Particle Swarm Optimization for Warehouse Design Problem

Conference Paper · December 2010

DOI: 10.13140/2.1.2435.2001

Particle Swarm Optimization for Warehouse Design Problem

Natanaree Sooksaksun † [1] **and Voratas Kachitvichyanukul** [2]
Industrial and Manufacturing Engineering
Asian Institute of Technology, Pathumthani 12120, THAILAND
Email: st105997@ait.ac.th[1]
voratas@ait.ac.th[2]

Abstract – The design of warehouse is traditionally done in two steps by first determining the aisle layout and dimension followed by the allocation of storage for items. The design process is performed iteratively until a design with appropriate performance criterion is found. This paper proposes a mathematical modeling approach for warehouse design to help the designer save time. A mathematical model is formulated to include the number of aisles, the length of aisle and the length of each pick aisle to allocate to each product class as decision variables. The objective function is to minimize the average travel distance of a warehouse that operates under a class-based storage with return routing policy. A particle swarm optimization algorithm is proposed for this highly non-linear mathematical model. The results show the optimum numbers of aisles in the designs from the proposed and the classical approaches are very close. However, the lengths of each pick aisle to allocate to each product class are different. Overall, the warehouse design of the proposed algorithm gives the design with lower average travel distance than that from the classical warehouse design and the computational time of the proposed algorithm is shorter.

Keywords: Warehouse design, Particle Swarm Optimization, Class-based storage policy

1. INTRODUCTION

The layout configuration of warehouse is one of many factors that influence the effectiveness of warehouse operations. A good warehouse layout configuration may significantly reduce the travel distance for order picking. Efficient warehouse operations are essential for a successful supply chain operation.

Generally, a common warehouse layout has a rectangular shape. Caron et al. (2000) presented three rectangular layouts that are commonly used in the warehouse design. The first layout is divided in two blocks with stocking aisle running parallel to the warehouse front-end where the I/O point is located. The second layout contains one block with stocking aisles running perpendicular to the warehouse front-end and the I/O point located in the middle of the front-end. The third layout is same as the second layout but the I/O point is located in the corner of front-end. However, this research considers only second layout.

Apple (1997) observed that the warehouse design is a complex task because of the interaction and relationships between each of the activities in warehouses. Rouwenhorst et al., (2000) pointed out another difficulty that is a potentially large number of feasible designs. Hassan (2002) proposed a general framework for warehouse design with arrangement of functional areas, number and location of docks and I/O points, aisle system design, flow pattern, space requirements, and assignment of items to storage locations. Gu et al. (2007) presented five major decisions that involve warehouse design. First is to determine the overall warehouse structure. Second is to determine sizing and dimensioning the warehouse and its departments. Third is to determine the detailed layout within each department. Fourth is to select warehouse equipments. Last one is to select operational strategies. In this paper, the focus is on a storage department that is the primary part of the warehouse. There are two decision steps in the classical method of the warehouse design that use class-based storage policy. First is to design the aisle systems that are to determine the number of aisles, their location, and length. These decision variables have major impacts on the required space, operations, material handling and storage assignment. Second is an assignment of the product in each product class to a storage location. This step is an important step in the design of a warehouse because its impact on movement time and cost, throughput, productivity of pickers, and congestion. The purpose of this research is to propose the PSO algorithm for the design of warehouse with class-based storage policy. In addition, this research compare between the proposed and classical approaches. The remainder of this paper is organized as follow. In section 2, the assumptions, the objective function and the constraints for mathematical model for warehouse design

† : Corresponding Author

are proposed. The PSO framework for warehouse design is proposed in section 3. Comparison between PSO and calculation is presented in section 4. Finally, the summary is presented.

2. MATHEMATICAL MODEL

2.1 Assumptions

This mathematical model is built based on following assumptions.

