Sensors & Transducers, Vol. 157, Issue 10, October 2013, pp. 141-146

www.sensorsportal.com

Sensors & Transducers

http://www.sensorsportal.com

Research on Adaptive Crossroads Control Model Based on Wireless Sensor Networks

[1,2] Musong GU, [1] Yusong YAN, [1] Yong YANG, [1] Hongwei WANG, [1] Long WEI, [3] Zhen ZUO

[1] College of Information Science and Technology, Southwest Jiaotong University, Chengdu, 610000, China

[2] College of Information Science and Technology, Chengdu University, Chengdu 610106, China

[3] The Chuandongbei Gas Project, Unocal East China Sea, Ltd., Chengdu 610021, China

[1] Tel.: (86) 139 8213 8748

[1] E-mail: gms2020@163.com

Received: 13 August 2013 /Accepted: 25 September 2013 /Published: 31 October 2013

Abstract: Traffic information collection and traffic flow control around the crossroads are main topics of current traffic related research. The most commonly used method of traffic signal control at present is fixed-time control which only applies to small traffic flow circumstances with low efficiency. Therefore, we propose a new method which is an adaptive crossroads control model based on Wireless Sensor Network, to accurately monitor the traffic flow around the crossroads. Chemical Reaction Optimization is used to optimize the proposed new model and this optimized model is used for real-time traffic control. We simulated the optimized model and research results show that the new model can significantly decrease vehicles' average delay time and effectively improve the traffic capacity. *Copyright © 2013 IFSA.*

Keywords: Traffic information collection, Traffic flow control, Wireless sensor network, Chemical reaction optimization.

1. Introduction

Automobile brings great convenience to human's travel, but rapid increment of the automobile results in severe traffic congestion as well as a series of social and economic problems. Thus researchers are researching many key problems in transportation with the aim of alleviating traffic congestion and enhancing the capacity of the existing infrastructure. Among the key problems, traffic information collection [1] and traffic flow control [2] around the crossroads is the main research topic. At present, fixed-time control is the most popular method in crossroads traffic control. But fixed-time control cannot be adjusted according to the change of traffic flow and it only applies to small traffic flow circumstances. Therefore, a new method needs to be introduced into traffic information collection and traffic flow control.

The earliest research on traffic signal optimization control begins from 1960s. Webster (1958) and Miller (1963) established models based on fixed period signal control separately, aiming at the minimal average vehicle delay. In 1977, Pappis and

Mamdani proposed two-phase fuzzy control for one-way single crossing [3, 4]. However, it was based on the idealized crossing model and cannot be applied in real traffic circumstances. Kiln introduced Genetic Algorithm into fuzzy control in crossing signal in 2001, which optimized parameters of fuzzy control model and improved the performance of the model effectively. Most of currently used traffic control strategies are based on the forecast of the traffic flow. But these strategies still cannot avoid the situation that “green light on for no traffic and red light on for heavy traffic”. If we know the accurate time that each vehicle arrives at the crossroads, then the control model can be easily established. Therefore we expect to propose a new model which can accurately monitor the vehicles to the crossroads and control the traffic light according to the monitored results to improve the traffic efficiency.

In order to realize accurate control, we can use Wireless Sensor Network (WSN) [5-8]. A WSN is composed of numerous sensor nodes with the abilities of sensing, data processing and wireless communication. Because of its powerful function and low energy cost, the WSN has been used in a wide range of fields for monitoring and long distance controlling. The information of position is very important in many applications, such as object tracking, traffic management and location based routing.

In the next section, an adaptive crossroads control model is introduced. In section 3, Chemical Reaction Optimization (CRO) [9-12] is used to optimize the model. In Section 4, the computational results along with a comparison to other algorithms are presented. The last section is to conclude this paper.

2. Adaptive Crossroads Control Model

There are three types of wireless sensor network nodes in the adaptive crossroads control model: monitor node, fixed node and mounted node. The main function of the monitor node is to collect information of the vehicles on the road and send it to the control center. The monitor node is deployed every 100 to 200 meters along the road according to the coverage of the node. The fixed node is installed beneath the roadbed to monitor vehicle information and transmit it to the monitor node. The mounted node is installed in a vehicle, collecting information of the vehicle itself and transmitting it to the monitor node. The control center analyses all the information collected and controls the traffic light according to the signal control algorithm.

In order to monitor the actual arriving traffic flow, we will install a set of sensor nodes at the entrance of each lane to monitor the traffic flow and the length of vehicles waiting in queue, as illustrated in Fig. 1. A typical crossroads consists of four entrances: east, south, west and north. Each entrance can do right turn, left turn and going straight. Each entrance has a underpass, thus, pedestrian and non-motorized vehicles will not affect this model.

