

Pod Layout Problem in Kiva Mobile Fulfillment System Using Synchronized Zoning

Mengcheng Guan*, Zhenping Li

College of Information, Beijing Wuzi University, Beijing, China

Email: *gmengcheng@163.com, lizhenping66@163.com

How to cite this paper: Guan, M.C. and Li, Z.P. (2018) Pod Layout Problem in Kiva Mobile Fulfillment System Using Synchronized Zoning. *Journal of Applied Mathematics and Physics*, 6, 2553-2562.
<https://doi.org/10.4236/jamp.2018.612213>

Received: October 20, 2018

Accepted: December 17, 2018

Published: December 20, 2018

Copyright © 2018 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

This article studies the pod layout problem in the Kiva mobile fulfillment system which adopts the synchronized zoning strategy. An integer programming model for the pod layout problem is formulated under the premise of knowing the relationship of the pods and items. A three-stage algorithm is proposed based on the Spectral Clustering algorithm. Firstly, the pod similarity matrix and the Laplacian matrix are constructed according to the relationship of the pods and items. Secondly, the pods are clustered by the Spectral Clustering algorithm and assigned to each zone based on the cluster results. Finally, the exact locations of pods in each zone are determined by the historical retrieval frequency of items, using the real data of a large-scale Kiva mobile fulfillment system to simulate and calculate the order picking efficiency before and after the adjustment of the pod layout. The results showed that the pod layout using synchronized zoning strategy can effectively improve the picking efficiency.

Keywords

Pod Layout Problem, Spectral Clustering Algorithm, Kiva Mobile Fulfillment System

1. Introduction

Kiva mobile fulfillment system, in which robots carry pods to workstations, has been gradually applied [1]. The movement of pods replaces walking of pickers, which significantly improves the picking efficiency and reduces the labor cost and response time. Kiva mobile fulfillment system is different from the traditional picker-to-part warehouse. There were relatively few studies devoted to the operation problems in Kiva mobile fulfillment system. Previous studies focused on the order schedule, path planning and task assignment of robots. Wu studied

order schedule problem in Kiva mobile fulfillment system [2]. Han studied the path planning problem of robots [3]. Jiang proposed a new task assignment algorithm to solve the task assignment problem of robots [4]. Shen established a reconfigurable storage space model for the dispatch and path planning problem [5]. Yan considered the deadlock and collision problem of robots [6]. Li examined the issue of pod selection which is to use as few pods as possible to complete order picking operation [7]. All of these are based on a small-scale Kiva mobile fulfillment system, which has only one picking zone.

With the expansion of the scale of the warehouse, the number of pods increased, and the pod layout directly affects the picking efficiency. According to the survey of different warehouses, we found that most of them execute pod layout randomly or by experience. It exposes some problems, such as the long-distance movement of robots, and the traffic congestion problem.

The traditional picker-to-part warehouse usually uses synchronized zoning which divides the order picking area into zones. Each picker is assigned to a zone. The advantages of synchronized zoning include the fact that each picker is responsible for order picking in the zone, and reduced traffic congestion, and pickers are familiar with the item locations in the zone [8].

We can imagine that if we adopt synchronized zoning in Kiva mobile fulfillment system, there are several benefits include the fact that it will reduce the long-distance movement, traffic congestion, total moving distance, and the order completion time, and it can decrease the cost of warehouse management by opening one or several picking zones when the number of orders is at a low peak. At present, there is no direct research achievement on the synchronized zoning in Kiva mobile fulfillment system.

Synchronized zoning in Kiva mobile fulfillment system divides the large-scale warehouse into several picking zones so that each zone basically contains the same number of item types and robots in each zone are fixed. The pod layout problem aims to separate the pods into several groups and put each group in a zone, on the premise of knowing the items stored in each pod. In order to ensure each zone basically contains the same number of item types, we cluster the pods and assign the pods to each zone. Therefore, the suitable clustering method is the key to the research, directly affects the results.

