

# Genetic Algorithm and Particle Swarm Optimization on Fertilizer Production Planning Optimization

Mohamad Yusak Anshori<sup>1</sup>, Teguh Herlambang<sup>2</sup>, Dinita Rahmalia<sup>3</sup>  
{yusak.anshori@unusa.ac.id<sup>1</sup>, teguh@unusa.ac.id<sup>2</sup>}

Management Department, University of Nahdlatul Ulama Surabaya<sup>1</sup>, Information System Department, University of Nahdlatul Ulama Surabaya<sup>2</sup>, Mathematics Department, University of Islam Darul Ulum Lamongan<sup>3</sup>

**Abstract:** Production planning is the important part of controlling the cost spent by the company. In this research, production planning model is linear integer programming model with constraints : production, worker, and inventory. Linear integer programming as optimization problem can be solved by exact method like branch and bound, cutting plane or heuristic method. In this paper, we use heuristic like Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) for solving production planning optimization in approaching. GA uses natural selection process of chromosomes while PSO is inspired by the behavior of flocks of birds, swarm of insects, or school of fish. The simulations show that both GA and PSO can find optimal solution of fertilizer production planning in approaching.

**Keywords:** Linear Integer Programming, Constrained Optimization, Genetic Algorithm, Particle Swarm Optimization, Production Planning Optimization

## 1. Introduction.

The applications of optimization with any constraints have been applied in many problems, including production planning. Production planning is the important part of controlling the cost spent by the company. Production planning used in this research is obtained from case study of fertilizer production in East Java, Indonesia. In Indonesia, there are many farmers as livelihood because the demand of food. Because the demand of agriculture production, fertilizer manufacturer should produce fertilizer and distribute to farmers. In production process, there are some costs and manufacturer must make the planning for revenue, cost, and production.

Production planning model is linear integer programming model with constraints : production, worker, and inventory. Linear integer programming as optimization problem can be solved by exact method like branch and bound, cutting plane [1-2] or heuristic method like Genetic Algorithm (GA) [3], Particle Swarm Optimization (PSO) [4], Artificial Bee Colony (ABC) [5], Firefly Algorithm (FA) [6].

Genetic Algorithm (GA) was discovered by Goldberg at 1989. How GA works is using natural selection process mechanism. Some process of GA are initialization chromosome population, selection, crossover and mutation chromosome [7]. Chromosome used in GA is a

matrix where the element is the decision variable-the number of number of units produced, the number of workers available, the number of hired workers, the number of fired workers, the number of inventory.

Particle Swarm Optimization (PSO) was discovered by Kennedy and Eberhart in 1995. PSO is inspired by the behavior of flocks of birds, swarm of insects, or school of fish in which individuals are called particles and the population is called a swarm. The PSO is initialized with a group of random candidate solutions as a swarm of particles. Each particle is given initial position and velocity. When particle finds a direction to the source of food, other particles will follow them [8].

In this paper, we use heuristic like GA and PSO for solving production planning optimization with constraints : production, worker, and inventory. The simulations show that both GA and PSO can find optimal solution of fertilizer production planning in approaching.

## 2. Method

### 2.1. Mathematical Model of Production Planning

Production planning optimization is optimization of cost spent by company with constraints: production, worker, and inventory. Mathematical model of production planning can be constructed as follows :

$$\min \sum_{t=1}^T C_t^P P_t + C_t^W W_t + C_t^H H_t + C_t^L L_t + C_t^I I_t \quad (1)$$

Subject to

$$P_t \leq n_t W_t, \quad t = 1, 2, \dots, T \quad (2)$$

$$W_t = W_{t-1} + H_t - L_t, \quad t = 1, 2, \dots, T \quad (3)$$

$$I_t = I_{t-1} + P_t - D_t, \quad t = 1, 2, \dots, T \quad (4)$$

$$P_t, W_t, H_t, L_t, I_t \geq 0, \quad t = 1, 2, \dots, T \quad (5)$$

with  $T$  is planning period,  $D_t$  is the number of units demanded in period  $t$ ,  $n_t$  is the number of units produced by a worker in period  $t$ ,  $C_t^P$  is the production cost per unit in period  $t$ ,  $C_t^W$  is the worker cost in period  $t$ ,  $C_t^H$  is the hired worker cost in period  $t$ ,  $C_t^L$  is the fired worker cost in period  $t$ ,  $C_t^I$  is the inventory holding cost in period  $t$ ,  $P_t$  is the number of units produced in period  $t$ ,  $W_t$  is the number of workers available in period  $t$ ,  $H_t$  is the number of hired workers in period  $t$ ,  $L_t$  is the number of fired workers in period  $t$ ,  $I_t$  is the number of inventory in period  $t$ ,

