International Journal of Applied Operational Research
Vol. 1, No. 1, pp. 19-33, Summer 2011

Journal homepage: www.ijorlu.ir

A Fuzzy Multi-Objective Class Based Storage Location Assignment

B. Maleki Vishkaei*, M. EbrahimNezhad Moghadam Rashti, B. Maleki Vishkaei

Received: March 1, 2011 ; **Accepted:** June 20, 2011

Abstract The required storage space and the material handing cost in a warehouse hinge on the storage implementation decision. Effects of storage area reduction on order picking and storage space cost are incorporated. Moreover, merchandises which are in the same shape and can be stored beside each other easily or goods that don't cause any danger like causing a fire if be in touch with each other, can be stored in one class together. In this paper first a multi-objective class based storage model is presented in which two objectives are considered; one is the sum of storage space cost and handing cost and the other is the quantitative objective "efficiency of storing products in one class". The demand rates and the second objective are evaluated with linguistic values. Fuzzy dynamic approach will be used to solve the proposed model considering an illustrative example to clarify it.

Keywords Warehouse Storage Planning, Class Based Storage, Storage Space, Multi-Objective Decision Making, Fuzzy Optimization, Dynamic Programming.

1 Introduction

All key performance indicators of a warehouse such as order picking time and cost, productivity accuracy and storage density are influenced by warehouse storage decisions [1]. Warehouse storage planning involves determining the storage policy, space requirement and specific location within the warehouse for each product. Random storage, dedicated storage and class based storage can be named as the common storage policies [2]. The implementation of class based storage location assignment involves determining the number of classes, product assignment to classes and storage location for each class. Cube-per-order index (COI) is used to assign product classes to storage location which capture item popularity and its storage space requirement [3]. COI is defined as the ratio of the item's storage space requirement (Cube) to its popularity (number of storage /retrieval request for the items).It is proved that n-class assignment in which products with lowest COI are stored in the most desirable location (i.e. closer to input/output points) gives optimal allocation in terms of order picking/storage time [4]. Muppani et al. [5] compared branch and bound algorithm with baseline dynamic programming algorithm in solving a class based storage location assignment which is an integer programming model with the goal of minimizing the sum of storage space cost and holding cost.

* **Corresponding author. (✉)**
E-mail: B.maleki.v@gmail.com (B. Maleki Vishkaei)

B. Maleki Vishkaei, M. EbrahimNezhad Moghadam Rashti
Department of Industrial Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran.

B. Maleki Vishkaei
Department of Electrical Engineering, Lahijan Branch, Islamic Azad University, Lahijan, Iran.

[Downloaded from ijaor.ir on 2026-11-28]

In real world, the products can't be assigned to classes freely. The shape of the products that are selected for a defined class should be similar so that they can be stored beside each other easily. Considering the efficiency of storing products that are in the same class as a new objective, changes the Muppani's model to a more realistic problem. In this paper the demand rates and the efficiency are evaluated with linguistic values which convert the crisp model to a fuzzy model. Therefore, we will face a fuzzy multi-objective class based storage model which needs a fuzzy algorithm to solve it. Dynamic programming (DP) is a powerful optimization approach for dealing with a large spectrum of complex problems involving sequential or multi-stage decision-making in many areas. For obvious reasons, the analysis of multi-stage decision-making problems by conventional DP is rather difficult under fuzzy environments. Assuming that Zadeh's fuzzy sets theory was an appropriate way to deal with uncertainties and imprecision in real-word problems, DP was one of the earliest fundamental methodologies to which fuzzy sets theory was applied leading to what might be called fuzzy dynamic programming (FDP). Excellent reviews of FDP appear in the literature [6, 7, 8]. Multi-objective dynamic programming relies heavily on the conventional DP technique and is used for solving problems that involve various objectives [9, 10, 11, 12, 13]. An effective and efficient fuzzy dynamic programming approach for hybrid multi-objective decision making problems with quantitative and qualitative objectives is presented by Lushu Li et al [14] which is used to solve the proposed model in this paper.

The paper is structured as follows: In section 2, the proposed model is formulated. In section 3, solving method will be described. An illustrative example is given in section 4 and section 5 concludes the paper.

2 Model Description

Using the literature [5] as a baseline, we assume that the storage and retrievals are performed in single command cycles and all products are stored and transported on identical storage media (e.g., pallets or totes). Other assumptions are described as follows:

- A class based storage policy is used
- An open location for an incoming load is selected randomly within its class.
- Each location is uniformly utilized and the assigned products are distributed homogeneously in the space allocated for the class which implies that the geometric center of the class is the same as the load center.
- An inventory decision has been made independently to the storage decision and all times required in the storage/ retrieval process, except travel times, are considered independent of storage allocation.
- There is no congestion between vehicles/cranes and loads are not relocated.
- Demand rates and the efficiency of two products being in the same class are considered as linguistic variables (New assumption).
- Once the storage locations are assigned they cannot be reshuffled during the planning horizon.
- There is a single input/output point.

