

Solving Inventory Routing Problems with Crow Search Algorithm

Maria Krisnawati^{1*}, Ayu Chris Monita², Amanda Sofiana^{3*}

Received: 9 February 2025/ Accepted: 5 July 2025 / Published online: 5 July 2025

* Corresponding Author Email, amanda.sofiana@unsoed.ac.id

1, 2, 3 – Department of Industrial Engineering, Universitas Jenderal Soedirman, Purwokerto, Indonesia

Abstract

The intensifying competition in the business environment compels companies to proactively retain customers by minimizing costs across all aspects while upholding product or service quality. Given the pivotal role of inventory control in cost optimization, a good management of inventory levels becomes priority. Employing a strategic approach within supply chain management, specifically the Inventory Routing Problem (IRP), proves instrumental in achieving an optimal balance in inventory that subsequently impacts cost minimization. The IRP integrates inventory management with vehicle routing to enhance efficiency. In this research, the IRP was applied through the application of the Crow Search Algorithm (CSA), a method inspired by the foraging and caching behaviors of crows. By emulating these natural habits, the CSA method aims to deliver an optimal distribution schedule that minimizes both inventory and transportation costs. The study relied on secondary data in the form of a comprehensive database encompassing relevant data and solution outcomes. The solutions derived from the database serve as a benchmark for evaluating the efficacy of the CSA method. The findings of the research revealed that the total cost incurred through the implementation of the CSA method is notably lower than that obtained from the database results. Furthermore, the simulation results indicate no significant difference between the CSA method and the database, affirming the applicability of the CSA method for effectively addressing IRP.

Keywords - Crow search algorithm; Inventory cost; Inventory routing problem; Transportation costs.

INTRODUCTION

Intense competition has emerged in many business fields, but companies often seem to lack the potential to compete[1]. To be able to deal with this, the company must try its best so that it can maintain the satisfaction and loyalty of its customers so that these customers are not interested in using services or products provided by other companies. Today, business companies have realized that customer satisfaction is an important part of business success. At the same time, customers are also an important role in expanding market value. Generally, a customer is a person who purchases goods and services from the market or a company that can meet their needs at an acceptable price. Therefore, companies must create high-quality products at acceptable prices to attract customers and maintain long-term relationships with customers [2]. Every company must try to reduce all costs without lowering product quality or standards set in order to offer attractive products at competitive prices.

Research [3] shows the most expensive asset of a company is inventory because inventory represents 50% of the total invested capital. Inventory control is very important for almost every type of business, both in manufacturing and services. Inventory balance must be achieved to maintain proper inventory at minimum costs. Inventory control is also one of the keys to improve services by managing inventory levels in every part of the supply chain [4].

An approach that can be used to ensure the production and distribution of goods in the right amount, location, and time to minimize inventory costs is to use a supply chain management strategy [5]. Meanwhile, the supply chain includes not only suppliers and customers, but also transportation, warehouse, retail, and customers themselves. The primary goal of managing the supply chain is to reduce expenses across the system while also focusing on service levels [6]. According to [6], to achieve this goal, one of the well-known concepts used in supply chain management is Vendor Managed Inventory (VMI). VMI entails a transfer of inventory decision-making authority from the retailer to the supplier. Essentially, under the VMI system, the supplier takes on the responsibility for managing retailer inventory. This involves the supplier determining both the quantity and period of inventory distribution, aligning these decisions with insights derived from retailer inventory levels and demand information. In such a scenario, the retailer is relieved of the need to place orders, as the inventory is already in stock upon arrival at the retailer warehouse [7].

In VMI, there is an approach, namely Inventory Routing Problem (IRP) model. Inventory Routing Problem model is the result of a combination of inventory management and vehicle routing [8]. Between suppliers and buyers, both have different objectives from one another. The supplier wants the minimum frequency of visits to reduce shipping costs, but the buyer wants shipments to be made as often as possible even in small quantities so that they do not accumulate too large inventory but still avoid stockout. Thus, it requiring optimal delivery scheduling. The use of the IRP model makes it possible to simultaneously determine the optimal inventory level, route, and distribution schedule based on minimum cost criteria [9]. Research [10] stated that the purpose of the IRP is to minimize inventory costs and transportation costs by taking into account several constraints such as vehicle and customer storage capacity, availability of customer demand at suppliers, etc. IRP is one of the combinatorial optimization problems and is classified as a Non-deterministic Polynomial-time hard (NP-hard)[11]. Due to this nature, especially if the complexity of the problem is large, it cannot be solved by standard linear methods. The metaheuristic approach is a technique that can be used to solve combinatorial problems, where this metaheuristic approach is designed to be flexible enough to handle as many different combinatorial problems as possible [12]. According to [13], several metaheuristic methods can be used to solve various problems, including Genetic Algorithm (GA), Cross-Entropy (CE), Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO), and Crow Search Algorithm (CSA) [14],

