Optimization for Input/Output Routing of Multi-arms Stacker Crane in Automated Warehouse, Based on Tabu Search

Hu Guiyan, Du Zhiping, Sun Weihua

(School of Logistics, Bejing Wuzi University, Beijing, 101149, China)

Abstract

Automated Storage/Retrieval System is the essential component for the efficiency of a logistics centre. The optimal input and output routing problem of stacker cranes with multi-arms in AS/RS was treated. To assign the SC, lots of items must be partitioned into several identical sub-groups. A routing of an SC is defined as an association of storing items of a sub-group into designed cells and retrieving items of a sub-group stored simultaneously. The mean tour time of SC was minimized by dividing into groups, pairing of items and routing of SC at the same time. The computational complexity of the problem including the size of solution space was discussed and an approximation algorithm based on heuristic and tabu search was also proposed. Then, we perform the proposed approximation algorithm by some numerical experiments based on practical conditions, and demonstrate it is efficient in a real automated warehouse.

Key words: optimal input/output routing, multi-arms stacker crane, mean tour time, tabu search

1. Introduction

AS/RS is the essential component of a logistics centre. In an AS/RS, there are many racks with high bay pallet setting in parallel and many SCs running in aisles between racks. The input/output scheduling is the most crucial for efficiency of an AS/RS of big scale. In this study, a SC considered holding a few items can simultaneously and automatically store and retrieve equivalent items to and from racks, respectively. To assign SC, lots of items input/output must be partitioned into several identical sub-groups. A routing of a SC is defined as an association of storing items of a sub-group into designed cells and retrieving items of a sub-group stored simultaneously. We discuss the optimal input/output problem of items partition and SC routing to minimize the mean tour time and propose an approximation algorithm based on tabu search and heuristic. Then we perform some numerical experiments based on practical conditions and demonstrate the algorithm efficiency in real automated warehouse.

2. Model of optimal input/output routing problem

We consider an AS/RS composing of $2L$ parallel racks with same function and L identical SCs. Each rack with w width and h height consists a columns and b rows composing $c=ab$ cells. The entrance/exit locates at low left of racks, i.e., a=1， b=0.

The capacity of a SC is defined by its held items, generally referred to the number of its arms. Each arm is considered to hold an item. We assume a SC running in both horizontal and vertical directions simultaneously can hold m items at one time. At most, a routing of a SC can input m items and output m items, respectively.

In the condition of M items should be stored and N items should be retrieved, the number (n) of SC routing will be

$$n = \max\left(\lceil M / m \rceil ,\ \lceil N / m \rceil \right)$$

Without losing of generality, items input (*M*) are presumed to be equivalent to items output (*N*), i.e., $M=N$, otherwise necessary dummy items should be added at the input/output location. To keep high efficiency, SC is considered to store and retrieve items to and from racks simultaneously.

firstly, the M input/output items are partitioned into n subgroups (n is natural number) each having m items, referred as $I_1, I_2, I_3, \cdots, I_n$ and $O_1, O_2, O_3, \cdots O_n$, respectively. Secondly, an input subgroup and an output subgroup form an I/O pair. Then, we define the routing of SC as starting with items to store and ending with items to retrieve at the input/output locations of racks. In any feasible tour, the SC is constraint to hold at most m items. The optimal input/output routing of multi-arms SC is depended on the I/O partitions, the I/O pairs and the routings of SC to be performed.

There are many literatures discussing the scheduling models of AS/RS [1]. Most of them assume a man-aboard SC serving many items in a single tour and focus on sequencing (i.e., touring) for such a SC to visit many cells in a rack or racks. The most fundamental one is the order picking problem in which only retrieving operations is performed[2]. However, our model concentrates on unmanned multi-arms SC which cannot serve many items in a single tour. We substantiate the optimal sequencing is most fundamental for efficiency of AS/RS as discussed in the next.