The problem is to establish classes of products and allocate them to storage locations so that total cost of order picking/ handling and storage space is minimized and the efficiency of

[Downloaded from ijaor.ir on 2026-11-28]

the selected classes is maximized. In this section, we establish notations and then formulate a model for the problem defined.

2.1 Notation

The following notation is used in this paper:
Indices

c and c' : for classes $(c, c' = 1,2,...,C)$

l and l' : for storage locations $(l, l' = 1,2,...,L)$

p and p' : for products $(p, p' = 1,2,...,P)$

t : for time periods $(t = 1,2,...,T)$

Parameters:

a_l : footprint area for location l.

COI : cube-per-order index for product p.

d_l : distance of location l from the input and output point.

$\tilde{D}_p$: total number of picks for product p in the planning period which is a fuzzy number.

f : space cost in $ per square foot.

f_p : footprint density, that is, footprint area required to store one unit load of product p considering the stacking height.

h : order picking /handling cost in $ per foot.

$\tilde{I}_p^t$: storage level in unit loads planned for product p during period t which is a fuzzy number.

$\tilde{F}_{ij}$: efficiency of storing products I and j in one class which is a fuzzy number.

$\tilde{F}_i$: efficiency of storing product i in one class lonely, which is a fuzzy number.

$\tilde{F}_c$: efficiency of dedicated products to class c which is a fuzzy number.

Decision variables:

$$x_{pc} = \begin{cases} 1, & \text{if product p is assigned to class c} \\ \circ, & \text{otherwise} \end{cases}$$

$$y_{lc} = \begin{cases} 1, & \text{if location l is assigned to class c} \\ \circ, & \text{otherwise} \end{cases}$$

2.2 Storage space and handling cost

According to notation, the storage space cost for class C can be formulated as $f.\sum_{l}(a_l.y_{lc})$.

[Downloaded from ijaor.ir on 2026-11-28]

The handling cost depends on unit handling cost, centroid distance of the allocated space from the input/output point which is gained from $\sum_l (a_l.d_l.y_{lc}) / \sum_l a_l.y_{lc}$ and total number of picks $\sum_p \widetilde{D}_p.x_{pc}$ .Therefore, the sum of storage cost and handling cost equals to equation 1.

$$Z_1 = f.\sum_l (a_l.y_{lc}) + 2\sum_c \left[\left\{ \frac{\sum_l (a_l \times d_l \times y_{lc})}{\sum_l (a_l \times y_{lc})} \right\} \times (\sum_p (\tilde{D}_p \times x_{pc})) \right] \quad (1)$$

2.3 Average efficiency

The average efficiency of class c depends on the products which are located together in this class. In the following equation, the first term denotes the sum of linguistic variables when there are more one product in class C and the second one shows the efficiency of class C when only one product is dedicated to class c.

$$F_c = \sum_{i=1}^{p-1} \sum_{j=i+1}^{p} \tilde{F}_{ij} x_{ic} x_{jc} + \sum_{i=1}^{p} \tilde{F}_i x_{ic} \prod_{\forall j, j \neq i} (1 - x_{jc}) \quad (2)$$

To gain the average efficiency of class c, equation 2 should be divided to $\sum_{i=1}^{p-1} \sum_{j=i+1}^{p} x_{ic} x_{jc} + \sum_{i=1}^{p} x_{ic} \prod_{\forall j, j \neq i} (1 - x_{jc})$. Therefore, the total efficiency can be described as the following equation.

$$Z_2 = \frac{\sum_{\forall c} F_c}{\sum_{i=1}^{p-1} \sum_{j=i+1}^{p} x_{ic} x_{jc} + \sum_{i=1}^{p} x_{ic} \prod_{\forall j, j \neq i} (1 - x_{jc})} \quad (3)$$

2.4 The proposed model

According to the defined problem the model can be formulated as follows:

$$\min \quad Z_1 = f.\sum_l (a_l.y_{lc}) + 2\sum_c \left[\left\{ \frac{\sum_l (a_l \times d_l \times y_{lc})}{\sum_l (a_l \times y_{lc})} \right\} \times (\sum_p (\tilde{D}_p \times x_{pc})) \right]$$

$$\max \quad Z_2 = \frac{\sum_{\forall c} F_c}{\sum_{i=1}^{p-1} \sum_{j=i+1}^{p} x_{ic} x_{jc} + \sum_{i=1}^{p} x_{ic} \prod_{\forall j, j \neq i} (1 - x_{jc})}$$

[Downloaded from ijaor.ir on 2026-11-28]

Subject to:

$$F_c = \sum_{i=1}^{p-1} \sum_{j=i+1}^{p} x_{ic} x_{jc} + \sum_{i=1}^{p} \tilde{F}_i x_{ic} \prod_{\forall j, j \neq i} (1 - x_{jc})$$

$$COI_p x_{pc} \leq COI_{p'} x_{p'c'} \quad \forall p = p' \quad \text{and} \quad c < c'$$

$$l_{y_{lc}} \leq l'_{y_{l'c'}} \quad \forall l = l' \quad \text{and} \quad c < c'$$

$$\sum_{c} y_{lc} \leq 1 \qquad \forall l$$

$$\sum_{c} x_{pc} \leq 1 \qquad \forall p$$

$$\max_t \left[\sum_{p} (\tilde{I}_p^t f_p x_{pc}) \right] \leq \sum_{l} (a_l y_{lc}) \qquad \forall c$$

$$COI_P = \frac{f_p \max_t \left\{ I_p^t \right\}}{\tilde{D}_p / T}$$

$$x_{pc} \in \{\circ , 1\} \qquad \forall p , c$$

$$y_{lc} \in \{\circ, 1\} \qquad \forall l,c$$

1. Minimize the sum of storage space cost and handling cost.
2. Maximize the total efficiency(new objective)
3. Linguistic variables for products that are dedicated to class C.(new constraint)