- The total space floor of the warehouse is known.
- The warehouse consists of multiple identical rectangular racks. Each rack can be used to store more than one product type.
- The warehouse has one dock which located at the middle.
- The class-based storage strategy and return routing policy are used in this warehouse.
- Horizontal travel system, i.e. the picker moves only along the aisle floor (low-level system).
- The order pickers can pick items from both sides of the aisle by one pass; no addition time is needed for changing picking from one aisle side to the other (i.e. narrow aisle). Therefore, travel distances are measured along the aisle center line.
- Items in the same class have the same order frequency. The order frequency of each item-class is defined as the number of times that an item from that class is required in a planning period; it is known and constant throughout the planning period.

2.2 Notations

Input data

q	number of picks in a picking tour
c	number of product classes
w_a	width of the cross aisle
w_b	centre-to-centre distance between two consecutive aisles
w_c	width of the storage rack
S	total floor space
S_i	percentage of the total storage space used for class i
f_i	order frequency of product class i
L_{min}	minimum length of a pick aisle
u_0	minimum floor space utilization of layout

Intermediate variable

p_{ij}	the probability that an item of class i located in aisle j is ordered

Decision variable

a	number of storage aisles
l_{ij}	partial length of pick aisle j used for storing of product class i

2.3 Objective function and Constraints

2.3.1 Objective function

Generally, the travel distance is considered as a primary objective in the optimization warehouse design (De Koster et al., 2007). Therefore, the objective of this model is to minimize the average travel distance of a picking tour. The average tour length will consider both the within-aisle distance and cross-aisle travel distance. Le-Duc and De Koster (2005) proposed the formula to estimate the average within-aisle travel distance of a tour as follows:

$$TD^{WA} = 2\sum_{j=1}^{a}\left\{\left\{\begin{array}{l}\dfrac{w_a}{2}+\left(p'_{1j}\right)^{q_j}\dfrac{l_{1j}q_j}{q_j+1}+\\ \sum\limits_{i=2}^{c}\left\{\begin{array}{l}\psi\left(p'_{kj},q_j\right)\\ \left[\begin{array}{l}\sum\limits_{k=1}^{i-1}l_{kj}+\\ \dfrac{l_{ij}q_j p'_{ij}}{q_j p'_{ij}+\psi\left(p'_{kj},q_j\right)\sum_{k=1}^{i}p'_{kj}}\end{array}\right]\end{array}\right\}\end{array}\right\}\times\left[1-\left(1-\sum_{i=1}^{c}p_{ij}\right)^{q}\right]\right\} \quad (1)$$

where $p'_{ij} = p_{ij} \Big/ \sum_{i=1}^{c} p_{ij} \quad \left(\forall_j = 1 \; , \; 2a\right)$, and

$$\psi\left(p'_{kj},q_j\right)=\left(\sum_{k=1}^{i}p'_{kj}\right)^{q_j}-\left(\sum_{k=1}^{i-1}p'_{kj}\right)^{q_j}, i\geq 2$$

To calculate the cross aisle travel distance, Sooksaksun and Kachitvichyanukul (2009) proposed the formula as follows:

$$TD^{CA}=2\left[\sum_{j=\frac{a+1}{2}}^{a}\left\{\left(2j-a-1\right)\left(w_b\right)\left(m'_j,q\right)\right\}\right] \quad (2)$$

where $\psi\left(m'_j,q\right)=\left(\sum_{j=\frac{a+1}{2}}^{j}m'_j\right)^{q}-\left(\sum_{j=\frac{a+1}{2}}^{j-1}m'_j\right)^{q}$

The formula for estimating the average tour length can determine by adding TD^{WA} and TD^{CA}. Therefore, the objective of this model is showed below.

