Applied Soft Computing 19 (2014) 333–342

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Application of cellular automata and type-2 fuzzy logic to dynamic vehicle path planning

Chenn-Jung Huang[a,b,*], Yu-Wu Wang[a], Heng-Ming Chen[b], Han-Wen Tsai[a], Jui-Jiun Jian[b], Ai-Lin Cheng[a], Jia-Jian Liao[c]

[a] Department of Computer Science and Information Engineering, National Dong Hwa University, Hualien, Taiwan
[b] Department of Electrical Engineering, National Dong Hwa University, Hualien, Taiwan
[c] Institute of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan

ARTICLE INFO

Article history:
Received 23 July 2011
Received in revised form 10 January 2014
Accepted 25 February 2014
Available online 13 March 2014

Keywords:
Optimal path planning
A* Search algorithm
Fuzzy logic
Cellular automata

ABSTRACT

Nowadays, most road navigation systems' planning of optimal routes is conducted by the On Board Unit (OBU). If drivers want to obtain information about the real-time road conditions, a Traffic Message Channel (TMC) module is also needed. However, this module can only provide the current road conditions, as opposed to actually planning appropriate routes for users. In this work, the concept of cellular automata is used to collect real-time road conditions and derive the appropriate paths for users. Notably, type-2 fuzzy logic is adopted for path analysis for each cell established in the cellular automata algorithm. Besides establishing the optimal routes, our model is expected to be able to automatically meet the personal demands of all drivers, achieve load balancing between all road sections to avoid the problem of traffic jams, and allow drivers to enjoy better driving experiences. A series of simulations were conducted to compare the proposed approach with the well-known A* Search algorithm and the latest state-of-the-art path planning algorithm found in the literature. The experimental results demonstrate that the proposed approach is scalable in terms of the turnaround times for individual users. The practicality and feasibility of applying the proposed approach in the real-time environment is thus justified.

1. Introduction

In the past, when people were bound for unfamiliar locations, they first had to find the destination and identify their current location on a map, and then searched for a suitable path to take. Recently, with rapid advances in the development of web-based geographical information systems (WebGIS), numerous free geographical information services are accessible on the Internet, such as Google Maps, TomTom, PaPaGo, etc. These GISs provide useful information such as satellite imagery, vector mesh routing and address positioning. Accordingly, users can acquire a wealth of information by merely clicking the mouse. However, although On Board Units (OBU) currently found in vehicles claim to provide "real-time" navigation functionality, a traditional Traffic Message Channel (TMC) is still utilized to acquire the road information via data fusion techniques, and traditional static path planning strategies still lack the precision and reliability to fully meet users' real-time requirements.

Vehicular navigation systems can be classified as either static navigation or dynamic navigation based on the type of information used in the algorithms. Static navigation plans routes with preloaded graphic information and topology, while dynamic navigation changes routes dynamically to fit traffic conditions and network topology. Hence, dynamic navigation needs to rely on an information exchange center to collect traffic data, and it can be observed that dynamic navigation is more tolerable to the variations of traffic flows.

The architecture of a navigation system can be designed for either autonomous navigation or centralized navigation [1]. The greatest difference between the two is whether or not there is a service center responsible for implementing path planning for all cars. In autonomous navigation, a planner in a vehicle carries out path planning based on data stored in the vehicle or data received from the information center. In centralized navigation, the service center receives applications from vehicles, conducts path planning, and then sends the results back to applicants. It has been found that road utilization is higher and the cost for vehicle equipment is lower in centralized navigation. Thus, it can be expected that

---

* Corresponding author at: Department of Computer Science and Information Engineering, National Dong Hwa University, Hualien, Taiwan. Tel.: +886 912796179.
E-mail addresses: cjhuang@mail.ndhu.edu.tw, chennjunghuang@gmail.com (C.-J. Huang).

http://dx.doi.org/10.1016/j.asoc.2014.02.023

this kind of navigation architecture will be widely deployed in the future, although this will come at the cost of increasing the traffic load on wireless communications.

Researchers have devised various algorithms to deal with path planning problems in the past. These can be classified into four kinds of shortest path problems, including Single-Pair Shortest-Path (SPSP), Single-Source Shortest-Paths (SSSP), All-Pairs Shortest-Paths (APSP), and Single-Destination Shortest-Paths (SDSP). In terms of solutions for the above-mentioned four kinds of problems, the Bi-directed method [2] can solve a SPSP problem; Dijkstra [3] proposed the shortest path algorithm to solve a SSSP problem; and Floyd [4] and Warshall [5] solved APSP and SDSP problems, respectively.

There are a lot of solutions dealing with shortest path problems between two single nodes in the literature. Among them, A^* algorithm [6] is one of the most effective ones. By this algorithm, the optimal path is obtained by computing the minimum path length from every node to the destination. As for multiple shortest path algorithms, it was reported that the *k*th shortest path algorithm developed in 1984 by Martins et al. [7] is 20 times faster than the traditional Dreyfus' algorithm in searching the 20th shortest path while the number of nodes reaches 1000.

