



## Optimasi Ukuran Penampang pada Struktur Rangka Batang Ruang dengan *Binary Dragonfly Algorithm*

R. Frans<sup>1\*</sup>, S. Thioritz<sup>2</sup>

<sup>1\*,2</sup>Program Studi Teknik Sipil, Fakultas Teknik, Universitas Atma Jaya Makassar, Makassar,  
Indonesia

Email : <sup>1\*</sup>[richardfrans.rf@gmail.com](mailto:richardfrans.rf@gmail.com), <sup>2</sup>[stevythio@gmail.com](mailto:stevythio@gmail.com)

### ARTICLE INFO

#### Article history :

Artikel masuk : 05 – 06 – 2023  
Artikel revisi : 18 – 06 – 2023  
Artikel diterima : 27 – 06 – 2023

#### Keywords :

Bamboo, Binary Dragonfly Algorithm, Optimization, Space Truss Structures.

#### Style IEEE dalam mensitasi artikel ini:

R. Frans, and S. Thioritz, "Optimasi Ukuran Penampang pada Struktur Rangka Batang Ruang dengan Binary Dragonfly Algorithm," *Jurmateks*, vol.6, no.1, pp. 45-57, 2023, doi: 10.30737/jurmateks.v6i1.4709

### ABSTRACT

The cross-section area of a structural element plays a vital role in designing and planning processes. Each structural element must meet design criteria while maintaining an effective cross-sectional area, ensuring the structure is economical and lightweight. This Economic aspect refers to the structure's cost and weight, minimizing any lateral effects that may arise. This research aimed to determine the optimum cross-sectional for bamboo elements, with the objective of minimizing the weight of the space truss structure but still meets the design criteria. Four different cross-sectional areas of bamboo were used, 4272,566 mm<sup>2</sup>, 3382,71 mm<sup>2</sup>, 3084,259 mm<sup>2</sup> and 2827,433 mm<sup>2</sup>. The optimization algorithm used is the binary dragonfly algorithm. There are two configurations of space truss structures considered, the first space truss structure is 25 members, and the second space truss is 32 members (pyramid module). The program was run four times to ensure the convergence and consistency of the results obtained. The results show that the weight of the first space truss structure is 86.98 kg while the weight of the second space truss structure is 64.6015 kg. The results can be considered optimal as the critical element's actual stress is close to the allowable stress. Therefore, the binary dragonfly algorithm can be an alternative method to optimize the cross-sectional area of structural element especially for space truss structure.

## 1. Pendahuluan

Bambu menjadi material yang berpotensi digunakan dalam konstruksi bangunan. Bambu dapat ditanam dengan investasi kecil, memiliki pertumbuhan cepat, dan merupakan sumber daya yang berkelanjutan untuk konstruksi baik skala kecil maupun besar. Kekuatan alamiah bambu menjadi salah satu faktor utama penggunaannya sebagai bahan bangunan [1]–[5]. Bambu memiliki kekuatan yang cukup tinggi meskipun memiliki kepadatan yang relatif

rendah. Hal ini disebabkan oleh struktur seratnya yang unik, terdiri dari serat-serat serong yang memberikan kekuatan dan kestabilan pada bambu. Bambu juga memiliki modulus elastisitas yang tinggi, yang memungkinkannya untuk menyerap dan mendistribusikan beban dengan baik [6], [7]. Penggunaan bambu sebagai bahan bangunan dapat membantu mengurangi jejak karbon yang menjadi masalah saat ini [8], [9]. Pada kondisi tertentu, kekuatan bambu bahkan dapat menjadi salah satu alternatif pengganti material baja [10].

Dalam penggunaannya dalam material konstruksi, penentuan ukuran penampang yang efektif menjadi bagian yang sangat penting. Hal ini dimaksudkan untuk mengoptimalkan penggunaan material, kekuatan, stabilitas, dan mengurangi beban serta efek gaya lateral yang timbul. Ukuran penampang yang tidak optimal cenderung menghasilkan struktur yang lebih berat daripada yang diperlukan. Berat struktur yang tidak efektif juga dapat mempengaruhi kinerja struktural secara keseluruhan, seperti respons terhadap beban lateral. Dalam hal ini, kurangnya penggunaan metode optimasi menjadi salah satu dari banyak faktor yang menyebabkan tidak efektifnya penentuan ukuran penampang struktur.