4, 5. Together ensure that if a product that has lower COI is assigned to class C and products with higher COI assigned to class C' then C is located nearer to the I/O point that the C'.

6. Ensure that a storage location can be assigned to one class.
7. Ensure that each product is assigned to one and only one class.
8. Ensure that there is adequate storage space to hold the products in a class in each planning period.
9. Calculate the cube-per-order index
10, 11. Impose binary restrictions on decision variables.

3 The Solving Method

The proposed model has nonlinear objective function involving integer variables. The following steps are used to solve the model.

1. In order to get a marginal evaluation for demand rates, the concept of the total expected value of a fuzzy number, which identifies with the total integral value method is used. In this method for a triangular fuzzy number $\widetilde{A} = (a, b, c)$ and a level of optimism $\alpha \in [0,1]$ the total expected value $E(\widetilde{A})$ equals to $\frac{1}{2}(\alpha b + a + (1 - \alpha)c)$.
2. Products are indexed from p=1,2,…,P in increasing order of COI_p computed by following equation.

[Downloaded from ijaor.ir on 2026-11-28]

$$COI_P = \frac{f_p \max_t \left\{ I_p^t \right\}}{\frac{\widetilde{D}_p}{T}} \tag{4}$$

3. In the increasing order of distance from I/O point, the locations are indexed from l=1,2,…,L.
4. Fuzzy dynamic programming approach by Lushu Li et al[14] is used to solve the problem considering the priorities that are gained in steps 1,2 and other constraints of the model that are described in section 2.

In this proposed dynamic programming method, the amounts of $Z_{1c}(i,j)$ and $\widetilde{Z}_{2c}(i,j)$ are the amounts of the first and second objective functions where i is the last product assigned and j is the last location to class C.

As the second function is a qualitative objective, the total expected value of these fuzzy numbers will be used instead of them. For a triangular fuzzy number $\widetilde{Z}_{2c}(i,j) = (a,b,c)$ and a level of optimism $\alpha \in [0,1]$ the second objective function will be changed to $O_{2c}(i,j)$ according to equation 5.

$$Q_{2,c}(i,j) = E(\widetilde{Z}_{2c}(i,j)) = \frac{1}{2}\left(\alpha \mathrm{b} + \mathrm{a} + (1-\alpha)\mathrm{c}\right) \tag{5}$$

In the next step the objective functions should be normalized by equations 6, 7. Z_1^+, O_2^+ which are the biggest amounts and Z_1^-, O_2^- are the smallest amounts that are gained.

Using Equation 7 the second objective function is expected to be minimized as the first objective function.

$$Q_{1c}(i,j) = \frac{Z_1^+ - Z_{1c}(i,j)}{Z_1^+ - Z_1^-} \tag{6}$$

$$Q_{2c}(i,j) = \frac{O_{2c}(i,j) - O_2^+}{O_1^+ - O_1^-} \tag{7}$$

Equations 8, 9 represent the total sum of each objective function at each stage of dynamic programming.

$$a_{1c} = \mathrm{a}_{1,c\text{-}1}(\mathrm{i}', \mathrm{j}') + \mathrm{Q}_{1c}(\mathrm{i}, \mathrm{j}) \tag{8}$$

$$a_{2c} = \mathrm{a}_{2,c\text{-}1}(\mathrm{i}', \mathrm{j}') + \mathrm{Q}_{2c}(\mathrm{i}, \mathrm{j}) \tag{9}$$

[Downloaded from ijaor.ir on 2026-11-28]

Finally, Equation 10 will be calculated for the last nodes (in last nodes all the products are dedicated and no more products are remained). $\sum_{j=1}^{2} w_{ij} = 1$ and if the first objective function is prior to the second objective function, $w_1 > w_2$ and vice versa.

$$\mu = \frac{1}{1 + \frac{\sum_{j=1}^{2} [w_j (1 - a_{1p}(i,j)]^2}{\sum_{j=1}^{2} (w_j . a_{2p}(i,j))^2}} \tag{10}$$

The bigger μ is reached, the better solution is gained.

4 A numerical example

Consider a company that stores five different products and its warehouse contains 16 locations for storing. In each location 200 products can be stored and we use the rectilinear distance to calculate the distance of each location from the I/O point which is located in center of side. Table 1 shows the warehouse and the distances. Fig 1, 2 show the linguistic variables for the demand rates and efficiency.