Research [14] showed that the CSA operates as a population-based method inspired by the foraging behavior of crows, which involves locating food and subsequently storing the acquired food in hidden nests. Compared with GA, PSO, and HS, the CSA method has fewer parameters to adjust, therefore the CSA method is easier to apply. From the simulation results in research conducted by [12], it showed that CSA's performance is promising because it produces competitive results compared to other algorithms studied. Also, the time required for CSA to find the optimal solution of a problem under study is faster than the PSO, where previously the PSO method was known as a fast method of finding solutions. Previously, the CSA method has been used for research on cases of production scheduling [15], aircraft maintenance check problems, and continuous airworthiness maintenance programs[16], but research has not been found on the case of inventory routing problems.

Thus, this study uses the Crow Search Algorithm method to solve the inventory routing problem case to find an optimal scheduling model and route determination from one warehouse to many retails by taking into account the demand and capacity of each retail.

MODEL

In this study, we used the Multi-period Inventory Routing Problem (MIRP) model developed by [17] by using the crow search algorithm with MATLAB 2017a software. The mathematics model is as follows:

1. Notations

The notation used in this study can be described as follows :

- $\mathcal{A}$ set of retail, indexed by $j = \{1, \dots, V\}$.
- $\mathcal{T}$ set of time periods, indexed by $t = \{1, \dots, T\}$.
- $\mathcal{K}$ set of vehicles, indexed by $k = \{1, \dots, K\}$.
- $\mathcal{V}$ denotes the set of all facilities (customers and supplier).
- 0 denoted a single supplier, $\mathcal{V}' = \mathcal{V} \setminus \{0\}$ denoted a retail.

Decision variables:

- x_{ij}^{kt} Equal to 1 if vehicle k travels directly between node i and node j in period t , 0 otherwise.
- y_i^{kt} Equal to 1 if node i is visited by vehicle k in period t , 0 otherwise

q_i^{kt} Quantity delivered to customer i with vehicle k in period t .

Parameters:

h_i Holding cost at node i (\$).

I_i^t Inventory level (piece)

c_{ij} Transportation cost between nodes i and j (\$).

d_i^t Demand at customer i in period t (piece).

C_i Inventory-holding capacity (piece).

Q_k Vehicle capacity (piece).

II. Mathematical Formulations

Reference [17] showed that the IRP can be formulated as:

$$\text{Min} \sum_{i \in \mathcal{V}} \sum_{t \in \mathcal{T}} h_i I_i^t + \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}, i < j} \sum_{k \in \mathcal{K}} \sum_{t \in \mathcal{T}} c_{ij} x_{ij}^{kt} \quad (1)$$

S.t.

$$I_i^t = I_i^{t-1} + \sum_{k \in \mathcal{K}} q_i^{kt} - d_i^t \quad i \in \mathcal{V}' \quad t \in \mathcal{T} \quad (2)$$

$$I_i^t \geq 0 \quad i \in \mathcal{V} \quad t \in \mathcal{T} \quad (3)$$

$$I_i^t \leq C_i \quad i \in \mathcal{V} \quad t \in \mathcal{T} \quad (4)$$

$$\sum_{k \in \mathcal{K}} q_i^{kt} \leq C_i - I_i^{t-1} \quad i \in \mathcal{V}' \quad t \in \mathcal{T} \quad (5)$$

$$q_i^{kt} \leq C_i y_i^{kt} \quad i \in \mathcal{V}' \quad k \in \mathcal{K} \quad t \in \mathcal{T} \quad (6)$$

$$\sum_{i \in \mathcal{V}'} q_i^{kt} \leq Q_k \quad k \in \mathcal{K} \quad t \in \mathcal{T} \quad (7)$$

$$q_i^{kt} \geq 0 \quad i \in \mathcal{V}' \quad k \in \mathcal{K} \quad t \in \mathcal{T} \quad (8)$$

$$x_{ij}^{kt} \in \{0,1\} \quad i \in \mathcal{V}' \quad k \in \mathcal{K} \quad t \in \mathcal{T} \quad (9)$$

$$y_i^{kt} \in \{0,1\} \quad i \in \mathcal{V} \quad k \in \mathcal{K} \quad t \in \mathcal{T} \quad (10)$$

Constraints (2) define the inventory at the retail, whereas constraints (3) prevent stockouts at the retail. Constraints (4) defines that the level of inventory at retail should not be more than its capacity. This is a usual assumption in IRP models. Constraints (5)–(6) link the quantities delivered to the routing variables. In particular, they only allow a vehicle to deliver products to a customer if the customer is visited by this vehicle. Constraints (7) ensure that the vehicle capacities are respected. Constraints (8)–(10) enforce integrality and nonnegativity conditions on the variables.

