

Optimasi Biaya dan Nutrisi Pakan Ayam Ras Menggunakan Algoritma Firefly dan *Reinforcement Learning*

David

Teknik Informatika

STMIK Pontianak

Pontianak, Indonesia

david@stmikpontianak.ac.id

Diajukan: 13 Januari 2025; Direvisi: 8 Juli 2025; Diterima: 9 Juli 2025

Abstrak

Penelitian ini mengembangkan metode optimasi komposisi pakan ayam ras menggunakan kombinasi Algoritma Firefly dan *Reinforcement Learning*. Masalah utama yang dihadapi peternak adalah tingginya biaya pakan dan kebutuhan pemenuhan nutrisi spesifik berdasarkan jenis ras, usia, dan berat ayam. Data dikumpulkan melalui studi literatur dan wawancara peternak terkait komposisi nutrisi dan harga pakan. Metode penelitian menggunakan perancangan model optimasi dengan parameter berupa bahan pakan (jagung, dedak padi, bungkil kedelai, dan lainnya), yang diujikan dalam simulasi Python dengan variasi jumlah fireflies dan iterasi. Hasil penelitian menunjukkan bahwa pendekatan ini menurunkan biaya pakan rata-rata sebesar 84% per kg dan mengurangi deviasi nutrisi rata-rata dari 3,5% menjadi 1,4% dibandingkan Algoritma Firefly biasa. Meskipun waktu proses sedikit lebih lama, metode Firefly dengan *Reinforcement Learning* menunjukkan konvergensi lebih cepat dan stabil. Temuan ini membuktikan integrasi Firefly dengan *Reinforcement Learning* efektif dalam optimasi komposisi pakan ayam ras, menawarkan solusi yang lebih baik bagi peternak untuk meningkatkan efisiensi dan produktivitas.

Kata kunci: Algoritma Firefly, *Reinforcement Learning*, Optimasi Komposisi Pakan, Pakan Ayam Ras.

Abstract

This study developed an optimization method for broiler chicken feed composition using a combination of the Firefly Algorithm and *Reinforcement Learning*. The main problems faced by farmers include the high cost of feed and the need to meet specific nutritional requirements based on breed type, age, and chicken weight. Data were collected through literature studies and interviews with farmers regarding nutritional composition and feed prices. The research method involved designing an optimization model with parameters consisting of feed ingredients (corn, rice bran, soybean meal, and others), tested through Python-based simulations with variations in the number of fireflies and iterations. The results showed that this approach reduced feed costs by an average of 84% per kg and decreased the average nutritional deviation from 3,5% to 1,4% compared to the standard Firefly Algorithm. Although the processing time was slightly longer, the Firefly Algorithm with *Reinforcement Learning* demonstrated faster and more stable convergence. These findings prove that integrating the Firefly Algorithm with *Reinforcement Learning* is effective for optimizing broiler chicken feed composition, offering a better solution for farmers to improve efficiency and productivity.

Keywords: Firefly Algorithm, *Reinforcement Learning*, Feed Composition Optimization, Broiler Chicken Feed.

1. Pendahuluan

Perkembangan peternakan ayam menunjukkan bahwa pakan merupakan komponen utama dalam memproduksi daging dan telur ayam. Saat ini pakan ayam merupakan salah satu masalah bagi para peternak ayam dikarenakan harga pakan yang cukup tinggi dan susah mendapatkan pakan ayam sehingga berpotensi terhentinya usaha peternakan ayam. Peternakan ayam ras di Kabupaten Mempawah memainkan peran penting dalam memenuhi kebutuhan protein hewani masyarakat. Salah satu tantangan utama yang dihadapi peternak adalah tingginya biaya pakan, yang dapat mencapai hingga 70% dari total biaya produksi [1]. Efisiensi pakan menjadi faktor krusial dalam menentukan profitabilitas usaha peternakan. Semakin efisien pemberian pakan ayam akan menghasilkan pertumbuhan dan produksi telur yang maksimal dan biaya produksi yang rendah. Di sisi lain, komposisi pakan yang tidak tepat dapat berdampak negatif pada

kesehatan dan produktivitas ayam [2], sehingga penting bagi peternak untuk menemukan formula pakan yang optimal yang memenuhi kebutuhan nutrisi ayam dengan biaya yang minimal.

Berdasarkan survei awal, didapatkan informasi bahwa peternak ayam ras seringkali menghadapi kesulitan dalam menentukan komposisi pakan yang tepat. Berbagai faktor seperti jenis ras ayam, usia, dan berat badan harus dipertimbangkan untuk memastikan bahwa pakan yang diberikan sesuai dengan kebutuhan nutrisi spesifik setiap kelompok ayam. Kesalahan dan kekurangan dalam formulasi pakan dapat menyebabkan masalah kesehatan pada ayam, mengurangi laju pertumbuhan, dan menyebabkan ketidakseimbangan nutrisi, yang pada akhirnya menurunkan produktivitas dan profitabilitas [3]. Selain itu, fluktuasi harga bahan pakan di pasar membuat proses formulasi pakan menjadi semakin kompleks dan tidak menentu.

