行政院國家科學委員會補助專題研究計畫成果報告

※※※※※※※※※※※※※※※※※※※※※※※※
※　　　　　　　　　　　　　　　　　　　　　　※
※　　　　　　　　　　　　　　　　　　　　　　※
※　　　　**分組式無人搬運車路軌網路設計**　　　　※
※　　　　　　　　　　　　　　　　　　　　　　※
※　　　　　　　　　　　　　　　　　　　　　　※
※※※※※※※※※※※※※※※※※※※※※※※※

計畫類別：☒個別型計畫　　□整合型計畫
計畫編號：NSC 89-2213-E-009-040
執行期間：88 年 8 月 1 日至 89 年 7 月 31 日

計畫主持人：劉復華

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
☒出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

執行單位：國立交通大學工業工程與管理學系

中 華 民 國 八 十 九 年 九 月 二 十 九 日

行政院國家科學委員會專題研究計畫成果報告

分組式無人搬運車路軌網路設計

Designing Path Network for Automatic Guided Vehicle System

計畫編號：NSC 89-2213-E-009-040

執行期限：88 年 8 月 1 日至 89 年 7 月 31 日

主持人：劉復華　國立交通大學工業工程與管理學系

計畫參與人員：本系學生 賴慶祥、洪碧涓、李紹彰、韓慧林

中文摘要

根據一給定的巷道佈置圖，本研究討論分組式無人搬運車系統的路軌設計問題。此分組式雙向的無人搬運車系統，裝卸站可被分到幾個運輸單元(組)內。各分組分別以一簡單的路軌連接組中之各裝卸站，並以一輛搬運車服務之；另設若干轉運站於各組路軌間，以便傳遞跨組之搬運需求。首先利用一數學規劃模式，找出一總成本最低的路軌佈置。在路軌的佈置上，總本成包含巷道的成本、路軌的成本以及車輛的旅行成本。此問題先前之研究提出兩種求解方法：數學規劃及兩階段的演算法。由數學規劃在取得最佳解時，往往花費相當多的時間。兩階段演算法的兩項結果：目標值及各，可能並不一定與數學規劃所求之最佳解一致。數學規劃法以分支界限法求解，大型題目耗時甚久始能得解。本研究主要在探討利用兩階段演算法的目標值作為以數學規劃法求解時的低限值；利用兩階段演算法的決策變數之值判斷決策變數的優先次序，此次序作為以數學規劃法求解時的分支次序；甚至混合上述兩種方法。本研究以亂數產生許多題目，每個題目均進行數學規劃、兩階段的演算法、以數學規劃法在三種設定下求解，共五種方法進行計算，記錄其所須之計算時間，及解答。比較可減少的計算時間以及解答之準確度。此演算法所求得的最低成本雖不能與數學規劃模式所得之最佳解完全相同，但應可十分接近。主要的是計算時間將可能是數千倍的縮短量，並可以求解更大的題目。同時也可利用兩階段演算法的快速所得之結果使得以數學規劃法大量降低計算時間，求得精確解。

關鍵詞：無人搬運車；路軌佈置；設施規劃；物料搬運系統

Abstract

In a divided AGV system, the pickup/deposit (P/D) stations are divided into several transport cells, with each cell having a simple flow-path traversed bidirectionally by a dedicated vehicle. Several transit stations may be set up to construct a transit system for handling the intercell transport requests. This current research assumes that the P/D stations for each transport cell and the transit station for each cell are given. Based on a given block layout, this research addresses the design problem for constructing the flow-path layout on a divided AGV system. The boundaries of the block layout were treated as the candidate aisles to place flow-paths. In order to find which candidate aisles should be selected, the authors propose a mathematical programming model to find the flow-path layout with the lowest total cost. The total cost includes the cost for aisles, the cost for flow-path and the travel cost for vehicles. Since the mathematical programming model takes time to reach the optimal solution, the authors also developed a two-phase heuristic algorithm to speed the process.

