SITEKIN: Jurnal Sains, Teknologi dan Industri, Vol. 21, No. 2, June 2024 pp.216 - 227
ISSN 2407-0939 print/ISSN 2721-2041 online

Comparison Genetics Algorithm and Particle Swarm Optimization in Dietary Recommendations for Maternal Nutritional Fulfillment

Diva Kurnianingtyas[1], Nathan Daud[2], Indriati[3], Lailil Muflikhah[4]
[1,2,3,4] Department of Informatics Engineering, Faculty of Computer Science, Universitas Brawijaya
Jl. Veteran No.10-11, Malang, East Java, 65145, Indonesia
Email: divaku@ub.ac.id

ABSTRACT

Fulfilling maternal nutrition is an NP-hard problem. Optimization techniques are required to solve its complexity. This issue is crucial as it affects the number of stunted toddlers in Indonesia. Stunting begins in the womb due to inadequate maternal nutrition during pregnancy. Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are optimization methods applied to NP-hard problems, including medicine. Their performance has not been compared in this field. This study aims to identify an alternative method for recommending daily menus based on maternal nutritional needs. There are 55 food ingredients used to fulfill five menu parts: staple food (SF), vegetables (VG), plant source food (PS), animal source food (AS), and complementary (CP). Nutritional adequacy for prenatal is determined by Total Energy Expenditure (TEE) based on basal energy, daily activity, and stress levels. Results show PSO outperforms GA in average fitness values, 30.45 to 102.51, while GA excels in execution time, 0.33 to 23.22 seconds. PSO is preferred for effectiveness, and GA for efficiency, but given the problem's urgency, PSO is recommended. Exploring other metaheuristic methods is advised to enhance menu recommendation solutions for maternal nutrition. Additionally, expanding the food database is necessary for more varied maternal menu to support stunting prevention.

Keywords: *Evolutionary Algorithm, Food Menu, Metaheuristics, Pregnancy, Swarm Intelligence.*

Introduction

NP-hard problems require increasing computational effort in line with the increasing complexity of the problem scope. One of the case studies in the health sector, including NP-hard, is the fulfillment of patient nutrition. Stunting is a nutrition problem that concerns the government and the public because its prevalence is still relatively high, reaching 21.6% by 2022 [1]. Based on these data, stunting in Indonesia is still classified as chronic. Stunting begins to occur when the fetus is still in the womb due to the mother's food intake during pregnancy, which is less nutritious. As a result, the nutrition obtained by children in the womb is insufficient. Malnutrition will inhibit the baby's growth and can continue after birth. Nutritional intake of maternal is one of the essential things to do to prevent stunting in children [2], [3].

Metaheuristic methods can solve NP-hard problems. The development of metaheuristic methods is rapid because the need for optimization in various fields is increasing, thus providing opportunities for researchers to develop metaheuristic methods continuously. However, although metaheuristic methods continue to develop, Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are still the leading methods of researchers because they are easy to understand and implement in solving NP-hard problems [4]. In its application, GA uses the concept of evolution by performing natural selection and natural genetics to find the optimal solution [5]. Meanwhile, PSO mimics the concept of the nature and behavior of a group of living things to obtain an optimal solution [6].

GA and PSO have been widely implemented in the medical sector to fulfill patient nutrition needs. Researchers [7] successfully found low-cost food recommendations for the elderly. Later, researchers [8] found recommendations for food combinations to increase immunity and early prevention of contracting Covid19 in young adults. The results are that the cost of expenses can save up to 33% per day for the average male and 42% per day for the average female. GA was also implemented to optimize food ingredients' nutrition for children's growth and development by [9]. Based on the results, GA can find the difference in food prices of IDR 37,722 for male patients and IDR 32,040 for female patients. The problem in the study was also carried out by [10] but using PSO to obtain 90% accuracy. In addition, GA [11] and PSO [12] were implemented to find food menus based on the number of calories and their content. Furthermore, researchers [13] found food recommendations for pregnant women with GA. Based on the exposure of several researchers, GA and PSO have indeed been widely implemented but have yet to be compared. The selection of methods affects the results of the menu recommendations obtained.

Based on the previous explanation, a study is needed to find a daily menu recommendation based on the maternal nutritional intake by the best performance algorithm. This study aims to find an alternative method to recommend a daily menu for pregnant women. GA and PSO metaheuristic models are implemented to solve this problem.

This study is divided into sections. Section 2 describes problems determining the daily maternal menu—furthermore, Sections 3 and 4 focus on explaining the design of GA and PSO. The following section (Section 5) shows the results of GA and PSO implementation and is then discussed. Finally, Section 6 explains the conclusions and limitations for future studies.

Problem Description

Pregnant women need adequate nutrition to prevent stunting. Nutrition can be fulfilled by completing five parts in each food menu: staple food (SF), vegetables (VG), plant source food (PS), animal source food (AS), and complementary (CP). The dataset used is obtained from [14], which contains 55 food ingredients consisting of 11 per section. Nutritional adequacy criteria for pregnant women can be determined by calculating Total Energy Expenditure (TEE) as a calculation of individual calorie needs based on basal energy needs, daily activities, and stress levels. The characteristics of the activity and stress factors are adjusted to the mother's condition, as shown in Table 1. Then, the calculation of TEE is shown in Equation (1).

$$\begin{gathered} BEE = 655 + (9.6 \times weight) + (1.85 \times height) - (4.68 \times age) \\ TEE = BEE \times Activity\ factor \times Stress\ factor \end{gathered} \tag{1}$$