Beberapa penelitian terdahulu telah membahas efektifitas penggunaan algoritma optimasi dalam bidang teknik sipil [11]. Pengkajian mengenai topologi pada struktur portal menggunakan material bambu dengan enam tipe topologi struktur yang dibandingkan, yaitu M Shape (3 tipe), W Shape (3 tipe) didapatkan struktur tipe *Double Howe* memiliki berat paling minimum dan nilai defleksi masih lebih kecil dibandingkan defleksi yang diijinkan. Penentuan tegangan ijin pada elemen selain meninjau gaya aksial yang terjadi pada masing-masing elemen tetapi juga meninjau momen lentur yang terjadi pada masing-masing elemen dikarenakan struktur diasumsikan sebagai rangka portal [11]. Penelitian lain melakukan optimasi topologi, ukuran penampang dan geometri dari suatu struktur rangka batang bidang dengan menggunakan material baja dan menggunakan algoritma genetika *hybrid* dimana algoritma ini merupakan gabungan antara algoritma biner dan algoritma riil [12]. Algoritma ini juga diterapkan pada penelitian dengan algoritma biner yang berbeda [13]. Algoritma optimasi dapat berupa algoritma yang bersifat riil maupun biner yang bergantung pada aplikasinya [14], [15]. *Binary dragonfly algorithm* merupakan bentuk algoritma diskrit dari *dragonfly algorithm* [16]. Sesuai dengan nama algoritmanya, *binary dragonfly algorithm* meniru perilaku dari kumpulan *dragonfly* yang berada di alam. Penggunaan algoritma optimasi telah banyak digunakan untuk menyelesaikan masalah-masalah di bidang ketekniksipil [17]–[19]. Namun, belum terdapat penelitian yang membahas mengenai penentuan ukuran penampang dari elemen menggunakan material bambu pada struktur rangka batang ruang dengan teknik optimasi, yaitu *binary dragonfly algorithm*.

Penelitian ini bertujuan untuk menentukan ukuran penampang bambu optimum pada elemen struktur rangka batang ruang menggunakan algoritma *binary dragonfly*. Algoritma tersebut akan memilih ukuran penampang yang optimum terhadap beban yang diterima berdasarkan tegangan yang terjadi dan tegangan ijinnya. Melalui hasil tersebut akan dicapai bobot struktur yang optimal serta meningkatkan efisiensi penggunaan material. Dengan mengaplikasikan algoritma *binary dragonfly*, diharapkan dapat memberikan kontribusi baru dalam domain optimasi ukuran penampang struktur rangka batang ruang.

## 2. Metodologi Penelitian

Dalam penelitian ini, terdapat dua jenis tipe struktur rangka batang ruang yang dianalisis. Model struktur pertama mengacu pada penelitian sebelumnya, struktur ini memiliki ukuran panjang, lebar dan tinggi yang sama, yaitu 2540 mm dengan 25 elemen [20]. Sedangkan model kedua juga mengacu pada penelitian sebelumnya yang memiliki bentuk mirip modul piramida dengan panjang, lebar dan tinggi struktur adalah masing-masing 2000 mm, 2000 mm dan 707 mm dengan 32 elemen [21]. Dalam penelitian ini terdapat 4 (empat) ukuran penampang bambu yang digunakan, yaitu  $4272,566 \text{ mm}^2$ ,  $3382,71 \text{ mm}^2$ ,  $3084,259 \text{ mm}^2$ ,  $2827,433 \text{ mm}^2$ . Keempat ukuran penampang bambu tersebut telah melalui uji kekuatan secara eksperimental untuk mendapatkan properti material bambu yang kemudian digunakan sebagai kriteria perencanaan [22]. Perhitungan dan analisis struktur serta algoritma optimasi yang digunakan (*binary dragonfly*) diprogramkan dengan menggunakan *software* MATLAB R2022a [22]–[24].

### 2.1 Kriteria Perencanaan

Dalam kriteria perencanaan terdapat 3 variabel yang dipertimbangkan yaitu tegangan aksial berupa tegangan tarik, tegangan tekan serta faktor kelangsingan dari elemen struktur. Tegangan tarik yang terjadi harus lebih kecil atau sama dengan tegangan tarik ijin, begitu pula dengan tegangan tekan harus lebih kecil atau sama dengan tegangan tekan ijin yang memperhitungkan faktor kelangsingan. Dalam penelitian ini, tegangan tekan ijin, tegangan tarik ijin yang digunakan adalah 40 MPa dan 280 MPa dengan modulus elastisitas sebesar 15.000 MPa.

### 2.2 *Binary Dragonfly Algorithm*

Mekanisme eksplorasi dan eksploitatif dari *dragonfly algorithm* dimodelkan oleh interaksi *dragonfly* dalam menemukan solusi yang optimal. Pada *dragonfly algorithm*, terdapat 5 (lima) perilaku yang digambarkan dari *dragonfly*, yaitu pemisahan, kesejajaran, kohesi, ketertarikan dan distraksi. Kelima perilaku ini digunakan untuk memperbarui posisi dari

*dragonfly*. Pemisahan bertujuan untuk menghindari terjadinya tumbukan statis (*static collision*) pada individu-individu di lingkungannya. Kesejajaran memungkinkan pencocokan kecepatan tiap individu dalam kawanan atau sub-kawanan. Kohesi mengacu pada penyimpangan individu saat ini menuju pusat massa kelompok *dragonfly*. Ketertarikan menunjukkan individu harus tertarik sumber makanan. Gangguan menyiratkan individu harus terganggu keluar dari predator.