Objective $\quad$ Minimize $TD = TD^{WA} + TD^{CA}$ $\quad$ (3)

2.3.2 Constraints

The constraints of the model are given in equations (4) to (9).

$$1 \le a \le \left\lfloor \frac{S}{(w_a + L_{min})\, w_b} \right\rfloor \quad (4)$$

$$\frac{2w_c a}{S}\left[\frac{S}{aw_b} - w_a\right] \ge u_0 \quad (5)$$

$$\sum_{i=1}^{c} l_{ij} = \frac{S}{aw_b} - w_a \qquad \forall_j = 1, 2, ..., a \quad (6)$$

$$\sum_{j=1}^{a} l_{ij} = S_i a \left[\frac{S}{aw_b} - w_a\right] \qquad \forall_i = 1, 2, ..., c \quad (7)$$

$$l_{ij} = l_{i,a-j+1} \qquad \forall (i = 1, 2, ..., c,\ j = 1, 2, ..., a) \quad (8)$$

$$l_{ij} \ge 0 \quad (9)$$

Equation (4) set the upper and lower bounds of the number of aisles. Equation (5) ensures that the floor space utilization is more than the minimum usable floor space. Equation (6) is the restriction on the aisle's length. The total space for each class is presented in equation (7). Equation (8) enforces the symmetrical property of the layout. Lastly, equation (9) ensures that the lengths of aisles are non-negative.

The number of aisles, the length of aisle and the length of each pick aisle to allocate to each product class can be determined that will minimize the travel distance from the mathematical model. Unfortunately, the objective function in this model is non-linear and it is difficult to solve. Therefore, the particle swarm optimization algorithm is proposed to solve this mathematical model.

3. PSO FOR WAREHOUSE DESIGN

In this section, a particle swarm optimization (PSO) algorithm is proposed for solving the mathematical model that is described in section 3. Key feature of the algorithm that is a decoding method is explained.

Particle swarm optimization (PSO) is one of the more recent evolutionary algorithms (Eberhart and Kennedy, 1995). It is a population based search method that is inspired by social behavior of organisms such as bird flocking or fish schooling. A brief and complete survey on PSO mechanism, technique, and application is provided by Kennedy and Eberhart (2001).

The PSO algorithm is described below following the definitions of the indices and notation.

Indices

n	index of particle, $n = 1, 2, ..., N$
d	index of dimension, $d = 1, 2, ..., D$
t	index of iteration, $t = 1, 2, ..., T$

Notation

X_n	position vector of particle n, $X_n = [x_{n1}, x_{n2}, ..., x_{nD}]$
V_n	velocity vector of particle n, $V_n = [v_{n1}, v_{n2}, ..., v_{nD}]$
P_n	personal best position so far of particle n
P_g	global best position so far of the swarm
$\phi(X_n)$	objective function value of particle n
$\phi(P_n)$	objective function value of P_n, the best objective function of particle n
$v_{nd}(t)$	velocity of particle n at the dimension d in the iteration t
$x_{nd}(t)$	position of particle n at the dimension d in the iteration t
w	Inertia weight
R_1, R_2	uniform random number on the interval [0,1]
c_p	personal best position acceleration constant
c_g	global best position acceleration constant
p_{nd}	personal best position of particle n at the dimension d
p_{gd}	global best position at the dimension d

3.1 PSO algorithm

Algorithm: PSO algorithm for warehouse design

1. Read the necessary input data for warehouse design.
2. Calculate the upper bound for maximum numbers of aisles.
3. Randomly generate the number of aisle between minimum and maximum number of aisles.
4. Initialize N particles as a swarm, generated the n^{th} particle with random position X_n in the range [X^{min}, X^{max}], velocity $V_n = 0$ and personal best $P_n = X_n$ where $n = 1, 2, ..., N$. Set iteration $t = 1$
5. $[x_{n1}, x_{n2}, ..., x_{nD}]$ is decoded into the number of aisle and the length of each pick aisle to allocate to each product class (see detail in section 3.2).
6. For $n = 1, 2, ..., N$, compute the objective function of X_n, $\phi(X_n)$
7. Update personal best. For $n = 1, 2, ..., N$, update $P_n = X_n$ if $\phi(X_n) < \phi(P_n)$

