Ecommerce Logistic Finding: Preliminary Work

G M Al Imran, Edwin Mit and Syahrul Nizam Junaini
Faculty of Computer Science and Information Technology
Universiti Malaysia Sarawak
94300 Kota Samarahan, Sarawak.
gimrancse@gmail.com, edwin@fit.unimas.my, syahruln@fit.unimas.my

***Abstract* - Electronic Commerce (EC) Logistics is a buzz word in the present day world economy. Proper delivery of products to the customer in a cost effective way and ensuring customer satisfaction is very much important. This research analyses the ecommerce logistics, makes a discussion on the development of logistics in the EC environment and establish an enhanced algorithm based on road network system, with the aid of the Logistics Information Management System (LIMS). It mainly focuses on faster and cost effective ecommerce logistics distribution.**

Keywords – E-commerce, Logistics Distribution, Shortest Path, Haversine Distance, A* Search, Road Network

I. INTRODUCTION

Logistics is a significant division of electronic commerce, while physical distribution systems have become a weakness that restraining the development[1] of electronic commerce as very few numbers of logistics service provider able to meet the clients requirements and demands. But still there remains a big problem which is last mile delivery[2]. Some points should be considered like locating delivery point, get delivery points spatial data, prepare road network system closely related with delivery path calculation or delivery scheduling[3]. In case of road transportation, GIS application is used widely used as GIS used to practice and analyse regularly on road network management system. Here comes the term shortest path as in road network system finding shortest path is a classic term.

There are lots of algorithm exist that are used to find out shortest path. We will analyse on the system of e-commerce logistics distribution with different algorithm like Dijkstra, Symmetrical Dijkstra Algorithm, A* Search, weighted A* Search, Radius Search to find out the optimal solution. Most of the algorithms rely on greedy technique but in our proposed algorithm we will add some real life parameter with the weighted value in case of distance and time calculation to make the system more real times.

In this paper, we proposed an efficient technique to find out the shortest path for logistics distribution Management system. Most of the time people forget to count some factor in case logistics delivery system like weather condition, vehicle category, high altitude, traffic history data which are very important factor. Also in case of implementation of the algorithm will use better data structure technique and run time index searching to enhance the overall system.

II. LITERATURE REVIEW

Logistics distribution has been a major issue since long and currently many scholars and academicians are also researching on it to bring out something efficient. For the e-commerce industry, logistics is a very important aspect, for it directly determines the progress of e-commerce terminal link, thus determines the progress along the entire chain. In case of well-organized ecommerce logistics system actually define whether it is going to be successful or failed[4] . Logistics has become the bottleneck in the development of electronic commerce. Logistics does have a huge impact on supply chain management and e-commerce system. In case of decision making during ordering a product or goods client want to have something that will help them and these things are closely related with logistics system like when the product will arrive and how long the client need to wait for and what will be shipment cost as well. Cost effective logistics system, exact time scheduling like expected delivery date or times, user/client friendly forecasting of delivery are the very fundamental areas of global logistics[2]. In B2C commerce system customer end people still feel shaky in case of online shopping as for uncertainty of delivery time. For delivering products to clients home and this is closely related with vehicle routing problems where timing is a problematic issue [5]. People are getting busy day by day and they need just-in-time delivery of goods or services and customer want many options of delivery mode as business provider or ecommerce centre can provide. On the other hand reverse logistics also getting a form of upcoming threat in e-commerce system. Now a days every enterprise that provides logistics support can follow common criteria which will increase the competitiveness among them and their service quality will be increased and off course in a faster way [3]. Reverse logistics already becomes a problem for the enterprises.

There are numbers of shortest path algorithm available; among them Dijkstra Algorithm is most popular[1]. Any kind of graph that consists of non-negative weights can be solved by Dijkstra algorithm. On the other hand A* search is the updated version of Dijkstra algorithm where heuristic is used to speed up the processing actually it makes quicker to reach the final node [1]. Floyed-Warshall algorithm is another algorithm which provides a generalize solution for speeding up the search process. It helps to find out shortest path like others among all vertices in a directed graph but here the negative cycles are forbidden. Thus the execution time enhancement has been done which was not in the Dijkstra algorithm, another algorithm named as BellMan-Ford algorithm also used where no negative cycle is allowed[6]. In previous system, a search technique was used which is known as control strategy. In that case searching started from a specific part of the search area and goes on through the all parts until it found the destination and it's a kind of step by step formula [7]. Here, the main technique was to generate state space tree considering that it will consist of less branches. Heuristic search is a core concept in Artificial Intelligence (AI) research work and huge numbers of algorithms are using it for planning, agent controlling and video games playing. In case of path finding between two points agents have to response quickly to the surroundings. For this reason, constant time limit is used to impose on number of path planned during each moves. But researchers found that it good not impose any limit on per move as it reduces the range of heuristics search that can be applicable. A*, IDA* and PRA*, re-planning algorithms such as D*, anytime algorithms such as ARA* and anytime re-planning algorithms such as AD* all have worked on static search algorithm but they cannot able to provide guarantee things like a constant bound on planning time per action [8]. It is because they used to finalize full solution before completing first action. That why when the problem increases in their sizes, planning time increases inevitably.