Pada *dragonfly algorithm*, posisi dari *dragonfly* diperbarui dengan menggunakan beberapa persamaan dikarenakan *dragonfly algorithm* merupakan sebuah algoritma untuk kasus kontinu maka perlu ada penyesuaian atau modifikasi untuk menjadi bentuk diskrit. Proses *binary dragonfly algorithm* dapat dilihat pada **Gambar 1**

---

**Algorithm 1. Binary Dragonfly Algorithm**

- 1) Randomly initialize positions of  $N$  dragonflies,  $X$
- 2) Initialize the step vectors,  $\Delta X$  to zeros
- 3) **while** (Maximum number of iterations is not met)
- 4) Evaluate the fitness values of dragonflies
- 5) Update the food source,  $Xf$  and enemy,  $Xe$
- 6) Update  $s$ ,  $a$ ,  $c$ ,  $f$ ,  $e$ , and  $w$
- 7) **for**  $i = 1$  to number of dragonflies,  $N$
- 8) Calculate  $S$ ,  $A$ , and  $C$  using (1), (2), and (3)
- 9) Compute  $F$  and  $E$  using (4) and (5)
- 10) Update step vector using (6)
- 11) Update the position of dragonfly ( $i$ -th) using (8)
- 12) **end for**
- 13) **end while**

**Output:** Best food source,  $Xf$

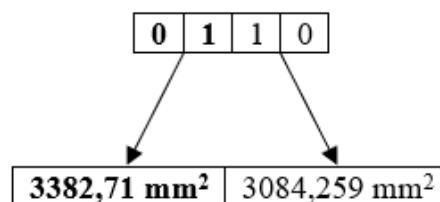
---

Sumber: J. Too and S. Mirjalili, "A Hyper Learning Binary Dragonfly Algorithm for Feature Selection: A COVID-19 Case Study," (2020).

**Gambar 1.** Algoritma Binary Dragonfly [25].

### 2.3 Konversi Angka Biner menjadi Angka Riil dan Fungsi Objektif

Dalam algoritma *binary dragonfly*, angka biner digunakan dalam proses optimasi. Namun, untuk melakukan analisis lebih lanjut perlu dilakukan konversi angka biner menjadi angka riil. **Gambar 2** menunjukkan contoh konversi angka biner menjadi angka riil untuk ukuran penampang tertentu. Vektor ukuran penampang yang dihasilkan dari konversi disusun sebagai berikut:  $\{A\} = \{4272,566 \text{ mm}^2, 3382,71 \text{ mm}^2, 3084,259 \text{ mm}^2, 2827,433 \text{ mm}^2\}$ .



Sumber: Peneliti (2023).

**Gambar 2.** Contoh Konversi Angka Biner menjadi Angka Riil.

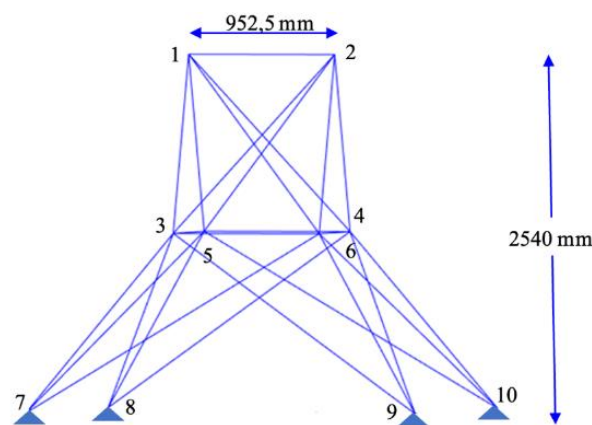
**Gambar 2** menunjukkan nilai angka biner adalah [0 1] akan menghasilkan vektor ukuran penampang yang digunakan, yaitu 3382,71 mm<sup>2</sup> sedangkan jika nilai angka biner adalah [1 0], ukuran penampang riil yang digunakan adalah 3084,259 mm<sup>2</sup> dan seterusnya. Seperti yang telah disampaikan sebelumnya bahwa tujuan dari algoritma ini adalah mendapatkan struktur rangka batang ruang dengan ukuran penampang sekecil mungkin namun tetap memenuhi kriteria perencanaan. Oleh karena itu, fungsi objektif yang digunakan untuk menghitung *fitness* adalah berat dari struktur rangka batang ruang. Persamaan (1) menunjukkan nilai *fitness* yang merupakan representasi dari berat struktur rangka batang ruang itu sendiri.

$$fitness = W = \sum_{k=1}^n \rho A_k l_k \quad (1)$$

Dimana  $W$  merupakan berat struktur rangka batang ruang (kg),  $\rho$  massa jenis bambu, diambil sebesar 710 kg/m<sup>3</sup>,  $A_k$  ukuran penampang profil ke- $k$  (m<sup>2</sup>), dan  $l_k$  adalah panjang batang ke- $k$  (m).

### 3. Hasil dan Diskusi

#### 3.1 Model 1: Struktur Rangka Batang Ruang Dengan 25 Elemen



Sumber: *Discrete Optimization of Structures Using Genetic Algorithms*, (1992).

#### **Gambar 3.** Struktur rangka batang ruang dengan 25 batang [20].

Tinggi struktur rangka batang ruang 2,54 m dengan panjang dan lebar masing-masing sebesar 2,54 m arah-x dan arah-y. Dimensi struktur yang digunakan yaitu setengah dari dimensi struktur pada penelitian yang dirujuk [20]. Hal ini dikarenakan kelangsingan batang yang tidak terpenuhi jika dibandingkan dengan dimensi penampang bambu yang tersedia. Perletakan sendi pada node 7, 8, 9 dan 10. Hubungan antar *node* untuk merepresentasikan elemen dapat dilihat pada **Tabel 1**.