CROW SEARCH ALGORITHM

Research [14] introduced an innovative metaheuristic optimization approach called the crow search algorithm (CSA), inspired by the intelligent behavior observed in crows. CSA, as a population-based method, operates on the premise that crows stored surplus food in hidden locations, and retrieving it as needed. This algorithm can be applied to address both continuous and discrete problems. For instance, in addressing discrete issues like scheduling problems, CSA was suggested for solving the Traveling Salesman Problem (TSP) [15]. In paper [18], CSA was compared to genetic algorithm (GA) and particle swarm optimization (PSO) techniques, revealing its rapid convergence capability and effectiveness in solving the optimal design challenges of third-order resonance-free passive filters in distribution networks.

I. CSA Implementation

The implementation of the CSA in this study was carried out using the MATLAB 2017a software.

- **Set Initial parameter:** set the adjustable parameter of CSA: flock size (N), maximum number of iterations ($\text{iter}_{\max}$), flight length (ℓ), and awareness probability (AP).

- **Randomly generate crows position:** Generate random numbers for the route and initial q as the initial position of crows, generate X values for route sequences with random numbers 1 to n data and generate q_0 values for the number of supplies sent with integer random numbers between 50 to 140 as many as n data.

Calculate the fitness function: In this study, the fitness function is a combination of the objective and boundary functions combined with the exterior penalty function method. The limits are then given a large value, to control the existing constraints to obtain optimal results and still meet the requirements or constraints. An example of coding to calculate limits with an exterior penalty function can be seen in Figure 1.

For each population, the position's quality is assessed through the fitness function value, and the objective is to identify the optimum value by seeking the best outcome from the fitness function. The following is an example of coding in MATLAB which is used to calculate the fitness function value. It is an example of coding in MATLAB which is used to calculate the fitness function value which can be seen in Figure 2

```
%exterior penalty function
c12(xn(j)) = abs(II(xn(j)) - Inv(xn(j)) - (sum(q(xn(j)))) + d(xn(j))) ;
c22(xn(j)) = max(0, -II(xn(j))) .^2;
c32(xn(j)) = max(0, II(xn(j)) - C(xn(j))) .^2;
c42(xn(j)) = max(0, (q(xn(j)) - C(xn(j)) + Inv(xn(j)))) .^2;
c72(xn(j)) = max(0, -q(xn(j))) .^2;

c12x=sum(c12);
c22x=sum(c22);
c32x=sum(c32);
c42x=sum(c42);
c72x=sum(c72);
```

FIGURE 1
EXAMPLE OF THE EXTERIOR PENALTY FUNCTION CODING

```
for i=1:N
[c12x(i),c22x(i),c32x(i),c42x(i),c72x(i),inv(i,:),q(i,:)] = invent2(q0(i,:),d,Inv,xn(i,:),C,hc); %function inventory
[c5x(i),c6x(i),Xi(i,:),Y(i,:),Z(i,:),jarak(i,:),jarakXi(i),jarakYi(i),jarakZi(i),xb] = menghitungkap(q(i,:),xn(i,:),xy,Inv,C,Q); %funct
ft(i) = ((jarak(i)*biyakirim) + (sum(inv(i,:)*hc))) + 1000000000*(c12x(i)+c22x(i)+c32x(i)+c42x(i)+c72x(i)) + 1000000000*(c5x(i)+c6x(i));
end
```

FIGURE 2
EXAMPLE OF CALCULATING AND EVALUATING THE FITNESS FUNCTION CODING

Population position as the best fitness function value and the best value is stored in memory. The following is an example of coding in Matlab which is used to store into the population the best position and the best fitness function.

```
mem=X; % Memory initialization
fit_mem=ft; % Fitness of memory positions
```

