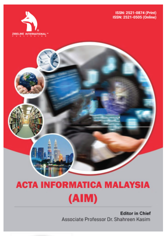
ZIBELINE INTERNATIONAL PUBLISHING
ISSN: 2521-0874 (Print)
ISSN: 2521-0505(Online)
CODEN: AIMCCO

Acta Informatica Malaysia (AIM)

DOI: http://doi.org/10.26480/aim.01.2021.16.20



CrossMark

RESEARCH ARTICLE

IMPLEMENTATION OF PARTICLE SWARM OPTIMIZATION ALGORITHM IN CROSS-DOCKING DISTRIBUTION PROBLEM

Samuel Ro Paian Purba[a], Harummi Sekar Amarilies[a], Nur Layli Rachmawati[a], A.A.N Perwira Redi[b]

[a] Department of Logistics Engineering, Universitas Pertamina, Indonesia.
[b] Industrial Engineering Department - Binus University
**Corresponding Author Email: samuelropaian@gmail.com*

ARTICLE DETAILS

Article History:

Received 16 February 2021
Accepted 20 March 2021
Available online 28 April 2021

ABSTRACT

In order to increase customer satisfaction and maintain customer loyalty, logistics service providers must pay attention to the quality of service provided, one of which is effecive warehouse management, especially in scheduling the arrival and departure of products transporting vehicles. Therefore, this study discusses warehouse management in form of delivery and pickup scheduling at PT XYZ's cross-docking warehouse. This study aims to obtain effective delivery and pickup scheduling and minimize operational costs. The Cross-docking Distribution Problem is an np-hard problem, so the Particle Swarm Optimization algorithm is used, which is a metaheuristic method in finding solutions. Based on the result, it was found that effective delivery and pickup scheduling was able to save inventory cost by 3.12% and reduce the percentage of delays from 73% to 0%. The scheduling process using Particle Swarm Optimization requires an average computation time of 26.2 seconds.

KEYWORDS

Delivery, pickup, cross-docking distribution problem, particle swarm optimization.

1. Introduction

In order to increase customer satisfaction and maintain customer loyalty, logistics service providers must pay attention to the quality of service provided, one of which is effective warehouse management, especially in scheduling the arrival and departure of products transporting vehicles Activities in a warehouse include receiving, storing and shipping goods or materials from one place to another (Yu and Egbelu, 2008). PT XYZ is a digital platform company that is engaged in the fulfillment of food commodities such as vegetables, fruit, meat and fish whose covers the DKI Jakarta area and its surroundings. In meeting all customers requirement, PT XYZ collaborates with suppliers in providing various types of commodities owned. The distribution system problem in the cross-docking warehouse is an NP-Hard Problem (Ma et al., 2011). Determination of delivery and pickup scheduling at PT XYZ's cross-docking warehouse with the aim of minimizing operational costs will make the complexity of the problem high so it is difficult to find a solution using an exact method approach.

The search for the best solution to this type of problem can be done with a metaheuristic approach (Miao et al., 2014) which is used to find vehicle assignment schemes to minimize total operational costs including penalty costs for unfulfilled shipments. One of the algorithms of the metaheuristic method is Particle Swarm Optimization The application of the Particle Swarm Optimization algorithm for multi-period cross-docking by considering time windows has proven to be more effective in terms of the quality of the solution and computational results obtained by CPLEX to minimize transportation, inventory and penalty costs (Yu et al., 2015). Based on this explanation, a good synchronization from inbound and outbound goods is required. This synchronization is realized by the existence of a good scheduling for arrival and departure of the vehicle with the aim of minimizing costs considering the time it reaches the customer. Therefore, this research is focused on the problem of delivery and pickup scheduling in order to minimize delays from both sides.