Table 1 The distances of the warehouse's locations from I/O

30	24	24	30
24	18	18	24
18	12	12	18
12	6	6	12

I/O

[Downloaded from ijaor.ir on 2026-11-28]

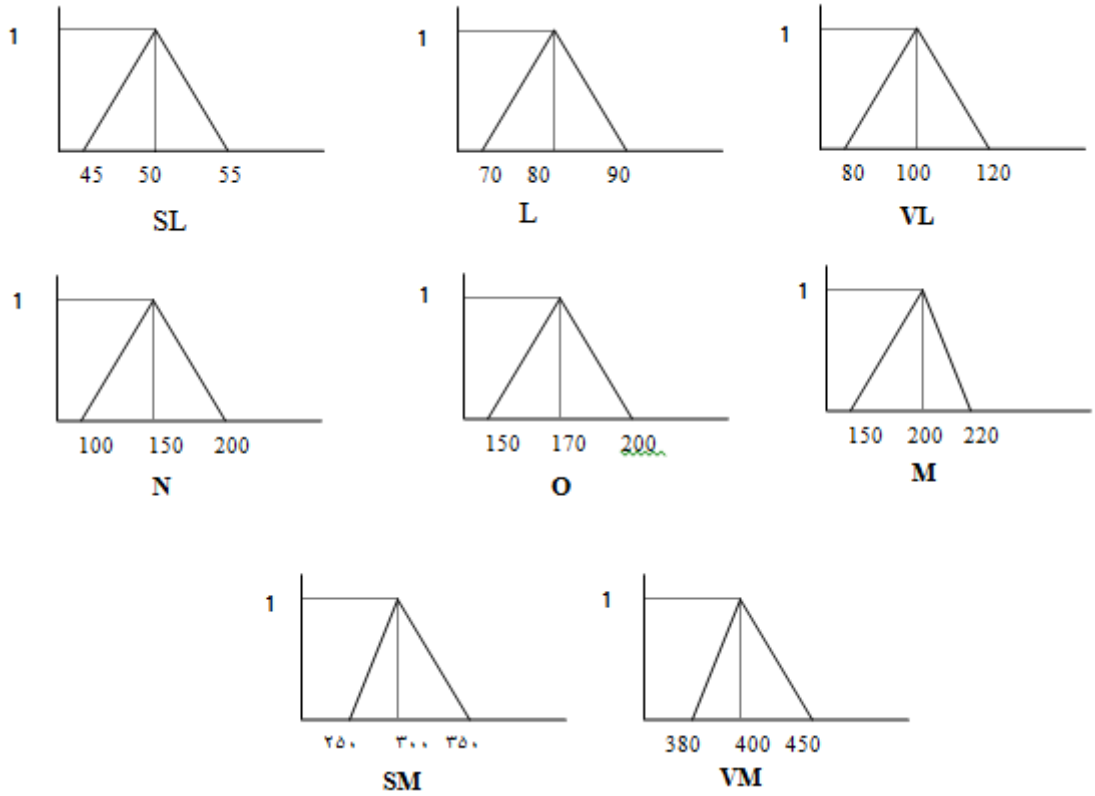


Fig. 1 Linguistic variables for demand rates

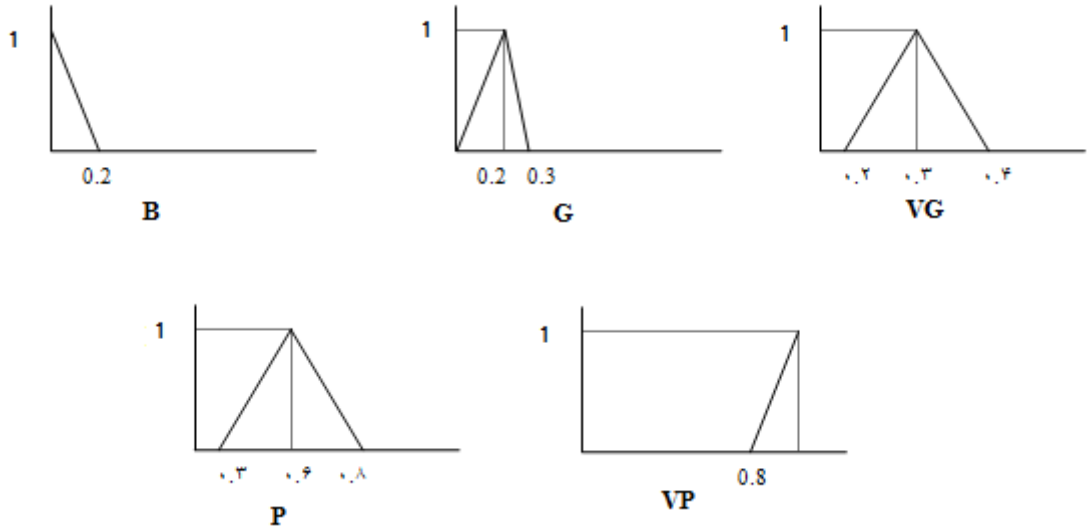


Fig. 2 Linguistic variables for $\widetilde{F}_{i,j}$, $\widetilde{\mathrm{F}}_{\mathrm{i}}$

[Downloaded from ijaor.ir on 2026-11-28]

T is fixed at 6 periods. In Table 2 the fuzzy demand rates for the next 6 periods are given and Table 3 contains the linguistic variables for each per of products and efficiency of storage each product in one class lonely.