Keywords: AGV System, Flow-path Layout, Facilities Planning, Materials Handling System

The problem and the objective

This project is an extension of previous NSC projects in the years of 1994, 1995, and 1997. Liu & Chen [1,2,3] address the problem. In a divided AGV system, the pickup/deposit (P/D) stations are divided into several transport cells, with each cell having a simple flow-path traversed bidirectionally by a dedicated vehicle. Several transit stations may be set up to construct a transit system for handling the intercell transport requests. It was assumed that the P/D stations for each transport cell and the transit station for each cell are given. Based on a given block layout, this research addresses the design problem for constructing the flow-path layout on a divided AGV system. The boundaries of the block layout were treated as the candidate aisles to place flow-paths. In order to find which candidate aisles should be selected, Liu and Chen [1,2,3] propose a mathematical programming model to find the flow-path layout with the lowest total cost. The total cost includes the cost for aisles, the cost for flow-path and the travel cost for vehicles. Since the mathematical programming model takes time to reach the optimal solution, Liu and Chen [1,2,3] also developed a two-phase heuristic algorithm to speed the process. The authors assume the aisles are wide enough for placing multiple flow-paths, the vehicles traverse bidirectionally, and the unit cost for aisles and the unit cost for flow-paths are constant. They further assume that a block layout for the departments of a manufacturing system is given; in that layout the transport cells for the divided AGV system and the location of the transit station for each cell are known. Based on the block layout, they attempt to design an aisle layout and the flow-path for each cell so as to minimize the total cost which includes the cost for aisles, the cost for flow-paths, and the travel cost for vehicles.

Construct a candidate-aisle network. The location of the s P/D stations, the location of the r intersections, and the linkage of a edges between P/D stations and intersections are determined. The number of pairs of P/D stations that have positive transport requests, p, should also be determined. In other words, there are p elements in the from/to matrix having $f_{lm}>0$. The value of p is about $s^2/4$. The authors denoted a network configuration as (*sr_ap*). For instance, associated with table 1 is denoted as (0808_2016); it has eight stations, eight intersections, 20 edges, and 16 transport requests.

Each cell has only one known transit station. Values for all the transport requests, f_{lm}, are randomly generated in the range of (2, 8). The length of the a edges, d_{ij}, is randomly generated in the range of (1, 20). Hence, for a given network, problems with different f_{lm} and d_{ij} would be generated.

Assume the cost for aisles was \$150/m, the cost for flow-paths was \$50/m, and the travel cost for vehicles was \$10/m. A Pentium II-300 PC was used to compute the optimal and heuristic solutions. CPLEX software was used for the MP problem. The heuristic algorithm was coded in C language.Two performance parameters provided by CPLEX were reset to speed up the resolutions of the MP models, denotes as Parameter α and β, respectively. Parameter α gave priority orders for integer variables for the branch-and-bound procedure. In this research, all the variables A_{ij} were assigned higher priorities. Parameter β gave *an upper cutoff* for restricting the search. In this study, the solution obtained by the heuristic was used as an upper bound. We recorded the computation times for CPLEX under two parameter settings: with α only, and with both α and β. It is interesting to compare the CPLEX computation times under the two settings: with α only, and with α and β at the same time.

The (0808_2016) configuration was tested. The computation results for the thirty problems are summarized in table 1. Columns (1) and (2) are the solutions obtained by MP and heuristic approaches, respectively. Column (3) shows the deviation of the heuristic solution. Eighteen out of the thirty heuristic solutions are equal to their optimal values. On average, a 1.21%

deviation was found.

The (0609_2009) problems were also tested in the cases of 2 cells and 3 cells. In the case of 2 cells, P/D stations were partitioned into cells {1,3,6} and {2,4,5} with transit stations {1,4}. See table 2 for the computation results. On average, the heuristic had a 0.39% deviation. In column (7), with the contribution of the heuristic solution, the MP was 2.59 times faster.

To learn how the number of cells would effect the computation, the (0609_2009) problems were also tested with 3 cells. P/D stations were partitioned into cells {1,3}, {2,6}, and {4,5} with transit stations {3,6,5}. See table 3 for the computation results. Compare the average values of columns (4), (5), and (6) in tables 7 and 8: (0.053, 0.062), (18.81, 79.62), and (9.31, 60.17). One can observe how the number of cells greatly increases the computation time, especially for the MP model.