Which is:
BEE = Basal Energy Expenditure
TEE = Total Energy Expenditure

Table 1. Activity and Stress Factor Characteristics

Activity		Stress	
Type of activity	Rate	Stress level	Rate
Bed rest	1.1	No stress, normal nutritional status	1.0-1.1
Bed rest, but limited mobility	1.2	Mild stress	1.2-1.4
Get out of bed	1.3	Moderate stress	1.4-1.5
Moderate activity	1.4-1.5	Severe stress	1.5-1.6
Strenuous activity	1.75	Extremely severe stress	1.7

In this study, GA and PSO were implemented using the same dataset to calculate fitness values shown in Equation (2) The chromosomes obtained by GA and the particles obtained by PSO are compared for fitness value to find the best performance. The function that determines how good the solution is called fitness. The implementation stage of GA and PSO to determine the daily intake of nutrients in pregnant women is shown in Figure 1.

$$\begin{gathered} \sum Penalty = |Energy\ Needs - Calories| + |Carbohydrates\ Need - Carbohydrates| \\ + |Protein\ Need - Protein|\,|Fat\ Need - Fat| \\ fitness = \frac{C}{\Sigma Penalty} \end{gathered} \tag{2}$$

Which is:
$C = Constanta(1000)$

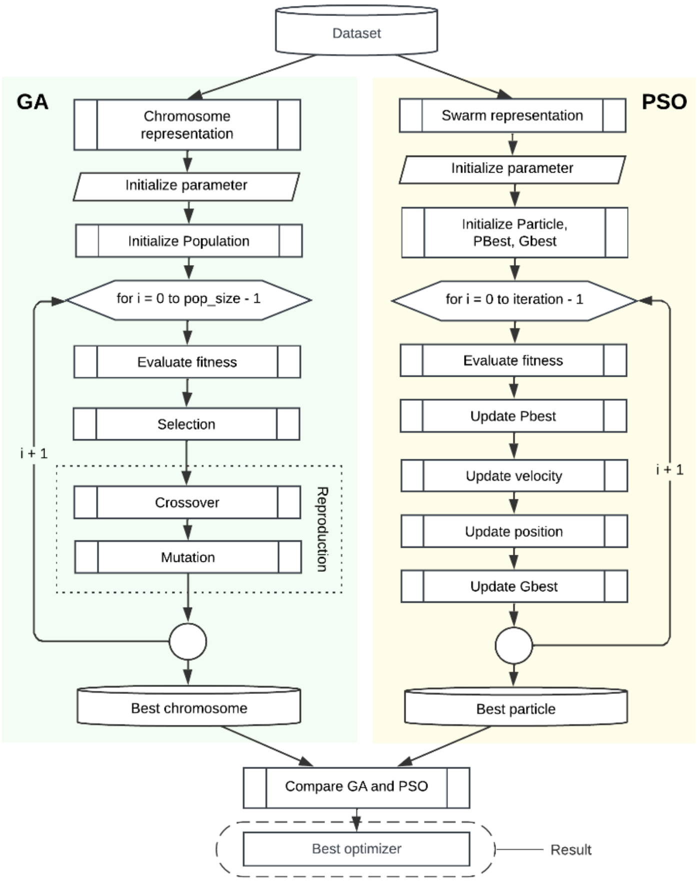

Figure 1. Problem Description

Genetics Algorithm

GA is a method for solving complex problems based on the principle of genetic selection [15]. The success rate in generating individuals is directly proportional to the solution outcome represented by the fitness value. This success can ensure that the next generation's quality will improve [16]. There are four stages in implementing GA: chromosome representation, reproduction, evaluation, and selection. The GA for this study is explained in detail as follows.

GA parameter

One strategy to optimize GA performance is to determine the parameters correctly. The parameters used are population size, generation size, crossover rate (Cr), and mutation rate (Mr). Population size is the number of individuals in a population that influences the search for space exploration. Furthermore, the generation size represents the population's number of iterations or evolutionary cycles to determine how long the algorithm should run. In addition, the correct values of Cr and Mr can influence how effectively the algorithm can explore and utilize variations in a population. The initialization of GA parameters in this study is shown in Figure 2.

Number of population	Number of generation	Cr	Mr
10	20	0.5	0.5

Figure 2. GA parameter

Chromosome representation

Formulating the problem solution into a form the GA can process is called the chromosome representation [17]. In this study, genes are represented by randomly generated integers 1-15. Each gene represents a type of food ingredient. One chromosome consists of 15 genes representing the daily morning, afternoon, and evening diet. The combination of genes in this chromosome forms a solution to meet the nutritional needs of pregnant women. An example of chromosome representation is shown in Figure 3.

Breakfast					Lunch					Dinner				
SF	PS	AS	VG	CP	SF	PS	AS	VG	CP	SF	PS	AS	VG	CP
6	11	6	2	4	8	7	6	5	7	6	8	4	7	3

Which is:
SF = Staple food
PS = Plant source food
AS = Animal source food
VG = Vegetables
CP = Complimentary

Figure 3. Chromosome representation

Crossover and Mutation initialization

The process of generating offspring in the population is done through two stages: crossover and mutation. The crossover stage involves two parents randomly selected from the population to produce a new chromosome [18]. The application of the two-point crossover technique is used to perform genetic mixing. This technique involves using two random individual chromosomes and determining two random gene positions as cut points. After determining the cut points, the chromosome segment between the first and second cut points will be swapped between the two randomly selected individuals [19]. Meanwhile, mutation generates new individuals in a randomized manner [20]. The mutation used in this study is insertion mutation. The segment to be mutated is determined randomly on the chromosome. One gene is randomly selected, and then the selected gene will be inserted into a randomly selected position on the chromosome. An illustration of the crossover and mutation process in this study is represented in Figure 4.