**Tabel 1.** Elemen Member pada Struktur Rangka Batang Ruang Aplikasi Pertama.

Elemen	Node-i	Node-j	Elemen	Node-i	Node-j
1	1	2	14	3	10
2	1	4	15	6	7
3	2	3	16	4	9
4	1	5	17	5	8
5	2	6	18	4	7
6	2	4	19	3	8
7	2	5	20	5	10
8	1	3	21	6	9
9	1	6	22	6	10
10	3	6	23	3	7
11	4	5	24	4	8
12	3	4	25	5	9
13	5	6			

Sumber: Data Olahan, (2023).

Untuk menguji algoritma yang digunakan, pengujian struktur tanpa beban dilakukan guna memverifikasi bahwa ukuran-ukuran penampang yang dihasilkan adalah ukuran penampang terkecil (sesuai yang ditentukan yaitu 2827, 433 mm<sup>2</sup>). Nilai tegangan tarik dan tegangan tekan yang didapatkan akan memiliki nilai 0 MPa yang berarti bahwa tidak terdapat beban luar yang bekerja. **Tabel 2** menunjukkan bahwa seluruh ukuran penampang untuk elemen-1 sampai elemen-25 adalah 2827,433 mm<sup>2</sup>. Algoritma *binary dragonfly* dapat menghasilkan ukuran penampang terkecil yang diharapkan, itu menunjukkan bahwa algoritma tersebut dapat digunakan secara efektif dan dapat menyelesaikan permasalahan yang lebih kompleks sehingga selanjutnya dapat mempertimbangkan beban ke dalam struktur. Oleh karena itu, beban vertikal diaplikasikan pada node 1, 2, 3, 4, 5 dan 6 sebesar 3 kN.

**Tabel 2.** Hasil Ukuran Penampang Tanpa Pembebanan untuk Aplikasi Pertama.

Elemen	Ukuran Penampang (mm <sup>2</sup> )	Tegangan Elemen (MPa)	Elemen	Ukuran Penampang (mm <sup>2</sup> )	Tegangan Elemen (MPa)
1	2,827,433	0	14	2,827,433	0
2	2,827,433	0	15	2,827,433	0
3	2,827,433	0	16	2,827,433	0
4	2,827,433	0	17	2,827,433	0
5	2,827,433	0	18	2,827,433	0
6	2,827,433	0	19	2,827,433	0
7	2,827,433	0	20	2,827,433	0
8	2,827,433	0	21	2,827,433	0
9	2,827,433	0	22	2,827,433	0
10	2,827,433	0	23	2,827,433	0
11	2,827,433	0	24	2,827,433	0
12	2,827,433	0	25	2,827,433	0
13	2,827,433	0			

Sumber: Data Olahan, (2023).

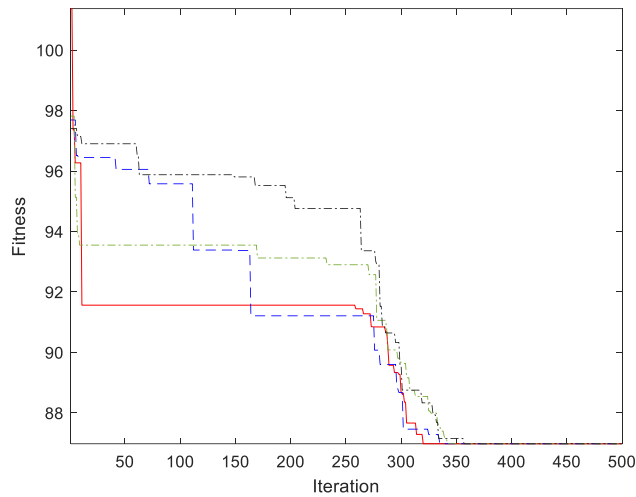


Berdasarkan **Tabel 3**, hasil optimasi dengan pembebanan vertikal sebesar 3 kN pada node 1, 2, 3, 4, 5 dan 6 menghasilkan ukuran penampang sebesar 2827,433 mm<sup>2</sup> pada elemen selain elemen 20, 21, 22 dan 25, yang merupakan ukuran penampang terkecil yang diharapkan. Tegangan tarik maksimal yang didapatkan pada elemen lainnya adalah 17,4459 kg/cm<sup>2</sup> sedangkan tegangan ijinnya adalah sebesar 628,1475 kg/cm<sup>2</sup>. Hal yang sama untuk tegangan tekan, tegangan tekan maksimal adalah 10,3041 kg/cm<sup>2</sup> sedangkan tegangan ijin tekan adalah untuk elemen tersebut adalah sebesar 20,4186 kg/cm<sup>2</sup>. Tegangan yang terjadi pada elemen-elemen tersebut juga jauh lebih kecil dibandingkan dengan tegangan ijin yang disyaratkan. Hal ini menunjukkan bahwa optimasi telah berhasil dalam menghasilkan ukuran penampang yang efisien dan tegangan yang aman sedangkan untuk elemen 20, 21, 22 dan 25 memiliki ukuran penampang yang lebih besar namun tetap optimal. Hal ini terlihat dari nilai tegangan yang mendekati nilai tegangan ijin yang disyaratkan pada keempat elemen tersebut. Tegangan maksimum yang terjadi pada elemen 20, 21, 22 dan 25 adalah sebesar 7,042 kg/cm<sup>2</sup>, 7,7106 kg/cm<sup>2</sup>, 15,1922 kg/cm<sup>2</sup> dan 15,2448 kg/cm<sup>2</sup> sedangkan nilai tegangan ijin untuk masing-masing elemen tersebut adalah sebesar 8,3092 kg/cm<sup>2</sup>, 7,8132 kg/cm<sup>2</sup>, 15,3064 kg/cm<sup>2</sup> dan 15,3064 kg/cm<sup>2</sup>. Meskipun tegangan maksimal yang terjadi pada elemen tersebut tidak melebihi tegangan ijin, namun ukuran penampang yang dihasilkan lebih besar dari ukuran penampang terkecil yang diharapkan.