3. Computational complexity of optimal input/output routing

To inquiry the solution space of the optimal input/output routing, numbers of I/O partitions, I/O pairs and SC tours are calculated as follow:

1). The number of different input (output) partitions is given by

$$g(M,m)=\frac{M!}{(m!)^n n!} \tag{1}$$

2). The number of different input /output pairs is

$$h(M,m)=\{g(M,m)\}^2 n!=\frac{(M!)^2}{n!(m!)^{2n}} \tag{2}$$

3). We assume that at any time in a tour the number of items stored must be equal to or more than that retrieved. The number of resulted feasible tours is

$$r(m)=\frac{(2m)!}{m+1} \tag{3}$$

4). Thus, according to the previous, the size of solution space results in

$$s(M,m)=h(M,m)\times r(m)=\frac{(2m)!(M!)^2}{(m+1)n!(m!)^{2n}} \tag{4}$$

According to formula 4, we enumerate the sizes of solution space in different M and m values showed in table 1. We perceive that the size of solution space becomes enormous quickly as M or m enlarging and consider an efficient approximation algorithm is needed in practice.

Table1. Number of different input/output routing (size of solution space)

m / M	1	2	3	4
$4m$	24	2.12×10^{6}	1.02×10^{12}	1.34×10^{18}
$6m$	720	6.22×10^{11}	4.71×10^{21}	1.18×10^{32}
$8m$	4.03×10^{4}	1.33×10^{18}	6.09×10^{32}	1.14×10^{48}
$10m$	3.63×10^{6}	1.24×10^{25}	9.55×10^{44}	3.68×10^{65}
$15m$	1.31×10^{12}	4.01×10^{44}	8.91×10^{78}	1.68×10^{114}

4. Approximation algorithm

The list of input items $I=(i_1,i_2, \ldots , i_M)$ is partitioned into a list of n input subgroup, i.e., $I=(I_1,I_2,\ldots,I_n)$, while $I_k=(i_{k1},i_{k2},\ldots,i_{km}),k=1,2,\ldots,n$. stands for a list of m

input items that to be visited by k-th SC. Heuristics algorithm is determined by an input/output scheduling problem, and I is investigated by tabu search. The list of practical input/output scheduling $S=$（$G_1,G_2,\ldots,G_n$）is determined by heuristics algorithm, while G_k stands for k-th input/output subgroup and its routing. $tt\ (G_k)$ is presented for the tour time of G_k SC, thus objective functions are as follow:

Total tour time is given by

$$\mathrm{TT}=\sum_{k=1}^{n} tt(G_k) \qquad (5)$$

Mean tour time is:

$$MT=TT/n\ [\mathrm{s}] \qquad (6)$$

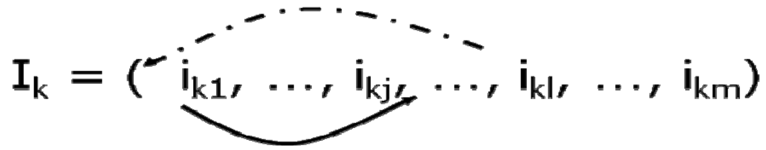
The valuation indicator, average throughput (TP) rate defining as the input/output items SCs held from entrance to exit per hour is given by

$$TP=7200m/MT\ [\mathrm{items/h}] \qquad (7)$$

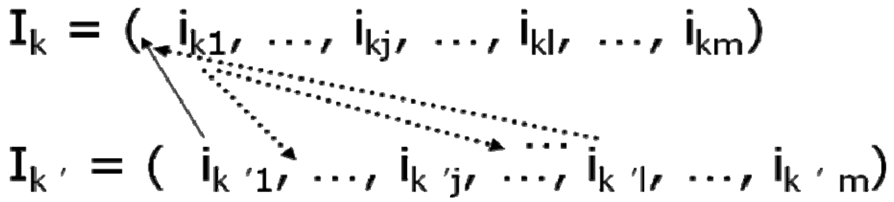
Neighborhood:

Given solution I，its neighborhood solution I' is obtained by two procedures. Let I_k be a subgroup, I' has items i_{kj}, $j=1,2,\ldots m$, which is moved in I as follow:

1). Inserting within a subgroup: i_{kj}($j=1,2,\ldots m$)moves to the first position in I_k showed as follow:



2). Inserting between subgroups: i_{kj}($j=1,2,\ldots m$) and every element in other subgroups are exchanged.