8. Update global best. For $n = 1, 2, ..., N$, update $P_g = P_n$ if $\phi(P_n) < \phi(P_g)$
9. Update velocity and position of each particle $n = 1, 2, ..., N$ and dimension $d = 1, 2, ..., D$.
- $v_{nd}(t+1) = wv_{nd}(t) + c_p R_1 (p_{nd} - x_{nd}(t)) + c_g R_2 (p_{gd} - x_{nd}(t))$
- $x_{nd}(t+1) = x_{nd}(t) + v_{nd}(t+1)$
- If $x_{nd}(t+1) < X^{min}$, then set $x_{nd}(t+1) = X^{min}$
- If $x_{nd}(t+1) > X^{max}$, then set $x_{nd}(t+1) = X^{max}$

10. If the stop criterion is met, i.e. $t = T$, stop. Otherwise, set $t = t+1$ and return to step 4.

3.2 Decode method

Decode method is one of the key elements for effective implementation of PSO for warehouse design. The dimension of the particle is determined by formula as follow.

$$\left(\frac{\text{No. of aisle } +1}{2} - 1\right) \times (\text{No. of class } -1) \tag{10}$$

For example, if the number of aisle equals 5 and the number of class equal 3; the dimension in each particle is 4. Each dimension is decoded to the length of each class in each aisle. It is sufficient to decode only the length of each class in aisle in the right hand side of the I/O point because the layout of the warehouse is symmetrical. Table 1 shows an example of the decoded data.

Table 1: The example of the data for decoded

Aisle	Class 1	Class 2	Class 3	Length of aisle
3	l_{13}	l_{23}	l_{33}	10
4	l_{14}	l_{24}	l_{34}	10
5	l_{15}	l_{25}	l_{35}	10
Storage space	5	10	15	

To decode, $l_{13} = x_{n1} \times \text{Min}(10, 5)$, $l_{23} = x_{n2} \times \text{Min}(10 - l_{13}, 10)$, $l_{14} = x_{n3} \times \text{Min}(10, 5 - l_{13})$ and $l_{24} = x_{n4} \times \text{Min}(10 - l_{14}, 10 - l_{23})$. After that, calculate $l_{33} = 10 - (l_{13} + l_{23})$, $l_{34} = 10 - (l_{14} + l_{24})$, $l_{15} = 5 - (l_{13} + l_{14})$, $l_{25} = 10 - (l_{23} + l_{24})$ and $l_{35} = 15 - (l_{33} + l_{34})$ In addition, $l_{11}, l_{21}, l_{31}, l_{12}, l_{22}$ and l_{32} are calculated by using equation (8).

4. COMPUTATION RESULT

This section describes how PSO algorithm for warehouse design can be used. Moreover, the results from this algorithm are compared with the classical method. The details are shown below.

4.1 Warehouse Assumptions

The warehouse is planned to store multiple product group. The total floor space of the warehouse is 495 square units. The width of the cross aisle is 3 units. The width of the storage rack is 2 units. The centre-to-centre distance between two consecutive aisles is 5 units. The minimum length of the storage rack is 1 unit. The minimum floor space utilization is 60 percent.

Furthermore, the annual throughput of the product is classified into 3 classes; A, B and C, based on their turnover rated. The ordering probabilities of each class are 0.8, 0.15 and 0.05 respectively. The percentage of the total storage space used for each class equal 20, 30 and 50 respectively. In this research, four number of picks per tour is considered; 10, 20, 40 and 80 respectively.

4.2 PSO parameters

The PSO for warehouse design is implemented in C# programming language using the PSO object library from ET-Lib (Nguyen, et. al. 2010). The PSO parameters are set as follow. The number of particle is fixed at 50. The stopping criterion is set at 100 iterations. The inertia weight is linearly decreasing from 5 to 0.9. The acceleration constants for self-learning and social learning are 0.1 and 1, respectively. The re-initialization is done every five cycles of iteration. Ten replications are used in this experiment.

4.3 Results from PSO for warehouse design

The results from the program show that the minimum number of aisle is 1 and the maximum number of aisle is 7. Moreover, the results from the PSO for warehouse design are shown in tables 2 and 3. Table 2 shows the example of the length of each aisle for storing each product class with the number of pick per tour equals ten. Moreover, figure 1 shows the optimal number of aisle that gives the minimum travel distance.