**Tabel 3.** Hasil Ukuran Penampang dengan Pembebanan untuk Aplikasi Pertama.

Elemen	Ukuran Penampang (mm <sup>2</sup> )	Tegangan ijin (kg/cm <sup>2</sup> )	Tegangan yang terjadi (kg/cm <sup>2</sup> )	Elemen	Ukuran Penampang (mm <sup>2</sup> )	Tegangan ijin (kg/cm <sup>2</sup> )	Tegangan yang terjadi (kg/cm <sup>2</sup> )
1	2,827,433	6,281,475	0,007	14	2,827,433	70,979	60,623
2	2,827,433	6,281,475	62,572	15	2,827,433	6,281,475	7,013
3	2,827,433	6,281,475	6,217	16	2,827,433	70,979	60,466
4	2,827,433	136,748	62,693	17	2,827,433	6,281,475	70,008
5	2,827,433	136,748	62,291	18	2,827,433	6,281,475	85,469
6	2,827,433	6,281,475	10,314	19	2,827,433	6,281,475	85,329
7	2,827,433	204,186	103,041	<b>20</b>	<b>3382,71</b>	<b>83,092</b>	<b>7,042</b>
8	2,827,433	6,281,475	102,811	<b>21</b>	<b>3,084,259</b>	<b>78,132</b>	<b>77,106</b>
9	2,827,433	204,186	102,711	<b>22</b>	<b>3382,71</b>	<b>153,064</b>	<b>151,922</b>
10	2,827,433	414,043	0,5248	23	2,827,433	6,281,475	173,855
11	2,827,433	414,043	0,5267	24	2,827,433	6,281,475	174,459
12	2,827,433	414,043	40,002	<b>25</b>	<b>3382,71</b>	<b>153,064</b>	<b>152,448</b>
13	2,827,433	6,281,475	38,725				

Sumber: Data Olahan, (2023).

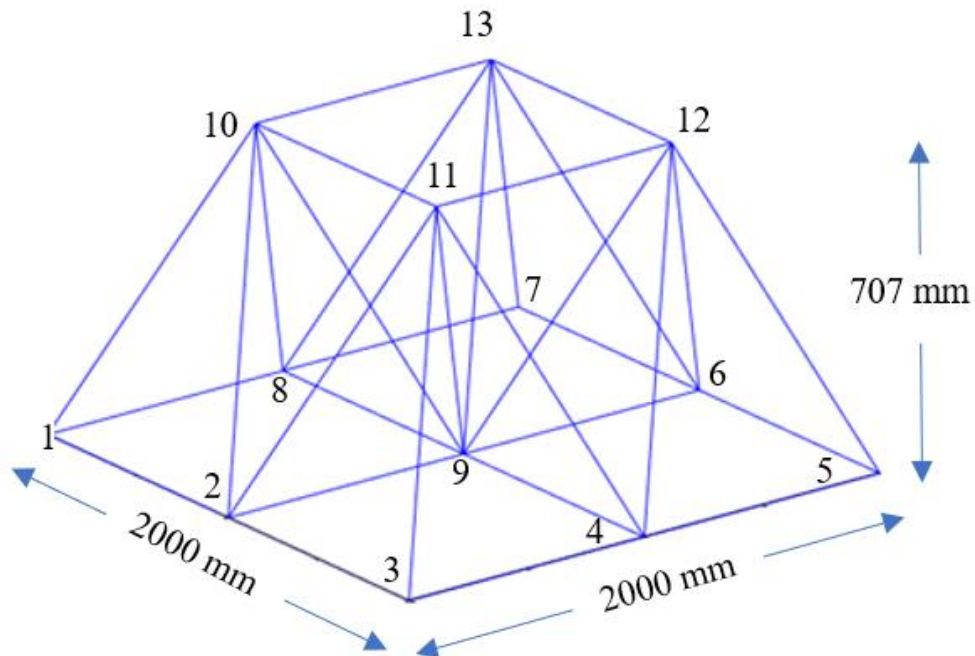


Sumber: Data Olan, (2023).

**Gambar 4.** Kurva *Fitness* untuk Setiap Iterasi Aplikasi Pertama.

Berdasarkan **Gambar 4**, dapat dilihat bahwa nilai *fitness* yang merepresentasikan tiap berat dari struktur rangka batang ruang itu sendiri mengalami penurunan setiap iterasi yang didapatkan. Hal ini menunjukkan bahwa algoritma *binary dragonfly* berhasil dalam mengoptimalkan struktur rangka batang ruang untuk mencapai berat yang lebih rendah. Total berat struktur yang didapatkan setelah optimasi adalah sebesar 86,98 kg.