Cluster analysis is an important branch of unsupervised learning in machine learning. It divides the target into multiple clusters, which attempts to achieve the target that the same cluster has a higher similarity and different clusters have lower similarity. Kiva mobile fulfillment system stores items which always have a tiny size. The large-scale warehouse always has thousands of pods and item types, which make the pod-item matrix high-dimension and sparse. The traditional clustering methods are ineffective and easy to converge to the local optimum when handling the high-dimension data. After comparison, we choose Spectral Clustering algorithm to cluster the pods, which is a hotspot in machine learning field in recent years. As an improved K-means algorithm with the advantages of converging to global optimization and high efficiency, it has been

widely used in the field of data dimensionality reduction, irregular data clustering, and image segmentation [9].

The rest of the article is organized as follows. Section II describes the problem and formulates an integer programming model. Section III proposes a three-stage algorithm for the model. Section IV describes the experiments and presents the results, Section V draws conclusions from the study.

2. Problem Description and Mathematical Model

2.1. The Warehouse Layout

The layout of Kiva mobile fulfillment system using synchronized zoning is shown in **Figure 1**. Pods are arranged above the warehouse. Each pod has several bins to store items, and the blank areas next to the pods are aisles. The picking workstations and the robot's initial docking area are below the warehouse. The dotted line is the dividing line of each zone, dividing the warehouse into several zones. Each zone has one workstation. In the warehouse which does not adopt synchronized zoning, when the warehouse receives an order, the system assigns the order to a workstation randomly while system assigns the order to the workstation