2. Literature Review

According to a study, Warehouse is a part of a company's logistics system that stores products at and between the point of source and point of consumption, and acts as a provider of information to management regarding the status, condition and disposition of many stored items (Lambert et al., 1998). The cross-docking system consists of 3 parts, namely the inbound side which functions as a provider (supplier), the cross-dock, and the outbound side which functions to distribute commodities to customers (Utami, 2017). In reducing excess costs in these activities, a warehouse management system innovation was carried out, namely a cross-docking system. The cross-docking system is an important logistics strategy, especially in retail, retail and industries engaged in distribution (Mulyawan and Suprapto, 2017). The application of a cross-docking system in Indonesia is not new, especially in the manufacturing industry which is an order fulfillment system using a cross-docking system to speed up the process of sending goods. A cross-docking system is used

Quick Response Code	Access this article online	
	Website: www.actainformaticamalaysia.com	**DOI:** 10.26480/aim.01.2021.16.20

Cite the Article: Samuel Ro Paian Purba, Harummi Sekar Amarilies, Nur Layli Rachmawati, A.A.N Perwira Redi (2021). Implementation of Particle Swarm Optimization Algorithm in Cross-Docking Distribution Problem. *Acta Informatica Malaysia, 5(1): 16-20.*

to linearly reduce inventory levels as well as consolidate activities.

The problem focuses on how to determine the flow of commodities from manufactures to customers through cross-dock warehouses or what is often called the cross-docking distribution problem. Problems in the cross-docking distribution problem associated with the operation of the cross-dock center focus on how to manage the cross-docking center to avoid lost sales (Tang and Yan, 2010). Research on cross-docking distribution problems is also carried out to reduce transportation and inventory costs and increase customer satisfaction (Ma et al., 2011). Another problem related to the cross-docking distribution problem is how to plan the network flow in a cross-docking system, which is a cross-docking distribution problem or a cross-docking network problem. (Belle et al., 2012). Research discusses several metaheuristic methods to find inbound and outbound sequences with the aim of minimizing the total operation time called makespan (Arabani et al., 2011). In the cross-dock distribution problem model is used to minimize internal operating costs and transportation costs (outbound) seen from the number of vehicles and tested using CPLEX with small instances and it is proven that the model used is easily adapted to various cross-dock configurations (Serrano et al., 2016).

The application of the Particle Swarm Optimization algorithm for multi-period cross-docking by considering time windows has proven to be more effective in terms of the quality of the solution and computational results obtained by CPLEX to minimize transportation, inventory and penalty costs (Yu et al., 2015). Research related to truck scheduling in multi-door cross-docking uses the Particle Swarm Optimization algorithm with the aim of minimizing operational time in the cross-docking warehouse and is proven to be able to find quality solutions with fast convergence. (Warisa and Piya, 2017). Based on previous research that discusses the application of the Particle Swarm Optimization algorithm to the cross-docking distribution problem, there is still a research gap for writers in developing algorithms by adjusting the conditions in the cross-docking warehouse.

3. RESEARCH METHODOLOGY

The author uses a cross-docking distribution problem model where in this situation the demand must be fulfilled even though the truck capacity is still not full in order to maintain the level of customer satisfaction (Putri, 2018). Manufactures send various types of products to customers through a cross-dock warehouse to classify according to customer demand for each point of destination. The cross-dock illustration model can be illustrated as shown below.

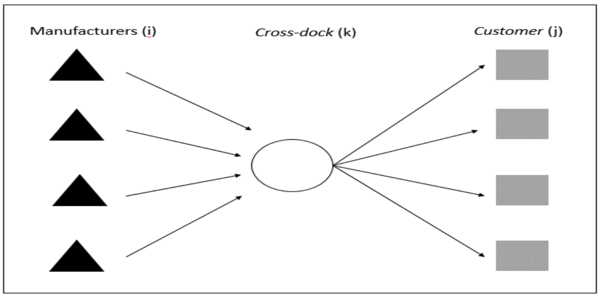


Figure 1: Illustration of cross-docking distribution problem

3.1 Mathematical Model

The mathematical model was adopted from a research by which discussed the multi period cross-docking distribution problem with time windows (Chen et al., 2006). The mathematical model was simplified according to actual conditions in the field.

3.1.1 Input

D set m *delivery*, index by i
P set n *pickup*, index by j
C set c *cross-docks*, index by k
G set d product, index by r
T set time, index by t

3.1.2 Parameter

DP Binary matrix index, where $DP_{i,r}$ is 1 if r is sent from delivery i, and 0 vice versa.
DA Quantity shipped by delivery i.
DD Matrix of distance from delivery i to crossdock k.
DS Delivery start time i DE Time finished delivery i
PP Binary matrix index, where $PP_{j,r}$ is 1 if r is picked up from pickup j and 0 vice versa.
PA Pickup quantity by pickup j.
PD Distance matrix from pickup j to crossdock k.
PS Pickup start time i PE Time to finish pickup i
CAP Cross-dock capacity k
COST Inventory costs
Tmin The minimum time horizon.
Tmax Maximum time horizon.

