J. Software Engineering & Applications, **2009, 2: 344-349**
doi:10.4236/jsea.2009.25045 Published Online December 2009 (http://www.SciRP.org/journal/jsea)

Research on OD Matrix Calculation Based on Quantum Behaved Particle Swarm Optimization Algorithm

Lianyu WEI, Jianfu DU

Institute of Civil Engineering, Hebei University of Technology, Tianjin, China.
Email: {wly57, batdujianfu}@126.com

Received June 20th, 2009; revised August 25th, 2009; accepted September 2nd, 2009.

ABSTRACT

Traffic information is so far less than the number of OD variables, that it is difficult to obtain the satisfactory solution. In this paper, a method based on Quantum behaved Particle Swarm Optimization (QPSO) algorithm is developed to obtain the global optimal solution. It designs the method based on QPSO algorithm to solve the OD matrix prediction model, lists the detailed steps and points out how to choose the PSO operator. Moreover, it uses MATLAB programming language to carry out the simulation test. The simulation results show that the method has higher efficiency and accuracy.

Keywords: *OD Matrix Prediction Model, QPSO, Simulation Analysis, Optimal Design*

1. Introduction

The traffic between import and export at the intersection is an important data for urban traffic management and control, and also impacts on the control of traffic lights at the intersection directly. Therefore, how to obtain accurate intersection OD and avoid the time-delay of control and decision-making system are very important issues. As the OD matrix calculation based on traffic has advantages, such as convenience, rapidness, low-cost, effectiveness, it has been more and more used in transport planning recently. However, when the number of surveyed sections is less than the number of OD variables, the solution of OD matrix calculation will be non-unique and cannot guarantee the accuracy [1]. Moreover, the objective function and fitness function of OD matrix calculation are mostly nonlinear equations. In light of the above issues, taking the reality situation that observation error and random error exist into account, the relationship between OD traffic and road traffic is not absolute linear. Although through rigorous mathematical methods, it cannot get an exact solution to the problem.

In pursuit of finding solution to these problems many researchers have been drawing ideas from the field of biology. A host of such biologically inspired evolutionary techniques have been developed namely Genetic Algorithm (GA) ([Baykasoğlu et al., 2008], [Costa et al., 2004], [Grosset et al., 2001], [Gürdal Soremekun et al., 2001], [Le Riche and Haftka, 1993], [Park et al., 2001], [Rajendran and Vijayarangan, 2001] and [Walker and Smith, 2003]), Artificial Neural Networks (ANN) (Garg, Roy Mahapatra, Suresh, Gopalakrishna & Omkar, 2007), Artificial Immune System (AIS) (Omkar, Khandelwal, Santhosh Yathindra, Narayana Naik & Gopalakrishna, 2008) and Particle Swarm Optimization (PSO) (Omkar, Mudigere, Narayana Naik & Gopalakrishna, 2008; Parsopoulos, Tasoulis & Vrahatis, 2004) which are widely used for solving such optimization problems. All of these algorithms with their stochastic means are well equipped to handle such problems [2].

Particle Swarm Optimization was introduced by Eberhart and Kennedy [3], inspired by the social behavior of animals such as bird flocking, fish schooling, and the swarm theory. Compared with GA and other similar evolutionary techniques, PSO has some attractive characteristics and in many cases proved to be more effective (Hassan, Cohanim, Weck & Venter, 2005). Both GA and PSO have been used extensively for a variety of optimization problems and in most of these cases PSO has been proven to have superior computational efficiency ([Hassan et al., 2005] and [Sun, 2008]; Zhang et al., 2003 L. B. Zhang, C. G. Zhou, X. H. Liu, Z. Q. Ma, M. Ma & Y. C. Liang (2003) solved multi-objective optimization problems using particle swarm optimization in Proceedings of the IEEE congress on evolutionary computation (CEC). Since

1995, many attempts have been made to improve the performance of the PSO (Clerc, 2004; Zheng, Ma, & Zhang, 2003). (Sun et al., 2004) and (Sun and Xu et al., 2004) introduced quantum theory into PSO and proposed a Quantum-behaved PSO (QPSO) algorithm, which are guaranteed theoretically to find good optimal solutions in search of space. The experiment results on some widely used benchmark functions show that the QPSO works better than standard PSO ([Sun et al., 2004] and [Sun and Xu et al., 2004]) and is a promising algorithm. Hence in the current work we propose to employ a multi objective optimization method based on QPSO and compare it to its predecessor PSO, which has been already implemented by Omkar and Mudigere et al. (2008) [2].