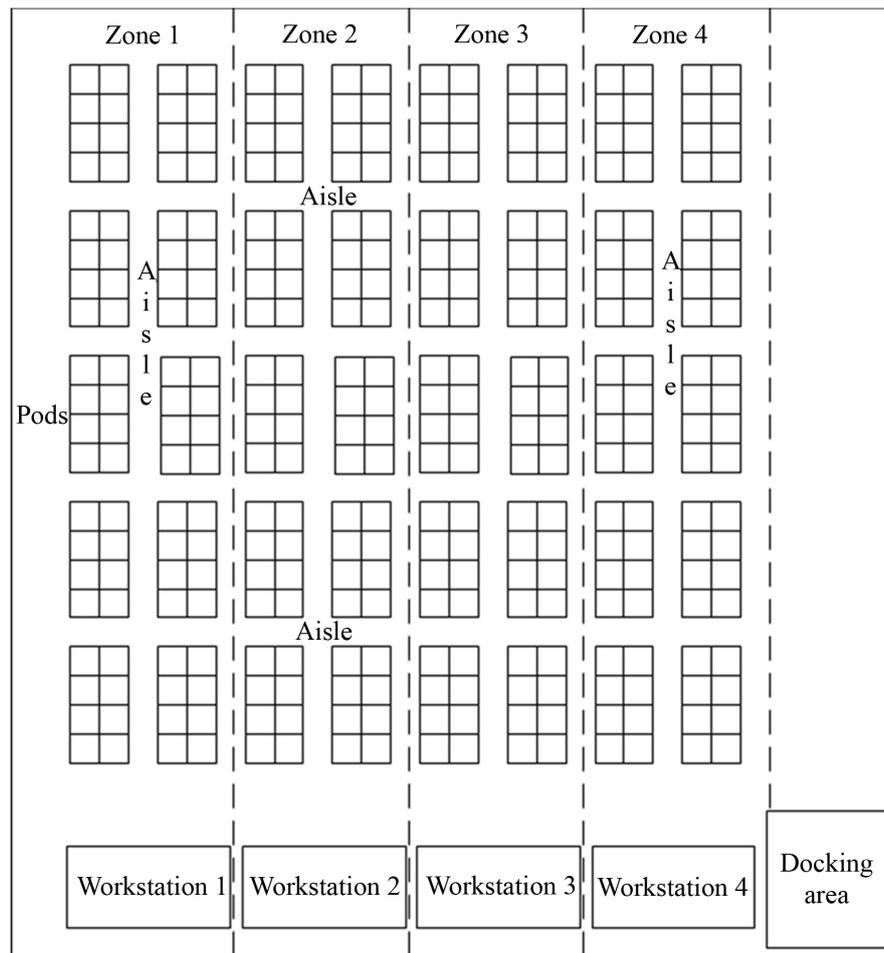


Figure 1. The layout of Kiva mobile fulfillment system using synchronized zoning.

which can complete the order with the least number of pods to be moved in the warehouse adopting synchronized zoning. Then the system sends tasks to the robots, and they depart from the docking area to the targeting pods and carry the pods to the corresponding workstations waiting for the picking. After completing the picking operation, the robot will return the pods to their original locations.

2.2. Problem Description

The Pod layout problem in the large-scale Kiva mobile fulfillment system can be described as: there are P zones in the warehouse, and each zone has L pods positions and a workstation. The pods and the workstation positions coordinate are known. There are M pods and N items in the warehouse. Where should M pods be placed in can improve warehouse operation efficiency and reduce the order completion time?

The following are parameters.

Indices

i , l —Number of pods, such that $i = 1, 2, \dots, M$.

j —Number of items, such that $j = 1, 2, \dots, N$.

k —Number of pod positions in each zone, such that $k = 1, 2, \dots, L$.

p —Number of zones, such that $p = 1, 2, \dots, P$.

o —Number of orders, such that $o = 1, 2, \dots, R$.

Parameters

M —Total number of pods.

N —Total number of item types.

P —Total number of zones.

L —Total number of pod positions of each zone.

R —Total number of orders.

$$s_{ij} = \begin{cases} 1, & \text{if item } j \text{ is put on the pod } i \\ 0, & \text{otherwise} \end{cases}$$

S —The relationship matrix of pods and items.

$$b_{oj} = \begin{cases} 1, & \text{if order } o \text{ contains the item } j \\ 0, & \text{otherwise} \end{cases}$$

B —The relationship matrix of orders and items.

w_{ij} —pod similarity value between pods i and l .

W —pods similarity matrix.

$D_p = \{d_{1p}, d_{2p}, \dots, d_{lp}\}$ —the distance between the position and its corresponding workstation in zone p .

The relationship matrix S of pods and items is defined as:

$$S = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1N} \\ s_{21} & s_{22} & \cdots & s_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ s_{M1} & s_{M2} & \cdots & s_{MN} \end{bmatrix}$$

The relationship matrix B of orders and items is defined as:

$$B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1N} \\ b_{21} & b_{22} & \cdots & b_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ b_{R1} & b_{R2} & \cdots & b_{RN} \end{bmatrix}$$

Pod similarity matrix W is defined as:

$$W = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1M} \\ w_{21} & w_{22} & \cdots & w_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ w_{M1} & w_{M2} & \cdots & w_{MM} \end{bmatrix}$$

Decision variables

$$y_{oi} = \begin{cases} 1, & \text{if order } o \text{ needs the pod } i \\ 0, & \text{otherwise} \end{cases}$$

$$x_{ikp} = \begin{cases} 1, & \text{if pod } i \text{ place in the } k \text{ position of zone } p \\ 0, & \text{otherwise} \end{cases}$$

2.3. Integer Programming Model

The integer programming model is as follows:

$$\min \sum_{o=1}^R \sum_{p=1}^P \sum_{k=1}^L \sum_{i=1}^M d_{ip} y_{oi} x_{ikp} \quad (1)$$

Subject to:

$$\sum_{k=1}^L \sum_{p=1}^P x_{ikp} = 1, \quad \forall i \quad (2)$$

$$\sum_{i=1}^M x_{ikp} \leq 1, \quad \forall p, k \quad (3)$$

$$\sum_{i=1}^M s_{ij} y_{oi} \geq b_{oj}, \quad \forall o, j \quad (4)$$

$$x_{ikp} = 0, 1; y_{oi} = 0, 1, \quad \forall i, k, p \quad (5)$$

Equation (1) aims to minimize the total distance of robots. (2) indicates that a pod can only be placed in one pod location. (3) indicates that one pod location can be placed at most one pod. (4) ensures each item of the order can be picked up from the pods which are carried to the workstation. (5) is a basic constraint.

