguatan  $k_v$  dan  $k_f$  adalah koefisien redaman kecepatan dan penguat derivatif untuk mengikuti profil pergerakan agen yang diinginkan. Gambar 1 memperlihatkan diagram blok sistem kontrol *swarm* dua tingkat yang telah diusulkan di (Joelianto & Sagala, 2012). Pengontrol PD akan menghasilkan pusat *swarm*  $\{x^i, y^i, z^i, \psi^i\}_d$  sebagai lintasan *swarm* yang harus dilacak oleh setiap agen.

Untuk memudahkan penjejakan dalam orientasi *NEA* (*North East Altitude*) (Joelianto & Sagala, 2012), model matriks karakteristik  $A \in R^{9 \times 9}$  diperluas menjadi matriks  $A \in R^{12 \times 12}$ , begitu juga dengan matriks input  $B \in R^{9 \times 4}$  diperluas menjadi  $B \in R^{12 \times 4}$ .



Gambar 1. Diagram blok sistem kontrol *swarm* dua tingkat

**2.2. Perancangan Pengontrol Agen**

Pengontrol pada agen (gambar 3) dilakukan oleh RLK diskrit. Pergerakan setiap agen adalah melacak lintasan (pusat *swarm*)  $\{x_x^i, x_y^i, x_z^i, \psi^i\}_d$  berdasarkan informasi agen  $\{x_x^i, x_y^i, x_z^i, \psi^i\}$ , dengan  $\psi^i$  adalah sudut geleng (*yaw*). Sehingga kesalahan  $e$  pada lup kedua menuju nol. Penguatan linier kuadrat  $K_{RLK}$  diperoleh dengan mengubah-ubah matrik bobot  $Q = Q^T \geq 0$  dan  $R = R^T > 0$  pada fungsi obyektif.

$$J_\infty = \sum_{n=1}^\infty [x^T(k) Q x(k) + u^T(k) R u(k)] \quad (8)$$

dengan persamaan dinamik setiap quadrotor adalah diskritisasi persamaan (4), diberikan oleh:

$$x(k + 1) = Ax(k) + Bu(k) \quad (9)$$

Sinyal kontrol optimal kuadratik diberikan oleh

$$u(k) = -K_{RLK} x(k) \quad (10)$$

dengan  $K_{RLK}$  adalah penguatan umpan balik penuh memenuhi persamaan

$$K_{RLK} = (R + B^T P_{ss} B)^{-1} B^T P_{ss} A \quad (11)$$

Matrik  $P_{ss}$  adalah solusi keadaan tunak (*steady state*) dari persamaan aljabar Riccati diskrit berikut:

$$P_{ss} = Q + A^T P_{ss} A - A^T P_{ss} B (R + B^T P_{ss} B)^{-1} B^T P_{ss} A \quad (12)$$

Pada persoalan penjejakan lintasan, nilai keluaran  $y$  dibandingkan dengan nilai referensi  $y_{ref}$ , dengan tujuan diperoleh nilai kesalahan antara  $y$  dan  $y_{ref}$  menuju nol. Nilai kesalahan dapat dikecilkan dengan menambahkan, secara umum, integrator pada sinyal kesalahan (Joelianto, 2017). Cara lain adalah dengan menggunakan turunan kesalahan. Jika diasumsikan pengukuran tanpa gangguan, maka kesalahan sistem dalam variable keadaan ( $e$ ) dapat dirumuskan sebagai:

$$\dot{e}(t) = \dot{x}_{ref}(t) - \dot{x}(t) \quad (13)$$

Jika referensi bernilai tetap, maka  $\dot{x}_{ref}(t) = 0$ , sehingga  $\dot{e}(t) = -\dot{x}(t)$ . Maka dinamika kesalahan penjejakan lintasan dapat dirumuskan dengan

$$\dot{e}(t) = -\eta \dot{x}(t) \quad (14)$$

dimana  $\eta$  adalah konstanta bobot kinerja penjejakan. Jika dirumuskan dalam bentuk matriks, persamaan diatas menjadi

$$\dot{e}(t) \triangleq \begin{bmatrix} \dot{x}_{error}(t) \\ \dot{y}_{error}(t) \\ \dot{z}_{error}(t) \end{bmatrix} = -\eta \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \end{bmatrix} \quad (15)$$

Pada model quadrotor nilai  $\{\dot{x}(t), \dot{y}(t), \dot{z}(t)\}$  dapat disubstitusikan dengan  $\{u(t), v(t), w(t)\}$  sehingga persamaan diatas berubah menjadi

$$\dot{e}(t) = \begin{bmatrix} \dot{x}_{error}(t) \\ \dot{y}_{error}(t) \\ \dot{z}_{error}(t) \end{bmatrix} = -\eta \begin{bmatrix} u(t) \\ v(t) \\ w(t) \end{bmatrix} \quad (16)$$

Sinyal kontrol umpan balik keadaan penuh  $u(k)$  pada persamaan (10) dapat diterapkan untuk permasalahan pelacakan lintasan dengan memberikan referensi yang selalu berubah, dan pelacak secara sempurna dengan  $e \rightarrow 0$  jika memenuhi kondisi *internal model principle (IMP)* (Joelianto, 2017).