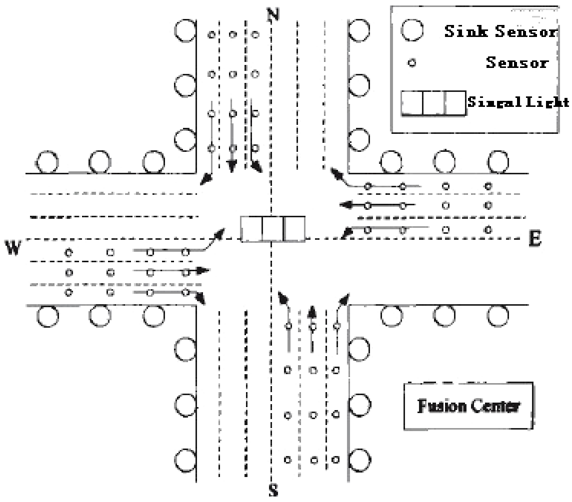

Fig. 1. Illustration of the crossroads.

2.1. Crossroads Phase-Hopping Induction Control Algorithm

TPx represents the period of one phase that allows vehicles to pass and it controls the algorithm’s execution time. TQ is the passing time of the fleet in each lane. PX is passing phase. The minimum TPx is 15 s, with the maximum of 90 s considering drivers’ tolerance. Based on the aforementioned assumptions, we can implement the Crossroads Phase-Hopping Induction Control Algorithm. The algorithm can be described as follows:

Step l: Collect real-time data to acquire L_{queue} which represents the real-time fleet length. T_{wait} means the average waiting time. Calculate the required passing time of each phase according to the data received by WSN and the longest required passing time is chosen as PX.

Step 2: T_{max} is the maximum simulation passing time of each phase. Set T_{max}=90, TPx=MIN(TQ(PX), T_{max});

Step 3: Switch to non-chosen phase controllers and let them wait for TPx;

Step 4: If TQ(PX) > 0, then let T_{max} = T_{max} -TPx, TPx = MIN(TQ(PX), T_{max}), go to Step 3;

Step 5: Execute Step l, if there is any TQ(PX) ≠ 0, then T_{max} =90s, TPx= MIN(TQ(PX), T_{max}), go to Step3; otherwise, if the passing time of non-chosen phase is 0, let T_{max} =90s, TPx=MAX(MIN(1, TQ(PX)),T_{max}), go to Step 3.

In order to evaluate the performance of this control algorithm, we use the Webster formula to calculate the delay time of crossroads.

$$\bar{d} = \frac{C(1-\lambda)^2}{2(1-y)} + \frac{x^2}{2q(1-x)} - 0.65(\frac{C}{q^2})^{\frac{1}{3}} x^{2+5\lambda} \quad (1)$$

By considering the delay time at each entrance of each phase, we minimize the overall delay time at all entrances of all phases. Therefore, we take the four-phase four-lane crossroads as an example, the objective function can be described as:

$$\min W = \sum_{i=1}^{4} \sum_{j=1}^{4} \{[\frac{C(1-\lambda_i)^2}{2(1-\lambda_i y_{ij})} + \frac{x_{ij}^{\ 2}}{2q_{ij}(1-x_{ij})}]q_{ij}\}, \tag{2}$$

where C is the duration (s); λ_i represents the green ratio of phase *i; q_{ij}* means the traffic flow at entrance *j* of phase *i*; x_{ij} is the traffic flow saturation at entrance *j* of phase *i*. This simulation uses the Chemical Reaction Optimization algorithm to optimize the model above.

3. Chemical Reaction Optimization Algorithm

3.1. Chemical Reaction Optimization Algorithm Introduction

Chemical Reaction Optimization is a relatively new algorithm used to solve optimization problems, which simulates the chemical reaction process. Because molecules tend to stay at the most stable energy state, products are more stable than the reactants. CRO performs a sequence of chemical reactions to reach the global minimum in solutions.

Every molecule has two important properties, potential energy (*PE*) and kinetic energy (*KE*). *PE* and *KE* represent the objective function value and the ability to accept worse solution, respectively. There are four types of chemical reactions, including on-wall ineffective collision, decomposition, intermolecular ineffective collision, and synthesis which take place in a closed container. There is only one molecule colliding with the wall of container in the on-wall ineffective collision. In the decomposition, one molecule is divided into several. The intermolecular ineffective collision involves more than one molecule interacting with each other. The synthesis combines many molecules into one. In the two aforementioned ineffective collisions, the number of molecule(s) remains the same and only the neighborhoods of original solution are searched. The other two reactions help the algorithm jump out of the local optimums by generating new solutions very different from the original ones. The potential energy change in the aforementioned chemical reactions can be depicted by Fig. 2.