Before crossover

P1	6	11	6	2	4	8	7	6	5	7	6	8	4	7	3
P2	3	8	11	2	4	3	7	11	7	7	9	6	4	2	7

After crossover

C1	6	11	6	2	4	3	7	11	7	7	9	6	4	7	3
C2	3	8	11	2	4	8	7	6	5	7	6	8	4	2	7

Before mutation

						x				y					
C1	6	11	6	2	4	3	7	11	7	7	9	6	4	7	3

After mutation

						x				y					
C1	6	11	6	2	4	7	3	7	11	7	6	8	4	7	3

Figure 4. Crossover and mutation initialization

GA fitness calculation

The evaluation of a population is used to check how well each individual performs. The value of the fitness function (see Equation (2)) significantly impacts the performance of the GA. Here is an example of calculating each gene to find the fitness in individual P1 in Figure 4.

$$Calories = 696 + 299.0 + 153.6 + 176 + 87.0 + 124 + 93.5 + 153.6 + 58 + 73.5 + 696 + 110.0 + 138.4 + 54 + 45.0 = 2957.6\ kcal$$

$$Charbohydrates = 164.2 + 4.15 + 10.16 + 28.0 + 22.35 + 27.0 + 11.3 + 10.16 + 10.0 + 17.7 + 164.2 + 4.0 + 26.72 + 2.0 + 10.2 = 512.14\ grams$$

$$Proteins = 9.4 + 21.9 + 10.16 + 6.4 + 0.45 + 4.2 + 6.5 + 10.16 + 1.4 + 0.9 + 9.4 + 5.3 + 5.76 + 4.0 + 0.75 = 96.68\ grams$$

$$Fat = 0.2 + 21.3 + 8.08 + 4.2 + 0.6 + 0.4 + 3.0 + 8.08 + 1.2 + 0.3 + 0.2 + 9.0 + 0.96 + 2.6 + 0.3 = 60.42\ grams$$

$$fitness = \frac{1000}{|2249.4046 - 2957.6| + |377.411 - 512.14| + |94.352 - 96.68| + |64.783 - 60.42|} = \frac{1000}{849.6155} = 1.177$$

SITEKIN: Jurnal Sains, Teknologi dan Industri, Vol. 21, No. 2, June 2024 pp.216 - 227
ISSN 2407-0939 print/ISSN 2721-2041 online

Selection initialization

In creating a new generation, it is necessary to form a new chromosome by selecting individuals from the parent population based on their fitness value. Fitness value plays an essential role in determining the quality of the resulting solution. The selection process in this study uses the elitism selection method. This process focuses on selecting individuals with the highest fitness level from a predetermined population. The following illustration of elitism selection is shown in Figure 5.

Parent	Breakfast					Lunch					Dinner					Fitness
	MP	SN	SH	SY	PLK	MP	SN	SH	SY	PLK	MP	SN	SH	SY	PLK	
P1	6	11	6	2	4	8	7	6	5	7	6	8	4	7	3	1.177
P2	3	8	11	2	4	3	7	11	7	7	9	6	4	2	7	5.63631
C1	6	11	6	2	4	7	3	7	11	7	6	8	4	7	3	0.51228
C2	3	8	11	2	4	8	7	6	4	5	7	6	8	2	7	2.115533

Figure 5. Selection initialization

GA results

The implementation of GA in this study is used to find a daily menu according to the nutritional needs of pregnant women. The following daily menu solution obtained from GA with the highest fitness of 68.27 is shown in Table 2.

Table 2. GA results

Mealtime	Gen number	Meal code	Meal name	Meal weight	Calories	Carbohydrates	Proteins	Fat
Breakfast	1	SF	White rice	200 grams	360 kcal	79.6 grams	6 grams	0.6 grams
	2	PS	Boiled soybeans	50 grams	95 kcal	6.35 grams	10.1 grams	4.1 grams
	9	AS	Carp pepes	80 grams	167 kcal	9.44 grams	12.2 grams	9 grams
	5	VG	Sour vegetable soup	200 grams	58 kcal	10 grams	1.4 grams	1.2 grams
	9	CP	Sweet orange	150 grams	68 kcal	16.8 grams	1.35 grams	0.3 grams
Lunch	10	SF	Boiled maize	200 grams	682 kcal	170 grams	0.6 grams	0 grams
	9	PS	Steamed mung beans	50 grams	55 kcal	9.15 grams	4.35 grams	0.3 grams
	10	AS	Fried eel	80 grams	334 kcal	25.6 grams	20.7 grams	16 grams
	6	VG	Mixed vegetables	200 grams	194 kcal	8.4 grams	11.6 grams	13 grams
	9	CP	Sweet orange	150 grams	68 kcal	16.8 grams	1.35 grams	0.3 grams
Dinner	9	SF	Boiled macaroni	200 grams	706 kcal	157.4 grams	17.4 grams	0.8 grams
	10	PS	Tempe crackers	50 grams	291 kcal	20.85 grams	6.05 grams	20 grams
	9	AS	Carp pepes	80 grams	167 kcal	9.44 grams	12.2 grams	9 grams
	9	VG	Jengkol stew	200 grams	424 kcal	58.2 grams	12 grams	20 grams
	10	CP	Mango	150 grams	78 kcal	18.45 grams	1.05 grams	0 grams

Particle Swarm Optimization

PSO is a computational method to solve a problem by improving candidate solutions iteratively. PSO is one of the simple bio-inspired algorithms for finding optimal solutions [21]. In the PSO algorithm, individuals are particles that fly through the search space, looking for the best position globally [22]. A set of particles keeps moving towards promising areas until it gets a global optimum to solve the optimization problem [23]. The steps of PSO are described in detail as follows.