In equation (2), the number of units produced (work force) limits the the number of units produced by available workers. In equation (3), the number of available workers is the function of previous period workers, hired worker, and fired worker. In equation (4), the number of current inventory is the function previous period inventory, the number of units produced, and the current demand

## 2.2. Initialization Population

For the production planning problem, the decision variable is the matrix where the elements are :  $P_t$  is the number of units produced in period  $t$ ,  $W_t$  is the number of workers available in period  $t$ ,  $H_t$  is the number of hired workers in period  $t$ ,  $L_t$  is the number of fired workers in period  $t$ ,  $I_t$  is the number of inventory in period  $t$ . The design of decision variable can be seen in equation (6).

$$X^k = (P, W, H, L, I) = \begin{bmatrix} P_1 & P_2 & \dots & P_T \\ W_1 & W_2 & \dots & W_T \\ H_1 & H_2 & \dots & H_T \\ L_1 & L_2 & \dots & L_T \\ I_1 & I_2 & \dots & I_T \end{bmatrix} \quad (6)$$

In the first step of both GA or PSO, we need to initialize the population of chromosome in GA or the population of particle in PSO. The initialization must satisfy constraints (2) - (5). In order to satisfy constraints, the initialization in GA or PSO can be constructed as follows:

```

for  $k = 1 : \max \text{ pop}$ 
1. Generate  $H_t^k \sim U(0, \max H)$  and  $L_t^k \sim U(0, \max L)$ ,  $t = 1, 2, \dots, T$ 
2. Set  $W_0^k$ 
   for  $t = 1 : T$ 
        $W_t^k = W_{t-1}^k + H_t^k - L_t^k$ 
   end
3. for  $t = 1 : T$ 
        $P_t^k \leq n_t W_t^k$ 
   end
4. Set  $I_0^k$ 
   for  $t = 1 : T$ 
        $I_t^k = I_{t-1}^k + P_t^k - D_t$ 
   end
end

```

### 2.3. Genetic Algorithm (GA)

Genetic Algorithm (GA) was discovered by Goldberg at 1989. How GA works is using natural selection process mechanism. Some process of GA are initialization chromosome population, selection, crossover and mutation chromosome [7].

#### Selection

In GA, selection process is replacing some chromosomes with the others to create new population [7]. Selection process works like roulette wheel. Selection process can be explained as follows:

1. Calculate the fitness value  $f(X^k)$ ,  $k = 1, 2, \dots, \max pop$
2. Calculate the total fitness for the population

$$F = \sum_{k=1}^{\max pop} f(X^k) \quad (7)$$

3. Calculate the selection probability  $p_k$

$$p_k = \frac{f(X^k)}{F}, \quad k = 1, 2, \dots, \max pop \quad (8)$$

4. Calculate cumulative probability  $q_k$

$$q_k = \sum_{j=1}^k p_j \quad (9)$$

5. Generate a random number  $r$  randomly between 0 and 1
6. If  $r \leq q_1$ , select first chromosome  $X^1$ , otherwise, select the  $k$ -th chromosome  $X^k$  such that  $q_{k-1} < r \leq q_k$

#### Crossover

In GA, crossover process is exchanging the element position of two parent chromosomes to become new chromosomes based on crossover probability  $p_c$  [7]. Crossover operator can improve optimal solution resulted by GA because crossover operator can explore the new solution. Crossover operator must be designed to satisfy constraints (2) – (5). The algorithm of crossover operator is :