Banyak penelitian sebelumnya menggunakan Algoritma Genetika (GA) untuk optimasi komposisi pakan [4], namun GA sering kali memerlukan banyak iterasi untuk mencapai konvergensi dan dapat terjebak pada solusi lokal. Berbagai studi menunjukkan hasil yang bervariasi saat menggunakan GA untuk optimasi, yang menunjukkan bahwa performa GA dapat sangat bergantung pada parameter dan konfigurasi yang digunakan [5]. Beberapa penelitian belum menggabungkan algoritma metaheuristik modern seperti Algoritma Firefly dengan teknik pembelajaran mesin seperti *Reinforcement Learning* (RL) untuk optimasi komposisi pakan.

Untuk mengatasi masalah ini, penelitian ini mengusulkan model penggunaan kombinasi Algoritma Firefly dengan *Reinforcement Learning* (Firefly-RL) sebagai metode optimasi baru. Algoritma Firefly, yang didasarkan pada perilaku kunang-kunang dalam mencari pasangan, memiliki kelebihan dalam hal konvergensi yang lebih cepat dan kemampuan untuk menghindari solusi lokal [6]. Metode ini, ketika dikombinasikan dengan *Reinforcement Learning*, mampu secara adaptif menyesuaikan parameter optimasi berdasarkan *feedback* dari lingkungan, sehingga meningkatkan efisiensi pencarian solusi optimal [7]. Penelitian ini juga mempertimbangkan parameter spesifik seperti jenis ras ayam, usia, dan berat badan dalam formulasi pakan, yang memberikan solusi dan menyusun komposisi pakan sesuai dengan kebutuhan nutrisi ayam berdasarkan jenis, berat dan usia ayam ras yang ditenakkan.

2. Metode Penelitian

2.1 Langkah-langkah penelitian

Penelitian ini menggunakan pendekatan kuantitatif dengan mengembangkan dan menguji algoritma metaheuristik Firefly yang digabungkan dengan *Reinforcement Learning* untuk mengoptimalkan komposisi pakan ayam ras. Bentuk penelitian studi kasus dengan metode eksperimental research. Data bersumber langsung dari peternakan ayam ras. Langkah-langkah penelitian yang dilakukan meliputi pengumpulan data, desain model optimasi, implementasi algoritma, inialisasi parameter algoritma, pelaksanaan eksperimen, dan analisis hasil.

2.1.1 Pengumpulan Data

Data dalam penelitian ini dikumpulkan melalui dua cara yaitu studi literatur. Pengumpulan data nutrisi dan komposisi bahan pakan ayam ras (seperti kandungan protein, karbohidrat, lemak, kalsium, dan fosfor) dilakukan dengan telaah literatur dari jurnal ilmiah, standar SNI, dan publikasi pakan ternak di Indonesia. Selain itu, informasi biaya masing-masing bahan pakan diperoleh dari laporan harga pasar komoditas terbaru, laporan asosiasi peternakan, dan publikasi harga pakan di tingkat lokal. Cara kedua yaitu dengan wawancara dan survei peternak. Untuk menyesuaikan kebutuhan nyata di lapangan, data juga dihimpun melalui wawancara dengan peternak ayam ras di beberapa sentra produksi, guna memperoleh data empiris tentang kebutuhan pakan per fase usia ayam, rasio konsumsi pakan serta preferensi formula pakan. Data ini digunakan untuk memvalidasi kebutuhan nutrisi dan memastikan model optimasi sesuai dengan praktik di lapangan.

Data yang digunakan dalam penelitian ini terdiri dari informasi nutrisi dan biaya dari berbagai bahan pakan seperti jagung, dedak padi, bungkil kedelai, tepung ikan, *Cpo*, *Dcp*, Garam, *CaCO₃*, *Premix*, DL Metinon, *L Lysin*, *Polard*, dan *CGM*. Selain itu, data mengenai jenis ras ayam, usia ayam, dan berat bobot ayam juga dikumpulkan untuk menentukan kebutuhan nutrisi spesifik. Data ini diperoleh dari literatur ilmiah dan sumber peternakan terpercaya yang memberikan rincian komposisi dan biaya bahan pakan.