In table 1, the warehouse that has the number of aisle equals 3 gives a lowest travel distance in each a number of pick per tour. Therefore, the optimal number of aisle is 3 and the length of each aisle is 30 units.

Table 2: Travel distance from the PSO algorithm

No. of pick/ tour	No. of aisle	Length of aisle	Travel distance		
			Max.	Min.	Mean
10	1	96	107.09	107.09	107.09
	3	30	87.80	87.80	87.80
	5	16.8	90.15d	90.06	90.09
	7	11.14	95.69	95.64	95.66
20	1	96	133.85	133.85	133.85
	3	30	111.43	111.43	111.43
	5	16.8	114.87	114.54	114.63
	7	11.14	123.69	123.61	123.64
40	1	96	157.12	157.12	157.12
	3	30	137.35	137.35	137.35
	5	16.8	138.24	138.20	138.22
	7	11.14	147.01	146.80	146.90
80	1	96	174.78	174.78	174.78
	3	30	161.02	161.02	161.02
	5	16.8	162.29	162.25	162.28
	7	11.14	170.04	169.95	169.98

Table 3: The detailed aisle length of each product class (Proposed Method)

No. of aisle in W/H	Aisle	Length for storing product		
		Class A	Class B	Class C
1	1	19.2	28.8	48
3	1	5.55	9.63	14.82
	2	6.90	7.75	15.35
	3	5.55	9.63	14.82
5	1	0.70	6.04	10.06
	2	1.12	5.72	9.96
	3	13.16	1.67	1.97
	4	1.12	5.72	9.96
	5	0.70	6.04	10.06
7	1	0.36	2.18	8.60
	2	0.68	1.83	8.63
	3	1.79	7.58	1.77
	4	9.92	0.23	1.00
	5	1.79	7.58	1.77
	6	0.68	1.83	8.63
	7	0.36	2.18	8.60

4.4 Results from the classical method

The same case is solved with classical warehouse design. The first step is to determine the number of aisle and dimension. The second step is to assign the product to storage. In this case, there are 4 available alternatives for the number of aisles; 1, 3, 5 and 7. The lengths of aisles for each alternative are 96, 30, 16.8 and 11.14 respectively. After that, products for each class are assigned for each aisle based on the travel distance. The products in class A are located close to the I/O point and in class B and class C are located further away. The details are shown in table 4.

After the assign the product, the average travel distances are calculated. The details are shown in table 5.

In table 5, the warehouse that has the number of aisle equals 3 gives a lowest travel distance in each a number of pick per tour which the result is same as PSO for warehouse design. Therefore, the optimal number of aisles is 3 and the length of each aisle is 30 units.

Table 4 The detailed aisle length for each product class (Classical Design)

No. of aisle in W/H	Aisle	Length for storing product		
		Class A	Class B	Class C
1	1	19.2	28.8	48
3	1	4.33	9	16.67
	2	9.34	9	11.66
	3	4.33	9	16.67
5	1	0	4.4	12.4
	2	3.9	5.5	7.4
	3	9	5.4	2.4
	4	3.9	5.5	7.4
	5	0	4.4	12.4
7	1	0	0	11.14
	2	0	4.46	6.68
	3	3.53	5.94	1.67
	4	8.54	2.6	0
	5	3.53	5.94	1.67
	6	0	4.46	6.68
	7	0	0	11.14

Table 5: The average travel distance from classical design

No. of pick/ tour	No. of aisle	Average travel distance
10	1	107.09
	3	87.88
	5	97.30
	7	96.15
20	1	133.85
	3	111.85
	5	123.72
	7	124.44
40	1	157.12
	3	137.83
	5	148.65
	7	149.67
80	1	174.78
	3	161.54
	5	171.12
	7	175.06

4.5 Compare between PSO for warehouse design and classical method

The average travel distances of picking tour that is the objective from PSO for warehouse design are compared in a figure 1. From figure 1, the optimum numbers of aisle for both methods equal three. In other words, the proposed algorithm and classical method give the same answer in term of the number of aisle. However, the proposed algorithm gives the lower travel distance than the classical method.