3.2. Algorithm Design

In this paper, we take the four-phase four-lane crossroads as an example. Tij represents the traffic flow delay time at entrance j of phase i. Therefore, the overall delay time is $\sum_{i=1}^{4} \sum_{j=1}^{4} T_{ij}$. Then we can encode this overall delay time in binary form.

We are going to describe the operators in the four types of chemical reactions. The operators can be described by binary coding as follows:

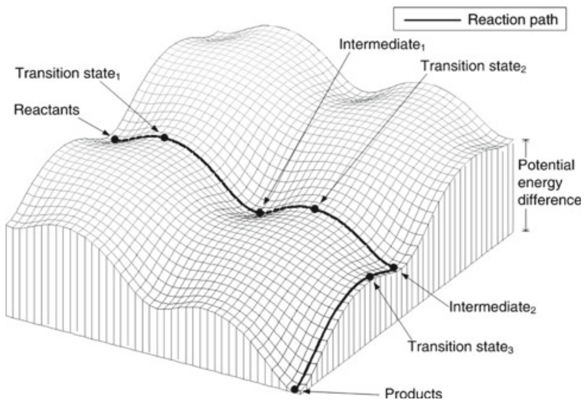

Fig. 2. Illustrative representation of a chemical reaction.

3.2.1. On-wall Ineffective Collision

In the on-wall ineffective collision, one molecule hits the container wall. Then we add little perturbation to it. This process can be described by randomly exchanging two bits in an existing ω to get the new solution ω'. For example:

$$\underbrace{00\underline{1}0100110000\underline{0}101}_{\omega} \rightarrow \underbrace{00\underline{0}0100110000\underline{1}101}_{\omega'} \tag{3}$$

3.2.2. Inter-molecular Ineffective Collision

In the inter-molecular ineffective collision, two molecules ω_1 and ω_2 collide with each other. Then we add small perturbations to both of them which results in two new solutions ω'_1 and ω'_2. To make it happen, we simply apply the mechanism used for the on-wall ineffective collision to ω'_1 and ω'_2 separately. For example:

$$\begin{aligned} &\underbrace{[00\underline{1}0100110000\underline{0}101]}_{\omega_1} \text{ AND } \underbrace{[00101001\underline{1}000\underline{0}0101]}_{\omega_2} \rightarrow \\ &\underbrace{[00\underline{0}0100110000\underline{1}101]}_{\omega_1'} \text{ AND } \underbrace{[00101000\underline{0}100\underline{1}0101]}_{\omega_2'} \end{aligned} \tag{4}$$

3.2.3. Decomposition

In the decomposition, one molecule *ω* splits into two, ω'_1 and ω'_2 which are quite different from *ω*. In order to make it happen, we test every bit of *ω*. Consider the *i*-bit of *ω as ω*(*i*). If it equals to 1, then its value is copied to the same position of ω'_1 and the rest bits of ω'_1 are set at random. Otherwise, the value of $\omega(i)$ is copied to the same position of ω'_2 and the rest bits of ω'_2 are set at random. In this way, ω'_1 is very different from ω'_2. This mechanism helps the CRO algorithm jump out of the local optimums. For example:

$$\underbrace{0010100110000101}_{\omega} \rightarrow \underbrace{1010101110110101}_{\omega_1'} \quad AND\ \underbrace{0000100100000100}_{\omega_2'} \tag{5}$$

3.2.4. Synthesis

In the synthesis, two molecules ω_1 and ω_2 combine into a new one ω'. In order to make ω' very different from ω_1 and ω_2, we adopt "exclusive or" to ω_1 and ω_2 to produce ω'. For example:

$$\underbrace{[0010100110000101]}_{\omega_1} \oplus \underbrace{[0100000110101001]}_{\omega_2} \rightarrow \underbrace{[0110100000101100]}_{\omega'} \tag{6}$$

Fig. 3 demonstrates the flow chart of the CRO algorithm. The major steps of the algorithm are summarized as follows:

The major steps of the algorithm are summarized as follows:

Step 1: In the initialization, we initialize a set of molecules with the size of popsize. The objective function is the *PE* of the molecule. The initial *KE* of every molecule is set to the value of *InitialKE*.