There are up to thousands of pods and items in the large-scale Kiva mobile fulfillment system, so it is necessary to propose an efficiency algorithm to solve the problem.

3. Three-Stage Algorithm

The items on each pod are known, in order to make the item types of each zone are as similar as possible, this article proposes a three-stage algorithm: 1) the pod similarity matrix and the Laplacian matrix are constructed according to the pods

and items relationship; 2) the pods are clustered using Spectral Clustering algorithm and assigned to each zone; 3) the exact pod position in each zone is determined by the historical retrieval frequency of items.

Spectral Clustering is a hot topic in machine learning in recent years [10] [11] [12] [13] [14]. This method utilizes the eigenvector of the Laplacian matrix to cluster. The Spectral Clustering algorithm can be regarded as a problem of finding the optimal graph, and each sample is considered as a vertex V in the graph. The similarity between samples is set to the weight of the edge E between the vertices, so we obtain an undirected weighted graph $G = (V, E)$ based on sample similarity. In graph G , we can convert the clustering problem of a sample point into a graph partitioning problem [15]. The objective of partitioning based on graph theory is to maximize the internal similarity of each subgraph and minimize the similarity among the subgraphs [16].

3.1. The First Stage

This stage aims to construct the pod similarity matrix and the Laplacian matrix. Each pod in Kiva mobile fulfillment system has several bins to store items. Therefore, the similarity value between two pods can be calculated by the item types of the two pods. Similarity measure determines the similarity, and we choose the cosine similarity measure to compute the similarity value among the pods after comparing different similarity measure.

Cosine similarity calculation method is shown in the following:

$$w_{ij} = \cos(X_i, Y_j) = \frac{X_i \cdot Y_j}{|X_i| \cdot |Y_j|}$$

Row vectors X_i and Y_j represent the item stored in two pods, corresponding to the i th and j th row vectors of matrix S , respectively. So we can obtain pod matrix W according to matrix S .

Laplacian matrix $L = D - W$, D is the degree matrix, and the elements of its diagonal D_{ii} is the sum of the elements in the i th row or j th column of the matrix W . The normalized Laplacian matrix L_{sym} is calculated as follows:

$$L_{sym} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$$

$I \in R^{n \times n}$ is the unit matrix.

The specific steps of the first stage:

Input: matrix S , historical order data.

Step 1: Calculate matrix W , degree matrix D and Laplacian matrix L .

Step 2: Normalized the Laplacian matrix L .

Step 3: Calculate the historical retrieval frequency of each item.

Output: Pod similarity matrix W , normalized Laplacian matrix L_{sym} , the historical retrieval frequency of each item.

3.2. The Second Stage

This stage uses the Spectral Clustering algorithm to cluster the pods and assigns

them to each zone by a proposed greedy algorithm. Assigning different clusters of pods to the same zone ensures the similarity of item types among different zones and ensures that the number of pods does not exceed the pod capacity of each zone.

The specific steps of the second stage:

Input: Normalized Laplacian matrix L_{sym} , number of clustering k .

Step 1: Do eigen-decomposition of L_{sym} and obtain the smallest k eigenvectors and eigenvalues.

Step 2: Cluster the eigenvectors by K-means algorithm and obtain the pod clusters.

Step 3: Assign the pods to zones. From the first cluster to the last cluster, pods are assigned to zones. If a cluster has unallocated pods, these pods will be assigned to zones in general according to the total number of pods in the cluster.

