

# Rekomendasi Spesifikasi Personal Computer Berdasarkan Kemampuan Finansial Menggunakan *Binary Particle Swarm Optimization*

Burhanu Sultan Ramadan <sup>a,1,\*</sup>, Ardiansyah <sup>b,2</sup>

<sup>a,b</sup> Program Studi Informatika Universitas Ahmad Dahlan, Jl. Ringroad Selatan, Daerah Istimewa Yogyakarta 55191, Indonesia

<sup>1</sup> burhanu1900018242@webmail.uad.ac.id\*; <sup>2</sup> ardiansyah@tif.uad.ac.id

## ABSTRAK

PC atau *Personal Computer* merupakan salah satu perangkat teknologi yang banyak digunakan hingga saat ini. Dalam menentukan PC yang diinginkan, konsumen biasanya mempertimbangkan berdasarkan kebutuhan dan kemampuan finansialnya. Namun menentukan spesifikasi PC yang tepat sesuai *budget* cukup sulit dilakukan karena banyaknya pilihan komponen yang tersedia. Oleh karena itu, penelitian ini bertujuan untuk mengembangkan aplikasi rekomendasi spesifikasi PC berbasis web menggunakan Algoritma BPSO (*Binary Particle Swarm Optimization*) guna membantu konsumen dalam memilih komponen PC yang sesuai dengan kemampuan finansialnya. Penelitian ini menggunakan data 8 komponen PC yang berisi model-model dari tiap komponen PC tersebut. Data-data tersebut akan dioptimasi menggunakan Algoritma BPSO, di mana proses optimasi akan diulangi sebanyak 30 kali menggunakan jumlah partikel yang berbeda yaitu 5, 15, 20, 25, dan 30 partikel. Setiap eksperimen akan dilakukan evaluasi menggunakan fungsi *fitness*, dengan validasi berupa kuisioner terhadap beberapa responden. Hasil dari penelitian ini adalah berupa aplikasi web yang mampu memberikan rekomendasi spesifikasi PC berdasarkan kemampuan finansial konsumen. Aplikasi dapat memberikan hasil yang optimal serta cepat dibandingkan dengan hasil yang diperoleh pada responden. Berdasarkan hasil eksperimen, ukuran populasi yang optimal adalah sebesar 30 partikel, dengan rata-rata nilai *fitness* yaitu 0,000159722 namun memiliki konsekuensi waktu pencarian yang lebih lama yaitu 12,85 detik.



## Kata Kunci

Personal Computer  
BPSO  
Sistem Rekomendasi  
Optimasi  
Fungsi Fitness  
Optimasi Diskrit



This is an open-access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license

## 1. Pendahuluan

PC atau *Personal Computer* merupakan salah satu perangkat teknologi yang banyak digunakan hingga saat ini. Menurut IDC (*International Data Corporation*), pasar PC Indonesia meningkat 50% dari tahun ke tahun, dengan total 2.6 juta unit dikirim ke Indonesia selama 6 bulan pertama pada 2021. PC yang beredar terbagi menjadi dua jenis yaitu PC *built-up* dan PC rakitan. PC *built-up* adalah PC yang dibuat langsung oleh pabrikan PC. Sedangkan PC rakitan adalah PC yang komponen-komponennya dirakit sendiri oleh konsumen ataupun dirakit oleh teknisi sesuai permintaan dari konsumen [1].

Dalam menentukan PC yang diinginkan, konsumen biasanya mempertimbangkan berdasarkan kebutuhan dan kemampuan finansialnya. Konsumen yang memiliki kemampuan finansial yang cukup lebih cenderung akan memilih PC *built-up* karena dapat langsung dipakai dan tidak perlu mengetahui komponen-komponen pembentuk sebuah PC.