Matrik bobot RLK yang diperlukan selanjutnya dicari secara uji coba, dengan mengubah-ubah nilai matrik  $Q$  dan  $R$  dengan membandingkan *root mean square (RMS)* kesalahan pelacakan terhadap suatu pola lintasan yang diberikan. Hasil matrik bobot secara uji coba yang baik adalah:

$$Q = \begin{bmatrix} 1000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 100 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 100 \end{bmatrix}$$

dan

$$R = \begin{bmatrix} 0.001 & 0 & 0 & 0 \\ 0 & 0.001 & 0 & 0 \\ 0 & 0 & 0.001 & 0 \\ 0 & 0 & 0 & 0.001 \end{bmatrix}$$

Matrik bobot  $Q$  dan  $R$  di atas menghasilkan penjejakan lintasan dengan hasil kesalahan RMS sebesar:  $x = 0,7862$  m,  $y = 0,4462$  m,  $z = 0,3203$  m.

**3. LITERATURE REVIEW**

**3.1. Kecerdasan Kawanan (Swarm Intelligence)**

Misal terdapat sebuah *swarm* dengan jumlah anggota  $M$  individu dalam sebuah ruang *Euclidean* berdimensi- $n$ . Masing-masing individu dimodelkan sebagai titik dengan mengabaikan dimensinya. Posisi dari individu- $i$  disimbolkan sebagai  $x^i \in R^n$ . Asumsi pergerakan setiap individu sinkron (bergerak secara bersamaan) dan tidak ada waktu jeda. Seluruh anggota *swarm* bergerak secara simultan dan setiap individu mengetahui posisi relatif dari individu lainnya. Persamaan gerak dari setiap individu- $i$  (Gazi & Passino, 2004a) diberikan oleh persamaan (1) berikut:

$$x^i = -\nabla_{x^i} \sigma(x^i) + \sum_{j=1, j \neq i}^M g(x^i - x^j), i = 1, \dots, M \tag{1}$$

Pada suku pertama, dimisalkan  $\sigma(\cdot): R^n \rightarrow R^n$  merepresentasikan profil dari penarik ( $\sigma(\cdot) < 0$ ), penolak ( $\sigma(\cdot) > 0$ ), atau netral ( $\sigma(\cdot) = 0$ ), dikenal sebagai  $\sigma$ -profil, yaitu profil dari adanya suatu sumber makanan atau bahan beracun. Pada suku kedua,  $g(\cdot)$  merupakan fungsi tarikan dan tolakan antar-individu dalam *swarm*. Fungsi tolakan dapat dimodelkan dengan tolakan yang terjadi jika agen berada pada jarak yang terlalu dekat (Gazi & Passino, 2004a). Fungsi ini direpresentasikan dalam persamaan berikut:

$$u' = k_r \exp\left(\frac{-\frac{1}{2}\|x^i - x^j\|^2}{r_s^2}\right) \tag{2}$$

Notasi  $k_r > 0$  adalah besar penguat dari tolakan, dan  $r_s > 0$  adalah ukuran area di sekitar agen di mana fungsi tolakan terjadi. Jika selisih jarak antar agen terlalu jauh, yaitu ketika  $\|x^i - x^j\| = \sqrt{(x^i - x^j)^T(x^i - x^j)}$  bernilai sangat besar relatif terhadap  $r_s$ , maka fungsi tersebut akan bernilai nol (Gazi & Passino, 2004b).

Elemen-elemen dalam lingkungan akan berpengaruh terhadap perilaku pergerakan suatu *swarm*. Dalam makalah ini, profil lingkungan yang dipilih adalah profil Gaussian, yang didefinisikan dengan persamaan:

$$\sigma(y) = -\frac{A_\sigma}{2} \exp\left(\frac{\|y - c_\sigma\|^2}{I_\sigma}\right) + b_\sigma. \tag{3}$$

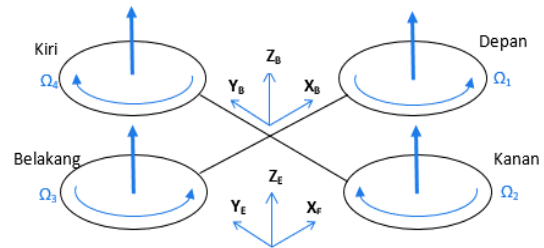
dengan  $A_\sigma \in R$ ,  $b_\sigma \in R$ ,  $I_\sigma \in R^+$ , and  $c_\sigma \in R^n$  yang memiliki suatu global maksimum atau minimum pada  $y = c_\sigma$  tergantung pada tanda  $A_\sigma$ . Fungsi Gaussian dengan mudah dibuat menjadi fungsi Gaussian jamak (Gazi & Passino, 2004a).