1. Select two chromosomes randomly of  $H_t$ ,  $t = 1, 2, \dots, T$  as first and second parents
2. Determine the element position of two parents  $1 \leq t \leq T$  randomly.
3. Exchange the element position of two parents so that result new  $H_t'$
4. Select two chromosomes randomly of  $L_t$ ,  $t = 1, 2, \dots, T$  as first and second parents
5. Determine the element position of two parents  $1 \leq t \leq T$  randomly.
6. Exchange the element position of two parents so that result new that result new  $L_t'$
7. Set  $W_0^k$

for  $t = 1 : T$

$$W_t^k = W_{t-1}^k + H_t^k - L_t^k$$

end  
8. for  $t = 1 : T$   
 $P_t^k \leq n_t W_t^k$   
end  
9. Set  $I_0^k$   
for  $t = 1 : T$   
 $I_t^k = I_{t-1}^k + P_t^k - D_t$   
end

#### Mutation

In GA, mutation process is changing some element of origin chromosome to become new chromosome based on mutation probability  $p_m$  [7]. Mutation operator can improve optimal solution resulted by GA because mutation operator can explore the new solution. Mutation operator must be designed to satisfy constraints (2) – (5). The algorithm of mutation operator is:

1. Select  $H_t$ ,  $t = 1, 2, \dots, T$
2. Determine the number of changed elements  $1 \leq t \leq T$  randomly.
3. Change with new elements between  $(0, \max H)$  uniformly distributed so that result new  $H_t'$
4. Select  $L_t$ ,  $t = 1, 2, \dots, T$
5. Determine the number of changed elements  $1 \leq t \leq T$  randomly.
6. Change with new elements between  $(0, \max L)$  uniformly distributed so that result new  $L_t'$
7. Set  $W_0^k$   
for  $t = 1 : T$   
 $W_t^k = W_{t-1}^k + H_t^k - L_t^k$   
end
8. for  $t = 1 : T$   
 $P_t^k \leq n_t W_t^k$   
end
9. Set  $I_0^k$   
for  $t = 1 : T$   
 $I_t^k = I_{t-1}^k + P_t^k - D_t$   
end

## 2.4. Particle Swarm Optimization (PSO)

PSO is an optimization method discovered by Kennedy and Eberhart in 1995. PSO is inspired by the behavior of flocks of birds, swarm of insects, or school of fish in which individuals are called particles and the population is called a swarm. The PSO is initialized with a group of random candidate solutions as a swarm of particles. When particle finds a direction to the source of food, other particles will follow them.

Local best particle is the best particle position selected during early time until current time because it keeps the best solution. Global best particle is selected from local best particle position on all swarms which has the best solution.

The basic PSO algorithm is as follows [9]:

1. Generate initialization particle position  $x_{ij}^k(0)$ ,  $k = 1, 2, \dots, \max pop$
2. Generate initialization particle velocity  $v_{ij}^k(0)$ ,  $k = 1, 2, \dots, \max pop$
3. Set local best particle  $p_{ij}^k = x_{ij}^k(0)$ ,  $k = 1, 2, \dots, \max pop$
4. Set global best particle  $g_{ij} = \arg \min_p (f(p_{ij}^k), k = 1, 2, \dots, \max pop)$
5. Update particle along time  $t$ .

for  $t = 0 : \max\_t$

for  $k = 1, 2, \dots, \max pop$

- Calculate the particle velocity  $v_{ij}^k(t+1)$

$$v_{ij}^k(t+1) = wv_{ij}^k(t) + c_1r_1(p_{ij}^k - x_{ij}^k(t)) + c_2r_2(g_{ij} - x_{ij}^k(t)) \quad (10)$$

- Update the particle position  $x_{ij}^k(t+1)$

$$x_{ij}^k(t+1) = x_{ij}^k(t) + v_{ij}^k(t+1) \quad (11)$$

- Calculate the fitness of particle  $f(x_{ij}^k(t+1))$

- Update local best particle  $p_{ij}^k$

$$p_{ij}^k = \arg \min_x (f(x_{ij}^k(0)), f(x_{ij}^k(1)), \dots, f(x_{ij}^k(t)), f(x_{ij}^k(t+1))) \quad (12)$$

end

Update global best particle  $g_{ij}$

$$g_{ij} = \arg \min_p (f(p_{ij}^k), k = 1, 2, \dots, \max pop) \quad (13)$$

end

Update the Particle

We need to modify update the particle and velocity on PSO so that new particle can satisfy constraints in equation (2) – (5). The modification of update particle and velocity can be designed as follows:

1. Update  $v_H^k$ ,  $k = 1 : \max pop$ , particle velocity of  $H^k$  with  $p_H^k$  is the its local best particle and  $g_H$  is the its global best particle.

$$v_H^k(t+1) = wv_H^k(t) + c_1r_1(p_H^k - H^k(t)) + c_2r_2(g_H - H^k(t)) \quad (14)$$

2. Update  $H^k$ ,  $k = 1 : \max pop$

$$H^k(t+1) = H^k(t) + v_H^k(t+1) \quad (15)$$

3. Update  $v_L^k$ ,  $k = 1 : \max pop$ , particle velocity of  $L^k$  with  $p_L^k$  is the its local best particle and  $g_L$  is the its global best particle.

$$v_L^k(t+1) = wv_L^k(t) + c_1r_1(p_L^k - L^k(t)) + c_2r_2(g_L - L^k(t)) \quad (16)$$

4. Update with  $L^k$ ,  $k = 1 : \max pop$

$$L^k(t+1) = L^k(t) + v_L^k(t+1) \quad (17)$$

5. Set  $W_0^k$

for  $i = 1 : T$

$$W_i^k = W_{i-1}^k + H_i^k - L_i^k$$

end

6. for  $i = 1 : T$

$$P_i^k \leq n_i W_i^k$$

end

7. Set  $I_0^k$

for  $i = 1 : T$

$$I_i^k = I_{i-1}^k + P_i^k - D_i$$

end

### 3. Result and discussion.

Data used in the simulations are obtained from one of the fertilizer manufacturer company in East Java, Indonesia during 6 periods : 2011, 2012, 2013, 2014, 2015, and 2016. From the data, we can make fertilizer production planning optimization. Table 1 shows cost per unit in each year which consists : production cost per unit  $C_t^P$ , worker cost  $C_t^W$ , hired worker cost  $C_t^H$ , fired worker cost  $C_t^L$ , and inventory holding cost  $C_t^I$ .

**Table. 1** Value of cost per unit (in million rupiahs)

Period	2011	2012	2013	2014	2015	2016
$C_t^P$	4.00	4.54	5.17	5.33	5.71	5.33
$C_t^W$	7	7	7	7	7	7
$C_t^H$	4	4	4	4	4	4
$C_t^L$	2	2	2	2	2	2
$C_t^I$	2.46	3.83	4.34	4.03	4.63	4.26

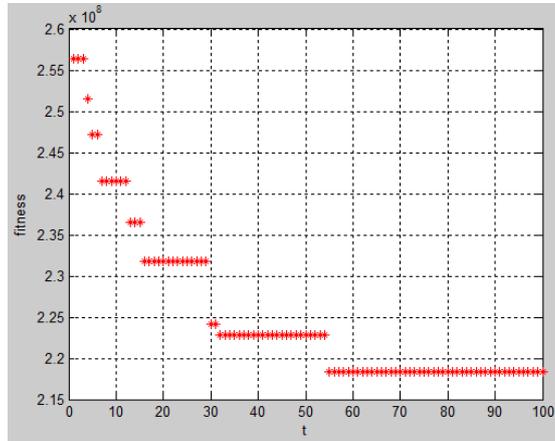
Table 2 shows the number of unit demanded  $D_t$  and the number of units produced by a worker  $n_t$ .

**Table. 2** The number of units demanded (in tons) and the number of units produced by a worker (in tons/worker)

Period	2011	2012	2013	2014	2015	2016
$D_t$	4328630	5008571	5409669	5375396	5546783	5357118
$n_t$	1060	1271	1249	1266	1285	1292

Simulations are implemented by both GA and PSO based on mathematical model of production planning optimization.