### 3.2 Model 2: Struktur Rangka Batang Ruang Dengan 32 Batang



Sumber: Data Olan, (2023).

**Gambar 5.** Struktur Rangka Batang Ruang dengan 32 Batang.



**Tabel 4.** Elemen Member pada Struktur Rangka Batang Ruang Aplikasi Kedua.

Elemen	Node-i	Node-j	Elemen	Node-i	Node-j
1	1	2	17	2	12
2	2	3	18	3	12
3	3	4	19	4	12
4	4	5	20	9	12
5	5	6	21	9	13
6	6	7	22	4	13
7	7	8	23	5	13
8	8	1	24	6	13
9	2	9	25	8	10
10	8	9	26	9	10
11	4	9	27	6	10
12	6	9	28	7	10
13	1	11	29	10	11
14	2	11	30	11	12
15	8	11	31	12	13
16	9	11	32	13	10

Sumber: Data Olahan, (2023).

Sama halnya dengan model 1, struktur rangka batang ruang tidak diberikan beban terlebih dahulu untuk mengecek algoritma yang telah dibuat. **Tabel 5** menunjukkan bahwa program telah berjalan dengan baik, karena tidak ada tegangan tarik dan tegangan tekan yang terjadi pada elemen – elemen tersebut sehingga menghasilkan ukuran penampang yang minimal, yaitu sebesar 2827,433 mm<sup>2</sup>. Tegangan untuk semua elemen adalah 0 MPa yang menandakan bahwa tidak terdapat beban luar yang bekerja.

**Tabel 5.** Hasil Ukuran Penampang Tanpa Pembebanan untuk Aplikasi Kedua.

Elemen	Ukuran Penampang (mm <sup>2</sup> )	Tegangan Elemen (MPa)	Elemen	Ukuran Penampang (mm <sup>2</sup> )	Tegangan Elemen (MPa)
1	2827,433	0	17	2827,433	0
2	2827,433	0	18	2827,433	0
3	2827,433	0	19	2827,433	0
4	2827,433	0	20	2827,433	0
5	2827,433	0	21	2827,433	0
6	2827,433	0	22	2827,433	0
7	2827,433	0	23	2827,433	0
8	2827,433	0	24	2827,433	0
9	2827,433	0	25	2827,433	0
10	2827,433	0	26	2827,433	0
11	2827,433	0	27	2827,433	0
12	2827,433	0	28	2827,433	0
13	2827,433	0	29	2827,433	0
14	2827,433	0	30	2827,433	0
15	2827,433	0	31	2827,433	0
16	2827,433	0	32	2827,433	0

Sumber: Data Olahan, (2023).

Optimasi Ukuran Penampang pada Struktur Rangka Batang Ruang dengan *Binary Dragonfly Algorithm*

<https://dx.doi.org/10.30737/jurmateks.v6i1.4709>



Selanjutnya, beban diaplikasikan vertikal pada tiap titik *node* kecuali pada *node* yang terdapat reaksi perletakan. Beban yang digunakan adalah sebesar 4,5 kN.

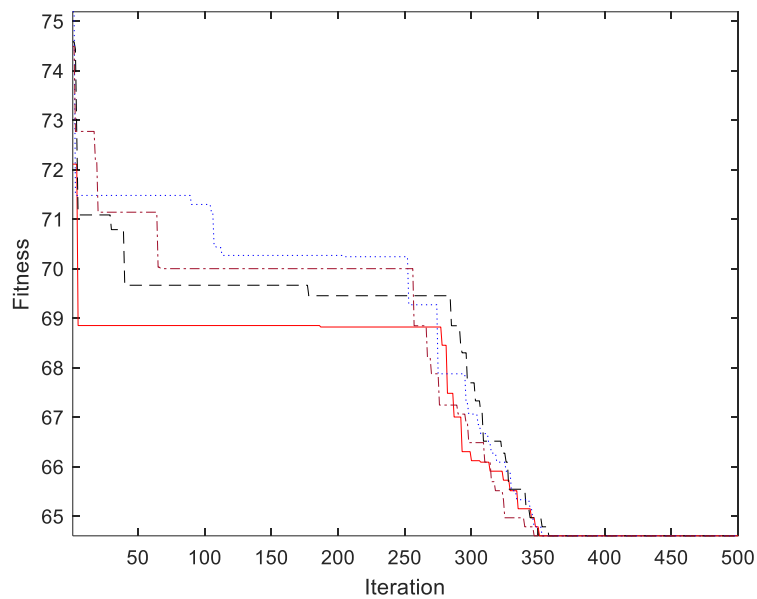
**Tabel 6.** Hasil Optimasi untuk Aplikasi Kedua.