3.1.3 Decision Variables

$X_{i,k,t}$ Binary solution, 1 if delivery for crossdock k at time t, 0 otherwise.
$Y_{j,k,t}$ Binary solution, 1 if pickup for crossdock k at time t, 0 otherwise.
$Z_{r,k,t}$ Integer, the number of products r in crossdock k at time t.

3.1.4 Objective Function

Minimize ($COST_{transportation} + COST_{inventory}$)
where
$COST_{transportation} = \sum_{i=1}^{m} \sum_{k=1}^{c} \sum_{t=Tmin}^{Tmaks} = x_{i,k,t} DD_{i,k} + \sum_{j=1}^{n} \sum_{k=1}^{c} \sum_{t=Tmin}^{Tmaks} y_{j,k,t} PD_{j,k}$
$COST_{inventory} = \sum_{k=1}^{c} COST_k \sum_{r=1}^{d} \sum_{t=Tmin}^{Taks} z_{r,k,t}$
Subject to:

$\sum_{k=1}^{c} \sum_{t=DS_i}^{DE_i} x_{i,k,t} \leq 1$ for all i (1)

$\sum_{k=1}^{c} \sum_{t=PS_j}^{PE_j} y_{j,k,t} = 1$ for all j. (2)

$z_{r,k,t} \geq 0$ for all r, k and $Tmin \leq t \leq Tmax$. (3)

$\sum_{r=1}^{d} z_{r,k,t} \leq CAP_k$ for all k and $Tmin \leq t \leq Tmax$. (4)

$z_{r,k,T_{\min}-1} = 0$ for all r, k. (5)

$z_{r,k,t} = z_{r,k,t-1} + \sum_{i=1}^{m} x_{i,k,t} DP_{i,r} DA_i - \sum_{j=1}^{n} y_{j,k,t} PP_{j,r} PA_j$ (6)

for all r, k and $Tmin \leq t \leq Tmax$.

(1) Ensures that each delivery is fulfilled within the specified time window at most one time. (2) determines the time windows limit for each pickup. (3) ensures that the flow of one product from each cross-dock at a time is non-negative. (4) is the limit on the capacity of each cross-dock. (5) sets the initial inventory for each product in each cross-dock. (6) calculates the change in inventory level at the cross-dock which ensures the conservation of product flow.

3.2 Particle Swarm Optimization

The Particle Swarm Optimization method was first introduced by Dr. Eberhart and Dr. Kenedy in 1995 at a neural network conference in Perth, Australia. Initially, PSO was inspired by the behavior of a swarm of insects, such as ants, termites, bees, or birds in searching for food randomly in an area (Molavi et al., 2018). There is only one piece of food but the birds do not know where the food is but know how far away the food is. One of the effective ways to obtain food is to follow the birds that are closest to the food (Golmakani and Fazel, 2011).

PSO Pseudocode

Random initialization of the whole swarm

Repeat

Evaluate f (x_i)

For all particle I

Update Velocities

v_i(t+1) = w*v_i(t) + r_1*c_1*v_i $(PBest_i - x_i(t))$ + r_2*c_2*v_i $(GBest - x_i(t))$

Move to new position x_i(t+1) = $x_i(t)$ + v_i (t+1)

If f (x_i) < f $(Pbest_i)$ **Then** $Pbest_i = x_i$

If f (x_i) < f $(Gbest_i)$ **Then** $Gbest_i = x_i$

Update (x_i, v_i)

End For

Until Stopping Criteria

3.3 Initial Solutions

Given the complexity of the problem in cross-docking, the approach taken is a local search technique. Simple methods such as assigning values randomly usually fail to produce viable solutions (Chen et al., 2006). The author uses the greedy method for the initial solution.
Generate initial solutions
Require *m* total deliveries
rearrange deliveries randomly
curr ← 1
while *curr_m* **do**
find set *S* "covered" by *Dcurr* greedily
assign *Dcurr, S* greedily
if solution feasible then
break
end if
curr ← *curr* − 1
end while
compute cost of solution
for *i* = 1 to *n* **do**
if *Pi* has not been fulfilled
cost←cost + penalty
end if
end for
return solution
return solution
Dcurr = (p,a,[s,e]),
S = {*Pi* |*Pi = (pi, ai ,* [*si, ei*]*)*} if it satisfies the following conditions
1. for all *Pi* ∈ *S, pi = p*
2. for all *Pi* ∈ *S,_ai_a*
3. for all *Pi* ∈ *S, ei_s.*

3.4 Solution Representation

Solution representation is one of the key elements to find a good solution. The following is a representative solution for delivery and pickup scheduling (Lambert et al., 1998). Before building a program using the Particle Swarm Optimization algorithm, a solution representation is needed to describe the desired solution in order to optimize the search for a solution in the Particle Swarm Optimization algorithm.