Table 2 The amounts of demands rates

Product				
1	2	3	4	5
SL	VL	VL	M	M
SL	SL	VL	SM	SL
M	SM	SL	N	N
M	L	VL	SM	N
SL	VM	N	SL	M
SL	O	SL	N	M

Table 3 The relation between products and the amounts of $\widetilde{F}_{i,j}$, $\widetilde{F}_i$

i,j	$\widetilde{F}_{i,j}$	i,j	$\widetilde{F}_{i,j}$	i	$\widetilde{F}_i$
1,2	B	2,4	B	1	G
1,3	G	2,5	VP	2	G
1,4	VG	3,4	P	3	G
1,5	P	3,5	G	4	G
2,3	G	4,5	G	5	G

The steps of solving the model are described as follows:

Step 1: The marginal evaluations for the demand rates are shown in Table 3.

Step 2: according to the cube-per-order index for each product (Table 4) the priority of the products for storing near the I/O point are as follows:

Table 4 The amounts of COI for the products

P	1	2	3	4	5
COI	1.976	2.203	1.636	1.574	1.246

$$5 \longrightarrow 1 \longrightarrow 4 \longrightarrow 3 \longrightarrow 2$$

[Downloaded from ijaor.ir on 2026-11-28]

Table 5 Indexed locations according to their distances from I/O

16	14	13	15
12	10	9	11
8	6	5	7
4	2	1	3

I/O

Step 3: In the increasing order of distance from I/O point, the locations are indexed in Fig.7

Step 4: Using the fuzzy dynamic programming approach, Fig 3 shows the amounts of $Z_{1c}(i,j)$ and $O_{2c}(i,j)$. $Q_{1c}(i,j)$ and $Q_{2c}(i,j)$ are given in Fig 4 and Fig 5 contains $a_{1c}(i,j)$ and $a_{2c}(i,j)$ .In Fig 6 equation 13 is considered for the last nodes and best answer is defined. Therefore, the optimal solution consists of two classes. The first class includes products 5, 1 storing in locations 1 to 5 and the second class only contains product 4,3,2 which is stored in locations 6 to 8.

[Downloaded from ijaor.ir on 2026-11-28]