Ding et al. [8] presented a real-time infrastructure-free route guidance algorithm called Vehicle-to-Vehicle Real-time Routing (V2R2). Although using the V2R2 algorithm can reduce the cost of the establishment and maintenance of infrastructure, it suffers from a lack of scalability. Besides, load balancing cannot be achieved by this algorithm because it attempts to avoid the disconnection problem by ignoring the routes with no traffic load. Recently, Li et al. [9] combined the hierarchy traits of road networks with the path planning algorithms known as the Graph Voronoi Diagram [10] and HSR-Based Route Planning [11]. This hierarchical Voronoi graph-based route planning algorithm [9] utilizes the characteristics of the road network hierarchy to reduce the searching time and the expanded node number of routes searched. However, as mentioned in [12], if a car accident occurs along the route generated by the shortest path algorithm, this planned path cannot be used at all. A road network database was adopted in [12] to store all paths that had been computed in the past. When a car accident occurs, the situation will be immediately updated in the road network database, and case-based reasoning is used to locate the path in the database by examining if the origin and the destination are the same as the data stored in the database. However, this method still needs to take into consideration the road status at different times or different locations, or else it cannot meet the requirements of path planning for every source/destination combination.

With the increasing demand for real-time information, dynamic path planning has become an essential research issue. For instance, Kanoh et al. [12] presented a solution which utilizes a genetic algorithm (GA) to solve dynamic path planning problems. Anwar [13] claimed that the shortest routes computed by existing algorithms in the literature are not optimal. Traffic accidents in real life often cause planned minimum-length paths to become useless. Therefore, Structured Query Language (SQL) was utilized in [14] to record the real-time road traffic information when traffic accidents occur. Simultaneously, case-based reasoning was used to record the previous planned data in the database. When the same path planning problem presents itself, it can be processed faster, without computation, again. However, actual traffic conditions still need be considered to assess whether the previously planned path can still meet the requirements of users, taking into account other factors. Moreover, a knowledge-based approach is adopted to reduce searching coverage. Features of road networks, such as railway crossings or bridges, are classified based on a hierarchy, and the resulting links are searched sequentially. Thus, a network can be segmented into several sub-networks and then Dijkstra's [3] shortest path algorithm is run in the sub-networks to decrease the number of computations. Furthermore, Huang et al. [15] attempted to improve the lifelong planning A^* algorithm (LPA^*) to solve path planning problems in a dynamic environment. With this method, linear distances from insert vertex to goal vertex and start vertex to goal vertex are considered. Minimum Bounded Rectangle (MBR) was utilized to eliminate unnecessary nodes in order to save searching space. Meanwhile, the planned shortest paths were recorded and the portions of related paths were computed when their path weights changed. However, computing time for both Dijkstra's algorithm and the A^* algorithm drastically increase with graph scale and variation of a dynamic graph.

Cellular automata are made up of cells like points in the lattice, and follow a simple rule [16]. They include large numbers of simple identical components with local interactions. Cellular automata can not only perform complex computations with a high degree of efficiency and robustness, but can also be addressed as massive collections of simple objects interacting locally with each other [17]. Each cell can assume a state from a finite set, changing its state simultaneously in discrete time steps according to a local transition rule. The new state of each cell depends on its previous state, as well the state of its adjacent cells [18]. Cellular automata have been used to develop a wide variety of complex systems in the real world, owing to their simple mathematical constructs and distinguishing features. For example, a method for spatial electric load forecasting using a reduced set of data was presented in [19]. The method used a cellular automata model for the spatiotemporal allocation of new loads in the service zone. The approach was tested in a real system in a mid-size city, showing good performance. Cheng et al. [20] proposed a cellular automata-based approach for clustering the interests of car drivers, increasing the lifetime of interest groups, and increasing the throughput in vehicle-to-vehicle environments. Simulation results revealed the strengths of the proposed cellular automata clustering algorithm in terms of increased group lifetime and throughput for vehicular networks. Tonguz et al. [21] introduced a new cellular automata approach to construct an urban traffic mobility model. Based on the developed model, characteristics of global traffic patterns in urban areas were studied. Their experimental results showed that different control mechanisms used at intersections, such as cycle duration, green split, and coordination of traffic lights have a significant effect on inter-vehicle spacing distribution and traffic dynamics.

To tackle the above-mentioned problems of traditional static path planning, this research proposes a novel dynamic optimal path planning method. The concept of cellular automata is used to collect the real-time road conditions and derive the appropriate paths for users. A local server is established in each region, and hierarchical servers are employed to plan routes. The real-time road information is first collected via servers. The information is analyzed, and a suitable route is then generated for drivers. Meanwhile, current users' driving paths in the same region are also considered in the decision making process to prevent too many most vehicles from moving on the same route.

Type-2 fuzzy logic is adopted for path analysis for each cell established in the cellular automata algorithm. The concept of a type-2 fuzzy set was introduced by Zadeh as an extension of an ordinary fuzzy set [24,25]. In an ordinary fuzzy set, the membership value for each fuzzy membership function is a crisp number in the interval [0,1]; whereas the membership value for each element of a type-2 fuzzy set is also a fuzzy set in the interval [0,1]. Recently, type-2 fuzzy logic has been successfully applied to many fields in various areas of science and technology [30–32]. In [30], the type-2 fuzzy logic based impulse detector, an image processing operator,