Table 1: Solution Representation

	period	1	2	3	4	5	6
Delivery	t (time)	14	..	..			
	K(number of cross-dock)	1	..				
Pickup	t (time)	14	14	..			
	K (number of cross-dock)	1	1	..			

4. Discussion

Cross-docking Distribution Problem with Particle Swarm Optimization algorithm solved by Microsoft Visual Studio 2019 with the C # programming language. The experiment was conducted using a laptop with an Intel (R) Core (TM) processor i3-5005U CPU @ 2.00GHz (4 CPUs), ~ 2.0GHz, 6144 MB RAM with the Windows 10 Pro 64-bit operating system.

In general, the program designed to solve the problem in this case was able to run well. The parameter values are determined by preliminary experiments refers to previous studies with multi-period cross-docking cases (He et al., 2016). The best parameter configuration according to in the case of multi period cross-docking distribution problem (Yu et al., 2015). The determination of the value of each parameter was carried out using previous research related to the best parameters used with almost the same cases, authors used the trial and error method in finding the best parameter combination.

Table 2: Best Parameter Configuration

Parameter	Value
Inertia weight	0.9
Cognitive learning	4
Social learning	4
Particle	50
Iteration	500

After configuring the best parameters, the data processing process is carried out, namely finding the best delivery and pickup scheduling. The data entered into the program is the daily scheduling data of each vendor in delivery and customer demand in pickup. The output of this result is a daily schedule of delivery and pickup.

Table 3: Results of Gbest Value and Computation Time

Run	CPU time	Gbest (Rp)
1	28.83	5.631.546,60
2	27.49	5.426.966,60
3	27.57	5.725.067,60
4	26.3	5.715.660,60
5	25.68	5.196.294,60
6	22.83	5.431.049,60
7	27.19	5.766.415,60
8	24.5	5.766.415,60
9	26.66	5.333.460,60
10	24.97	5.335.356,60
Average	**26.20**	**5.532.823,40**

The initial solution was built using the greedy method. This method is used because it has an uncomplicated mechanical process and yields decent results. The drawback of this algorithm is that it does not search for the whole solution so it cannot be ascertained whether the initial solution is good. However, the role of the Particle Swarm Optimization algorithm is able to find an overall solution that is better than the initial solution. The author adds an algorithm calculation regarding penalty cost. This calculation is used to ensure that no output from either delivery or pickup comes out of the range time windows for each. If there is an output that comes out of the respective time windows, a penalty fee will be charged according to the difference between the delivery (start-end) and pickup (start-end) time windows. So that the way this algorithm works minimizes the cost of penalties in order to produce output that matches the time windows (Damghani et al., 2017). As discussed in the previous algorithm analysis, the authors made a penalty cost in order to minimize the absence of delivery and pickup that is out of the time windows. After running from 24 deliveries, none of them violated the time windows, but from 363 pickups there were 9 pickups that came out of the time windows (Fitriani et al., 2014). The time windows for each delivery and pickup are defined as soft constraints. Soft constraints are limits that should not be violated, but if these limits are violated there are indications of being subject to penalty fees.

4.1 Sensitivity Analysis

Sensitivity analysis was performed to determine the effect of parameter changes on the objective function. The parameters to be analyzed are the number of delivery and pickups, changes in time windows for delivery and changes in time windows for pickup.