Furthermore, the lengths of each pick aisle to storage of each product class are different. Moreover, the computational time of the proposed algorithm is extremely short.

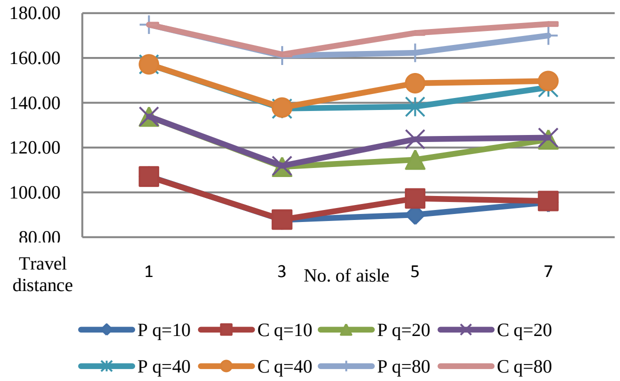

Figure 1 Compare the travel distance between PSO for warehouse design and classical warehouse design

5. SUMMARY

The objective of this paper is to propose the PSO algorithm for warehouse design that using class-based storage policy. Furthermore, the results from the proposed algorithm and that from the classical method are compared. The results show the optimal numbers of aisles from two methods are the same. However, the assignments of items to storage location are different. The average travel distance from the proposed algorithm is shorter than that from the classical method. In addition, the computational time of the proposed algorithm is extremely short.

ACKNOWLEDGMENT

The first author acknowledges the financial support from the National Science and Technology Development Agency.

REFERENCES

Apple, J. (1977). Plant Layout and Material Handling, third ed. John Wiley, New York.

Caron, F., Marchet, G., and Perego, A. (2000) Layout design in manual picking systems: a simulation approach, Integrated Manufacturing Systems, 11(2), 94-104.

De Koster R., Le-Duc, T. and Roodbergen. K.J. (2007) Design and control of warehouse order picking: A literature review, European Journal of Operational Research, 182, 481–501

Hasson, H. Mohsen., (2002) A framework for the design of warehouse layout, Facilities, 20(13/14), 432-440.

Kennedy, J., Eberhart, R., (2001) Swarm intelligence, San Francisco: Morgen Kaufmann Publishers.

Le-Duc, T. and De Koster R. (2005) Travel distance estimation and storage zone optimization in a 2-block class-based storage strategy warehouse, International Journal of Production Research, 43(17), 3561-3581.

Rouwenhorst, B., Sooksaksun, V., van Houtum, G.J., Mantel, R.J., and Zijm, W.H.M. (2000) Warehouse design and control: Framework and literature review, European Journal of Operational Research, 122, 515-533

Sooksaksun, N. and Kachitvichyanukul, V. (2009) Performance Evaluation of Warehouse with One-block Class-based Storage Strategy. Proceedings of Asia Pacific Industrial Engineering & Management Systems Conference, Kitakyushu, Japan.

AUTHOR BIOGRAPHIES

Natanaree Sooksaksun is a Doctoral Candidate in Industrial Engineering & Management, School of Engineering and Technology, Asian Institute of Technology, Thailand. She is on study leave from King Mongkut's University of Technology North Bangkok, Thailand. She received a Master of Engineering degree from King Mongkut's University of Technology Thonburi, Thailand.. Her research interests include warehouse design and operation. Her email address is <st105997@ait.ac.th>

Voratas Kachitvichyanukul is a Professor in Industrial & Manufacturing Engineering, School of Engineering and Technology, Asian Institute of Technology, Thailand. He has extensive experiences in modeling and simulation of manufacturing systems. His teaching and research interests include evolutionary algorithms for combinatorial optimization, simulation, planning and scheduling, high performance computing and applied operations research with special emphasis on industrial systems. His email address is <voratas@ait.ac.th>