### 3.2. Wahana Udara Nir-awak Quadrotor

Pada wahana udara nir-awak (WUNA) quadrotor, daya angkat dihasilkan oleh keempat rotor yang terpasang pada setiap lengan. Tiap rotor yang berhadapan membentuk satu pasang. Pasangan pertama terletak pada sumbu-x dan berotasi searah jarum jam sedangkan pasangan kedua terletak pada sumbu-y dan berotasi berlawanan arah jarum jam. Setiap rotor berada pada sudut *pitch* yang tetap dengan aliran udara mengarah ke bawah untuk menghasilkan daya angkat. Struktur quadrotor cukup kokoh dan bagian yang dapat divariasikan hanya kecepatan perputaran rotor (De Lellis Costa De Oliveira, 2011).

Kerangka acuan bumi ( $X_E, Y_E, Z_E$ ) digunakan untuk menentukan orientasi quadrotor, bukan

kerangka badan quadrotor ( $X_B, Y_B, Z_B$ ). Pada gambar 1, lambang  $\Omega_i, i = 1,2,3,4$  adalah kecepatan putar setiap rotor. Secara umum, pergerakan quadrotor dikontrol dengan melakukan variasi daya dan torsi pada pusat rotasinya (Sudiyanto, Muljowidodo, & Budiyo, 2009).



Gambar 2. Pemodelan dasar dinamika quadrotor

Sistem quadrotor dapat dimodelkan dengan persamaan tidak-linier yang kemudian dilinierisasi kedalam bentuk persamaan ruang keadaan berikut

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{4}$$

dengan  $x \triangleq \{du \ dv \ dw \ dp \ dq \ dr \ d\phi \ d\theta \ d\psi\} \in R^{9 \times 1}$  adalah variable ruang keadaan dan  $u \triangleq \{d\Omega_{R1} \ d\Omega_{R2} \ d\Omega_{R3} \ d\Omega_{R4}\} \in R^{4 \times 1}$  adalah masukan quadrotor. Model quadrotor yang digunakan adalah model linier saat keadaan terbang pada kecepatan 5m/s (Sudiyanto, Muljowidodo, & Budiyo, 2009).

### 3.3. Optimasi Menggunakan Algoritma Genetik

Algoritma genetik (AG) merupakan algoritma yang terinspirasi dari konsep seleksi alam. Di alam, persaingan antara individu-individu menyebabkan individu yang lebih kuat akan dominan atas individu yang lemah. Individu disini dianalogikan dengan solusi, sehingga solusi yang memiliki tingkat kebugaran atau *fitness* yang lebih tinggi akan lebih mudah untuk terseleksi (Chakraborty, 2010) (Kramer, 2017) (Mirjalili, 2019).

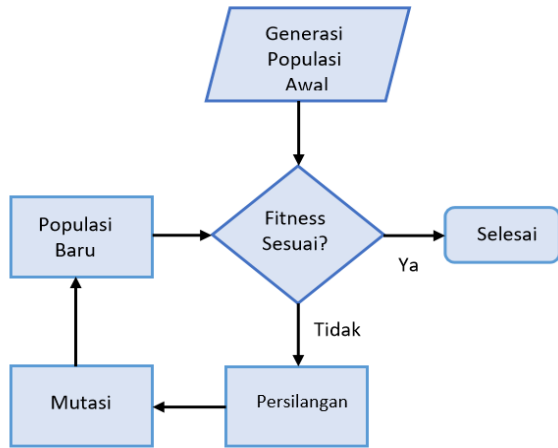
Dalam bidang teknik, algoritma genetik diaplikasikan untuk menyelesaikan permasalahan optimasi. Permasalahan awalnya dimodelkan sebagai model matematika berupa fungsi, kemudian untuk mencari solusi, dicari parameter yang meminimalkan atau memaksimalkan fungsi tersebut sehingga menghasilkan kinerja yang diinginkan (Calvez & Hutzler, 2005).

Permasalahan optimasi mencakup tiga faktor (Chakraborty, 2010), yaitu:

- Fungsi Obyektif: Fungsi yang ingin di minimalkan / maksimalkan
- Variabel tidak diketahui: nilai-nilai yang mempengaruhi fungsi obyektif
- Batasan: batasan-batasan pada nilai variabel

Dari tiga faktor di atas, dapat disimpulkan bahwa proses optimasi adalah mencari harga variabel yang meminimumkan / memaksimalkan fungsi obyektif

tanpa keluar dari batasan yang telah ditentukan. Proses AG secara singkat terdiri dari tiga proses utama, yaitu seleksi alam, persilangan, dan mutasi. Individu dengan kebugaran (*fitness*) yang tinggi akan terseleksi, untuk kemudian disilangkan sehingga terbentuk keturunan baru. Keturunan baru ini selanjutnya diberi mutasi, yang dapat mengembalikan gen unggul yang hilang saat terjadi persilangan. Proses ini terus diulang sampai ditemukan individu dengan *fitness* yang terbaik seperti diperlihatkan secara singkat pada gambar 3.