Elemen	Ukuran Penampang (mm <sup>2</sup> )	Tegangan ijin (kg/cm <sup>2</sup> )	Tegangan yang terjadi (kg/cm <sup>2</sup> )	Elemen	Ukuran Penampang (mm <sup>2</sup> )	Tegangan ijin (kg/cm <sup>2</sup> )	Tegangan yang terjadi (kg/cm <sup>2</sup> )
1	2827,433	0	37,5643	17	2827,433	0	628,1475
2	2827,433	0	628,1475	18	2827,433	16,2225	37,57
3	2827,433	28,3916	37,5643	19	2827,433	40,5563	628,1475
4	2827,433	28,3916	628,1475	20	2827,433	24,3338	37,57
5	2827,433	0	628,1475	21	2827,433	24,3338	628,1475
6	2827,433	0	37,5643	<b>22</b>	<b>3084,259</b>	<b>37,1792</b>	<b>41,3563</b>
7	2827,433	28,3916	628,1475	23	2827,433	16,2225	628,1475
8	2827,433	28,3916	37,5643	24	2827,433	0	628,1475
9	2827,433	16,2237	628,1475	<b>25</b>	<b>3084,259</b>	<b>37,1792</b>	<b>41,3563</b>
10	2827,433	0	628,1475	26	2827,433	24,3338	628,1475
11	2827,433	0	628,1475	27	2827,433	0	628,1475
12	2827,433	16,2237	37,5643	28	2827,433	16,2225	628,1475
13	2827,433	16,2225	37,57	29	2827,433	0	628,1475
14	2827,433	0	628,1475	30	2827,433	24,3356	628,1475
15	2827,433	40,5563	628,1475	31	2827,433	0	628,1475
16	2827,433	24,3338	37,57	32	2827,433	24,3356	37,5643

Sumber: Data Olahan, (2023).

Berdasarkan **Tabel 6** tegangan yang terjadi pada elemen-elemen selain elemen 22 dan 25 jauh lebih rendah daripada tegangan ijin, sehingga ukuran penampang yang digunakan adalah 2827,433 mm<sup>2</sup>. Tegangan tarik maksimal pada elemen-elemen lainnya adalah 40,5563 kg/cm<sup>2</sup>, sedangkan tegangan ijinnya adalah sebesar 628,1475 kg/cm<sup>2</sup>. Demikian pula, tegangan tekan maksimal adalah 28,3916 kg/cm<sup>2</sup> dengan tegangan ijin tekan untuk elemen tersebut adalah 37,5643 kg/cm<sup>2</sup>.

Pada elemen 22 dan 25, tegangan yang terjadi mendekati nilai tegangan yang diijinkan, dengan ukuran penampang yang digunakan sebesar 3084,259 mm<sup>2</sup>. Tegangan pada elemen 22 adalah 37,1792 kg/cm<sup>2</sup>, dengan nilai tegangan ijin sebesar 41,3563 kg/cm<sup>2</sup>. Sedangkan untuk elemen 25, tegangan yang terjadi adalah 37,1792 kg/cm<sup>2</sup> dengan tegangan ijin sebesar 41,3563 kg/cm<sup>2</sup>. Hasil ini menunjukkan bahwa algoritma optimasi berhasil dalam mencapai ukuran penampang yang sesuai dengan batasan tegangan ijin. Tegangan yang terjadi pada elemen 22 dan 25 mendekati nilai tegangan ijin, menunjukkan keefektifan algoritma dalam menemukan solusi yang memenuhi persyaratan tegangan. Dengan demikian, dapat disimpulkan bahwa ukuran penampang yang digunakan pada elemen 22 dan 25 telah optimum.



Sumber: Data Olan, (2023).

**Gambar 6.** Kurva *Fitness* untuk Setiap Iterasi Aplikasi Kedua.

Untuk memastikan konvergensi dan konsistensi hasil optimasi, program dijalankan sebanyak 4 (empat) kali. **Gambar 6** menunjukkan bahwa hasil yang diperoleh adalah konvergen dan konsisten dikarenakan ukuran penampang yang didapatkan pada keempat percobaan program adalah sama. Ini menunjukkan bahwa algoritma yang digunakan mampu menghasilkan solusi yang stabil dan tidak bergantung pada kondisi awal atau iterasi. Hasil optimasi tersebut menghasilkan total berat struktur sebesar 64,6015 kg.

#### 4. Kesimpulan

Berdasarkan hasil diskusi, algoritma *binary dragonfly* mampu menentukan ukuran penampang yang optimal untuk elemen-elemen struktur rangka batang ruang yang ditinjau. Hasil optimasi menunjukkan bahwa ukuran penampang yang diperoleh memenuhi kriteria perencanaan dan memiliki elemen struktur yang efisien. Hal tersebut ditunjukkan dengan nilai tegangan yang terjadi mendekati dengan tegangan yang diijinkan pada elemen-elemen yang kritis. Selanjutnya, penelitian dapat dilakukan untuk menerapkan algoritma *binary dragonfly* pada tipe struktur lainnya, seperti struktur rangka bidang ataupun struktur rangka ruang. Sehingga, dapat ditemukan ukuran penampang yang optimal untuk struktur-struktur tersebut dan dapat meningkatkan efisiensi penggunaan material.