Step 2: In each iteration, we decide the collision type according to whether it is an uni-molecular or an inter-molecular reaction. When it is an uni-molecular, we adopt the On-wall ineffective collision or decomposition. When it is an inter-molecular, we adopt the Inter-molecular ineffective collisions or Synthesis.

Step 3: Calculate the value of the objective function and check if it is the new minimum point.

Step 4: The iteration process continues until a stopping criterion is satisfied. We output the best-so-far solution in the final stage.

Stopping criterion: The program will be stopped when the iteration number reaches the predetermined maximum iteration number.

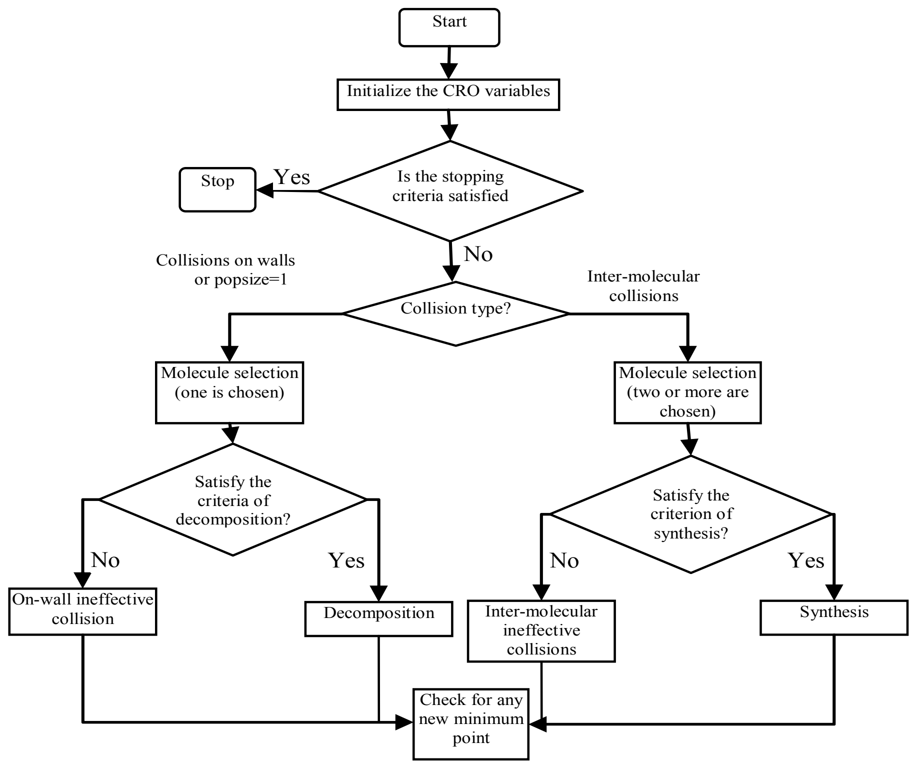

Fig. 3. Flow Chart of the CRO Algorithm.

4. Simulation Results

In this paper, we simulated a Wireless Sensor Network in Network Simulator version 2. The traffic information of the crossroads is shown in Table 1.

In this simulation, we compare the Chemical Reaction Optimization – optimized Adaptive Crossroads Control Model (CRO-ACCM), the Genetic Algorithm [13, 14] – optimized Adaptive Crossroads Control Model (GA-ACCM) and the Fixed-Time Control Model (FTCM). The strategy of the Fixed-Time Control Model is that green light is on for going straight/right turn for 60 s, then be on for left turn for 20 s. And this strategy rotates within four phases. We implemented the simulator in MATLAB. The experimental results are shown in Fig. 3. We implemented the simulator in MATLAB. All simulations were implemented on a PC with an Intel(R) Core(TM)2 Duo CPU @1.80 GHz and 4 GB of RAM. The parameter values of CRO are given as follows: PopSize = 16, KELossRate = 0.2, InitialKE = 800, MoleColl = 0.5, α = 3000, and β = 10.

Table 1. Traffic information of the crossroads.