Berbeda dengan PC rakitan, konsumen perlu untuk mengetahui komponen-komponen pembentuk sebuah PC. Hal ini membuat sebagian besar konsumen yang awam dengan hal tersebut datang langsung ke toko komputer untuk menanyakan spesifikasi komputer yang sesuai dengan kebutuhan serta sesuai dengan kemampuan finansialnya, dengan itu konsumen dapat langsung mengetahui komponen yang cocok berdasarkan arahan teknisi, akan tetapi harga dan spesifikasi yang disediakan



tentu berbeda antara toko komputer satu dengan yang lainnya. Tidak sedikit juga konsumen yang memilih untuk mencari referensi di internet sebelum membeli komponen-komponen pembentuk PC. Tetapi karena sudah banyaknya komponen *hardware* PC yang dikeluarkan oleh produsen-produsen *hardware*, membuat konsumen mengalami kesulitan dalam memilih komponen-komponen PC yang sesuai dengan kebutuhan serta kemampuan finansialnya.

Beberapa penelitian telah dilakukan untuk mengatasi permasalahan terkait spesifikasi PC. Penelitian yang dilakukan oleh [2] menggunakan Algoritma *Backtracking* menghasilkan aplikasi yang dapat merekomendasikan spesifikasi PC dengan keluaran kombinasi komponen PC tidak melebihi batas anggaran yang dimasukkan dan tetap menjaga kompatibilitas antara komponen PC dengan baik. Penelitian ini hanya menggunakan 5 komponen PC yaitu *motherboard*, *processor* (CPU), *memory* (RAM), *graphic card* (GPU), dan *storage* (HDD).

Penelitian lain dilakukan [1], aplikasi dapat melakukan pencarian menggunakan Algoritma Genetika untuk menghasilkan solusi yang optimal berupa spesifikasi PC. Penelitian ini menggunakan 7 komponen PC yaitu *motherboard*, *memory* (RAM), *processor* (CPU), *graphic card* (GPU), *power supply* (PSU), *storage* (HDD/SSD), dan monitor.

Pada penelitian ini, metode yang digunakan berupa Algoritma BPSO (*Binary Particle Swarm Optimization*), yang dipilih karena lebih efisien, efektif, dan lebih cepat mengalami konvergensi dibandingkan dengan Algoritma Genetika [3]. Komponen PC yang digunakan akan terdiri dari 8 komponen diantaranya adalah *motherboard*, *processor* (CPU), *graphic card* (GPU), *memory* (RAM), *storage* (HDD/SSD), *power supply* (PSU), *cpu cooler* dan *case*.

Tujuan dari penelitian ini adalah untuk menghasilkan aplikasi rekomendasi spesifikasi PC berbasis *web* untuk menampilkan hasil rekomendasi spesifikasi PC berdasarkan kemampuan finansial yang dimiliki oleh konsumen. Harapannya, aplikasi ini nantinya dapat membantu konsumen dalam memilih komponen PC yang sesuai dengan kemampuan finansialnya.

## 2. Metode

### 2.1. Tahapan Penelitian

Penelitian ini terbagi menjadi beberapa tahapan yang akan dijabarkan sebagai berikut :

1. Studi Literatur

Pencarian referensi melalui jurnal, buku dan sumber-sumber yang relevan lainnya untuk mendukung penelitian serta memberikan gambaran terkait penelitian yang akan dilakukan.

2. Penyiapan Alat

Dalam penelitian ini menggunakan perangkat keras dan perangkat lunak untuk mengembangkan sistem dapat dilihat pada Tabel 2.1 dan Tabel 2.2:

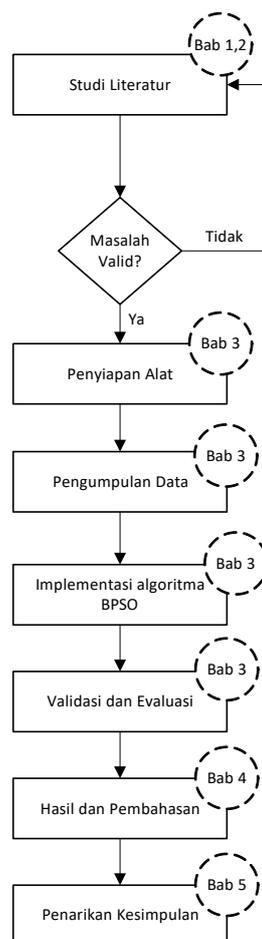
**Tabel 2. 1** Perangkat keras

Item	Spesifikasi
Laptop HP Pavilion Gaming-15	Processor Core i7, Memori 8 GB, SSD 256 GB