PSO parameter

In PSO studies, several parameters are crucial to optimizing the algorithm's performance. The number of particles affects the algorithm's ability to explore the search space, potentially covering a larger area [24]. The number of iterations is a parameter that determines the duration of the optimization process. Cognitive and social parameters are often represented as c_1 and c_2 influencing how particles are affected by the personal best ($pBest$) and global best ($gBest$) positions found by a set of particles. The inertia weight (w) is another crucial factor that helps control the balance between exploring the search space and exploiting the best solution. Proper adjustments to these parameters can significantly affect the efficiency and effectiveness of the PSO in finding optimal solutions [25]. The following initialization of the PSO parameters used is shown in Figure 6.

Number of particles	Itermax	C_1	C_2	ω
100	20	1.4	1.4	0.5

Figure 6. PSO parameter

Particle initialization

Particle initialization is the first step in PSO. In this stage, each particle in the population is randomly assigned an initial position and initial velocity in the search space. The position and velocity will be updated throughout the iterations based on the particle's knowledge of the best position ever encountered ($pBest$) and the best position ever encountered by all particles ($gBest$). Proper particle initialization is very important because it can affect the ability of the PSO algorithm to find the optimal solution [26]. An illustration of the representation of each particle is shown in Figure 7.

	SF	PS	AS	VG	CP	SF	PS	AS	VG	CP	SF	PS	AS	VG	CP	Fitness
X1	7	6	5	1	11	10	7	7	11	2	9	2	5	1	11	0.4327
X2	11	6	9	1	1	3	2	1	8	4	2	8	3	3	5	1.55231

Which is:
SF = Staple food
PS = Plant source food
AS = Animal source food
VG = Vegetables
CP = Complimentary

Figure 7. Particle initialization

PSO fitness calculation

The assessment of a group is done by evaluating the extent of each member's performance. The fitness function determines how good an individual is and significantly influences PSO performance. The fitness function implemented in PSO is shown in Equation (2)—the following example of calculating fitness in particle x_1 in Figure 7.

$$Calories = 916 + 69 + 151.2 + 282 + 117 + 682 + 93.5 + 228.8 + 150 + 127.5 + 706 + 94.5 + 151.2 + 282 + 117 = 4167.7\ kcal$$

$$Charbohydrates = 150.2 + 1.3 + 0.48 + 25.2 + 30.0 + 170.0 + 11.3 + 1.84 + 20.0 + 11.55 + 157.4 + 6.35 + 0.48 + 25.2 + 30 = 641.3\ grams$$

$$Proteins = 13.8 + 5.35 + 9.04 + 30.6 + 1.2 + 0.6 + 6.5 + 38.8 + 4.6 + 1.35 + 17.4 + 10.1 + 9.04 + 30.6 + 1.2 = 180.18\ grams$$

$$Fat = 28.8 + 0.55 + 12.56 + 6.6 + 0.6 + 0.0 + 3.0 + 7.36 + 5.6 + 9.75 + 0.8 + 4.1 + 12.56 + 6.6 + 0.6 = 99.48\ grams$$

$$fitness = \frac{1000}{|2249.4046 - 4167.7| + |377.411 - 641.3| + |94.352 - 180.18| + |64.783 - 99.48|} = \frac{1000}{2302.7086} = 0.4327$$

$pBest$ and $gBest$ initialization

The parameters $pBest$ and $gBest$ represent the best solution the individual particles and the group have found. $pBest$ indicates that each particle is set to the starting position because the starting position is the crucial solution that the particle has found. $gBest$ is set to $pBest$ when it has the best fitness among all particles. In other words, gBest is the best solution for the particle swarm found [27]. The initialization of $pBest$ and $gBest$ values is shown in Figure 8.

	SF	PS	AS	VG	CP	SF	PS	AS	VG	CP	SF	PS	AS	VG	CP	Fitness
pBest 1	7	6	5	1	11	10	7	7	11	2	9	2	5	1	11	0.4327
pBest 2	11	6	9	1	1	3	2	1	8	4	2	8	3	3	5	1.55231

	SF	PS	AS	VG	CP	SF	PS	AS	VG	CP	SF	PS	AS	VG	CP	Fitness
gBest	11	6	9	1	1	3	2	1	8	4	2	8	3	3	5	1.55231

Figure 8. pBest and gBest initialization

Velocity update

Velocity update is one of the critical aspects of PSO. Each particle in the swarm has a position and velocity that determine movement through the search space to find the optimal solution. This update combines information from $pBest$ and $gBest$ [28]. The calculation of the velocity update is shown in Equation (3).

$$V_i(t+1) = w \times v_i(t) + c_1 \times r_i \times \left(pBest_i - x_i(t)\right) + c_2 \times r_2 \times \left(gBest - x_i(t)\right) \tag{3}$$

which is:
$v_i(t+1)$: The particle speed in the next iteration
w : The inertia weight that controls the impact of the previous velocity on the updated velocity.
c_1, c_2 : The acceleration coefficient that regulates the influence of pBest and gBest
r_1, r_2 : Random number within a range
$Pbest_i$: The best position found by particle i.
$Gbest$: The best position found by the entire swarm.
$x_i(t)$: It is the current position of particle i at iteration t.

Next, a representation of the particle's initial velocity change is shown in Figure 9.