Gambar 3. Diagram alir algoritma genetik (AG)

Algoritma genetik (AG) mensimulasikan proses seleksi alam. Berikut adalah proses AG (Chakraborty, 2010):

1. Mulai: Dibuat sebuah populasi kromosom acak berjumlah  $p$
2. *Fitness* (Kebugaran): evaluasi *fitness*  $f(w)$  setiap kromosom  $w$  dalam populasi
3. Populasi baru: Buat populasi baru dengan mengulang langkah-langkah berikut sampai populasi baru terbentuk.
4. Seleksi Alam: Pilih dua kromosom orang tua menurut *fitness* (semakin besar nilai *fitness* semakin besar kemungkinan individu terseleksi) untuk kemudian dilakukan persilangan.
5. Persilangan: Dengan adanya probabilitas penyilangan, silangkan kedua orang tua yang didapatkan dari seleksi untuk menghasilkan keturunan.
6. Mutasi: Dengan probabilitas mutasi, lakukan mutasi pada setiap lokus (posisi dalam kromosom)
7. Penerimaan: Masukkan individu baru ke dalam populasi
8. Ganti: lakukan algoritma pada populasi baru yang terbentuk.

9. Uji: Jika hasilnya memuaskan hentikan komputasi, dan ambil solusi dengan *fitness* terbaik dalam populasi terakhir.

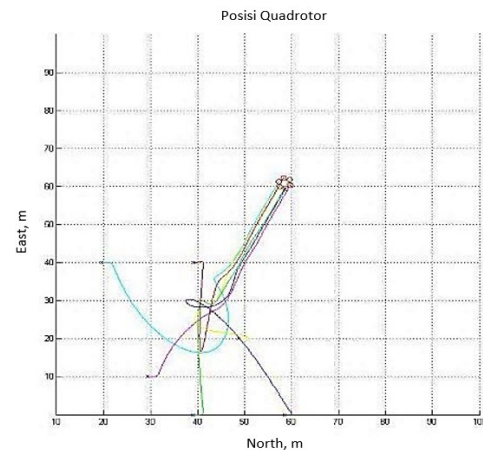
Persilangan, dan mutasi merupakan operator seleksi alam dengan pengaruh terbesar terhadap algoritma genetik.

#### 4. HASIL DAN PEMBAHASAN

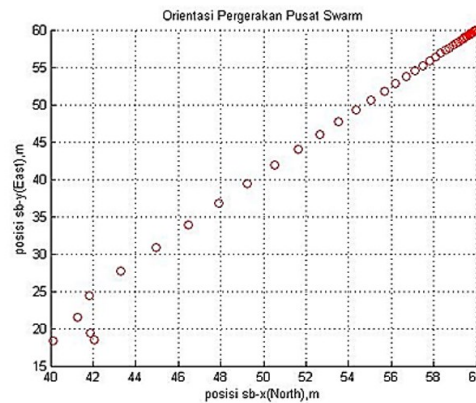
##### 4.1. Simulasi dan Optimasi Parameter Proporsional dan Derivatif dengan AG

Ketika terdapat gaya penarik, pergerakan anggota *swarm* akan berperilaku berdasarkan pengaruh penarik dan penolak tersebut yang mana quadrotor membentuk *swarm* (gambar 4). Keenam quadrotor bergerak bersama-sama menuju koordinat tujuan sambil menghindari tumbukan antar anggota. Pada gambar 5 digambarkan pusat *swarm*, yang pergerakannya juga terpengaruh adanya gaya tolakan dari rintangan dan tarikan dari tujuan. Pusat *swarm* dan kecepatan *swarm* mengikuti definisi (Passino, 2005) sebagai berikut

$$\bar{x} = \frac{1}{M} \sum_{i=1}^M x^i \text{ dan } \bar{v} = \frac{1}{M} \sum_{i=1}^M v^i \quad (17)$$



Gambar 4. Simulasi *swarm* dengan gaya penarik



Gambar 5. Orientasi pergerakan pusat *swarm*

Algoritma genetik (AG) selanjutnya digunakan untuk mencari parameter-parameter pengontrol *swarm* pada lup pertama diberikan oleh persamaan (7). Parameter-

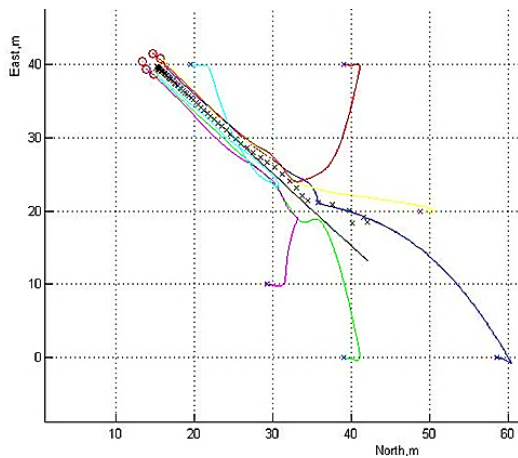
parameter yang digunakan untuk pembangkitan populasi awal AG adalah sebagai berikut: jumlah kromosom 8 dan jumlah gen setiap kromosom 10. Jumlah generasi yang dihasilkan adalah 30. Proses reproduksi dari kromosom-kromosom pilihan menggunakan probabilitas persilangan (*crossover*) 0,6. Mutasi pada suatu kromosom dalam suatu populasi memiliki probabilitas 0,05. Pengkodean kromosom yang berisi bilangan biner menjadi individu yang bernilai real dalam rentang interval yang ditentukan adalah [0,1;3].