**Tabel 2. 2** Perangkat lunak

Item	Spesifikasi
Sistem Operasi	Windows 11, 64-bit
<i>Text Editor</i>	Visual Studio Code
<i>Web Browser</i>	Google Chrome dan Mozilla Firefox
<i>Framework</i>	CodeIgniter 4 dan Flask
Bahasa Pemrograman	PHP 8.1.6 dan Python 3.11.1
Basis Data	MySQL / MariaDB 10.4.24

3. Pengumpulan Data  
Metode pengumpulan data yang digunakan adalah dokumen-dokumen berupa daftar harga komponen-komponen PC secara *online*.
4. Implementasi Algoritma  
Melakukan implementasi Algoritma BPSO (*Binary Particle Swarm Optimization*) dalam pembuatan aplikasi untuk memperoleh rekomendasi spesifikasi PC berdasarkan kemampuan finansial konsumen.
5. Validasi dan Evaluasi  
Uji Validasi dilakukan kepada 3 responden yang mengetahui jenis-jenis komponen pembentuk sebuah PC, dengan meminta responden untuk memilih model dari 8 komponen PC tanpa mempertimbangkan kompatibilitas dari tiap komponen, di mana setiap responden dimisalkan mempunyai *budget* 20 juta. Evaluasi adalah proses identifikasi untuk mengukur hasil yang telah didapatkan oleh suatu metode. Tujuan dari evaluasi adalah menilai efisiensi dan efektifitas metode. Pada penelitian ini, teknik yang digunakan untuk mengevaluasi tingkat akurasi dari nilai objektif yang dihasilkan adalah fungsi *fitness*.
6. Hasil dan pembahasan  
Hasil penelitian yang dilakukan digambarkan dalam grafik yang menunjukkan konvergensi dari nilai *fitness* dan grafik waktu eksekusi. Setiap hasil akan dianalisis dan dibahas dengan menjabarkan faktor-faktor yang mempengaruhi hasil tersebut.
7. Penarikan kesimpulan  
Tahapan terakhir adalah berupa penarikan kesimpulan dan saran terkait penerapan algoritma BPSO (*Binary Particle Swarm Optimization*) pada aplikasi yang dikembangkan.



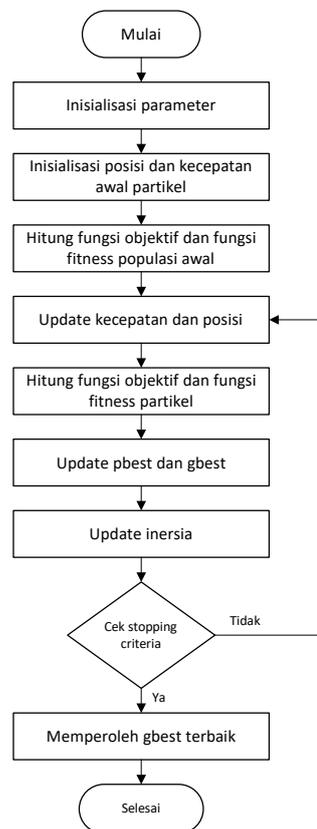
Gambar 2. 1 Diagram tahapan penelitian

## 2.2. Algoritma BPSO (*Binary Particle Swarm Optimization*)

BPSO atau *Binary Particle Swarm Optimization* merupakan modifikasi dari algoritma sebelumnya yaitu PSO (*Particle Swarm Optimization*) [4]. PSO merupakan algoritma metaheuristik yang melakukan pencarian solusi berdasarkan pada perilaku sosial dari kawanan burung, lebah, atau ikan. Algoritma ini mensimulasikan pergerakan kawanan burung ketika mencari makanan. Jika seekor burung menemukan makanan, informasi terkait makanan tersebut akan menyebar ke seluruh kawanan burung dan akan mempengaruhi pergerakan burung yang lain. Burung diasumsikan sebagai partikel dan kawanan burung dapat diasumsikan sebagai populasi partikel [5].