Iteration = 1															
V1	1.01	0.46	2.44	0.08	-12.18	-2.03	-1.33	-5.23	-1.10	0.67	-5.92	2.25	-1.84	2.16	-3.23
V2	0.45	0.43	0.25	0.36	0.03	0.04	0.25	0.17	0.41	0.3	0.35	0.46	0.42	0.13	0.15

Figure 9. Velocity update

Position update and Fitness calculation

Once the particle velocity is updated, the particle position must also be updated. The position update allows the particle to move through the search space based on the updated velocity. The particle position update is calculated using Equation (4).

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{4}$$

Which is:
$x_i(t+1)$: The position of particle i in the next iteration (t+1)
$x_i(t)$: The current position of particle i at iteration t
$v_i(t+1)$: The velocity of particle i in the next iteration (t+1)

The updated particle position will be calculated as a fitness function according to Equation (2). The fitness value determines how good the solution represented by the particle position is [29]. The new position and fitness value of the particle are shown in Figure 10.

Iteration = 1																Fitness
X1	8.01	6.46	7.44	1.08	-1.18	7.97	5.67	1.77	9.90	2.67	3.08	4.25	3.16	3.16	7.77	3.1653
X2	11.45	6.43	9.25	1.36	1.03	3.04	2.25	1.17	8.41	4.30	2.35	8.46	3.42	3.13	5.15	1.5523

Figure 10. Position update.

$pBest$ and $gBest$ update

In addition to updating the particle position, $pBest$ and $gBest$ must be updated. If the particle's fitness value is better than the $pBest$ fitness value, the $pBest$ is updated. If the particle fitness value is better than the $gBest$ fitness value, then the $gBest$ is updated. The particle update is used to remember the best solution found and provide a reference in the movement of particles in the search space [30]. The updated $pBest$ and $gBest$ are shown in Figure 11.

	Iteration = 1														Fitness
pBest1	8	6	7	1	1	8	6	2	10	3	3	4	3	3	8	3.1653
pBest2	11	6	9	1	1	3	2	1	8	4	2	8	3	3	5	1.5523

	Iteration = 1														Fitness
gBest	8	6	7	1	1	8	6	2	10	3	3	4	3	3	8	3.1653

Figure 11. $pBest$ and $gBest$ update

PSO result

PSO is implemented in this study to find a daily menu according to the nutritional needs of pregnant women. The following daily menu solution obtained from PSO with the highest fitness of 188.75 is shown in Table 3.

Table 3. PSO results

Mealtime	Dimension number	Meal code	Meal name	Meal weight	Calories	Carbohydrates	Proteins	Fat
Breakfast	3	SF	Steamed rice	200 grams	240 kcal	52 grams	5 grams	0.8 grams
	7	PS	Fermented soybean cake	50 grams	94 kcal	11.3 grams	7 grams	3 grams
	6	AS	Sweet and sour gourami fish	80 grams	154 kcal	10.16 grams	10 grams	8.1 grams
	2	VG	Fried bean sprouts	200 grams	176 kcal	28 grams	6 grams	4.2 grams
	4	CP	Apple	150 grams	87 kcal	22.35 grams	0 grams	0.6 grams
Lunch	5	SF	Black glutinous rice	200 grams	362 kcal	74.6 grams	8 grams	2.4 grams
	6	PS	Boiled black-eyed peas	50 grams	69 kcal	1.3 grams	5 grams	0.6 grams
	3	AS	Fried milkfish	80 grams	98 kcal	0 grams	16 grams	3.8 grams
	5	VG	Sour vegetable soup	200 grams	58 kcal	10 grams	1 grams	1.2 grams
	2	CP	Avocado	150 grams	128 kcal	11.55 grams	1 grams	9.8 grams
Dinner	8	SF	Boiled potatoes	200 grams	124 kcal	27 grams	4 grams	0.4 grams
	7	PS	Fermented soybean cake	50 grams	94 kcal	11.3 grams	7 grams	3 grams
	6	AS	Sweet and sour gourami fish	80 grams	154 kcal	10.16 grams	10 grams	8.1 grams
	4	VG	Papaya flower vegetable	200 grams	98 kcal	19.6 grams	3 grams	0.6 grams
	2	CP	Avocado	150 grams	128 kcal	11.55 grams	1 grams	9.8 grams

Results and Discussion

Parameter tuning

One factor that affects an algorithm's performance is the parameters. The algorithm with the best parameters will perform better than other parameter combinations. In this study, tests were conducted on GA and PSO parameters to improve the solution results, which have been shown in Figure 12 and Figure 13, respectively. GA parameters consist of population size, generation size, Cr value, and Mr value. Testing was conducted three times, namely testing population size, generation size, and a combination of Cr and Mr. Initial testing of population size was determined by ten populations and repeated in multiples of 10 to 130 populations. Initial testing of generation size was determined at 20 generations and repeated up to 220 generations. Furthermore, nine combinations were used to test the combination of Cr and Mr. The results of each test of the best GA parameters were obtained from an average of 10 trials. Based on the Figure 12, the best parameter values of population size, generation size, Cr, and Mr are 90, 160, 0.1, and 0.9, respectively.