Berdasarkan serangkaian pengujian terhadap parameter-parameter pada pengontrol model *swarm*, nilai parameter-parameter  $k_a$ ,  $k_b$ ,  $k_v$  dan  $k_f$  terbaik dirangkum pada pada tabel 1. Selanjutnya, parameter tersebut, dilakukan optimasi menggunakan AG. Karakteristik *swarm* yang diinginkan adalah cepat membentuk pusat *swarm*, dan dapat bergerak tanpa adanya tumbukan. Untuk itu, fungsi obyektif berupa persamaan garis diberikan yang harus dimaksimalkan oleh AG.

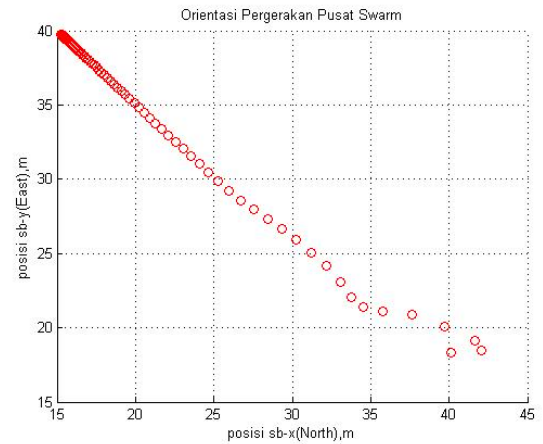
Pergerakan pusat *swarm* akan dibandingkan terhadap garis tersebut dan hasil terbaik adalah pergerakan pusat *swarm* yang paling mendekati persamaan garis yang dihasilkan. Dengan AG, optimasi menghasilkan parameter  $k_a$ ,  $k_b$ ,  $k_v$  dan  $k_f$  optimal seperti pada tabel 1. Simulasi dilakukan dengan posisi awal dari enam agen adalah  $X_1(40,40,0)$ ,  $X_2(40,0,0)$ ,  $X_3(60,0,0)$ ,  $X_4(20,40,0)$ ,  $X_5(30,10,0)$ ,  $X_6(50,20,0)$  dan nilai  $a = 1$ ,  $b = 10$ , dan  $c = 3$ . Parameter  $a$ ,  $b$  dan  $c$  tersebut didapatkan setelah melakukan serangkaian simulasi, sehingga dihasilkan pergerakan *swarm* terbaik.

Tabel 1. Nilai Parameter Hasil AG dan Uji Coba (Manual)

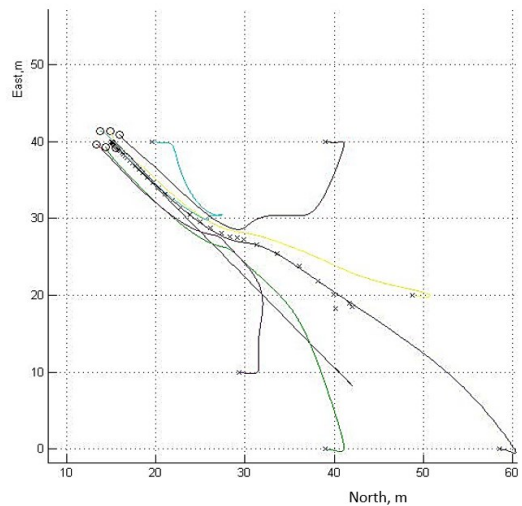
Hasil GA				Hasil Uji Coba			
$k_a$	$k_b$	$k_v$	$k_f$	$k_a$	$k_b$	$k_v$	$k_f$
0.7779	4.3194	2.4428	1.4214	1	10	1	1



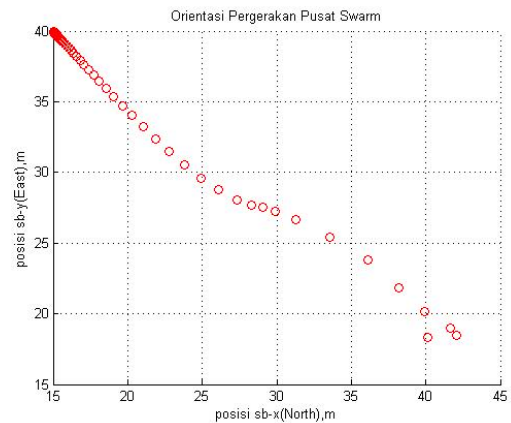
Gambar 6. Simulasi dengan parameter dari AG



Gambar 7. Pusat pergerakan *swarm* menggunakan parameter optimal AG



Gambar 8. Simulasi dengan parameter uji coba



Gambar 9. Pusat pergerakan *swarm* menggunakan parameter hasil uji coba manual

Pada gambar 6 yang dihasilkan, pusat *swarm* yang dilacak oleh agen-agen dilambangkan dengan tanda  $\times$ , garis hitam adalah persamaan garis fungsi obyektif, dan garis berwarna adalah pergerakan dari enam agen. Gambar 7 memperlihatkan lintasan pusat *swarm* saja dengan simbol lingkaran berwarna merah. Gambar lain selanjutnya mengikuti notasi yang sama.

Pada simulasi, pergerakan pusat *swarm* dibandingkan terhadap fungsi obyektif yang dibuat, yaitu garis hitam setelah *swarm* bergerak secara tunak.