FIGURE 3
EXAMPLE OF CODING TO SAVE MEMORY

- **Generate new position:** After going through one iteration, the next iteration needs to update the population with a new position for all populations. An example of coding for a new position update can be seen in the following Figure 4.

```

for i=1:N
    if rand>AP
        xnew(i,:)= ceil(X(i,:)+f1*rand*(mem(num(i),:)-X(i,:))).
    for j=1:pd
        if xnew(i,j) > ba
            xnew(i,j) = ba;
        end
        if xnew(i,j) < bb
            xnew(i,j) = bb;
        end
    end
end
else
    xnew(i,:)=X(i,:); % Generation of a new position for crow
end

```

FIGURE 4
EXAMPLE OF POSITION UPDATE CODING

- **Check the feasibility of new positions:** The latest position is compared to the optimal position from the preceding iteration. If the new position yields better results compared to the best position in the prior iteration, the current position is substituted with the new one. Conversely, if the new position does not demonstrate improvement, the position from the previous iteration is retained. An example of coding can be seen in the following figure.

```

G(i,:)=ismember(xnew(i,:),X(i,:));
for j=1:pd
    if G(i,j)==0
        xnew(i,j)=X(i,j);
    else
        xnew(i,j)=xnew(i,j);
    end
end
end

```

FIGURE 5
EXAMPLE OF CODING TO ENSURES THE ELIGIBILITY OF A NEW POSITION

- **Evaluate fitness function of new positions:** For the latest selected position, the best fitness function value is calculated. If the fitness function value is better than the fitness function value in the previous iteration, then the fitness function value is used. Otherwise, the fitness function value in the previous iteration is used.
- **Update memory:** For each population of selected positions, compute memory updates using the following coding.

```

if ft(i)<fit_mem(i)
    if R(i)<1
        mem(i,:)=xnew(i,:); % Update memory
        fit_mem(i)=ft(i);
    end
end
end

```

FIGURE 6
EXAMPLE OF UPDATING MEMORY CODING

- **Check termination criterion:** The step of generating a new position until the memory update is repeated until it reaches the maximum iteration. When it reaches certain criteria, the crow search algorithm process will stop and come up with the best memory-based solution.

RESULTS AND DISCUSSION

I. Optimal solution with CSA

The data used in this research is a database from the Management Science Laborator, which is then used as a comparison with the research results. Comparison of the best solutions generated from research and solutions from the database can be seen in table 1. The results show that the CSA method is performing quite well because the total costs generated from the research are smaller than the results in the database.

TABLE I
COMPARISON OF CSA RESULTS WITH DATABASE

	CSA	Database
Total Cost	\$ 31,452.68	\$ 33,004.36
Inventory Cost	\$ 12,296.00	\$ 12,094.50
Transportation cost	\$ 19,156.68	\$ 20,909.86

The total of CSA methods in Table I is the result of the best fitness function in the crow search algorithm simulation with MATLAB 2017a. The fitness function in the simulation is a combination of the objective and boundary functions which are made into unconstraint with the exterior penalty function method. According to [19] there are penalty functions to penalize infeasible solutions by reducing their fitness values in proportion to their degrees of constraint violation. If the resulting value violates predetermined limits, the algorithm will give a penalty value in the form of a cost added to the total cost so that the result is not selected as the optimum value. To ensure there is no boundary violation, it can be seen in the simulation results as shown in Figure 7, where the values of $c12x$, $c22x$, and so on have been defined as constraints of 0 (zero). Then the solution generated in this simulation does not violate the predetermined constraints.