2.1.2 Desain model Optimasi

Algoritma Firefly dikombinasikan dengan *Reinforcement Learning* dirancang untuk menyusun komposisi pakan ayam ras yang optimal. Algoritma Firefly bertugas mengeksplorasi ruang solusi,

sedangkan *Reinforcement Learning* memperbaiki proses eksploitasi solusi dengan memanfaatkan mekanisme *reward*.

2.1.3 Implementasi Algoritma

Implementasi dilakukan dalam bahasa pemrograman Python dengan data masukan berupa komposisi nutrisi dan harga masing-masing bahan pakan. Kebutuhan nutrisi minimum setiap ayam diinput berdasarkan hasil survei dan literatur.

2.1.4 Inisialisasi Parameter Algoritma

Parameter Algoritma Firefly dan *Reinforcement Learning* diinisialisasi berdasarkan studi literatur dan eksperimen awal. Parameter penting yang diatur meliputi jumlah kunang-kunang (n), jumlah iterasi maksimum (*max iter*), koefisien penyerapan cahaya, koefisien *attractiveness* awal, dan parameter langkah acak. Untuk *Reinforcement Learning*, *adjustment* awal dan fungsi *reward* ditentukan untuk mengarahkan proses pembelajaran.

2.1.5 Pelaksanaan Eksperimen dan Pengujian

Eksperimen dilakukan dengan menerapkan Algoritma Firefly tanpa dan dengan *Reinforcement Learning* untuk mengoptimalkan komposisi pakan ayam ras. Setiap eksperimen dijalankan untuk sejumlah iterasi yang telah ditentukan, dan hasilnya dibandingkan berdasarkan kriteria biaya dan nilai gizi yang diperoleh. Proses eksperimen melibatkan evaluasi *fitness* dari setiap solusi yang dihasilkan oleh kunang-kunang, pembaruan posisi berdasarkan kecerahan dan *adjustment Reinforcement Learning*, serta pembelajaran *Reinforcement Learning* untuk memperbaiki kebijakan berdasarkan *reward* yang diterima.

Pengujian dilakukan untuk mendapatkan pengaruh jumlah *Fireflies* dan jumlah iterasi terhadap performa algoritma pada *Firefly Algorithm* dengan *Reinforcement Learning*. Pengujian dilakukan dengan beberapa variasi jumlah *Fireflies* dan jumlah iterasi, untuk mengamati bagaimana model FA-RL mencapai minimum *cost* dan memenuhi batasan nutrisi. Sebagai pembanding, Algoritma Firefly tanpa RL juga diujikan dengan parameter serupa.

2.1.6 Analisis Hasil

Hasil eksperimen dianalisis dengan membandingkan konvergensi, biaya pakan, nilai gizi, dan waktu eksekusi antara Algoritma Firefly tanpa dan dengan *Reinforcement Learning*. Grafik perbandingan digunakan untuk memvisualisasikan perbedaan kinerja kedua algoritma. Analisis statistik dilakukan untuk memastikan signifikansi perbedaan hasil antara kedua pendekatan. Hasil optimasi dibandingkan dari aspek minimum *cost*, deviasi nutrisi (*error* nutrisi), waktu proses dan stabilitas konvergensi. Analisis dilakukan secara kuantitatif dengan nilai-nilai hasil optimasi serta grafik perbandingan performa antar algoritma. Berdasarkan analisis ini, efektivitas dan efisiensi dari integrasi *Reinforcement Learning* ke dalam Algoritma Firefly dalam mengoptimalkan komposisi pakan ayam ras dievaluasi.

2.2 Algoritma Firefly

Algoritma Firefly adalah salah satu algoritma metaheuristik yang terinspirasi oleh perilaku biologis kunang-kunang. Algoritma ini dikembangkan oleh Xin-She Yang pada tahun 2008 dan digunakan untuk menyelesaikan berbagai masalah optimasi. Konsep dasar dari algoritma ini adalah menggunakan sifat kecerahan (*attractiveness*) dari kunang-kunang untuk menarik satu sama lain, yang dianalogi dengan pencarian solusi optimal dalam ruang solusi [6][8].

Prinsip Dasar perilaku pada Algoritma Firefly terdiri dari *attractiveness* dan *movement*. Setiap kunang-kunang memiliki tingkat kecerahan (*attractiveness*) yang berkaitan dengan nilai *fitness* dari solusi yang diwakilinya. Kecerahan kunang-kunang menurun seiring dengan peningkatan jarak dari kunang-kunang lain karena efek penyerapan cahaya. Kunang-kunang yang kurang cerah akan bergerak menuju kunang-kunang yang lebih cerah (*movement*). Intensitas cahaya atau kecerahan berbanding terbalik dengan jarak; semakin jauh jaraknya, semakin lemah kecerahannya. *Attractiveness* (β) dari kunang-kunang dihitung berdasarkan jarak (r) antara dua kunang-kunang dan parameter penyerapan cahaya (γ). Fungsi *attractiveness* dapat dirumuskan sebagai [8][9]:

$$\beta(r) = \beta_0 e^{-\gamma r^2} \quad (1)$$

di mana (β_0) adalah *attractiveness* pada jarak nol.