Algoritma PSO pada awalnya dikembangkan untuk ruang permasalahan dengan nilai kontinyu [6], tetapi banyak permasalahan optimasi yang juga terjadi dalam ruang permasalahan dengan nilai diskrit [7]. Contoh sederhana adalah permasalahan yang membutuhkan untuk mengurutkan atau mengatur elemen diskrit, seperti penjadwalan atau pencarian rute [4].

Pada Algoritma BPSO, partikel merepresentasikan posisi-nya dalam bilangan biner yaitu 0 atau 1. Setiap partikel kemudian dapat mengubah nilai posisi-nya atau bermutasi dari 1 menjadi 0 ataupun sebaliknya. Perbedaan utama dari algoritma BPSO dan PSO adalah kecepatan sebuah partikel didefinisikan sebagai kemungkinan berubahnya nilai posisi partikel menjadi 1. Sehingga, kecepatan dibatasi dalam interval  $[0,1]$  [8]. Alur Algoritma BPSO dapat dilihat pada gambar di bawah ini:



Gambar 2. 2 Flowchart Algoritma BPSO

## 3. Hasil dan Pembahasan

### 3.1. Pengolahan Data

Data yang dibutuhkan berupa 8 komponen komputer yang meliputi adalah *motherboard*, *processor* (CPU), *graphic card* (GPU), *memory* (RAM), *storage* (HDD/SSD), *power supply* (PSU), *cpu cooler* dan *case*. Data tersebut diperoleh dari beberapa pasar online Indonesia diantaranya adalah

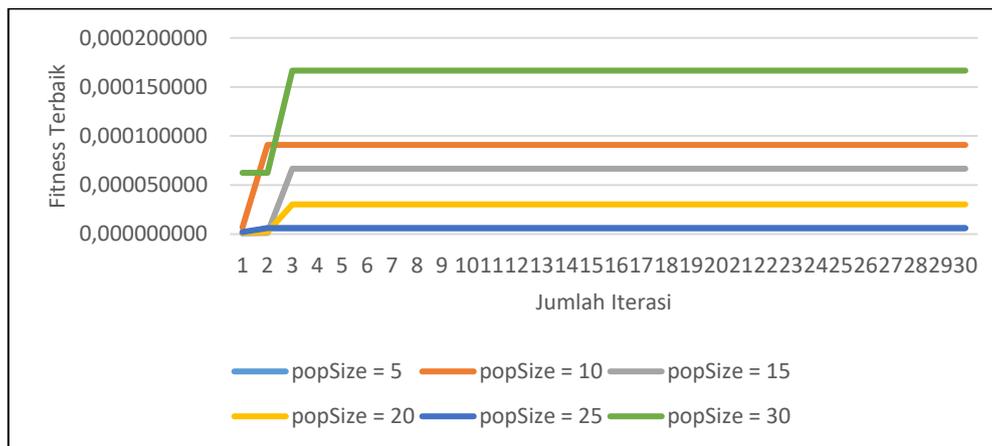
*Shopee, Tokopedia, Blibli, dan Bhineka*. Data yang diperoleh kemudian dimasukkan ke dalam basis data dan digunakan sebagai data utama yang kemudian diproses menggunakan Algoritma BPSO untuk mendapatkan spesifikasi PC yang sesuai dengan kemampuan finansial konsumen.