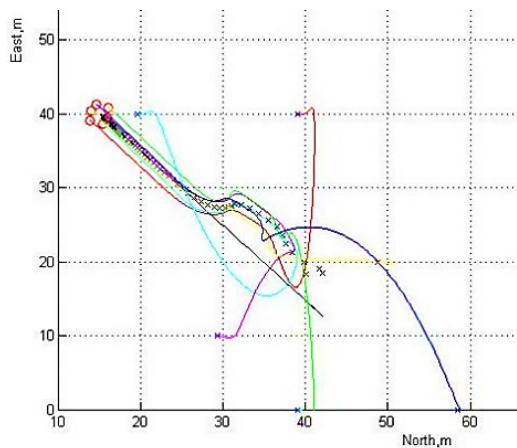
Simulasi *swarm* model quadrotor menggunakan nilai-nilai parameter pada tabel 1 menghasilkan *swarm* dengan galat RMS terhadap fungsi objektif sebesar 0,0094 m untuk parameter dioptimasi dengan AG (gambar 6 dan 7). Sedangkan tanpa optimasi (manual), *swarm* terbentuk dengan galat RMS terhadap fungsi objektif sebesar 0,0554 m (gambar 8 dan 9). Hasil optimasi parameter dengan AG jauh lebih baik karena menghasilkan RMS yang lebih kecil dan lintasan yang dihasilkan terlihat lebih stabil.

**4.2. Simulasi dan Optimasi Parameter Tarikan, Tolakan dan Jarak antar Agen dengan AG**

Selanjutnya dilakukan pengujian terhadap tiga parameter gaya tarikan dan tolakan, yaitu parameter *a* (tarikan), *b* (tolakan) dan *c* (jarak antar-agen). Pengujian sebelumnya terhadap ketiga parameter tersebut menghasilkan parameter-parameter terbaik dengan nilai  $a=1, b=10$  dan  $c=3$  yang telah digunakan pada simulasi-simulasi sebelumnya. Fungsi obyektif yang dimaksimumkan adalah fungsi saat posisi agen membentuk pusat *swarm* yang kohesif. Parameter yang menghasilkan *swarm* dalam waktu tercepat akan mendapatkan nilai *fitness* yang lebih tinggi. Optimasi dengan AG menghasilkan parameter optimal  $a=5,7067, b=26,5161$ , dan  $c=4,1760$  yang dirangkum pada tabel 2. Hasil simulasi diperlihatkan pada gambar 10 dan 11.

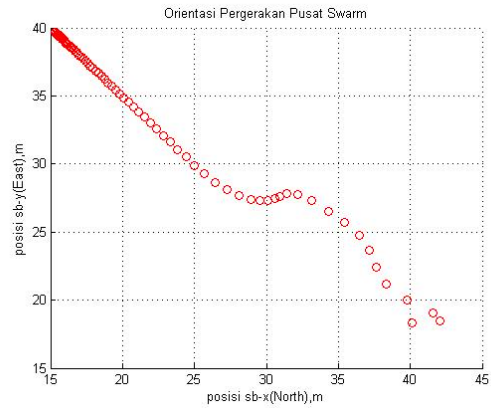
Tabel 2. Nilai Parameter *a, b, c* Hasil AG dan Uji Coba (Manual)

Hasil GA			Hasil Uji Coba		
<i>a</i>	<i>b</i>	<i>c</i>	<i>a</i>	<i>b</i>	<i>c</i>
5,7067	26,5161	4,1760	1	10	3



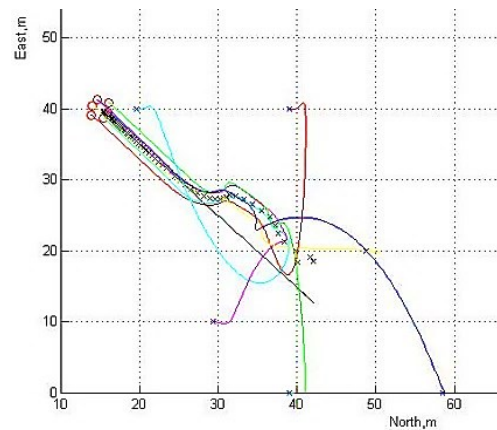
Gambar 10. Simulasi parameter *a, b* dan *c* dari AG

Hasil simulasi dengan parameter *a, b* dan *c* yang ditentukan dengan AG menghasilkan pergerakan pusat *swarm* dengan kesalahan RMS sebesar 0.0556 m, tidak jauh berbeda dengan hasil yang diperoleh dalam uji coba sebelumnya.