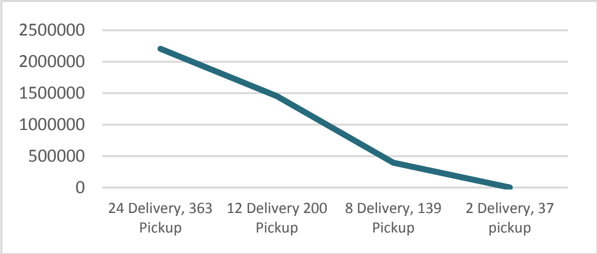


Figure 2: Graph Sensitivity Analysis of Changes in Delivery and Pickup

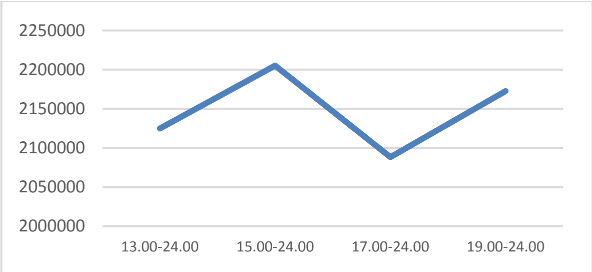


Figure 3: Graph Sensitivity Analysis of Changes in Time Windows Delivery

Based on Figure 2, the decrease and increase in the number of deliveries and pickups affects the gbest value in a straight line, meaning that the more the number of deliveries and pickups handled by the cross-docking warehouse, the gbest value will also increase, and vice versa. In the time windows delivery parameter, based on Figure 3, it can be seen that every change in the time windows value affects the gbest value (Bowersox et al., 1999). However, the change in time windows does not affect directly or significantly the gbest value.

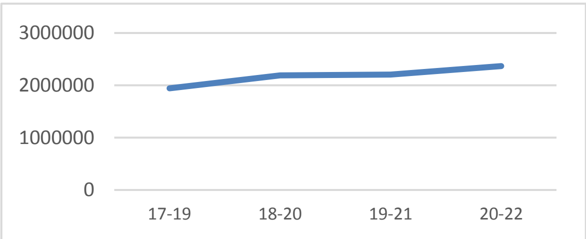


Figure 4: Graph Sensitivity Analysis of Changes in Time Windows Pickup

In the time windows pickup parameter, based on Figure 4, the decrease in time windows affects the Gbest value in a straight line, meaning that the faster the pickup time, the smaller the Gbest value that is obtained. This condition is in accordance with the initials of the solution built by the author using the greedy method, namely the faster the pickup time comes to the cross-docking warehouse, the faster the commodity will be carried out in the delivery process. The sensitivity analysis shows that the gbest value is influenced by the number of delivery and pickups and the decrease in time windows for pickup, but it is not significantly affected by changes in time windows for delivery.

4.2 Managerial Implications

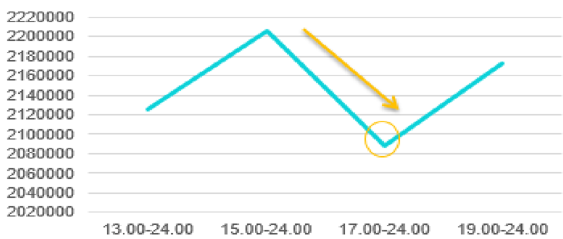


Figure 5: Managerial Implications for Time Windows Delivery

Considerations used in scheduling applications for PT XYZ, namely by reducing the window delivery time to 17.00 - 24.00 so that the Gbest value decreases by 6%.

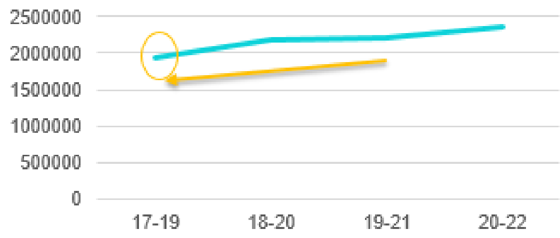


Figure 6: Managerial Implications for Time Windows Pickup

The second consideration is to determine the time windows pickup that starts at 17.00 so get on decrease in Gbest value.

5. Conclusion

Scheduling delivery and pick-up at PT XYZ was able to complete 24 deliveries and 363 pick-ups that are effective and can provide inventory costs of 3.12% and reduce the proportion from 73% to 0%. Making a scheduling program using the Particle Swarm Optimization algorithm is able to solve problems in big data in real cases and has passed the verification and validation tests both in terms of mathematical models and programs in Microsoft Visual Studio. Determination of the initial development of the solution using the greedy method is proven to show a viable solution and does not produce too random values for each iteration process. Determination of the correct parameter configuration will produce a good quality solution. This is important for testing methods in delivery and pickup scheduling.