### 3.2. Hasil Eksperimen

Eksperimen ini dilakukan untuk menguji konvergensi nilai *fitness* berdasarkan ukuran populasi yang berbeda. Proses pengujian akan dibagi menjadi dua bagian dengan setiap bagian menggunakan komposisi data model komponen PC yang berbeda. Hal ini dilakukan untuk mengecek hasil rekomendasi apabila data model pada setiap komponen tidak memiliki jumlah yang sama.

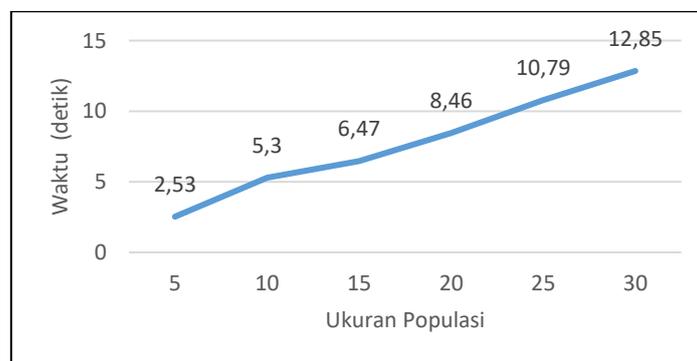
#### 3.1.1. Komposisi Data Spesifikasi PC 1

Pada bagian ini, komposisi data spesifikasi PC yang digunakan terdiri dari 8 komponen PC dengan setiap komponen memiliki 8 model. Berikut grafik konvergensi pada gambar 3.1:



Gambar 3. 1 Grafik konvergensi data komposisi PC 1

Berdasarkan grafik konvergensi pada gambar 3.1, rata-rata konvergensi setiap ukuran populasi terjadi pada iterasi ke 2 dan 3. Hal ini menunjukkan bahwa iterasi maksimum dapat diterapkan dengan studi kasus penelitian, serta dilihat dari kondisi pemberhentiannya, iterasi dapat berhenti setelah terjadi konvergen dan bukan sebaliknya. Melalui perbandingan setiap ukuran populasi, populasi dengan ukuran sebanyak 30 partikel memiliki rata-rata nilai *fitness* tertinggi yaitu 0,000159722, di mana semakin tinggi nilai *fitness* maka akan semakin rendah nilai objektif yang dihasilkan sebuah partikel. Tetapi dengan bertambahnya ukuran populasi maka akan menghasilkan waktu eksekusi yang lebih lama.