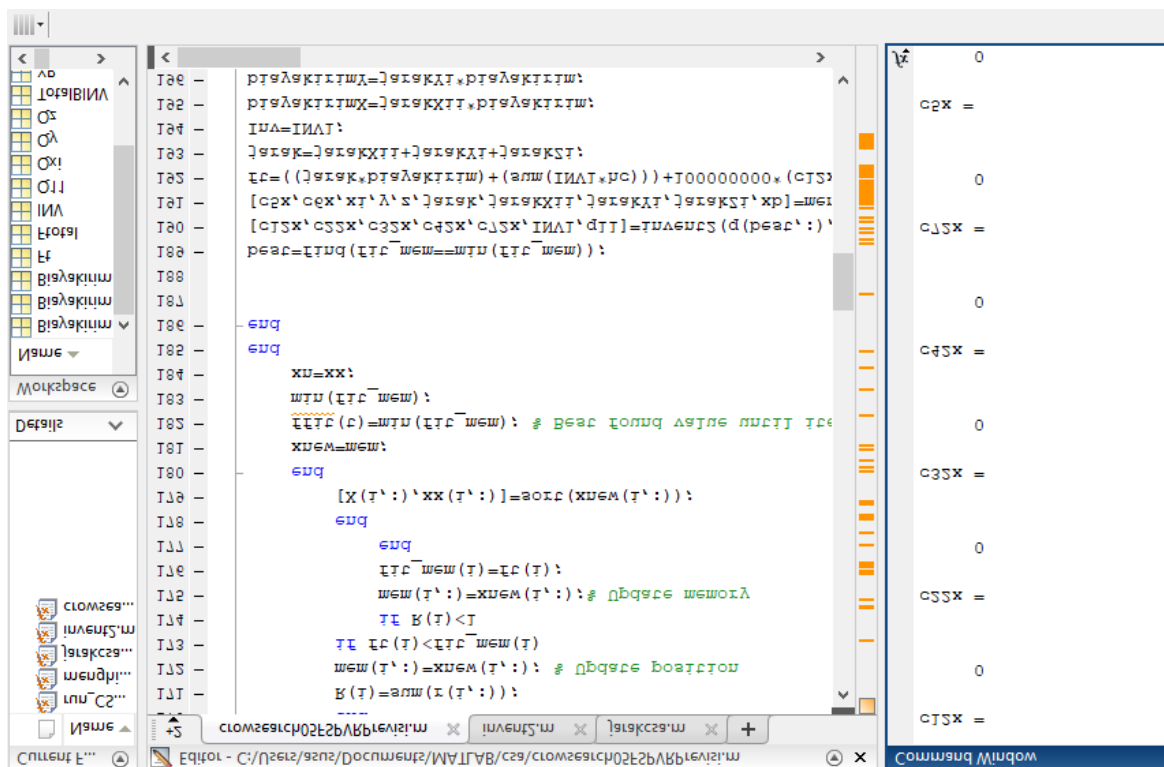


FIGURE 7
EXTERIOR PENALTY VALUE FOR EACH LIMIT

II. Validation of simulation results

According to [20], model validation is typically described as the confirmation that a computerized model, within the scope of its applicability, exhibits an acceptable level of accuracy aligning with the intended use of the model. Reference [21] showed proposed model validation can be done by two methods of testing, by comparing the mean value of the simulation results and the database, and also comparing the amplitude variations value. The comparison between the simulation results and the database can be seen in the following table.

TABLE II
COMPARISON OF MEAN VALUES AND VARIATIONS AMPLITUDE OF CSA RESULTS WITH THE DATABASE

	CSA	Database	Difference
Mean	\$ 20,968.46	\$ 22,002.91	4.7%
Amplitude variations	9706.004659	10497.69616	7.54%

When the model has no systematic error, the E1 value very rarely exceeds 5% and the E2 value does not exceed 30% [21]. The value of E1 and the value of E2 in this study were 4.7% and 7.54% respectively, which means that these values do not exceed the predetermined limits. Therefore, the simulation results of the inventory routing problem with the crow search algorithm can be said to be valid.

CONCLUSION

Drawing conclusions from the conducted research, it can be inferred that the inventory routing problem is effectively solvable through the Crow Search Algorithm method. This is supported by a comparison mean value below 5% and an amplitude variation ratio of less than 30%, affirming the validity of the model. Furthermore, applying the CSA method to solve the Inventory Routing Problem (IRP) yielded a smaller optimal total cost compared to the database while cohering to the predetermined constraints.

FUTURE RESEARCH

Considering the results of the conducted research, a recommendation for future studies would be to employ an alternative approach in addressing the Inventory Routing Problem while maintaining the same constraints and assumptions. This would enable a comparative analysis with the CSA method.