Pembaharuan Posisi kunang-kunang i diperbaharui menuju kunang-kunang j yang lebih cerah dengan rumus [8][9]:

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2} (x_j^t - x_i^t) + \alpha(\text{rand}-0.5) \quad (2)$$

di mana:

x_i adalah posisi kunang-kunang

r_{ij} adalah jarak antara kunang-kunang i dan j

α adalah parameter langkah acak,

rand adalah bilangan acak yang diambil dari distribusi *uniform* dalam interval $[0,1]$

2.3 Algoritma Firefly dengan *Reinforcement Learning*

Integrasi *Reinforcement Learning* ke dalam Algoritma Firefly bertujuan untuk meningkatkan kemampuan eksplorasi dan eksploitasi dari algoritma tersebut. Dengan menggabungkan *Reinforcement Learning*, Algoritma Firefly dapat belajar dari pengalaman iterasi sebelumnya untuk membuat keputusan yang lebih baik di iterasi berikutnya. *Reinforcement Learning* adalah metode *machine learning* di mana agen belajar untuk membuat keputusan melalui *trial and error* untuk memaksimalkan *reward* kumulatif. Komponen utama dari *Reinforcement Learning* antara lain [10]:

- a) Agen: Entitas yang mengambil tindakan.
- b) Lingkungan: Tempat di mana agen beroperasi.
- c) Tindakan (*Actions*): Langkah-langkah yang dapat diambil oleh agen.
- d) *Reward*: Umpan balik yang diterima agen setelah mengambil tindakan.
- e) Kebijakan (*Policy*): Strategi yang digunakan oleh agen untuk menentukan tindakan berdasarkan keadaan saat ini.

Langkah-Langkah Algoritma Firefly dengan *Reinforcement Learning*:

1. Inisialisasi
Tetapkan parameter algoritma dan inisialisasi posisi awal kunang-kunang yaitu jumlah kunang-kunang (n), jumlah iterasi maksimum (*max iter*), koefisien penyerapan cahaya (γ), koefisien attractiveness awal (β_0), dan parameter langkah acak (α). Inisialisasi komponen *Reinforcement Learning*, termasuk *adjustment* awal dan fungsi *reward*.
2. Evaluasi
Hitung nilai *fitness* untuk setiap kunang-kunang berdasarkan fungsi tujuan.
3. Pembelajaran *Reinforcement*
Untuk setiap kunang-kunang i , evaluasi tindakan berdasarkan kebijakan *Reinforcement Learning*. Lakukan tindakan yang memaksimalkan *reward* yaitu pergerakan ke arah kunang-kunang yang lebih cerah. Perbarui *adjustment* dan fungsi nilai berdasarkan *reward* yang diterima.
4. Perbarui Posisi Kunang-kunang
Perbarui posisi setiap kunang-kunang i berdasarkan *adjustment Reinforcement Learning* dan pergerakan ke arah kunang-kunang yang lebih cerah.
Formula pembaruan posisi diperbaiki untuk mempertimbangkan *reward* dan *adjustment Reinforcement Learning*:

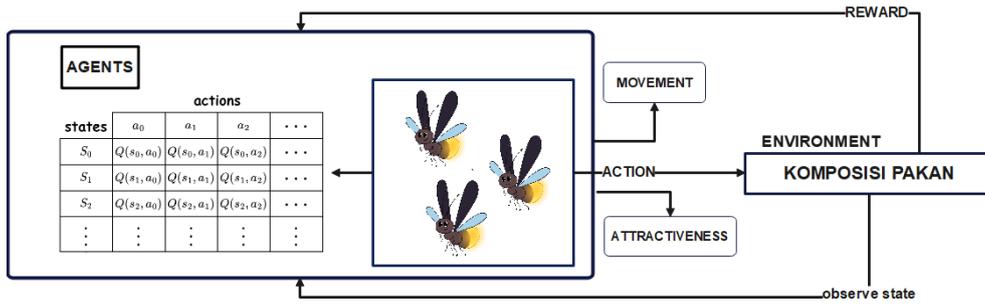
$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}} (x_j^t - x_i^t) + \alpha(\text{rand}-0.5) + \text{adjustment} \quad (3)$$

di mana *Adjustment* adalah penyesuaian posisi berdasarkan *adjustment Reinforcement Learning*.