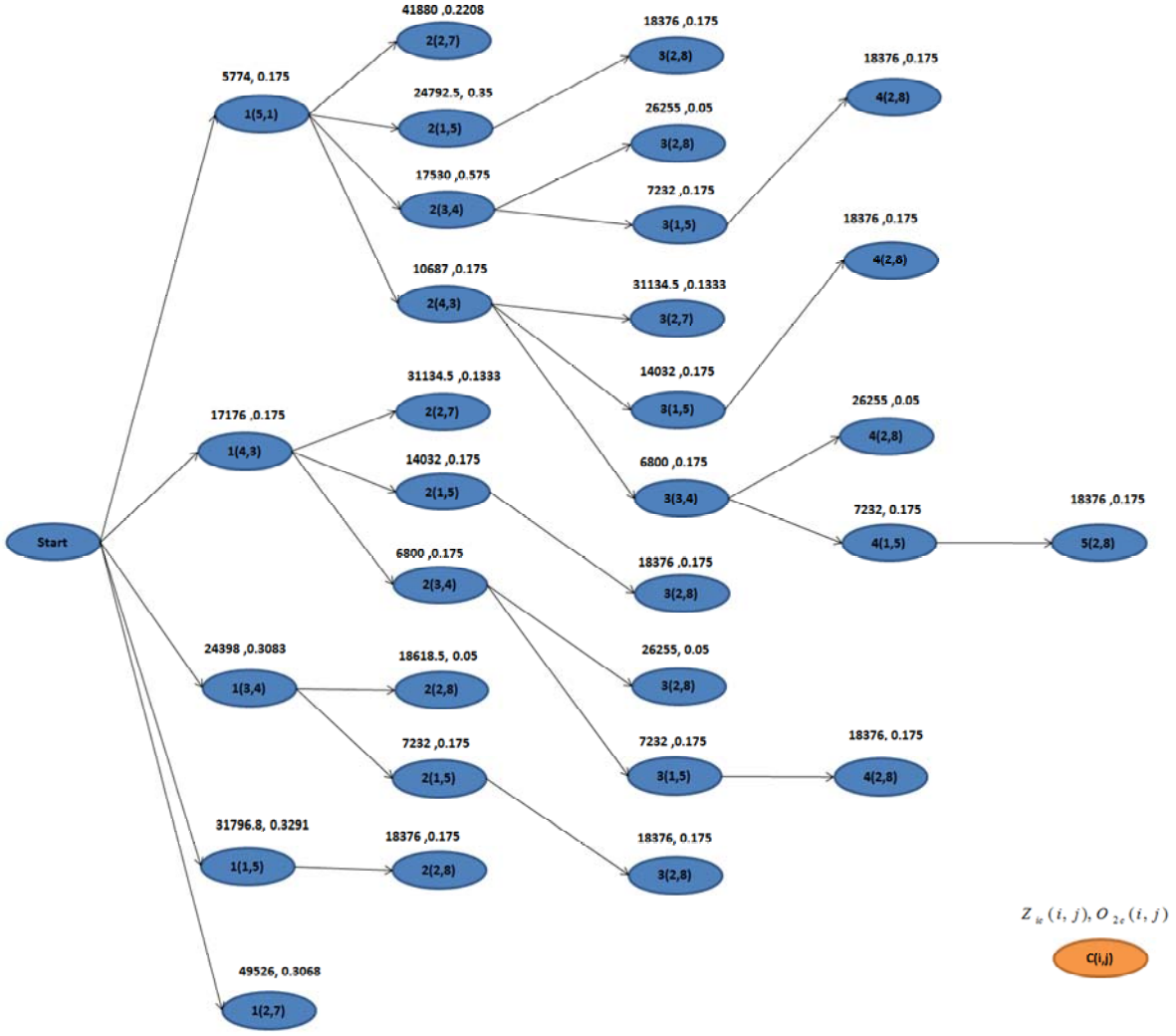


Fig. 3 Dynamic programming diagram for $Z_{1c}(i,j)$ and $O_{2c}(i,j)$

[Downloaded from ijaor.ir on 2026-11-28]

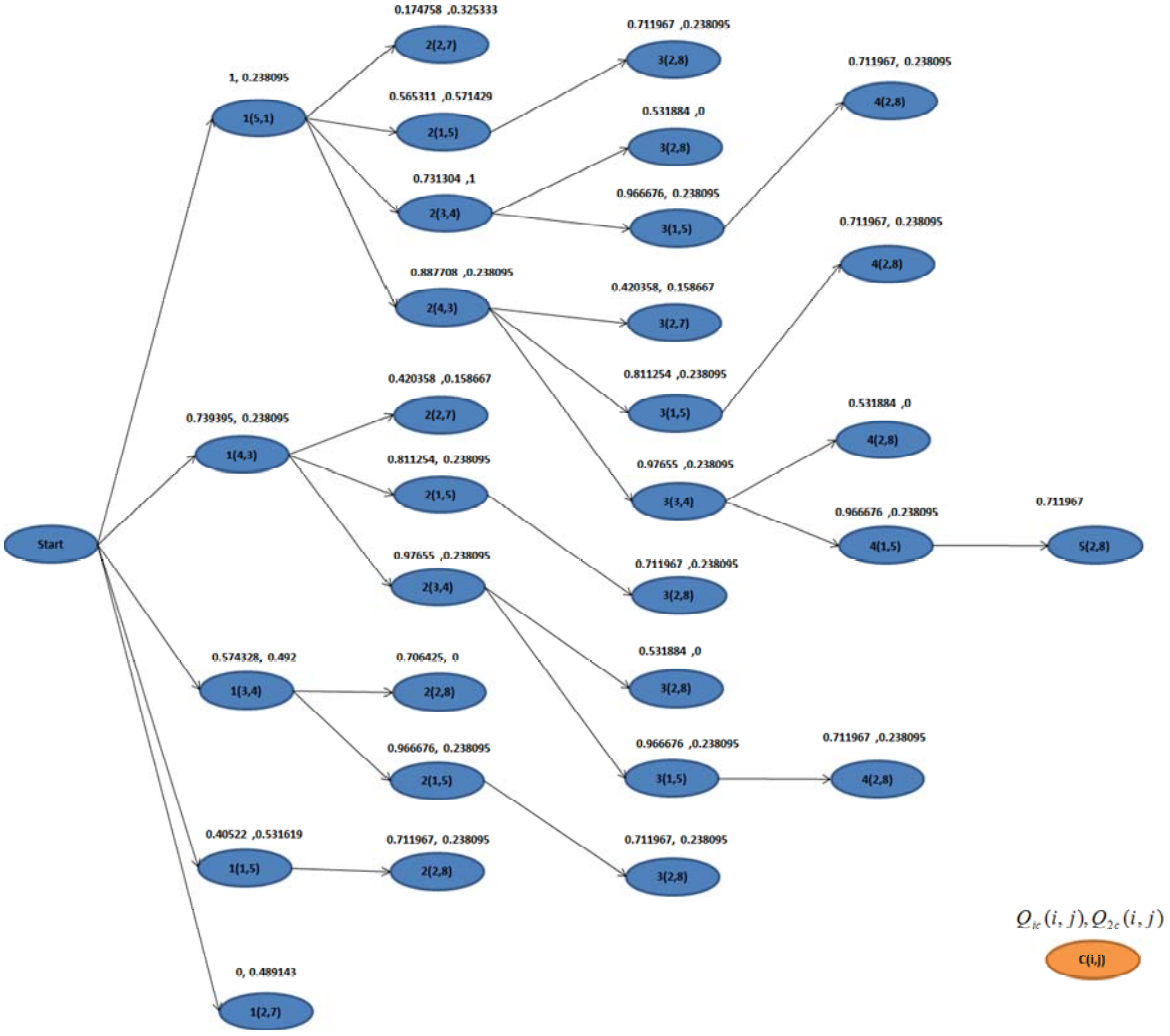


Fig. 4 Dynamic programming diagram for $Q_{1c}(i,j)$ and $\mathrm{Q}_{2c}(i,j)$

[Downloaded from ijaor.ir on 2026-11-28]