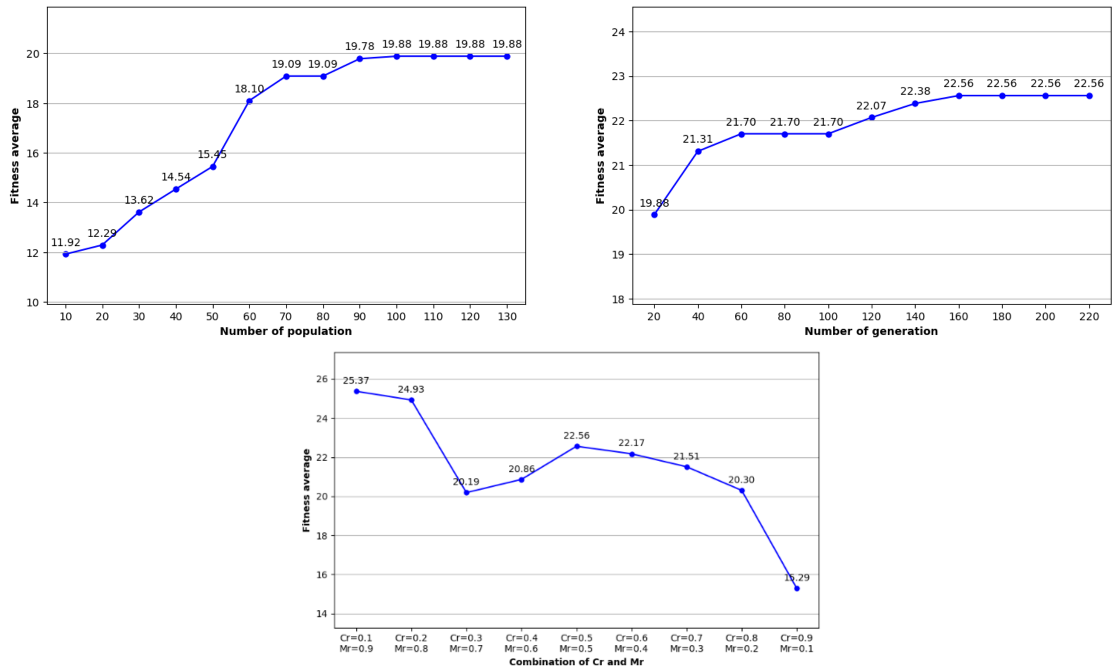

Figure 12. GA parameter tuning

PSO has parameters for the number of particles and iterations, w, c_1, and c_2. PSO parameter testing was conducted three times, namely testing the number of particles, the number of iterations, and the combination of w, c_1, and c_2 values. Initial testing of the number of particles was determined to be ten particles and repeated in multiples of 10 to 130 particles. The initial test of the number of iterations was set at 20 iterations and repeated up to 220 iterations. Furthermore, ten combinations are used to test the combination of w, c_1, and c_2 values. The results of each test of the best PSO parameters are obtained from an average of 10 trials. Based on the Figure 13, the best parameter values of particles, several iterations, w, c_1, and c_2, are 130, 220, 0.5, 1.4, and 1.4, respectively.

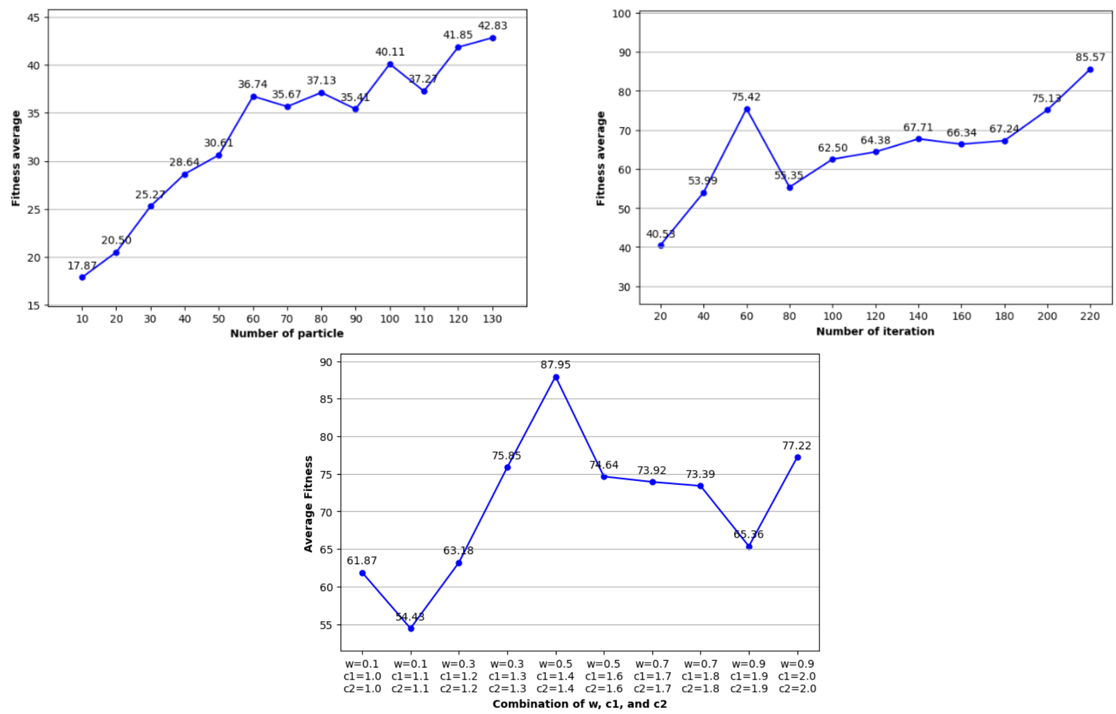

Figure 13. PSO parameter tuning

Method comparation

The parameter tuning results were used to improve the performance of GA and PSO in obtaining a daily menu to meet the nutritional needs of pregnant women. Based on the results (see Table 4), GA and PSO were run five times with the best parameters, obtaining an average fitness value of 30.45 and 102.51, respectively. PSO obtained an average fitness value approximately three times higher than GA. This reason is because PSO usually converges by finding the global optimum point. This PSO characteristic differs from GA, which can only find solutions at local optimum points or arbitrary points rather than global optimum points. This statement has been proven in studies [31], [32]. However, judging from the average execution time value, GA obtained a lower value than PSO, which was 0.33 and 23.22, respectively. Although the effectiveness of PSO is high, PSO has a low convergence rate, so it requires execution time that tends to be longer than GA. These results have been proven in [33]. Based on the problem's urgency, PSO is more recommended than GA because the effectiveness rate tends to improve.