I’ is likely to be identical with *I* obtained by unsophisticated inserting. We project some prerequisites modifying the inserting procedure to eliminate the identical subgroups. Calculation time for scanning neighborhood will be reduced dramatically.

Tabu list:

Tabu list is given by list of M input items, $TL= (TL_1, TL_2, \ldots TL_M)$, TL_i stands for the number of iterations in the tabu search during which item i is prohibited to move, i.e., only item i with $TL(_i)=0$ can move in the next iteration.

Heuristic algorithm:

The capacity of SC at any feasible tour is constraint to hold at most m items, and at any time in a tour the number of items stored must be equal to or more than that retrieved. SC firstly takes the first element i_1 from the list of input items $I= (I_1,I_2, \ldots , I_M)$, and then the nearest cell within $I \cup O$ and so on, until m input or output items are selected in a subgroup. Such an input subgroup and an output subgroup constitute an I/O pair. The left input items of $I= (I_1,I_2, \ldots , I_M)$, are manipulated repeatedly as previous until n subgroups are determined.

5. Numerical experiments

To test the efficiency of approximation algorithm proposed, we perform some numerical experiments of which program is coding by C++ with PC (Pentium 4, 2.4GHZ). At first, we assume some experimental conditions as follow:

1). Rack condition: Each rack with 1.5(m) width and 1.5(m) height (w=h=1.5(m)) consists of a=20 Columns and b=50 rows composing c=ab=1000 cells.

2). SC condition: Acceleration speed (α_x, α_y) is equal to deceleration on both horizontal and vertical directions. We assume $\alpha_x = \alpha_y =$ 0.3, 0.4, 0.5(m/s^2), the maximum speed on horizontal direction $Vxmax$=1.667(m/s) and the maximum speed on vertical direction $Vymax$=0.50(m/s).

3). Tabu list: $TL_{min}=7$, $TL_{max}=20+10\times(\lfloor M/100 \rfloor)$.

30 and 10 different examples are suggested when M=30, 60 respectively. Each example is tested repeatedly for 5 times. Experimental results are obtained as follow:

1). Effects of the different initial solutions and different problem sizes on Mean Time of routing (*MT*) based on tabu search.

Effects of 5 different initial solutions and 2 different problem sizes on *MT* search were tested. As presented in table 2, different initial solutions have no effects on the *MT*. On the other hand, as the problem size altered, the *MT*s varied respectively.

Table2. *MT* time with different initial solutions and different problem sizes (t_{max}=200).

Rule / *M*	RN	ST	LT	SD	LD
30	116.6	116.6	116.9	116.8	116.9
60	110.8	110.6	111.5	111.9	110.8

Rule RN: List of *M* input items in non-repetitive, random order.

Rule ST: List of *M* input items in moving time non-decreasing order from entrance.

Rule LT: List of *M* input items in moving time non-increasing order from entrance.

Rule SD: List of *M* input items in distance non-increasing order from entrance.

Rule LD: List of *M* input items in distance non-decreasing order from entrance.

2). Effect of the max number of iterations (t_{max}) on *MT* and the computation time(s) (*CPU* time(s)) (*M*=30).

As shown in table 3, when *M*=30, compared t_{max}=200 with t_{max}=500, *MT* reduced less than 0.2%, but *CPU* increased significantly. Therefore, the follow results are received at t_{max}=200.

Table3. Effect of t_{max} on *MT* and *CPU* time [s] (*M*=30)

t_{max}	0	1	10	50	100	200	300	400	500
MT	127.9	121.2	117.9	117.0	116.8	116.54	116.51	116.46	116.41
CPU	0.002	0.443	4.44	22.31	44.66	89.37	134.02	178.69	223.45

3). Effect of tabu search robustness on *MT*.

We suggested 10 examples each tested for 5 times. No significant difference on *MT* was obtained (table 4). The robustness of tabu search is suggested to be favorable.