Step 4: Repeat Step 3 until all the pods in all clusters are assigned.

Output: Pod assignment results.

3.3. The Third Stage

This stage determines the pod's exact position in each zone according to item's historical retrieval frequency.

The specific steps of the third stage:

Input: The corresponding pod assignment results for each zone, item's historical retrieval frequency, matrix S , the pod location coordinate data in each zone.

Step 1: Calculate the sum of historical retrieval frequency of the items in each pod as the pod's historical retrieval frequency.

Step 2: All pods in each zone are sorted in decreasing order according to the pod's historical retrieval frequency.

Step 3: Allocate the first pod to the pod position that has the shortest distance from its workstation, until all of the pods have been allocated.

Output: The pod layout results.

4. Computational Experiments

4.1. Experiment Datasets

Experiment datasets comes from a large-scale Kiva mobile fulfillment system in China. The warehouse has 1850 pods, nine workstations. The datasets include pod data, bin data and 40,000 historical order line data.

Due to the warehouse is large, the picking efficiency is not ideal; we are planning to adopt the synchronized zoning to improve the picking efficiency, without adjusting the item's location. Divide the warehouse into 5 zones to improve picking efficiency by adjusting the pod layout.

4.2. Experiment Results Analysis

The historical retrieval frequency of items can be obtained using 80% of the historical order line data. Using 20% of the historical order line data to verify the

Table 1. Simulation results.

Indicator	Unpartition	synchronized zone order picking and cluster num k						
		10	20	30	40	50	60	70
ACT ^a	298.3	212.4	215.7	216.9	219	214.8	217.6	216.9
ACTR ^b	0	-28.80%	-27.70%	-27.30%	-26.60%	-28.00%	-27.10%	-27.30%
ARD ^c	68.3	60.8	61.3	62.3	61.8	61.4	62.3	61.2
ARDC ^d	0	-10.90%	-10.20%	-8.80%	-9.50%	-10.00%	-8.80%	-10.30%

^aAverage order completion time(s); ^bAverage order completion time rate (compared to unpartitioned); ^cAverage robot moving distance(m); ^dAverage moving distance rate (compared to unpartitioned).

proposed model and algorithm. The simulation experiment is implemented with MATLAB 2017b on an Intel I5-7500, 8GB RAM computer.

From **Table 1**, we can see that when using the synchronized zoning, the average order completion time and the average robots moving distance corresponding to different k are slightly different but significantly lower than the results of not adopting the strategy. In this experiment, the picking efficiency is maximized in the case of 10 clusters. The average completion time of orders was reduced by 28.81%, and the average robot moving distance was reduced by 10.90%. Therefore, the synchronized zoning in large-scale Kiva mobile fulfillment system can effectively improve warehouse picking efficiency and shorten order completion time, decrease the average robot moving distance.

5. Conclusions

In this article, by analyzing the operation flow of a large-scale Kiva mobile fulfillment system, we proposed the synchronized zoning and an integer programming model for pod layout problem. A three-stage algorithm based on Spectral Clustering algorithm is proposed to solve the problem. The validity of the model and algorithm is validated by the simulation experiment based on the real data of an enterprise. The results showed that the synchronized zoning can not only reduce the order completion time but also reduce the average robot moving distance.

The number of clustering in the algorithm is artificially determined, which can utilize Elbow Method or other methods to automatically determine the number of clustering. This article studied the pod layout problem only if the items that are stored in each pod are known, and does not take into account whether the items stored on each pod are reasonable. We can study the joint optimization of the storage assignment problem and pod layout problem in the future. Kiva mobile fulfillment system is one of the main development directions of warehouse, and it is a major research direction to apply the method of data mining or machine learning to solve the problems.

Funded

This work is funded by the National Natural Science Foundation of China (Grant

No. 71771028, 71540028) and the High-Level Innovation Team Support Program in Universities of Beijing (Grant No. PXM2018-014214-000024).