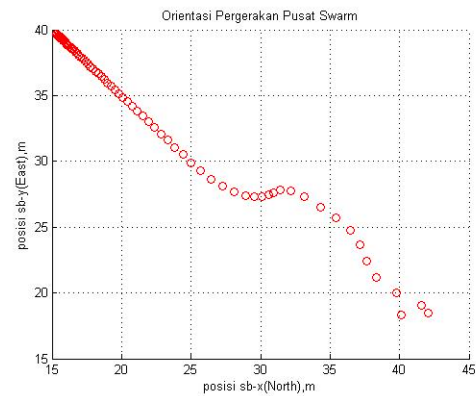


Gambar 11. Pusat pergerakan *swarm* dengan parameter *a, b* dan *c* dari AG

Selain itu, ketika seluruh parameter menggunakan parameter yang dihasilkan oleh AG (gambar 12 dan 13), didapatkan kesalahan RMS 0.0279 m, lebih buruk dari hasil dengan parameter  $k_a, k_b, k_v$  dan  $k_f$  saja yang dioptimasi. Hal ini menunjukkan parameter *a, b* dan *c* kurang memberikan hasil yang baik ketika dicari menggunakan AG.



Gambar 12. Simulasi dengan seluruh parameter *a, b, c, k<sub>a</sub>, k<sub>b</sub>, k<sub>v</sub>* dan *k<sub>f</sub>* dari AG



Gambar 13. Pusat pergerakan *swarm* dengan seluruh parameter *a, b, c, k<sub>a</sub>, k<sub>b</sub>, k<sub>v</sub>* dan *k<sub>f</sub>* dari AG



## 5. KESIMPULAN

Sistem kontrol *swarm* untuk menghasilkan *flocking* wahana nir-awak quadrotor dengan optimasi algoritma genetik telah dipaparkan dalam makalah ini. Permasalahan utama adalah kesulitan dalam pencarian nilai-nilai parameter terbaik dari tujuh parameter sistem kontrol. Penalaan parameter-parameter tersebut sulit dilakukan karena sifat sistem yang tidak linier dan rumit sehingga sulit mencari parameter terbaik. Optimasi parameter-parameter sistem kontrol *swarm* dengan menggunakan algoritma genetik menghasilkan parameter-parameter optimal yang sukar didapatkan secara uji coba manual dan memberikan respon *flocking* yang baik dengan nilai RMS yang minimum.

## DAFTAR PUSTAKA

- BLUM, C. & MERKLE, D., 2008. Swarm intelligence in optimization. In: Swarm intelligence, pp. 43-85, Berlin, Heidelberg: Springer.
- BUROHMAN, A.M., WIDYOTRIATMO, A. & JOELIANTO, E., 2016. Flocking for nonholonomic robots with obstacle avoidance. In 2016 International Electronics Symposium (IES), IEEE, pp. 345-350, September.
- CALVEZ, B. & HUTZLER, G., 2005. Automatic tuning of agent-based models using genetic algorithms. In International Workshop on Multi-Agent Systems and Agent-Based Simulation Springer, Berlin, Heidelberg, pp. 41-57, July.
- CAO, L., TANG, S. & ZHANG, D., 2017. Flight control for air-breathing hypersonic vehicles using linear quadratic regulator design based on stochastic robustness analysis. *Frontiers of Information Technology & Electronic Engineering*, 18(7), pp. 882-897.
- CHAKRABORTY, R.C., 2010. Fundamentals of genetic algorithms. *Reproduction*, 22, p.35.
- CHOUTRI, K., LAGHA, M., DALA, L. & LIPATOV, M., 2018. Quadrotors UAVs swarming control under Leader-Followers formation. In 2018 22nd International Conference on System Theory, Control and Computing (ICSTCC), IEEE, pp. 794-799.
- DAS, D., GURRALA, G. & SHENOY, U. J., 2016. Linear quadratic regulator-based bumpless transfer in microgrids. *IEEE Transactions on Smart Grid*, 9(1), pp. 416-425.
- DE LELLIS COSTA DE OLIVEIRA, M., 2011. M.S. Thesis: Modeling, Identification, and Control of a Quadrotor Helicopter, Czech Technical University in Prague, Prague, Ukraine.
- FAELDEN, G.E.U., VICERRA, R.R.P., LIM, L.A.G., SYBINGCO, E., DADIOS, E.P. & BANDALA, A.A., 2017. Implementation of Swarm Social Foraging Behavior in Unmanned Aerial Vehicle (UAV) Quadrotor Swarm. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 21(2), pp. 197-204.
- FERRANTE, E., TURGUT, A.E., HUEPE, C., STRANIER, A., PINCIROLI, C. & DORIGO, M., 2008. Self-Organized Flocking with a Mobile Robot Swarm. 7th ed. Padgham, Parkes, Müller and Parsons. Portugal, pp. 39-46.
- FERRANTE, E., TURGUT, A.E., HUEPE, C., STRANIERI, A., PINCIROLI, C. & DORIGO, M., 2012. Self-organized flocking with a mobile robot swarm: a novel motion control method. *Adaptive Behavior*, 20(6), pp. 460-477.
- GAZI, V. & PASSINO, K.M., 2003. Stability Analysis of Swarms. *IEEE Transactions on Automatic Control*, 48(4), pp. 692-697.
- GAZI, V. & PASSINO, K.M., 2004a. Stability Analysis of Social Foraging Swarms. *IEEE Transactions on System, Man and Cybernetics*, 34(1), pp. 539-557.
- GAZI, V. & PASSINO, K.M., 2004b. A Class of Attractions-Repulsion Functions for Stable Swarm Aggregations. *International Journal of Control*, 77(18), pp.1567-1579.
- GE, Z., WANG, Y. & LV, M., 2018. Three-dimensional Trajectory Tracking Guidance Law Based on Linear Quadratic Regulator. *Journal of Physics: Conference Series*. 1039(1), p. 012042. IOP Publishing.
- HAKIIM, K., DHARMAWAN, A. & FAIZAH, F., 2017. Optimasi Kendali PID menggunakan Algoritma Genetika untuk Penerbangan Quadrotor. *IJEIS (Indonesian Journal of Electronics and Instrumentation Systems)*, 7(2), pp. 173-184.
- HALCI, B., GAZI, V. & CIHAN, O., 2019. Modelling and Coordination of a Swarm of Quadrotors Using Lagrange Dynamics and Potential Functions. In 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, pp. 963-970.
- HÖNIG, W., PREISS, J.A., KUMAR, T.K. S., SUKHATME, G.S. & AYANIAN, N., 2018. Trajectory planning for quadrotor swarms. *IEEE Transactions on Robotics*, 34(4), pp. 856-869.
- İÇEN, M., ATEŞ, A. & YEROĞLU, C., 2017. Optimization of LQR weight matrix to control three degree of freedom quadcopter. In 2017 International Artificial Intelligence