5. Iterasi
Ulangi langkah evaluasi dan pembaruan posisi hingga mencapai jumlah iterasi maksimum atau kondisi berhenti lainnya.
6. Solusi Terbaik
Setelah mencapai iterasi maksimum atau kondisi berhenti, pilih kunang-kunang dengan nilai *fitness* terbaik sebagai solusi optimal.

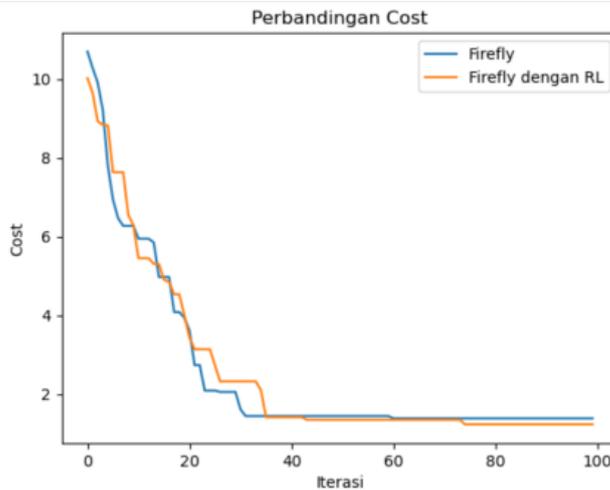
3. Hasil dan Pembahasan

Dalam optimasi komposisi pakan, Algoritma Firefly digunakan untuk mencari kombinasi bahan pakan yang memenuhi batasan nutrisi dan meminimalkan biaya. *Reinforcement Learning* akan membantu Algoritma Firefly dalam mempercepat konvergensi menuju solusi optimal dengan memberikan *reward* berdasarkan kualitas solusi yang dihasilkan pada setiap iterasi. Integrasi ini dilakukan dengan menggunakan hasil Algoritma Firefly sebagai *input* untuk *Reinforcement Learning*. Seperti yang ditampilkan pada arsitektur di Gambar 1. Setiap solusi yang dihasilkan oleh Algoritma Firefly akan dievaluasi oleh *Reinforcement Learning*, disimpan dalam *Q-Table* dan *feedback* dari *Reinforcement Learning* akan digunakan untuk memperbaiki solusi pada iterasi berikutnya. Proses ini berulang hingga solusi optimal ditemukan.



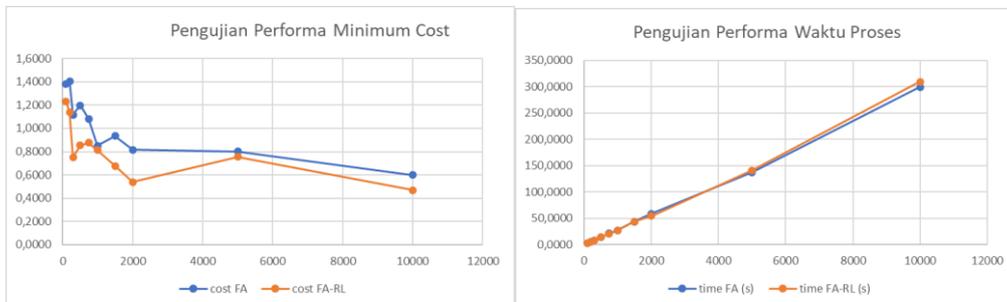
Gambar 1. Arsitektur Model Integrasi FA dengan RL.

Sejumlah pengujian berikut dilakukan dengan parameter jumlah *fireflies* sebanyak 30, koefisien penyerapan cahaya 1.0, koefisien *attractiveness* 1.0. Sedangkan parameter *Reinforcement Learning* dengan *learning rate* sebesar 0.1 dan nilai *gamma* 0.9. Gambar 2. menunjukkan hasil perbandingan *cost* dengan *Iterasi_maksimum*=100 dan jumlah *Fireflies*=30. Di mana didapatkan hasil bahwa Firefly dengan *Reinforcement Learning* memiliki *cost* yang minimum. Selanjutnya dilakukan pengujian performa berdasarkan jumlah iterasi yang dapat dilihat pada Tabel 1 berikut.



Gambar 2. Grafik Hasil Perbandingan *Cost* dengan *Iterasi_maksimum*=100 dan Jumlah *Fireflies*=30.

Gambar 3 menunjukkan bahwa semakin banyak jumlah iterasi maka akan menghasilkan model Firefly dengan *Reinforcement Learning* dengan performa yang lebih baik dimana minimum *cost* akan tercapai, namun jika dilihat dari waktu proses Firefly dengan *Reinforcement Learning* cenderung akan lebih lambat dibandingkan dengan Algoritma Firefly.



Gambar 3. Grafik Hasil Penguujian Performa *Cost* dan Waktu Proses Berdasarkan Jumlah Iterasi.