A large-sized problem (1019_3922) was also tested. Table 4 summarizes the results. On average, the heuristic solution was only 1.09% higher than the optimal solution, as shown in column (3). In column (4), the heuristic needs less than one second to obtain the solutions. In columns (5) and (6), CPLEX needs a very long time to obtain optimal solutions. Some problems even need more than a week. As shown in column (7), using the heuristic solution as an upper bound, CPLEX would have optimal solutions within a shorter time.

The size of MP models and computation times for the testing problem sets are summarized in table 5. The size of the MP model (1527_5631) problem is much larger than that of (1019_3922). It thus would require a very long computation time. The authors also solved 30 testing problems by the heuristic. The computation time is summarized as follows: average, 4.5 seconds; maximum, 6.37 seconds; minimum, 2.69 seconds; and standard deviations, 0.95. One can see that, even for such a large problem, the heuristic needs only a few seconds.

Problem (0808_2016) was tested with the cost parameters C_A:C_P:C_T under four combinations: 200:20:1, 100:20:1, 30:10:1, and 15:5:1. The deviations between the heuristic solutions and the optimal solutions are summarized in table 6. It shows that with a higher aisle cost, as in the first two cases, the deviations are higher, 5.18 % and 6.0 %. Since the aisle length was restricted by its higher cost in phase 1, the total length of flow-paths for all cells might take longer and travel cost are also increased.

Conclusion and Discussion

Assume the cell configuration and the location of each transit station are given for a divided AGV system. This paper aims to propose heuristic algorithm to determine the flow-path layout with a lower cost. The heuristic algorithm can provide a near optimal solution within a reasonable computation time. It can also serve as a tight upper bound for finding the optimal solution through the branch-and-bound procedure.

The heuristic, as currently stated and implemented, is a greedy procedure. The heuristic may be trapped in a local optimal solution. Ideas from methods such as the Tabu search, genetic algorithm, and simulated annealing may be used to enhance the heuristic. The main objective of this study is to introduce the MP model and structure of the heuristic. Using the heuristic solution as an upper bound for solving the optimal solution of the MP model can speed up the resolution time.

Due to the rapid development in guiding systems, the cost for aisles is usually higher than that for flow-paths. Thus, the heuristic approach determines an aisle layout first, then determines the flow-path for each cell on the aisle layout. If the costs for aisles and flow-paths are very low, each transport request tends to be handled by its shortest path. On the other hand, if the cost for aisles is quite high, the flow- path layout problem becomes a Steiner tree problem. Comparisons in tables 1~5 show the proposed heuristic algorithm handles testing problems well.

Self-evaluation

This research consumes a huge of computation time. We shows the heuristic approach would be used as a tool in the beginning of the design work. Once the solution is obtained, we suggest the MP model should applied to the design to get the accurate mathematical solution. Practical engineering conditions should also be considered before it is implemented. A full paper is completed and presented in the international conference of 2000 Applied Mathematical Programming and Modelling held in London, April 17~19, 2000. The paper is submitted to the Annual of Operations Research.

References

[1] Liu, F.H. and Chen, J.T. (1999). A Heuristic Approach to Determine Bidirectional Flow-path for AGV Systems. *International Journal of Industrial Engineering - Applications and Practice*, 6(2), pp.171-181.

[2] LIU, F. H. and CHEN, J. T., 1997, Analytical framework for designing the divided automated guided vehicles system, *International Journal of Industrial Engineering - Applications & Practice*, **4**(2), 90-102.

[3] Chen, J, T, 1998, A framework for designing the divided guided vehicle system, Doctoral Dissertation (advisor: Liu, F.H.), Department of Industrial Engineering and Management, National Chiao Tung University, Taiwan, Republic of China.

Part of the data is not listed in the following tables since the number of page for this report is limited. Readers are welcomed to contact the author to obtain the whole data.

Table 1: Results of (0808_2016) example problems.

	Solution (total cost)			Computation time (seconds)			
Col.	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Ex.	Optimal	Heuristic	$= \frac{[(2) - (1)]}{(1)}$	Heuristic	α	α & β	=(5)/(6)
1	4129	4221	2.23%	0.060	129	55	2.35
2	6606	6606	0.00%	0.100	32	12	2.79
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
29	7334	7354	0.27%	0.110	38	17	2.26
30	7798	8011	2.73%	0.110	37	9	4.36
Avg	6529	6607	1.21%	0.087	71	21	4.70
Max	8536	8636	13.12%	0.160	177	139	11.71
Min	4129	4221	0.00%	0.004	12	5	1.04
Std	1274	6722	1.62	0.088	70	20	5.07
19 of 30 heuristic solutions equal to the optimal solutions							

Table 2: Results of (0609_2009) example problems, with 2 cells.