**Daftar Pustaka**

- [1] K. Chaowana, S. Wisadsatorn, and P. Chaowana, "Bamboo as a sustainable building material—culm characteristics and properties," *Sustain.*, vol. 13, no. 13, 2021, doi: 10.3390/su13137376.
- [2] F. B. Haves, "The natural renewable building material-bamboo. A comparison between Germany and Indonesia.," *J. A Sustain. Glob. South*, vol. 3, no. 2, p. 25, 2019, doi: 10.24843/jsgs.2019.v03.i02.p05.
- [3] X. Yu, "Bamboo: structure and culture: Utilizing bamboo in the industrial context with reference to its structure and cultural dimensions," p. 179, 2007, [Online]. Available: <https://books.google.com.my/books?id=1UnnAQAACAAJ&dq=bamboo+structure&hl=en&sa=X&ved=0ahUKEwia4Jaz-83XAhULv7wKHTQJCLsQ6AEIMDAC>
- [4] G. Prajapati and S. Dua, "A Critical Review of Bamboo as a Building Material for Sustainable Development," vol. Volume 4, no. 3, 2022, [Online]. Available: <https://www.researchgate.net/publication/360701427>
- [5] M. Yadav and A. Mathur, "Bamboo as a sustainable material in the construction industry: An overview," *Mater. Today Proc.*, vol. 43, pp. 2872–2876, 2021, doi: 10.1016/j.matpr.2021.01.125.
- [6] M. J. Romero Méndez *et al.*, "Prototipo de invernadero con estructura de bambú: selección de materiales, diseño e impacto ambiental," *Rev. Mex. Ciencias Agrícolas*, vol. 8, no. 5, pp. 1059–1072, 2017, doi: 10.29312/remexca.v8i5.108.
- [7] H. Zhang, J. Zhong, Z. Liu, J. Mai, H. Liu, and X. Mai, "Dyed bamboo composite materials with excellent anti-microbial corrosion," *Adv. Compos. Hybrid Mater.*, vol. 4, no. 2, pp. 294–305, 2021, doi: 10.1007/s42114-020-00196-8.
- [8] A. K. Boity, H. Bhandari, and S. Shukla, "Bamboo as a sustainable building construction material," *Materialstoday:Proceedings*, vol. 71, no. 2, pp. 306–311, 2022, doi: <https://doi.org/10.1016/j.matpr.2022.09.218>.
- [9] V. Lescha and T. E. Darmayanti, "Application of Bamboo Material to the Characteristics of Residential Houses in the Tropics: Hideout Bali," *J. Arsit.*, vol. 12, no. 2, p. 165, 2022, doi: 10.36448/ja.v12i2.2367.
- [10] P. F. Borowski, I. Patuk, and E. R. Bandala, "Innovative Industrial Use of Bamboo as Key 'Green' Material," *Sustain.*, vol. 14, no. 4, 2022, doi: 10.3390/su14041955.
- [11] P. Ubolsook and S. Thepa, "Structural analysis of bamboo trusses Structure in greenhouse," *2nd Int. Conf. Environ. Sci. Technol.*, vol. 6, no. March, pp. 228–232, 2011.

- [12] R. Frans and Y. Arfiadi, "Sizing, shape, and topology optimizations of roof trusses using hybrid genetic algorithms," *Procedia Eng.*, vol. 95, no. Scescm, pp. 185–195, 2014, doi: 10.1016/j.proeng.2014.12.178.
- [13] R. Frans, "Batang Bidang Dengan Menggunakan Binary Bat Algorithm," vol. VII, no. Ii, pp. 74–81, 2022.
- [14] A. Jain, S. Sharma, and S. Sharma, *Firefly Algorithm*. 2021. doi: 10.1002/9781119681984.ch6.
- [15] X.-S. Yang, *Nature-Inspired Optimization Algorithms*. Elsevier, 2020.
- [16] S. Mirjalili, "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Comput. Appl.*, vol. 27, no. 4, pp. 1053–1073, 2016, doi: 10.1007/s00521-015-1920-1.
- [17] A. Kaveh, *Advances in metaheuristic algorithms for optimal design of structures, second edition*. 2016. doi: 10.1007/978-3-319-46173-1.
- [18] A. Kaveh, *Applications of metaheuristic optimization algorithms in civil engineering*. 2016. doi: 10.1007/978-3-319-48012-1.
- [19] T. Dede, M. Kripka, V. Toğan, V. Yepes, and R. V. Rao, "Advanced Optimization Techniques and Their Applications in Civil Engineering," *Adv. Civ. Eng.*, vol. 2018, 2018, doi: 10.1155/2018/5913083.
- [20] S. Rajeev and C. S. Krishnamoorthy, "Discrete Optimization Of Structures Using Genetic Algorithms By. S. Rajeev 1 and C. S. Krishnamoorthy 2," *October*, vol. 118, no. 5, pp. 1233–1250, 1992.
- [21] C. A. S. Freitas, W. V. Silva, L. M. Bezerra, F. S. M. Francisco, V. C. P. Neto, and B. A. T. Ribeiro, "Experimental analysis of space trusses with typical connections reinforced with steel and sisal-resin spacers," *Adv. Steel Constr.*, vol. 15, no. 4, pp. 398–405, 2019, doi: 10.18057/IJASC.2019.15.4.10.
- [22] Mathwork, "MATLAB: The Language of Technical Computing," *The Mathwork*, 2023.
- [23] Mathworks, "MATLAB: Programming Fundamental," 2023.
- [24] S. Nayak and S. Chakraverty, *Interval Finite Element Method with MATLAB*, Academic P. 2018. doi: <https://doi.org/10.1016/B978-0-12-812973-9.09987-2>.
- [25] J. Too and S. Mirjalili, "A Hyper Learning Binary Dragonfly Algorithm for Feature Selection: A COVID-19 Case Study," *Knowledge-Based Syst.*, vol. 212, p. 106553, 2021, doi: 10.1016/j.knosys.2020.106553.