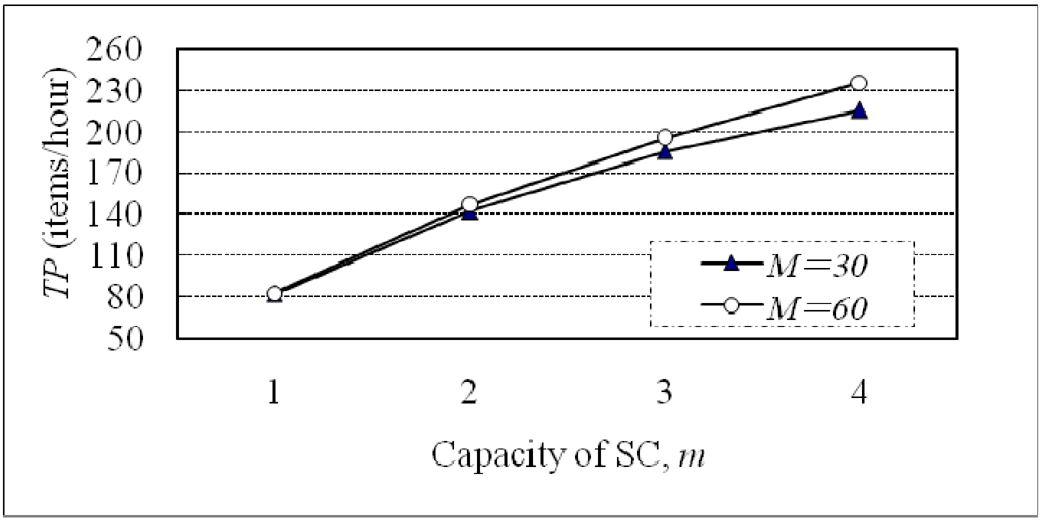
Table4. Effect of robustness on *MT*[s] (*M*=60).

ex. times	1	2	…	10
1	114.035	108.290	…	107.625
2	114.180	108.370	…	107.605
3	114.415	108.055	…	107.720
4	113.995	107.635	…	107.605
5	114.875	107.480	…	107.605

4). Effect of capacity of SC *m* on throughput *TP*.

As shown in fig1, the throughput increased about 75%, 125%, and 190% when the capacity of SC, *m* enlarged from 1 to 4. We also observed *TP* increased more significantly when *M*=60 than that when *M*=30.

Fig1. Effect of SC capacity, *m* on throughput *TP* (*M*=30, 60).



5). Effect of acceleration (α_{x}, α_{y}) on *TP*.

TP increased approximately 5% as the acceleration (α_x,α_y) increased 0.1(m/s) (table 5).

Table5. Effect of acceleration (α_{x}, α_{y}) on *TP* (*M*=30, 60).

M Acceleration	30	60
0.3	185.6	195.6
0.4	195.3	205.4
0.5	201.7	211.8
∞	234.4	248.2

6. Conclusions

In this study，we discussed the optimization for input/output routing of multi-arms SC in AS/RS to minimize the SC mean tour time. At first, we assume a model of AS/RS, and suggest the optimal input/output routing of SC with multi-arms is depended on the I/O partitions, the I/O pairs and the routings of SC to be performed. We enumerate the solution space of exponential function in different *M* and *m* values, and propose an approximation algorithm based on heuristic and tabu search. By some numerical experiments based on practical conditions, we demonstrate the robustness of proposed approximation algorithm, discuss the effect of capacity of SC *m* and acceleration/deceleration speed on throughput *TP*.

References

[1] Van den Berg, Jeroen P(1999): “A literature survey on planning and control of warehousing systems”. IEEE Transactions 31, 751-762.

[2] De Koster, Rene, and Edo van der Poor(1998): “Routing orderpickers in a warehouse: a comparison between optimal and heuristic solutions”. IEEE Transactions 30, 469-480.

Acknowledgement

The Project is sponsored by SRF for ROCS, SEM and supported by Center of Logistics Technology and Engineering, Bejing Wuzi University.