III. Proposed Methods

As heuristic searching technique is the most efficient way to find out the optimal shortest path between two points and we named it as CA* i.e Complete A* search algorithm. We will use the A* search as base algorithm. First we create a list of delivery point to specific region. First of all, regarding the entire road network we will generate a graph that will represent the exact result as real road network. Then we will find out the shortest path by using CA* search. To implement CA* search algorithm we need value of function f(n) = w(n) + h(n), where w(n) is the weighted value to visit from one point to its next adjacent point and h(n) is the heuristic value of cost required to go to destination node from current node. In our proposed method the node to node cost is the combination of motion of the vehicle, distance of node to node, time delay due to goods loading and unloading, traffic signal delay, weather, high altitude, and delay due to unknown factors. Another factor we are going to introduce in our system is historical data of traffic for node to node cost calculation. We will keep traffic data on hourly, daily, weekly and monthly basis. We take Haversine distance of current node to destination node as heuristic value h(n) because this heuristic is an admissible heuristic as we know that the shortest path between any two point is a straight line and we will use Haversine distance to work with more real distance calculation. Another heuristic we will use later is the historic data for reaching one node from another node. In our method we start search from initial node and search for its neighbour nodes, then we do necessary calculation of required cost w(n) to go to each neighbour and check for all pairs of neighbour node have other cost like delay due to weather, traffic, unload time and others. All these cost are then added with the cost w(n) to go to neighbour node from its initial node and this value with heuristic cost. Then we expand the node that have minimum cost and follow the rules that we have done for previous node. If we find larger value of any current node then we back track to previous node having less cost than current node value. When we find target node in neighbour of any node and its cost remain less than any other node value then the search is over. We keep track on the starting time which helps us to check which paths are facing traffic signal and calculate the approximate time to reach the destination because the delivery time will provided to the customer at the time of ordering goods for his/her satisfaction.

A. Establishment of Data Structure

We will use graph to implement our project. A graph will be created with nodes which represents Kuching city's road network. The Graph will contain Nodes that represents different road connections in Kuching. The Graph class will represent a graph, which will be composed of a collection of nodes and edges. This Graph class will maintain its collection of nodes using the NodeList class, which would be a collection of TNode objects. It will delegate the edge maintenance to the TNode class. The TNode class will maintain the edge information using the adjacency list technique. We will use special type of Hash function to store all instances that's related to each and every node in the graph. We will store nodes instance in such technique that it will generate unique key each time. As we will use hash table and generate unique key it will accelerate the internal searching process which will enhance the execution time of the program.

B. Proposed Mathematical Model

As our target is to find shortest path so it is very much important to find out the weighted value between from one graph node to another graph node. The weighted value are calculated via this equation, $T_{d=}\frac{di}{si}{}_{+}\,t_u{}_{+}\,K_1+K_2$ where, di= Distance from one node to another, s_i= Speed of the vehicle, t_u = Goods unloading time, K_1= total delay due to weather problem, slop, Ferry, traffic and others, K_2=Unexpected delay. Unexpected delay may occur due to any system failure in vehicle. If a node pair faces traffic signal then add traffic delay cost with T_d and then we get g(n) which represents node to node cost. After that we add g(n) with heuristic value and find our evaluation function f(n)=w(n) + h(n), we get the heuristic value from longitude and latitude value of two nodes using Haversine distance calculation technique[9]. We get distance value in miles or kilometres unit so it is converted into time unit by dividing it with average speed of vehicle.

C. Shortest Path Implementation

At first we will be listed down the all delivery point and run whole system to find each distance from source point to delivery point. Regarding the designed road network and giving it a digital graph representation, we will try to find shortest path which will start from a source point and look for final destination point. To implement CA* search algorithm we need value of function as value of cost required to go to destination node from current node which is described in Mathematical model section.