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Wurman, P.R., D'Andrea, R. and Mountz, M. (2008) Coordinating Hundreds of Cooperative, Autonomous Vehicles in Warehouses. *AI Magazine*, **29**, 9-20.
- [2] Wu, Y., Meng, X., Wang, Y. and Hu, J. (2016) Order Sequence Optimization for "Part-to-Picker" Order Picking System. *Journal of Mechanical Engineering*, **52**, 206-212.
<https://www.scopus.com/inward/record.uri?eid=2-s2.0-4962651709&partnerID=40&md5=55071d82e8f39afe5f05270f98f2ae4f>
- [3] Han, M.M., Liu, J., Wu, S. and Wang, J.T. (2017) Path Planning of Mobile Robot Based on Particle Swarm Optimization Algorithm. *Journal of Computer Applications*, **37**, 2258-2263.
- [4] Dong, J. and Xin, X. (2017) Multi-Robot Dynamic Task Allocation Algorithm Based on Pareto Improvement. *Journal of Computer Applications*, **37**, 3620-3624.
- [5] Shen, B., Ningbo, Y.U. and Liu, J. (2014) Intelligent Scheduling and Path Planning of Warehouse Mobile Robots. *CAAI Transactions on Intelligent Systems*, **9**, 659-664.
- [6] Yan, X., Zhang, C. and Qi, M. (2017) Multi-AGVs Collision-Avoidance and Deadlock-Control for Item-to-Human Automated Warehouse. 2017 *International Conference on Industrial Engineering, Management Science and Application (ICIMSA)*, Seoul, 13-15 June 2017, 1-5.
- [7] Li, Z.P., Zhang, J.L., Zhang, H.J. and Hua, G.W. (2017) Optimal Selection of Movable Shelves under Cargo-to-Person Picking Mode. *International Journal of Simulation Modelling (IJSIMM)*, **16**, 145-156. [https://doi.org/10.2507/IJSIMM16\(1\)CO2](https://doi.org/10.2507/IJSIMM16(1)CO2)
- [8] 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. <https://doi.org/10.1016/j.ejor.2006.07.009>
- [9] Guan, T. and Yang, T. (2014) Analysis of General Model and Classical Algorithms for Spectral Clustering. *Pattern Recognition & Artificial Intelligence*, **27**, 1015-1025.
- [10] Arias-Castro, E., Lerman, G. and Zhang, T. (2017) Spectral Clustering Based on Local PCA. *The Journal of Machine Learning Research*, **18**, 253-309.
- [11] DeFord, D.R. and Pauls, S.D. (2017) Spectral Clustering Methods for Multiplex Networks. arXiv:1703.05355.
- [12] Chen, X.j., Peng, S., Huang, J.Z., Nie, F.P. and Ming, Y. (2017) Local PurTree Spectral Clustering for Massive Customer Transaction Data. *IEEE Intelligent Systems*, **32**, 37-44. <https://doi.org/10.1109/MIS.2017.41>
- [13] Jiang, C., Xie, H. and Bai, Z. (2017) Robust and Efficient Computation of Eigenvectors in a Generalized Spectral Method for Constrained Clustering. In: Singh, A. and Zhu, J., Eds., *Artificial Intelligence and Statistics*, PMLR, Fort Lauderdale, FL, 757-766.
- [14] Han, X., Yao, H. and Zhong, G. (2017) Handwritten Text Line Segmentation by

Spectral Clustering. *Eighth International Conference on Graphic and Image Processing (ICGIP 2016)*, International Society for Optics and Photonics, 102251A.

- [15] Dhillon, I.S., Guan, Y. and Kulis, B. (2004) A Unified View of Kernel K-Means, Spectral Clustering and Graph Cuts. Citeseer.
- [16] Shi, J. and Malik, J. (2005) Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, **22**, 888-905.