- and Data Processing Symposium (IDAP), IEEE, pp. 1-6.
- JOELIANTO, E., 2017. Linear Quadratic Control: A State Space Approach. Bandung: ITB Press.
- JOELIANTO, E. & QURTHOBI, A., 2011. Optimal Control Design for A Formation Tracking with Leader-Follower Concept of Multi-Agent Autonomous Helicopter Model. In Proceedings of International Conference on Intelligent Unmanned Systems (Vol. 7).
- JOELIANTO, E. & SAGALA, A., 2012. Swarm tracking control for flocking of a multi-agent system. In 2012 IEEE Conference on Control, Systems & Industrial Informatics. IEEE, pp. 75-80, September.
- KRAMER, O., 2017. Genetic algorithm essentials (Vol. 679). Springer.
- KUSHLEYEV, A., MELLINGER, D., POWERS, C. & KUMAR, V., 2013. Towards a swarm of agile micro quadrotors. *Autonomous Robots*, 35(4), pp. 287-300.
- LAZIM, I.M., HUSAIN, A.R., MOHD SUBHA, N.A., MOHAMED, Z. & MOHD BASRI, M. A., 2017. Optimal formation control of multiple quadrotors based on particle swarm optimization. In Asian Simulation Conference, pp. 121-135. Springer, Singapore.
- LEONARD, J., SAVVARIS AL. & TSOURDOS, A., 2012. Towards a fully autonomous swarm of unmanned aerial vehicles. In Proceedings of 2012 UKACC International Conference on Control, IEEE. pp. 286-291.
- MARADA, T., MATOUSEK, R. & ZUTH, D., 2017. Design of linear quadratic regulator (LQR) based on genetic algorithm for inverted pendulum. *Mendel*, 23(1), pp. 149-156.
- MIRJALILI, S., 2019. Genetic algorithm. In *Evolutionary algorithms and neural networks* (pp. 43-55). Springer, Cham.
- MOHAMMED, I.K. & ABDULLA, A.I., 2018. Design of optimised linear quadratic regulator for capsule endoscopes based on artificial bee colony tuning algorithm. *International Journal for Engineering Modelling*, 31(1-2), pp. 77-98.
- OKYERE, E., BOUSBAINÉ, A., POYI, G.T., JOSEPH, A.K. & ANDRADE, J.M., 2019. LQR controller design for quad-rotor helicopters. *The Journal of Engineering*, 17, pp. 4003-4007.
- PASSINO, K.M., 2005. *Biomimicry for optimization, control, and automation*. London: Springer-Verlag.
- POSSIERI, C., SASSANO, M., GALEANI, S. & TEEL, A.R., 2020. The linear quadratic regulator for periodic hybrid systems. *Automatica*, 113, 108772.
- REYNOLDS, C.W., 1987, August. Flocks, herds and schools: A distributed behavioral model. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pp. 25-34, August.
- SUDIYANTO, T., MULJOWIDODO, M., & BUDIYONO, A., 2009. First principle approach to modeling of primitive quad rotor. *International Journal of Aeronautical and Space Sciences*, 10(2), pp.148-160.
- TRIZULJAK, A., DUCHOŇ, F., RODINA, J., BABINEC, A., DEKAN, M. & MYKHAILYSHYN, R., 2019. Control of a small quadrotor for swarm operation. *Journal of Electrical Engineering*, 70(1), pp. 3-15.
- TURGUT, A.E., ÇELIKKANAT, H., GÖKÇE, F., & ŞAHİN, E., 2008. Self-organized flocking in mobile robot swarms. *Swarm Intelligence*, 2(2-4), pp. 97-120.
- WIDYOTRIATMO, A., JOELIANTO, E., PRASDIANTO, A., BAHTIAR, H. & NAZARUDDIN, Y.Y., 2017. Implementation of Leader-Follower Formation Control of a Team of Nonholonomic Mobile Robots. *International Journal of Computers Communications & Control*, 12(6), pp.871-885.