Algoritma Firefly cenderung memperlihatkan performa yang baik dalam mencari solusi optimal melalui mekanisme cahaya yang memandu pergerakan kunang-kunang. Dengan jumlah iterasi yang cukup, algoritma ini dapat menemukan komposisi pakan yang memenuhi kebutuhan nutrisi dan meminimalkan

biaya. Akan tetapi, solusi yang dihasilkan tidak selalu berupa solusi global optimal, melainkan berpotensi hanya mencapai solusi lokal optimal. Integrasi Algoritma Firefly dengan *Reinforcement Learning* meningkatkan kemampuan algoritma dalam mengeksplorasi ruang solusi secara lebih efektif. *Reinforcement Learning* memberikan *reward* berdasarkan kualitas solusi di setiap iterasi, sehingga algoritma dapat secara konsisten memperbaiki solusi yang diperoleh.

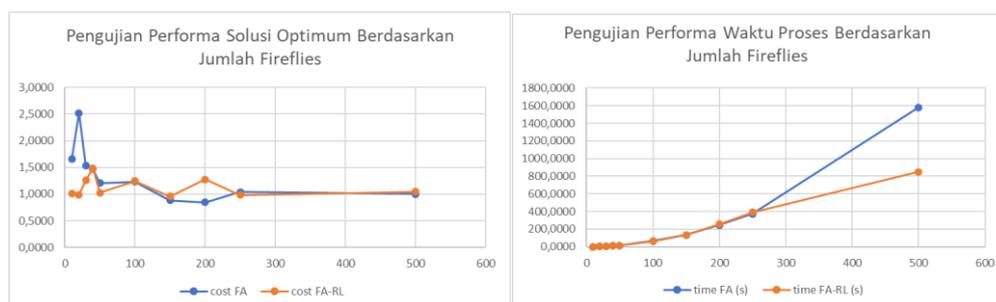
Tabel 1. Analisis Nilai Performa Berdasarkan Jumlah Iterasi.

Percobaan	Iterasi	Fireflies	cost FA	cost FA-RL	Error Nutrisi Firefly	Error Nutrisi FA-RL	Optimasi biaya pakan (%)
1	100	30	1,3826	1,2317	3,9819	1,2324	83,25
2	200	30	1,4058	1,1364	3,5869	1,0417	81,158
3	300	30	1,1175	0,7514	3,7885	1,3595	85,992
4	500	30	1,1973	0,8541	3,8703	1,4510	86,035
5	750	30	1,0833	0,8772	3,5635	1,0846	82,603
6	1000	30	0,8504	0,8129	3,6398	1,9484	82,286
7	1500	30	0,9335	0,6776	3,4192	1,5326	81,187
8	2000	30	0,8148	0,5389	3,0140	1,1949	82,952
9	5000	30	0,8028	0,7567	3,2252	1,4335	87,273
10	10000	30	0,5983	0,4686	3,7293	1,4381	87,714
Rata-rata					3,58186	1,37167	84,045

Tabel 1 menunjukkan hasil percobaan bahwa *Firefly Algorithm* dengan *Reinforcement Learning (FA-RL)* memiliki performa lebih baik dibandingkan *Firefly* biasa dalam hal *minimum cost* (optimasi biaya pakan per kg) dan akurasi pemenuhan nutrisi. Mulai awal iterasi, model belum memantapkan proses eksplorasi dan eksploitasi, sehingga *error* nutrisi cenderung tinggi. Alasannya, solusi-solusi awal masih sangat acak, belum mendekati kebutuhan target secara presisi. Model mulai stabil pada iterasi sedang, *error* nutrisi rata-rata turun sedikit karena algoritma memperoleh kesempatan cukup untuk training memperbaiki solusi. Bertambahnya jumlah iterasi, Algoritma *Firefly* dengan *Reinforcement Learning* menunjukkan penurunan yang lebih konsisten pada biaya pakan dibandingkan dengan Algoritma *Firefly* biasa. Ini dikarenakan *Reinforcement Learning* membantu algoritma untuk lebih cermat dalam mengeksplorasi dan mengeksploitasi solusi yang ada. Hasil penelitian menunjukkan bahwa *minimum cost* tercapai lebih efektif dengan penggunaan *Reinforcement Learning*.