### 3.1. Genetic Algorithm (GA) Simulation



**Fig. 1** Optimization Result by GA

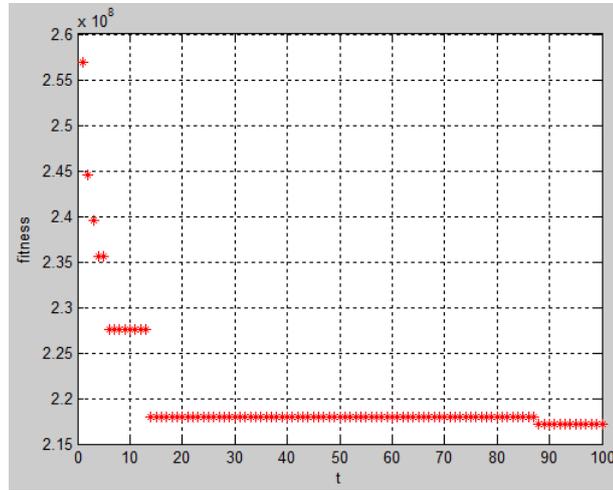
Fig. 1 shows the simulation of GA on production planning optimization with crossover probability is 0.8 and mutation probability is 0.7. In early time, there are population of random chromosomes. Based on natural selection process mechanism : selection, crossover, and mutation, they create new population of chromosomes. The fitness is updated so that fitness will decrease and converge. From the GA simulation with the number of population is 20 and maximum generation is 100, the objective value as fitness is 218296454. Decision variable as optimal solution of GA can be seen in Table 3.

**Table. 3** Optimal Solution by GA

$H_t$	111	123	114	113	342	295
$L_t$	225	273	236	201	206	118
$W_t$	3233	3083	2961	2873	3009	3186
$P_t$	2873699	3236018	3136140	3153824	3410087	3488524
$I_t$	10750236	8977683	6704154	4482582	2345886	477292

### 3.2. Particle Swarm Optimization (PSO) Simulation

Fig. 2 shows the simulation of PSO on production planning optimization. In early time, the particle positions are chosen randomly. In the optimization process, position and velocity particle are updated so that fitness will decrease and converge. From the PSO simulation with the number of population is 20 and maximum iteration is 100, the objective value as fitness is 217234859.



**Fig. 2** Optimization Result by PSO

Decision variable as optimal solution of PSO can be seen in Table 4.

**Table. 4** Optimal Solution by PSO

$H_t$	114	108	199	224	206	160
$L_t$	218	237	290	214	274	101
$W_t$	3243	3114	3023	3033	2965	3024
$P_t$	3011748	3247318	3157005	3123273	3261580	3186448
$I_t$	10888285	9127032	6874368	4622245	2337042	166372

#### 4. Conclusion

Production planning optimization is one of constrained optimization with constraint production, worker, and inventory. Heuristic method like GA and PSO has been applied in approaching optimal solution. GA works based on natural selection process mechanism of chromosome with modifying crossover and mutation so that chromosome can satisfy the constraints. PSO works with modifying the particle position and particle velocity so that particle can satisfy the constraints. From the simulations, both GA and PSO can find optimal solution of fertilizer production planning in approaching.

#### Reference

- [1] H.A Taha, *Operations Research: An Introduction*, New Jersey, USA: Prentice Hall, 2007

- [2] F.S Hillier, and G.J Lieberman, *Introduction to Operations Research*. New York, USA: Mc Graw Hill, 2001
- [3] D Rahmalia, Perbandingan Metode Analitik dan Metode Heuristik pada Optimisasi Masalah Transportasi Distribusi Semen, *Prosiding Seminar Nasional Matematika dan Pembelajarannya*, pp. 1164–1172, 2016
- [4] D Rahmalia, Particle Swarm Optimization-Genetic Algorithm (PSOGA) on Linear Transportation Problem, *AIP Conference Proceeding*, pp. (020030)1-12, 2017
- [5] D Rahmalia and T Herlambang, Optimisasi Masalah Transportasi Distribusi Semen Menggunakan Algoritma Artificial Bee Colony, *Multitek Indonesia*, 11 (2), 2018
- [6] D Rahmalia and A. M Rohmah, Optimisasi Perencanaan Produksi Pupuk Menggunakan Firefly Algorithm, *Jurnal Matematika MANTIK*, 4(1), 2018.
- [7] M Gen and R Chen, *Genetic Algorithms and Engineering Design*, New York, USA: John Wiley and Sons, 1997
- [8] J Kennedy and R C Eberhart, Particle Swarm Optimization, *Proc. IEEE Int. Conf. Neural Networks*, pp. 1942-1948, 1995