Table 4. Performance comparison of PSO and GA

Experiment	Fitness		Execution time (s)	
	GA	PSO	GA	PSO
1	21.4582036	119.410094	0.34458709	20.1431885
2	68.2739324	188.758438	0.3261373	22.3740525
3	31.3040988	88.5673414	0.34987569	24.7962832
4	12.6285749	54.7946515	0.33823681	24.7904711
5	18.596962	61.0325539	0.31342983	24.0167296
Average	30.4523543	102.512616	0.33445334	23.224145

Conclusion

GA and PSO are metaheuristic methods that can find solutions to NP-hard problems. In this study, GA and PSO are implemented to address the problem of fulfilling maternal nutritional needs to obtain a daily menu. The best parameters used by GA include population size, generation size, Cr, and Mr, which are 90, 160, 0.1, and 0.9, respectively. Afterwards, the best PSO parameters implemented include the number of particles and the number of iterations, w, c_1, and c_2 in the order of 130, 220, 0.5, 1.4, and 1.4. Based on the average fitness value results, PSO performance is better than GA, with values of 30.45 and 102.51, respectively. PSO characteristics that always try to find the global optimum point prove that the effectiveness of PSO is better than GA. In contrast to the comparison of execution time, GA performance is better than PSO with sequential values of 0.33 and 23.22. Although the effectiveness of PSO is high, PSO has a low convergence rate, so it requires execution time that tends to be longer than GA. The findings obtained from this study are that PSO tends to be suitable for effectiveness problems, and GA is used for efficiency problems. However, based on the problem's urgency, PSO is more recommended than GA. Applying metaheuristic methods other than GA and PSO is recommended to improve the daily menu recommendation solution results for maternal. In addition, future studies also need to add more food data to find a more varied maternal menu to support stunting prevention. The current study has provided valuable insights into the application of GA and PSO for optimizing maternal nutrition through daily menu planning, there is a clear indication that further exploration of alternative metaheuristic methods is warranted. The incorporation of additional metaheuristic approaches could potentially enhance the performance of nutritional recommendation systems in future research.

Acknowledge

This study was supported by a grant from Universitas Brawijaya with Number 611.76/UN10.C200/2023.

References

[1] F. Ali, "Angka Stunting Tahun 2022 Turun Menjadi 21, 6 Persen." 2023.

[2] U. Ramlah, "Gangguan kesehatan pada anak usia dini akibat kekurangan gizi dan upaya pencegahannya," *Ana'Bulava J. Pendidik. Anak*, vol. 2, no. 2, pp. 12–25, 2021.

[3] F. A. Sari *et al.*, "Peran Mahasiswa Kukerta dalam Meningkatkan Kesadaran Gizi untuk Mencegah

Stunting," *J. Pendidik. Tambusai*, vol. 7, no. 3, pp. 233–239, 2023.

[4] X. Fu, Y. Sun, H. Wang, and H. Li, "Task scheduling of cloud computing based on hybrid particle swarm algorithm and genetic algorithm," *Cluster Comput.*, vol. 26, no. 5, pp. 2479–2488, 2023.

[5] Y. E. A. Seputra, "Prediksi Pergerakan Harga Bitcoin menggunakan Algoritma Genetika," *J. Soc. Sustain. Manag.*, vol. 3, no. 2, pp. 22–36, 2023.

[6] A. W. Widodo, D. Kurnianingtyas, and W. F. Mahmudy, "Optimization of Healthcare Problem using Swarm Intelligence: A Review," in *2022 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, IEEE, 2022, pp. 747–751.

[7] A. Ramadhan, I. Cholissodin, and L. Muflikhah, "Optimasi Gizi pada Lanjut Usia dengan Alergi untuk Meningkatkan Kualitas Hidup menggunakan Algoritma Genetika," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 4, pp. 1709–1716, 2023.

[8] A. N. Sari, I. Cholissodin, and B. Rahayudi, "Optimasi Kombinasi Bahan Makanan untuk Meningkatkan Imunitas dan Pencegahan Dini Tertular Covid-19 pada Usia Dewasa Muda menggunakan Algoritma Genetika," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 11, pp. 4869–4877, 2021.

[9] P. R. Sari, I. Cholissodin, and B. Rahayudi, "Optimasi Gizi Bahan Makanan pada Anak-Anak untuk Tumbuh Kembang menggunakan Algoritma Genetika (Studi Kasus: Dinas Kesehatan Kabupaten Kediri)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 12, pp. 5429–5436, 2021.

[10] Q. D. A. Sulistiani, B. Irawan, and C. Setianingsih, "Dietary habits for toddler growth using particles swarm optimization algorithms," in *2020 2nd International conference on cybernetics and intelligent system (ICORIS)*, IEEE, 2020, pp. 1–6.

[11] A. Q. A'yun, "Implementasi Algoritma Genetika dalam Optimasi Menu Makanan Berdasarkan Jumlah Kalori dan Kandungannya".