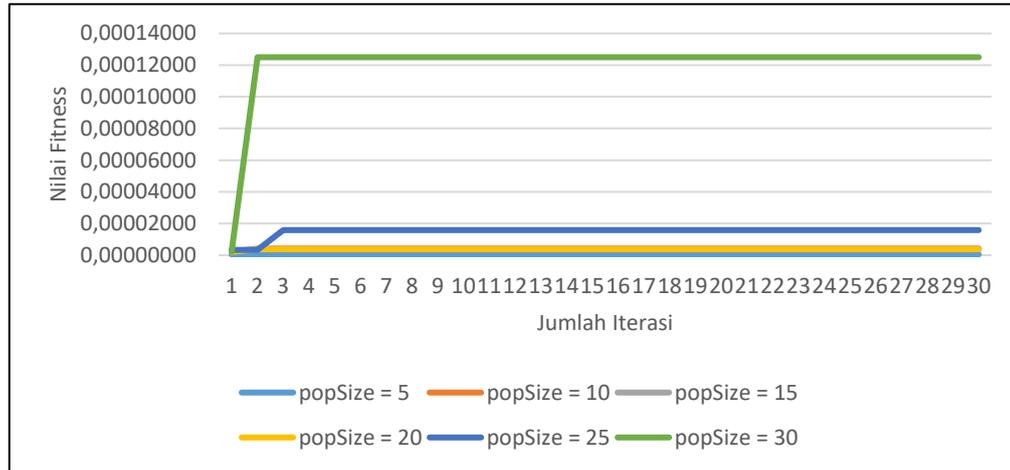


Gambar 3. 2 Grafik waktu eksekusi program komposisi data spesifikasi PC 1

Grafik pada gambar 3.2 merupakan perbandingan antara waktu eksekusi pada setiap ukuran populasi, di mana percobaan pertama dimulai dari ukuran populasi sebanyak 5 partikel, kemudian 10, 15, 20, 25, dan percobaan terakhir berupa ukuran populasi sebanyak 30 partikel. Dapat dilihat semakin tinggi ukuran partikel maka akan menghasilkan waktu eksekusi yang semakin lama.

### 3.1.2. Komposisi Data Spesifikasi PC 2

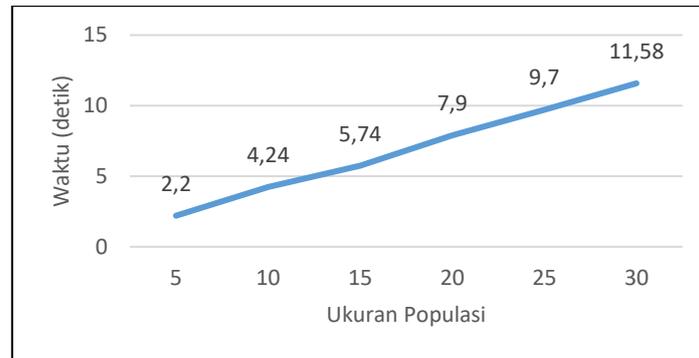
Komposisi data spesifikasi yang digunakan terdiri dari 4 model *motherboard*, 2 model CPU, 5 model GPU, 3 model *memory* (RAM), 7 model *storage* (HDD/SSD), 8 model *cpu cooler*, 6 model PSU, dan 3 model *case*. Grafik konvergensi dapat dilihat pada gambar 3.3 berikut:



Gambar 3. 3 Grafik konvergensi data komposisi spesifikasi PC 2

Berdasarkan grafik konvergensi pada gambar 3.3, nilai *fitness* yang dihasilkan jauh lebih rendah dibandingkan dengan pengujian sebelumnya, di mana pada pengujian ini, nilai *fitness* yang dihasilkan oleh ukuran populasi sebanyak 5, 10, 15, 20, dan 25 berada di bawah 0,00002000. Sama seperti pengujian sebelumnya, nilai *fitness* tertinggi terdapat pada ukuran populasi sebanyak 30 partikel, tetapi nilai *fitness* yang dihasilkan hanya mendapatkan angka 0,0001250 di mana nilai *fitness* pada pengujian sebelumnya berada pada angka 0,0001666.

Waktu eksekusi yang dihasilkan pada pengujian ini tidak jauh berbeda dari pengujian sebelumnya. Waktu eksekusi pengujian kedua dapat dilihat pada gambar 3.4 berikut:



Gambar 3. 4 Grafik waktu eksekusi program komposisi data spesifikasi PC 2

Melalui grafik waktu eksekusi pada gambar 3.14 dapat dilihat jika waktu eksekusi yang dihasilkan terus meningkat ketika ukuran populasi bertambah.

Berdasarkan nilai *fitness* yang dihasilkan dari kedua pengujian dengan komposisi data spesifikasi PC yang berbeda, dapat disimpulkan bahwa semakin banyak data model PC yang digunakan maka akan semakin tinggi pula nilai *fitness* yang dihasilkan. Hal ini dikarenakan data harga pada model PC yang bervariasi sehingga kemungkinan untuk diduplikasinya harga yang paling mendekati semakin tinggi.

Pengujian juga menghasilkan ukuran populasi optimal yang dapat digunakan pada aplikasi yaitu ukuran populasi sebanyak 30 partikel. Dengan kekurangan yaitu waktu eksekusi yang akan semakin lama.