Computational time in the program is very dependent on the size of the data. The larger the data size, the longer it will take the program to find the optimal solution. Solving real cases in this cross-docking warehouse requires an average computation time of 26.2 seconds. For further research, it is necessary to study the parameters in the Particle Swarm Optimization algorithm which is more about the combination of parameter combinations in order to obtain a more optimal solution from the current research. It is better if the program can use other metaheuristic algorithms such as Simmulated Annealing, Tabu Search etc. Calculation of cost savings optimization can be done in further research.

References

Arabani, B.A., Ghomi, S.M., Zandieh, M., 2011. Meta-heuristics implementation for scheduling of trucks in a cross-docking system with temporary storage . Expert System with Applications.

Belle, J.V., Valckeaners, P., Cattryse, D., 2012. Cross-docking: State of the art. Omega.

Bowersox, D.J., Stank, T.P., Daugherty, P., 1999. Lean Launch: Managing Product Introduction Risk Through Response-Based Logistics. Journal of Production Innovative Management.

Chen, P., Guo, Y., Lim, A., Rodrigues, B., 2006. Multiple crossdocks with inventory and time windows. Computers & Operations Research.

Damghani, K.K., Tavana, M., Arteaga, F.J., Dashti, M.G., 2017. A customized genetic algorithm for solving multi-period cross-dock. Measurement.

Fitriani, N., Yusuf, R.P., Rantau, I., 2014. Analisis Persediaan Beras Di Perusahaan Umum BULOG Divisi Regional Nusa Tenggara Timur. E-Jurnal Agribisnis dan Agrowisata.

Golmakani, H.R., Fazel, M., 2011. Constrained Portofolio Selection using Particle Swarm Optimization. Expert Systems with Application.

He, Y., Ma, W.J., Zhang, J. P., 2016. The Parameters Selection of PSO Algorithm influencing On performance of Fault Diagnosis. MATEC.

Lambert, D.M., Stock, J.R., Ellram, L.M., 1998. Fundamental of Logistics Management. United States: The McGraw-Hill Companies.

Ma, H., Miao, Z., Lim, A., Rodrigues, B., 2011. Crossdocking distribution networks with setup costand time. Omega.

Miao, Z., Cai, S., Xu, D., 2014. Applying an adaptive tabu search algorithm to optimize truck-dock. Expert Systems with Applications.

Molavi, D., Shahmardan, A., Sajadieh, M.S., 2018. Truck Scheduling In A Cross Docking Systems With Fixed Due Dates And Shipment Sorting. Computer & Industrial Engineering.

Mulyawan, T., Suprapto, B., 2017. Implementasi Sistem Crossdocking Untuk Meminimalkan Biaya Operasional Kendaraan (Studi Kasus Pada Distributor Unilever CV Berkat Abadi Dan CV Sinar Berkat Abadi. Jurnal String, 2 (3).

Putri, R.W., 2018. Metode Particle Swarm Optimization Untuk Optimasi Portofolio. LPPM UNINDRA.

Serrano, C., Moral, J., Delorme, X., Dolgui, A., 2016. Cross-Docking Operation Schedulling: Truck Arrivals, Shop Floor Activities and Truck Departures. IFAC-PapersOnLine.

Tang, S.L., Yan, H., 2010. Pre-distribution vs. post-distribution for cross-docking with transhipment. Omega.

Utami, A.F., 2017. Pengembangan Model Operasional Cross-Docking. Institut Teknologi Sepuluh November.

Warisa, W., Piya, H., 2017. Truck Scheduling in Multi-Door Cross Docking Terminal by Modified Particle Swarm Optimization. Computer & Industrial Engineering.

Yu, V.F., Jewpanya, P., Kachitvichyanukul, V., 2015. Particle Swarm Optimization for the Multi-Period Cross-Docking Distribution Problem with Time Windows. International Journal of Production Research .

Yu, W., Egbelu, P.J., 2008. Scheduling of inbound and outbound trucks in cross docking systems with temporary storage. European Journal of Operational Research.

Cite the Article: Samuel Ro Paian Purba, Harummi Sekar Amarilies, Nur Layli Rachmawati, A.A.N Perwira Redi (2021). Implementation of Particle Swarm Optimization Algorithm in Cross-Docking Distribution Problem. *Acta Informatica Malaysia, 5(1): 16-20.*