[12] A. Larrantuka, C. Setianingsih, and F. M. Dirgantara, "Sistem Penentuan Pola Makan Berat Badan Ideal Orang Dewasa Menggunakan Algoritma Particle Swarm Optimization," *eProceedings Eng.*, vol. 9, no. 3, 2022.

[13] N. S. Handayani, S. Kusumadewi, and E. Fitriyanto, "Rekomendasi Makanan untuk Ibu Hamil Menggunakan Algoritma Genetika (Food Recommendations for Pregnant Women Using Genetic Algorithms)," *JUITA J. Inform.*, vol. 8, no. 1, pp. 45–53, 2020.

[14] I. Cholissodin and E. Riyandani, "Swarm Intelligence," *Malang Fak. Ilmu Komput. Univ. Brawijaya*, 2016.

[15] J. M. García, C. A. Acosta, and M. J. Mesa, "Genetic algorithms for mathematical optimization," in *Journal of Physics: Conference Series*, IOP Publishing, 2020, p. 12020.

[16] A. Lambora, K. Gupta, and K. Chopra, "Genetic algorithm-A literature review," in *2019 international conference on machine learning, big data, cloud and parallel computing (COMITCon)*, IEEE, 2019, pp. 380–384.

[17] R. Andreswari, I. Syahputra, and M. Lubis, "Performance analysis of heuristic miner and genetics algorithm in process cube: a case study," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 11, no. 1.

[18] S. Regilan and L. K. Hema, "Optimizing environmental monitoring in IoT: integrating DBSCAN with genetic algorithms for enhanced clustering," *Int. J. Comput. Appl.*, pp. 1–11, 2023.

[19] M. Fogue, P. Garrido, F. J. Martinez, J.-C. Cano, C. T. Calafate, and P. Manzoni, "A novel approach for traffic accidents sanitary resource allocation based on multi-objective genetic algorithms," *Expert Syst. Appl.*, vol. 40, no. 1, pp. 323–336, 2013.

[20] Y. Hou, N. Wu, M. Zhou, and Z. Li, "Pareto-optimization for scheduling of crude oil operations in refinery via genetic algorithm," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 47, no. 3, pp. 517–530, 2015.

[21] A. G. Gad, "Particle swarm optimization algorithm and its applications: a systematic review," *Arch. Comput. methods Eng.*, vol. 29, no. 5, pp. 2531–2561, 2022.

[22] D. Freitas, L. G. Lopes, and F. Morgado-Dias, "Particle swarm optimisation: a historical review up to the current developments," *Entropy*, vol. 22, no. 3, p. 362, 2020.

[23] S. Sengupta, S. Basak, and R. A. Peters, "Particle Swarm Optimization: A survey of historical and recent developments with hybridization perspectives," *Mach. Learn. Knowl. Extr.*, vol. 1, no. 1, pp. 157–191, 2018.

[24] N. Hashim, N. F. N. Ismail, D. Johari, I. Musirin, and A. A. Rahman, "Optimal population size of particle swarm optimization for photovoltaic systems under partial shading condition," *Int. J. Elec. Comp. Eng*, vol. 12, pp. 4599–4613, 2022.

[25] M. A. Kunna, T. A. A. Kadir, M. A. Remli, N. M. Ali, K. Moorthy, and N. Muhammad, "An enhanced segment particle swarm optimization algorithm for kinetic parameters estimation of the main metabolic

model of Escherichia coli," *Processes*, vol. 8, no. 8, p. 963, 2020.

[26] R. F. Abdel-Kader, "Genetically improved PSO algorithm for efficient data clustering," in *2010 Second International Conference on Machine Learning and Computing*, IEEE, 2010, pp. 71–75.

[27] J. Ghosh, A. K. Poonia, and R. Varma, "Multi-objective hierarchical particle swarm optimization of linear antenna array with low side lobe and beam-width," *Int. J. Appl. Eng. Res*, vol. 12, no. 8, pp. 1628–1632, 2017.

[28] J. Xiang, D. Liu, S. Wang, and F. Wu, "An Improved Adaptive Velocity Update Particle Swarm Optimization Algorithm for Parameter Identification of Lithium-ion Battery," in *2023 3rd New Energy and Energy Storage System Control Summit Forum (NEESSC)*, IEEE, 2023, pp. 519–523.

[29] Y. Del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J.-C. Hernandez, and R. G. Harley, "Particle swarm optimization: basic concepts, variants and applications in power systems," *IEEE Trans. Evol. Comput.*, vol. 12, no. 2, pp. 171–195, 2008.

[30] K. E. Parsopoulos and M. N. Vrahatis, "Particle swarm optimization and intelligence: advances and applications: advances and applications," 2010.

[31] E. Mutiara, E. Nurlelah, E. Ermawati, and M. R. Firdaus, "Komparasi Metode ANN-PSO dan ANN-GA dalam Prediksi Penyakit Tuberkulosis," *KLIK-KUMPULAN J. ILMU Komput.*, vol. 9, no. 2, pp. 366–381, 2022.

[32] M. M. Munir, A. Pujianto, and H. A. M. Lamuru, "Optimisasi Algoritma Genetika dengan Particle Swarm Optimization (PSO) untuk Sistem Rekomendasi Diet Gizi bagi Penderita Diabetes," *J. Ris. Sist. dan Teknol. Inf.*, vol. 1, no. 2, pp. 38–48, 2023.

[33] R. Cazacu, "Comparison between the performance of GA and PSO in structural optimization problems," *Am. J. Eng. Res*, vol. 5, no. 11, pp. 268–272, 2016.