### 3.3. Hasil Uji Validasi

Hasil uji validasi setiap responden berupa total harga dari model-model yang dipilih dan total waktu yang dibutuhkan responden tersebut dalam memilih model komponen PC. Ini bertujuan untuk mengetahui apakah aplikasi yang dibuat dapat menghasilkan rekomendasi spesifikasi PC secara cepat atau tidak. Hasil uji validasi dapat dilihat pada tabel 3.1, tabel 3.2, tabel 3.3.

**Tabel 3. 1** Hasil uji validasi responden 1

Komponen	Model Terpilih	Harga
Motherboard	MSI MAG B760M MORTAR WIFI	Rp3.980.000,00
Processor (CPU)	AMD Ryzen 7 5800X	Rp4.078.000,00
Graphic Card (GPU)	MSI GeForce RTX 3060 Ventus 2X 12G	Rp7.600.000,00
Memory (RAM)	Kingston HyperX Fury DDR4 8 GB 2666 MHz	Rp350.000,00
Storage (HDD/SDD)	Western Digital Caviar Blue	Rp700.000,00
CPU Cooler	DeePCool AK400	Rp375.000,00
PSU	Thermaltake Toughpower GF1 750W	Rp1.800.000,00
Case	Cougar MX330	Rp598.000,00
<b>Total Harga</b>		<b>Rp19.481.000,00</b>
<b>Total Waktu</b>		<b>24 detik</b>

**Tabel 3. 2** Hasil uji validasi responden 2

Komponen	Model Terpilih	Harga
Motherboard	ASRock X570 Taichi Razer Edition	Rp5.390.000,00
Processor (CPU)	Intel Core i5-12600KF	Rp4.860.000,00
Graphic Card (GPU)	Asus DUAL OC GeForce GTX 1650	Rp2.605.000,00
Memory (RAM)	Kingston FURY Beast 16 GB	Rp1.039.000,00
Storage (HDD/SDD)	Samsung 970 Evo Plus	Rp1.200.000,00
CPU Cooler	Noctua NH-U14S	Rp1.285.000,00
PSU	MSI MPG A750GF	Rp1.700.000,00
Case	NZXT H5 Flow	Rp1.500.000,00
<b>Total Harga</b>		<b>Rp19.579.000,00</b>
<b>Total Waktu</b>		<b>2 menit 22 detik</b>

**Tabel 3. 3** Hasil uji validasi responden 3

Komponen	Model Terpilih	Harga
Motherboard	ASRock X570 Taichi Razer Edition	Rp5.390.000,00
Processor (CPU)	Intel Core i5-12600KF	Rp4.860.000,00
Graphic Card (GPU)	Asus DUAL OC GeForce GTX 1650	Rp2.605.000,00
Memory (RAM)	TEAMGROUP T-Force Delta TUF RGB 16 GB	Rp795.000,00
Storage (HDD/SDD)	Seagate Barracuda	Rp1.500.000,00
CPU Cooler	ARCTIC Freezer 34 eSports DUO	Rp745.000,00
PSU	Corsair RM850	Rp2.600.000,00
Case	Lian Li Lancool One	Rp1.280.000,00
<b>Total Harga</b>		<b>Rp19.775.000,00</b>
<b>Total Waktu</b>		<b>1 menit 19 detik</b>

Berdasarkan hasil yang didapatkan dari responden, apabila setiap total harga yang dihasilkan responden diuji menggunakan fungsi objektif dan fungsi *fitness*, maka akan didapatkan hasil sebagai berikut :

**Tabel 3. 4** Perhitungan manual fungsi objektif responden

	Fungsi Objektif	Fungsi Objektif BPSO
Responden 1	$20.000.000 - 19.481.000 = 519.000$	7.000
Responden 2	$20.000.000 - 19.579.000 = 421.000$	51.000
Responden 3	$20.000.000 - 19.775.000 = 225.000$	40.890