	Solution (total Cost)			Computation time (seconds)			
Col.	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Ex.	Optimal	Heuristic	$= \frac{[(2) - (1)]}{(1)}$	Heuristic	α	α &β	=(5)/(6)
1	4363	4397	0.78%	0.110	5.24	2.52	2.08
2	3636	3636	0.00%	0.050	8.11	3.14	2.58
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
29	4047	4047	0.00%	0.060	2.63	2.67	0.99
30	4141	4141	0.00%	0.050	24.37	8.59	2.84
Avg	4537	4554	0.39%	0.053	18.81	9.31	2.59
Max	7471	7471	3.18%	0.110	141.90	88.79	6.92
Min	3151	3151	0.00%	0.004	2.60	2.39	0.78
Std	976	974	0.82%	0.022	26.70	17.47	1.35
22 of 30 heuristic solutions equal to the optimal solutions							

Table 3: Results of (0609_2009) example problems, with 3 cells.

	Solution (total cost)			Computation time (seconds)			
Col.	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Ex.	Optimal	Heuristic	$=\frac{[(2)-(1)]}{(1)}$	Heuristic	α	α &β	=(5)/(6)
1	5075	5184	2.15%	0.060	27.63	16.15	1.71
2	4715	4715	0.00%	0.060	48.23	9.92	4.86
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
29	5629	5656	0.48%	0.060	10.98	12.48	0.88
30	5003	5021	0.36%	0.050	109.60	99.13	1.11
Avg	5208	5275	1.27%	0.062	79.62	60.17	2.36
Max	7508	7508	9.19%	0.110	1200.37	1110.14	5.73
Min	3586	3586	0.00%	0.004	8.16	4.00	0.88
Std	930	955	1.94%	0.026	215.04	200.36	1.29
11 of 30 heuristic solutions equal to the optimal solutions							

Table 4: Results of (1019_3922) example problems.

	Solutions (total cost)			Computation time (seconds)			
Col.	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Ex.	Optimal	Heuristic	$=\frac{[(2)-(1)]}{(1)}$	Heuristic	α	α &β	=(5)/(6)
1	17281	17333	0.30%	0.44	606537	323102	1.88
2	9359	9612	2.63%	0.28	84016	68133	1.23
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
14	11104	11122	0.16%	0.28	41090	29939	1.37
15	13939	13939	0.00%	0.27	5298	2305	2.3
Avg	12416	12520	0.79%	0.37	122137	66406	5.47
Max	17281	17333	2.95%	0.55	606537	323102	42.37
Min	7558	7558	0.00%	0.22	5298	164	1.23
Std	2404	2444	0.99%	0.11	178001	99342	10.44
5 of 15 heuristic solutions equal to the optimal solutions							

Table 5: Summary of the problem size and computation time.

Problem sets	(0808_2016) 3 cells	(0609_2009) 2 cells	(0609_2009) 3 cells	(1019_3922) 3 cells	(1527_5631) 3 cells
# of stations	8	6	6	10	15
# of intersections	8	9	9	19	27
# of edges	20	20	20	39	56
# of transport request	16	9	9	22	31
# of non-negatives var.	118	83	105	211	357
# of binary variables	2820	1440	1660	8853	28840
# of constraints	5323	2585	3063	16477	53976
CPLEX (seconds)*	21	9.31	60.17	66406	NA
Heuristic (seconds)	0.087	0.053	0.062	0.37	4.5

*: Parameters α &β are used.

Table 6. Cost parameters effect on the deviation of (0808_2016) testing problems.

# of cell	3	3	3	3
C_A: C_P: C_T / Item	200:20:1	100:20:1	30:10:1	15:5:1
Mean	5.18%	6.00%	2.53%	1.22%
Std	0.06	0.06	0.05	0.03
Max	19.17%	19.47%	19.81%	13.12%
Min	0.00%	0.00%	0.00%	0.00%