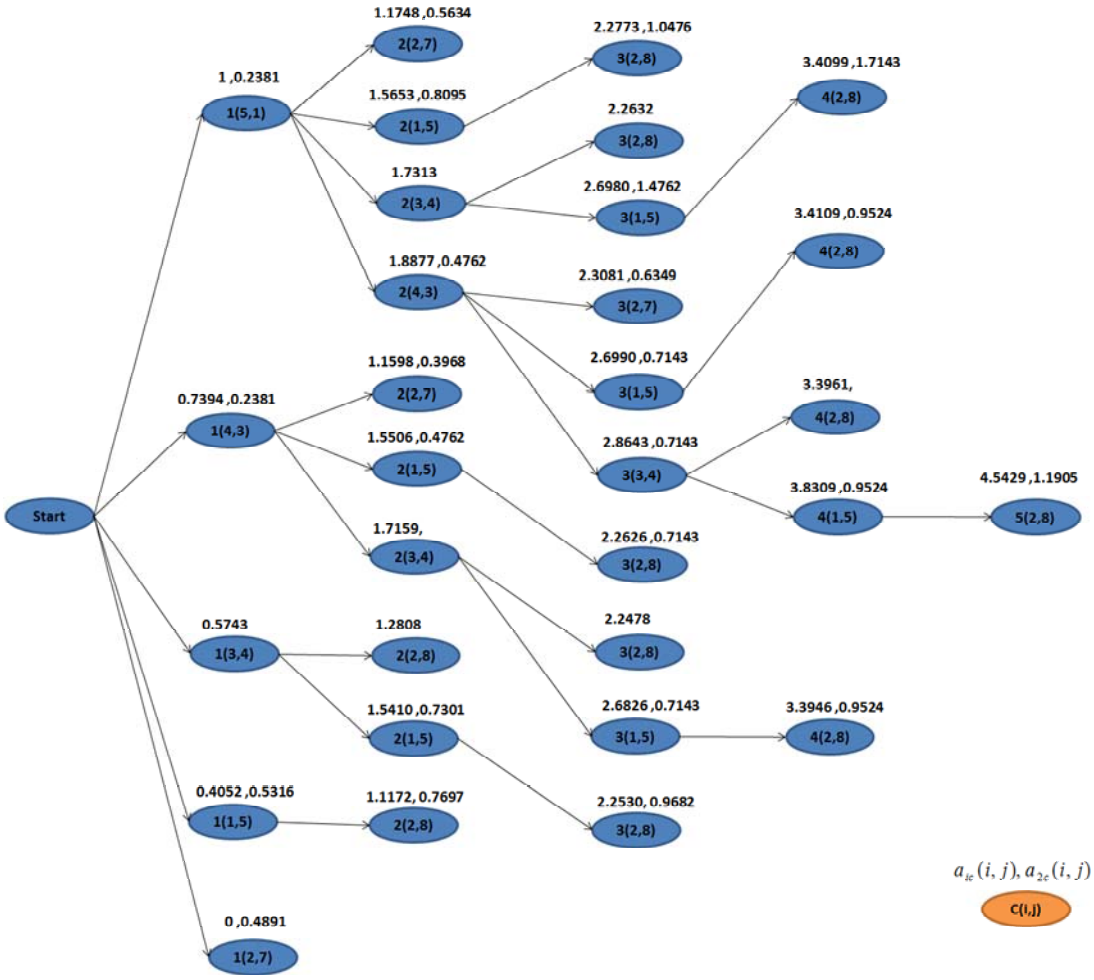


Fig. 5 Dynamic programming diagram for $a_{1c}(i,j)$ and $a_{2c}(i,j)$

[Downloaded from ijaor.ir on 2026-11-28]