**Tabel 3. 5** Perhitungan manual fungsi *fitness* responden

	Fungsi <i>Fitness</i>	Fungsi <i>Fitness</i> BPSO
Responden 1	$\frac{1}{(519.000 + 0,001)} = \frac{1}{519.000,001} = 0,00000192678$	0,000142857
Responden 2	$\frac{1}{(421.000 + 0,001)} = \frac{1}{421.000,001} = 0,00000237530$	0,0000196078
Responden 3	$\frac{1}{(225.000 + 0,001)} = \frac{1}{225.000,001} = 0,00000444444$	0,0000244558

Jika dibandingkan dengan nilai *fitness* yang dihasilkan berdasarkan eksperimen, maka nilai *fitness* dengan algoritma BPSO mempunyai hasil lebih tinggi dibandingkan nilai *fitness* yang dihasilkan oleh responden. Waktu eksekusi yang diperoleh dengan algoritma BPSO juga lebih cepat dibandingkan dengan total waktu yang diperoleh oleh responden.

#### 4. Kesimpulan

1. Algoritma BPSO (*Binary Particle Swarm Optimization*) berhasil diterapkan pada aplikasi dan dapat menghasilkan solusi yang optimal, di mana total harga dari spesifikasi PC yang direkomendasikan mendekati dengan kemampuan finansial yang telah diinputkan oleh *user*.
2. Semakin banyak data model komponen PC yang digunakan maka hasil rekomendasi yang dihasilkan akan semakin optimal.
3. Ukuran populasi sebanyak 30 partikel menghasilkan nilai *fitness* tertinggi di antara ukuran populasi sebelumnya yaitu 5, 10, 15, 20, dan 25 partikel. Dengan konsekuensi, ukuran populasi yang tinggi akan membuat waktu pencarian solusi lebih lama.

---

Daftar Pustaka

- [1] M. Michael and W. Winarno, "Design and Development of Computer Specification Recommendation System Based on User Budget With Genetic Algorithm," *Int. J. New Media Technol.*, vol. 5, no. 1, pp. 25–29, 2018, doi: 10.31937/ijnmt.v5i1.814.
- [2] R. H. Kiswanto, "Spesifikasi Komputer Rakitan Berdasarkan Kebutuhan dan Anggaran Menggunakan Algoritma Backtracking," *J. Eksplora Inform.*, vol. 10, no. 1, pp. 1–12, 2020, doi: 10.30864/eksplora.v10i1.358.
- [3] J. C. W. Lin, L. Yang, P. Fournier-Viger, T. P. Hong, and M. Voznak, "A binary PSO approach to mine high-utility itemsets," *Soft Comput.*, vol. 21, no. 17, pp. 5103–5121, 2017, doi: 10.1007/s00500-016-2106-1.
- [4] J. Kennedy and R. C. Eberhart, "Discrete binary version of the particle swarm algorithm," *Proc. IEEE Int. Conf. Syst. Man Cybern.*, vol. 5, pp. 4104–4108, 1997, doi: 10.1109/icsmc.1997.637339.
- [5] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, 1995, vol. 4, pp. 1942–1948. doi: 10.1109/ICNN.1995.488968.
- [6] Y.-J. Gong *et al.*, "Genetic Learning Particle Swarm Optimization," *IEEE Trans. Cybern.*, vol. 46, no. 10, pp. 2277–2290, Oct. 2016, doi: 10.1109/TCYB.2015.2475174.
- [7] Wei-Neng Chen, Jun Zhang, H. S. H. Chung, Wen-Liang Zhong, Wei-Gang Wu, and Yu-hui Shi, "A Novel Set-Based Particle Swarm Optimization Method for Discrete Optimization Problems," *IEEE Trans. Evol. Comput.*, vol. 14, no. 2, pp. 278–300, Apr. 2010, doi: 10.1109/TEVC.2009.2030331.
- [8] M. A. Khanesar, M. Teshnehlab, and M. A. Shoorehdeli, "A Novel Binary Particle Swarm Optimization," *Mediterr. Conf. Control Autom.*, vol. 1, no. 1, 2007.