REFERENCES

- [1] P. Ghemawat, "Competition and Business Strategy in Historical Perspective," *Bus. Hist. Rev.*, vol. 76, no. 1, pp. 37–74, 2002, doi: 10.2307/4127751.
- [2] K. Khadka and S. Maharjan, "CUSTOMER SATISFACTION AND CUSTOMER LOYALTY," *CENTRIA Univ. Appl. Sci.*, no. November, pp. 21–36, 2017, doi: 10.4337/9781781955970.00008.
- [3] J. Heizer and B. Render, "Operations Management," in *Salemba Empat*, 2014.
- [4] C. Kwak, J. Sung, C. Ouk, and I. Kwon, "Expert Systems with Applications Situation reactive approach to Vendor Managed Inventory problem," *Expert Syst. Appl.*, vol. 36, no. 5, pp. 9039–9045, 2009, doi: 10.1016/j.eswa.2008.12.018.
- [5] David Simchi-Levi, P. Kaminsky, and E. Simchi-Levi, *Design and Managing The Suppl Chain Concepts Strategies and Case Studies*. 2000.
- [6] S. H. R. Pasandideh, S. T. A. Niaki, and A. R. Nia, "A genetic algorithm for vendor managed inventory control system of multi-product multi-constraint economic order quantity model," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 2708–2716, 2011, doi: 10.1016/j.eswa.2010.08.060.
- [7] D. Choudhary, R. Shankar, P. K. Dey, H. Chaudhary, and L. S. Thakur, "Benefits of retailer-supplier partnership initiatives under time-varying demand: A comparative analytical study," *Int. J. Prod. Res.*, vol. 52, no. 14, pp. 4279–4298, 2014, doi: 10.1080/00207543.2013.879615.
- [8] S. M. J. Mirzapour Al-e-Hashem and Y. Rekik, "Multi-product multi-period Inventory Routing Problem with a transshipment option: A green approach," *Int. J. Prod. Econ.*, vol. 157, no. 1, pp. 80–88, 2014, doi: 10.1016/j.ijpe.2013.09.005.

- [9] N. H. Moin, S. Salhi, and N. A. B. Aziz, "An efficient hybrid genetic algorithm for the multi-product multi-period inventory routing problem," *Int. J. Prod. Econ.*, vol. 133, no. 1, pp. 334–343, 2011, doi: 10.1016/j.ijpe.2010.06.012.
- [10] O. Guemri, A. Bekrar, B. Beldjilali, and D. Trentesaux, "GRASP-based heuristic algorithm for the multi-product multi-vehicle inventory routing problem," *4or*, vol. 14, no. 4, pp. 377–404, 2016, doi: 10.1007/s10288-016-0315-1.
- [11] A. J. Kleywegt, V. S. Nori, and M. W. P. Savelsbergh, "The stochastic inventory routing problem with direct deliveries," *Transp. Sci.*, vol. 36, no. 1, pp. 94–118, 2002, doi: 10.1287/trsc.36.1.94.574.
- [12] M. T. K. dan Transmigrasi, "Peraturan Menteri Tenaga Kerja dan Transmigrasi Nomor Per.13/Men/X/2011 Tentang Nilai Ambang Batas Faktor Fisika dan Faktor Kimia di Tempat Kerja Tahun 2011," *Peratur. Menteri Tenaga Kerja Dan Transm.*, 2016.
- [13] A. P. S., "Nature Inspired Metaheuristic Algorithms," pp. 306–309, 2017.
- [14] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems : Crow search algorithm," *Comput. Struct.*, vol. 169, pp. 1–12, 2016, doi: 10.1016/j.compstruc.2016.03.001.
- [15] A. Adhi, B. Santosa, and N. Siswanto, "A meta-heuristic method for solving scheduling problem : crow search algorithm A meta-heuristic method for solving scheduling problem : crow search algorithm," 2018, doi: 10.1088/1757-899X/337/1/012003.
- [16] N. Siswanto, A. N. Adianto, H. A. Prawira, and A. Rusdiansyah, "A crow search algorithm for aircraft maintenance check problem and continuous airworthiness maintenance program," vol. 3, no. 2, pp. 115–123, 2019.
- [17] L. C. Coelho, J. F. Cordeau, and G. Laporte, "Thirty years of inventory routing," *Transp. Sci.*, vol. 48, no. 1, pp. 1–19, 2014, doi: 10.1287/trsc.2013.0472.
- [18] S. H. E. Abdel, A. F. Zobaa, and M. E. Balci, "Optimal resonance-free third-order high-pass filters based on minimization of the total cost of the filters using Crow Search Algorithm," *Electr. Power Syst. Res.*, vol. 151, pp. 381–394, 2017, doi: 10.1016/j.epsr.2017.06.009.
- [19] Ö. Yeniay, "Penalty function methods for constrained optimization with genetic algorithms," *Math. Comput. Appl.*, vol. 10, no. 1, pp. 45–56, 2005, doi: 10.3390/mca10010045.
- [20] R. G. Sargent, "Verification and validation of simulation models," *Winter Simul. Conf. Proc.*, vol. 1, pp. 121–130, 1998.
- [21] Y. Barlas, "Multiple tests for validation of system dynamics type of simulation models *," vol. 42, pp. 59–87, 1989.