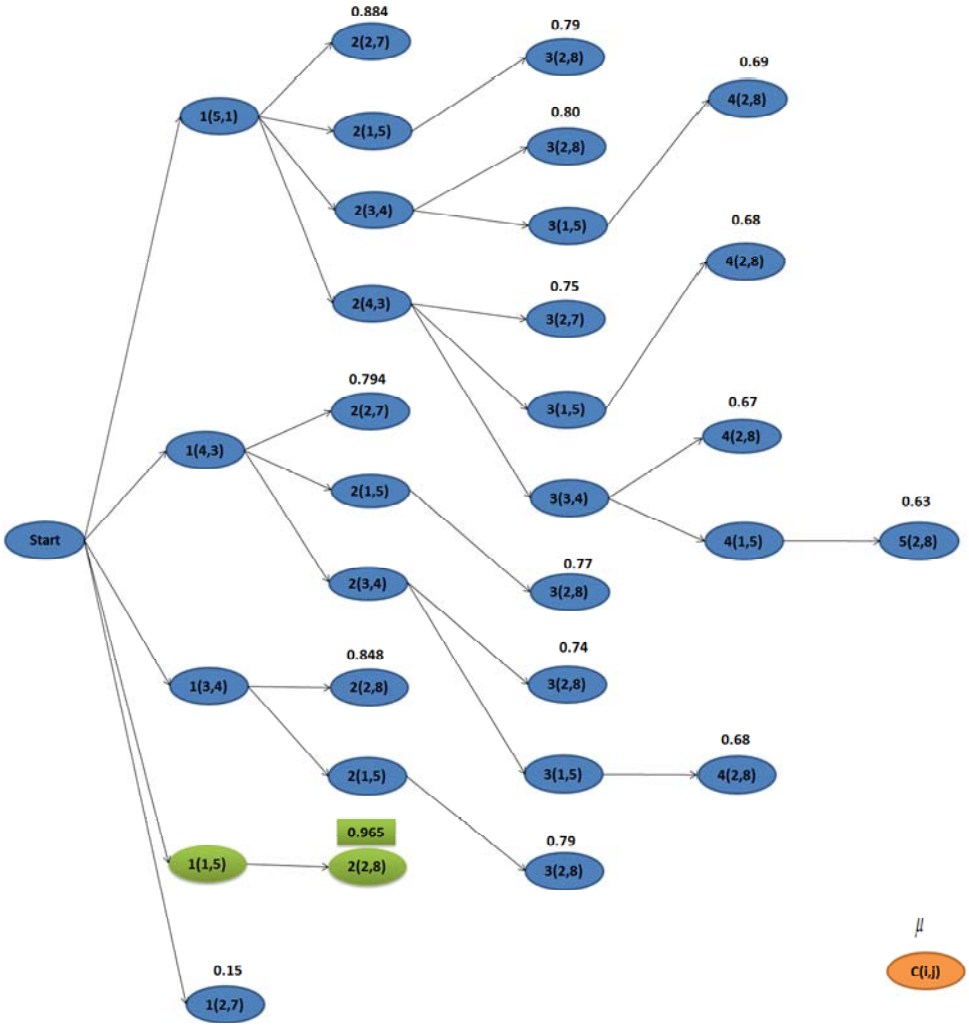


Fig. 6 The last dynamic programming diagram

5 Conclusion

In this paper a fuzzy dynamic programming approach is used to solve a nonlinear integer programming class based storage model to minimize the sum of storage and handling costs and maximize the efficiency of storing the products beside each other. To clarify the proposed fuzzy dynamic programming, an example considering five products is solved. The bigger the problem is, the longer time it takes to solve it. Therefore, fuzzy meta-heuristic algorithms could be a good solving method for very big problems which can be studied in future researches.

[Downloaded from ijaor.ir on 2026-11-28]

References

1. Frazelle, E., (2002). World-class Wareousing and Material Handling. McGraw-Hill, New York. USA.
2. Hausman, W. H., Schwarz, L. B., Graves, S. C., (1976). Optimal storage assignment in automatic warehousing systems. Management Science, 22(6), 629-638.
3. Heskett, J. L., (1963). Cube-Per-Order Index – A key to warehouse stock location. Transportation and Decision Management, 3, 27-31.
4. Goetschalckx, M., Ratliff, H. D., (1990). Shared storage policies based on the duration stay of unit loads. Management Science, 34(9), 1120-1132.
5. Muppani, V. R., Adil, G. K., (2008). A branch and bound algorithm for class based storage location assignment, Eouropean Journal of Operational Research, 189, 492-507.
6. Bellman, R. E., Zadeh, L. A., (1970). Decision making in a fuzzy environment. Management Sci, 17, 141-164.
7. Kacprzyk, J., (1997). Multistage fuzzy control: a model-Based approach to fuzzy control and decision making, Wile. New York, USA.
8. Kacprzyk, J., Esogbue, A. O., (1996). Fuzzy dynamic programming: main developments and applications. Fuzzy Sets and Systems, 81, 31-45.
9. Abo-Sinna, M. A., (1989). A dynamic programming approach for solving bi criterion allocation problems. Proc. 3rd Conf, on Operational Research and its Military Applications, Cairo, Egypt, 133-146.
10. Chankong, V., Haimes, Y. Y., (1983). Multi-objective Decision making theory and methodology. Series in Sysytem Science and Engineering, North-Holland, Amsterdam.
11. Hussein, M. L., Abo-Sinna, M. A., (1995). A fuzzy dynamic approach to the multi-criteria resource allocation problem. Fuzzy Sets and Systems, 69, 115-124.
12. Mine, H., Ohno, K., (1979). Decomposition of mathematical programming problems by dynamic programming. Internat. J. Systems Sci, 10, 370-385.
13. Su, C. C., Hsu, Y. Y., (1991). Fuzzy dynamic programming: an application to unit commitment, IEEE Trans. Power Systems, 6, 1231-1237.
14. Lushu Li, Lai, K. K., (2001). Fuzzy dynamic programming approach to hybrid multi-objective multistage decision-making problems. Fuzzy Sets and Systems, 117, 13-25.

[Downloaded from ijaor.ir on 2026-11-28]