Waktu proses Algoritma *Firefly* ini cenderung lebih cepat karena hanya mengandalkan mekanisme cahaya untuk pergerakan kunang-kunang. Tanpa adanya tambahan evaluasi dari *Reinforcement Learning*, proses iterasi berjalan lebih efisien. Algoritma *Firefly* dengan *Reinforcement Learning* menghasilkan solusi dengan biaya yang lebih rendah, waktu proses dari algoritma ini cenderung lebih lambat. Hal ini disebabkan oleh tambahan evaluasi dan update kebijakan yang dilakukan oleh *Reinforcement Learning* pada setiap iterasi. Setiap solusi yang dihasilkan harus dievaluasi dan diberi *reward*, yang menambah kompleksitas dan waktu proses. Waktu proses dari Algoritma *Firefly* dengan *Reinforcement Learning* lebih lama dibandingkan dengan Algoritma *Firefly* biasa, terutama pada jumlah iterasi yang lebih tinggi. Ini disebabkan oleh tambahan evaluasi dan update kebijakan yang dilakukan oleh *Reinforcement Learning*, yang meningkatkan kompleksitas dan durasi setiap iterasi. Meskipun waktu proses lebih lama, keunggulan Algoritma *Firefly* dengan *Reinforcement Learning* terletak pada kemampuannya untuk mencapai solusi yang lebih optimal dalam hal biaya dan pemenuhan nutrisi. Oleh karena itu, algoritma ini lebih cocok digunakan dalam skenario di mana kualitas solusi lebih penting daripada kecepatan waktu proses.

Pengujian berikutnya dilakukan dengan variasi jumlah *fireflies* dengan parameter koefisien penyerapan cahaya 1.0, koefisien *attractiveness* 1.0 dan jumlah iterasi sebanyak 100. Sedangkan parameter *Reinforcement Learning* dengan *learning rate* sebesar 0.1 dan nilai *gamma* 0.9.



Gambar 4. Grafik Hasil Pengujian Performa Cost dan Waktu Proses Berdasarkan Jumlah Fireflies.

Berdasarkan Gambar 4, jumlah *fireflies* tidak melebihi jumlah iterasi, performa Algoritma Firefly dengan *Reinforcement Learning (FA-RL)* cenderung lebih baik. Hal ini disebabkan oleh kemampuan *FA-RL* untuk memanfaatkan jumlah iterasi yang cukup untuk setiap firefly dalam proses eksplorasi dan eksploitasi solusi. Dengan jumlah *fireflies* yang moderat, algoritma memiliki keseimbangan yang baik antara eksplorasi ruang solusi dan konvergensi menuju solusi optimal. Ketika jumlah *fireflies* melebihi jumlah iterasi, performa *FA-RL* cenderung menurun. Hal ini terjadi karena setiap firefly tidak memiliki cukup iterasi untuk mengeksplorasi ruang solusi secara efektif. Akibatnya, proses konvergensi menjadi tidak efisien dan solusi yang dihasilkan mungkin tidak optimal.

Tabel 2. Analisis Nilai Performa Berdasarkan Jumlah *Fireflies*.

Percobaan	<i>Fireflies</i>	Iterasi	<i>cost</i> FA	<i>cost</i> FA-RL	<i>Error</i> Nutrisi Firefly	<i>Error</i> Nutrisi FA-RL	Optimasi biaya pakan (%)
1	10	100	1,6549	1,0146	3,7412	1,1737	84,2740
2	20	100	2,5168	0,9913	3,8807	1,4932	87,5130
3	30	100	1,5342	1,2562	3,1770	1,7341	82,9250
4	40	100	1,4711	1,4867	3,6147	1,7521	85,8860
5	50	100	1,2060	1,0248	3,2015	1,2624	80,2150
6	100	100	1,2306	1,2452	3,6215	1,4881	83,5510
7	150	100	0,8817	0,9610	3,3658	1,4888	88,7010
8	200	100	0,8476	1,2750	4,1399	1,3633	81,4200
9	250	100	1,0416	0,9852	3,4600	1,5926	88,5660
10	500	100	1,0029	1,0475	3,4177	1,1631	84,0800
Rata-rata					3,562	1,45114	84,7131

Hasil percobaan pada Tabel 2 menunjukkan bahwa *Firefly Algorithm* dengan *Reinforcement Learning (FA-RL)* memiliki performa lebih baik dibandingkan *Firefly* biasa dalam hal minimum *cost* (biaya pakan per kg) dan akurasi pemenuhan nutrisi berdasarkan jumlah *fireflies*. Jumlah *fireflies* yang kecil akan mengakibatkan minimum *cost* FA-RL yang selalu lebih rendah daripada *Firefly* biasa. Jumlah *fireflies* yang banyak akan menyebabkan *error* nutrisi semakin naik karena iterasi tidak cukup untuk seluruh *fireflies* mengeksplorasi ruang solusi secara optimal.