Direction of travel	Traffic flow in each lane (no. of vehicles per hour)			Original length of flit (no. of vehicles)		
	L	F	R	L	F	R
E	254	790	235	6	16	7
W	362	875	464	9	15	9
S	397	1022	305	7	9	8
N	427	902	227	9	13	11

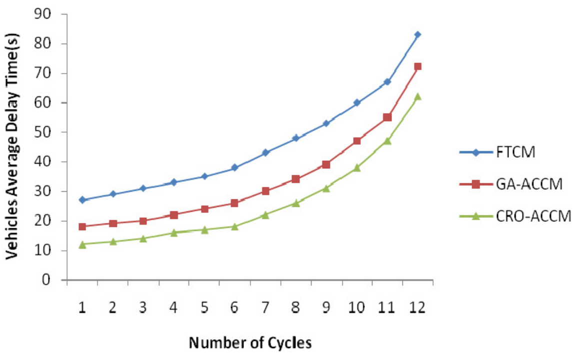

Fig. 4. Vehicles Average Delay Time – Number of Cycles.

With the CRO-ACCM, the average delay time of 12 cycles is 32.26 s. Compared with the FTCM and the GA-ACCM, the average delay time of the CRO-ACCM decreases 19.25 % and 7.5 % respectively. We can know from Fig. 3 that as the traffic flow increases at the crossroads, the performance of the CRO-ACCM decreases. In the unsaturated situation, the CRO-optimized model significantly decreases the average delay time compared to the Fixed-Time Control Model.

This CRO-ACCM can choose the phase as needed. This model not only inherits the advantages of induction control, but also overcomes the disadvantage of limited application range. In order to ensure the compatibility, we describe the crossroads model only by the lane and the phase. Simulation experiment proves that this model can solve the traffic jams effectively.

5. Conclusion

In this paper, we proposed an Adaptive Crossroads Control Model which is based on WSN. CRO is used to optimize this model. According to the results calculated from the model, we can realize real-time control of traffic lights at the crossroads.

Simulation experiment proves that the average delay time obtained from the CRO-ACCM is much shorter than the ones obtained from the GA-ACCM and FTCM. This means that the CRO-ACCM can effectively decrease vehicles' average delay time and significantly enhance the traffic efficiency at the crossroads.

Acknowledgement

This work is supported by the Scientific Research Foundation of Sichuan Provincial Education Department (No.13ZB0052).

References

[1]. T. Matsuo, Y. Kaneko and M. Matano, Introduction of intelligent vehicle detection, in *Proceedings of the IEEE Vehicle Navigation and Information Systems Conference*, 1993, pp. 529-533.

[2]. L. A. Klein, Traffic parameter measurement technology evaluation, in *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, 1999, pp. 709-713.

[3]. G. F. Newell, A simplified theory of kinematic waves in highway traffic, *Transportation Research*, Vol. 27, No. 4, 1993, pp. 281-314.

[4]. C. F. Daganzo, The cell transmission model: a dynamic representation of highway traffic consistent with the hydrodynamic theory, *Transportation Research*, Vol. 24, No. 3, 1994, pp. 104-112.

[5]. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey, *Computer Networks*, Vol. 38, No. 4, 2002, pp. 393–422.

[6]. C. Savarese, J. Rabaey, K. Langendoen, Robust positioning algorithm for distributed ad-hoc wireless sensor networks, in *Proceedings of the USENIX Technical Annual Conference*, 2002, pp. 155-159.

[7]. S. Zhang, A. G. Constantinides, Lagrange programming neural networks, *IEEE Transaction on Circuits and Systems-II: Analog and Digital Signal Processing*, Vol. 39, No. 7, 1992, pp. 441-452.

[8]. I. K. Altínel, N. Aras, E. Güney, C. Ersoy, Binary integer programming formulation and heuristics for differentiated coverage in heterogeneous sensor networks, *Computer Networks*, Vol. 52, No. 12, 2008, pp. 2419–2431.

[9]. A. Y. S. Lam, V. O. K. Li, Chemical-reaction-inspired metaheuristic for optimization, *IEEE*

Transactions on Evolutionary Computation, Vol. 14, No. 3, 2010, pp. 381–399.

[10]. A. Y. S. Lam, V. O. K. Li, Chemical reaction optimization for population transition in peer-to-peer live streaming, in *Proceedings of the IEEE Congress on Evolutionary Computation*, 2010, pp. 235-240.

[11]. J. Xu, A. Y. S. Lam, V. O. K. Li, Chemical reaction optimization for task scheduling in grid computing, *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 2011, pp. 103-107.

[12]. J. Xu, A. Y. S. Lam, V. O. K. Li, Chemical reaction optimization for the grid scheduling problem, in *Proceedings of the IEEE International Conference on Communication*, 2010, pp. 470-475.

[13]. D. E. Goldberg, Genetic algorithms in search, optimization and machine learning, reading, mass, *Addison Wesley*, 1989.

[14]. L. Davis (ed), Handbook of genetic algorithms, *Van Nostrand*, 1991.