In our we start search from initial node and search for its neighbour nodes, then we do necessary calculation of required cost w(n) to go to each neighbour and check for all pairs of neighbour edges weight. In the same time before making immediate next move will calculate next 3 steps in advance to reduce unwanted node visiting which is kind of look ahead search with the help of heuristic value. After checking all the weighted values and look ahead search data, our search agent moves to next node as per algorithm. Then we expand the node that have minimum cost and follow the rules that we have done for previous node. If we find larger value of any current node then we back track to previous node having less cost than current node value. When we find target node in neighbour of any node and its cost remain less than any other node value then the search is over. We will continue this and also keep track on the starting time which helps us to check which paths are facing traffic signal and calculate the approximate time to reach the destination because the delivery time will provided to the customer at the time of ordering goods for his/her satisfaction..

IV. Experimental Result And Discussion

A. Probable result

As we proposed that our system would be a better solution for finding shortest path for ecommerce logistics system, here we are showing some factor that would provide some idea about it. As all these factors have impacts on vehicle speed so the integration of them will make our searching process not only optimal but also more realistic and compatible to e-commerce logistics. The final result would be improved for using the below factors. Our system is slightly different from GPS system and Google navigation system as both of them have used Dijkstra's algorithm and we will use A* search algorithm.

Table I. Factors that will have effects on Delivery time

Applied Term	Effect
A* search as base algorithm	Faster and complete search
Heuristic	Graph's node visit will cut into half
Slops in Road	Used to reduce 20~30% vehicle speed
High Traffic	Used to reduce speed at least 5~10%
Traffic Jam	Increase delay time minimum 2~3 minutes on each jam
Traffic Signal	Have delay time of 30 seconds on average
Bad weather	On rainy season vehicle speed reduced 30~40%

V. Conclusion

In this paper, we mainly focus on different shortest path algorithm, ecommerce logistics system, and previous research works that are related to road network systems, ecommerce logistics solutions and also provide a sound logistics distribution system by finding shortest path. This paper will enhance the logistics distribution functionality of e-commerce with great degree. It can be applied in international logistics distribution system as well as vehicle routing. If we can establish a collaborative system for logistics distribution for multi groups of service provider companies then we can meet up both side demand and ensure better logistics distribution. So our future plan is to establish such kind of logistics distribution system where it will work as a group.

References

[1] I. Beker, V. Jevtić, and D. Dobrilović, "Shortest-path algorithms as a tools for inner transportation optimization," *Int. J. Ind. Eng. Manag.*, vol. 3, no. 1, pp. 39–45, 2012.

[2] R. E. Mays, R. Racadio, and M. K. Gugerty, "Competing constraints: The operational mismatch between business logistics and humanitarian effectiveness," *Proc. - 2012 IEEE Glob. Humanit. Technol. Conf. GHTC 2012*, pp. 132–137, Oct. 2012.

[3] Y. Zhuang, "Bayesian Network-Based Online Shopping Satisfaction Evaluation and Empirical Analysis of the Third-Party Logistics," *2012 Int. Conf. Manag. e-Commerce e-Government*, pp. 415–418, Oct. 2012.

[4] W. Xiaoyan and P. Kailing, "Current Situation, Problems and Countermeasures of China's B2C Logistics," 2012.

[5] M. Masmoudi, “Mathematical Modeling for a Rich Vehicle Routing Problem in E-commerce Logistics Distribution,” pp. 290–295, 2013.

[6] M. ~a. Anwar and T. Yoshida, “Integrating OO Road Network Database, Cases and Knowledge for Route Finding,” *Proc. 2001 ACM Symp. Appl. Comput.*, pp. 215–219, 2001.

[7] Z. Y. Z. Yong-mei, M. L. M. Li, and M. L. M. Lan, “Research and Implementation of Electronic Commerce Logistics Distribution Management,” *E-bus. E-Government (ICEE), 2010 Int. Conf.*, pp. 2089–2092, 2010.

[8] V. Bulitko, Y. Björnsson, and R. Lawrence, “Case-based subgoaling in real-time heuristic search for video game pathfinding,” *J. Artif. Intell. Res.*, vol. 39, pp. 269–300, 2010.

[9] S. E. Braden, “Figure 1. The haversine formula solved for distance where d is the distance in meters, r is the radius of the planetary body in meters, phi is latitude and lambda is longitude for two points.,” pp. 1–2, 2015.