Waktu proses Algoritma *Firefly* biasa relatif lebih cepat karena tidak adanya tambahan evaluasi dan *update* kebijakan dari *Reinforcement Learning*. Algoritma *Firefly* dengan *Reinforcement Learning* memiliki Waktu proses cenderung lebih lambat karena setiap *Firefly* harus dievaluasi dan diberikan *reward* oleh *RL*, yang menambah kompleksitas dan durasi setiap iterasi. Namun, terdapat titik di mana waktu proses mulai membaik ketika jumlah *fireflies* mencapai sekitar 500, karena dengan jumlah ini, algoritma mulai mencapai efisiensi dalam pengelolaan populasi *fireflies*. Performa optimal dari *FA-RL* dicapai saat jumlah *fireflies* tidak melebihi jumlah iterasi. Pada kondisi ini, algoritma mampu mengeksplorasi dan mengeksploitasi ruang solusi dengan lebih efektif, mencapai solusi dengan biaya yang lebih rendah. Namun, jika jumlah *fireflies* terlalu banyak, tanpa cukup iterasi, algoritma menjadi tidak efisien.

4. Kesimpulan

Firefly Algorithm dengan *Reinforcement Learning* menunjukkan performa yang lebih baik dalam optimasi komposisi pakan ayam ras, terutama ketika jumlah *fireflies* tidak melebihi jumlah iterasi. Algoritma ini tetap memerlukan waktu proses yang lebih lama dibandingkan Algoritma *Firefly* biasa, tetapi mampu menghasilkan biaya pakan lebih rendah, yaitu rata-rata 84,045 % per kg (berdasarkan jumlah iterasi) dan 84,7131 % per kg (berdasarkan jumlah *firefly*) pada Algoritma *Firefly* biasa, dengan deviasi nutrisi yang juga lebih kecil 1,37167 (berdasarkan jumlah iterasi) dan 1,45114 (berdasarkan jumlah *firefly*). Optimalitas dan efisiensi *FA-RL* tercapai ketika jumlah *fireflies* bersifat moderat, dan waktu proses mulai membaik saat jumlah *fireflies* banyak, dengan peningkatan rata-rata waktu proses pada jumlah *fireflies* lebih besar. Integrasi *Firefly Algorithm* dengan *Reinforcement Learning* ini memberikan pendekatan yang lebih efektif untuk memenuhi kebutuhan nutrisi dan menekan biaya pakan, sehingga menjadi alternatif yang lebih baik bagi peternak dalam meningkatkan efisiensi dan produktivitas.

Daftar Pustaka

- [1] N. Haryuni, “Analisis Kinerja Finansial Kenaikan Harga Dedak Padi Terhadap Tingkat Pendapatan Peternak Ayam Petelur Di Kabupaten Blitar Jawa Timur,” *Jurnal Ilmiah Fillia Cendekia*, vol. 3, no. 1, pp. 10–15, 2018.
- [2] J. Junaedi, “Analisis Faktor Penghambat Pengembangan Usaha Ayam Pedaging pada Peternakan Rakyat Pola Mandiri Di Kabupaten Buton Tengah,” *Musamus Journal of Livestock Science*, vol. 7, no. 1, pp. 31–40, 2024.

-
- [3] A. K. Armayanti *et al.*, *Nutrisi Ternak Dasar: Dinamika Teori dan Perkembangannya*. PT. Sonpedia Publishing Indonesia, 2024.
- [4] A. A. Bahar, “Optimasi Komposisi Pakan Ayam Broiler Sesuai Standarisasi Kebutuhan Nutrisi Menggunakan Algoritma Genetika,” *Jurnal Teknologi Informasi dan Terapan*, vol. 6, no. 1, pp. 30–34, 2019.
- [5] N. Rikatsih and W. F. Mahmudy, “Adaptive genetic algorithm based on crossover and mutation method for optimization of poultry feed composition,” in *2018 International Conference on Sustainable Information Engineering and Technology (SIET)*, Nov. 2018, pp. 110–114.
- [6] W. Pei, G. Huayu, Z. Zheqi, and L. Meibo, “A novel hybrid firefly algorithm for global optimization,” in *2019 IEEE 4th international conference on computer and communication systems (ICCCS)*, Feb. 2019, pp. 164–168.
- [7] J. Jia and W. Wang, “Review of reinforcement learning research,” in *2020 35th Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, Oct. 2020, pp. 186–191.
- [8] M. Hussain and W. K. Jenkins, “Effectiveness of the bio-inspired firefly algorithm in adaptive signal processing for nonlinear systems,” in *2019 IEEE international symposium on circuits and systems (ISCAS)*, May 2019, pp. 1–4.
- [9] J. Wang *et al.*, “Improvement and application of hybrid firefly algorithm,” *IEEE Access*, vol. 7, pp. 165458–165477, 2019.
- [10] N. Mazyavkina, S. Sviridov, S. Ivanov, and E. Burnaev, “Reinforcement learning for combinatorial optimization: A survey,” *Comput Oper Res*, vol. 134, p. 105400, 2021.