This paper firstly establishes maximum entropy OD matrix calculation model. It makes full use of the quantum behaved particle swarm optimization algorithm (QPSO algorithm) to solve the global optimization of the objective function, lists the detailed steps of calculation and shows how to choose the particle swarm operator, and then calculates the OD matrix of road intersection. Finally, compared to the results of conventional PSO algorithm, it verifies the superiority of QPSO algorithm. Therefore, this paper provides a more reliable method to solve the OD matrix of road intersection.

2. Origin—Destination Matrix Prediction Model

2.1 Maximum Entropy Model

The maximum entropy approach is motivated by 'information theory' and the work of Shannon, 1948. C. E. Shannon, A mathematical theory of communication. Bell Syst. Tech. J. (1948), pp. 379–423. Shannon (1948) who defined a function to measure the uncertainty, or entropy, of a collection of events, and Jaynes who proposed maximizing that function subject to appropriate consistency relations, such as moment conditions. The maximum entropy (ME) principle and its sister formulation, minimum cross-entropy (CE), are now used in a wide variety of fields to estimate and make inferences when information is incomplete, highly scattered, and/or inconsistent (Kapur and Kesavan, 1992). In economics, the ME principle has been successfully applied to a range of econometric problems, including nonlinear problems, where limited data and/or computational complexity hinder traditional estimation approaches. Theil (1967) provides an early investigation of information theory in economics. Mittelhammer et al. (2000) provide a recent text book treatment which is focused more tightly on the ME principle and its relationships with more traditional estimation criteria such as maximum likelihood [4].

In general, information in an estimation problem using the entropy principle comes in two forms: 1) information (theoretical or empirical) about the system that imposes constraints on the values that the various parameters can take; and 2) prior knowledge of likely parameter values. In the first case, the information is applied by specifying constraint equations in the estimation procedure. In the second, the information is applied by specifying a discrete prior distribution and estimating by minimizing the entropy distance between the estimated and prior distributions — the minimum Cross-Entropy (CE) approach. The prior distribution does not have to be symmetric and weights on each point in the prior distribution can vary. If the weights in the prior distribution are equal (e.g. the prior distribution is uniform), then the CE and ME approaches are equivalent.

The model is as follows:

$$\max E = -\sum_{i=1}^{n} \sum_{j=1}^{n} T_{ij} \left(\ln T_{ij} - \ln T \right) \tag{1}$$

$$s.t. \qquad V_k = \sum_{i=1}^{n} \sum_{j=1}^{n} T_{ij} P_{ij}^{k} \qquad k = 1, \mathbf{L}\, m, \tag{2}$$

$$T_{ij} \ge 0 \qquad \forall i, j \tag{3}$$

$$T = \sum_{i=1}^{n} \sum_{j=1}^{n} T_{ij} \tag{4}$$

Where n is the number of OD pairs; T_{ij} is the estimated OD matrix; V_k is the detected traffic on section k; p_{ij}^{k} is the distribution ratio of T_{ij} on section k, which is obtained by the traffic distribution.

2.2 Hopfield Neural Network Model

The Hopfield Neural Network model [4] is

$$\begin{cases} C\dfrac{du_i}{dt} = -\dfrac{U_i}{R} + I_i + \sum_{j=1}^{N} T_{ij} V_j, \\ V_i = g\left(u_i\right), i = 1, 2, \mathbf{K}, N. \end{cases} \tag{5}$$

Here in Equation (5), T_{ij} is the connection weight value between node I and node J, and $T_{\text{ij}} = T_{\text{ji}}$; $g(\cdot)$. is a function with $g'(\cdot) > 0$; U_i is the input of node I; V_i is the output of node I; I_i is the constant value of node I; C is a positive constant; and R is a positive constant.

Equation (5) can be abbreviated to be

$$\begin{cases} \dfrac{du_i}{dt} = I_i + \sum_{j=1}^{N} T_{ij} V_j, \\ V_i = g\left(u_i\right), i = 1, 2, \mathbf{K}, N. \end{cases} \tag{6}$$

It can be proved that when the energy function of system (6) is

$$E = -\frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} T_{ij} V_i V_j - \sum_{i=1}^{N} V_i I_i$$

We can have $\dfrac{dE}{dt} \le 0$, and only when $dV_i / dt = 0$,

$dE/dt = 0$, $(i = 1, 2, \mathbf{K}, N)$.

The stable state of system (6) is the local minimum of the above energy function. The computation processor of system (6) is a process to find the local minimum in fact, the goal function is the above energy function of system (6).

Maximum entropy model is recognized by majority of scholars among many OD matrix calculation models, because this model's structure is simple and principle is clear. It also suits the situation without a priori OD matrix and which is a congested network [5].

2.3 Model Simplification

In light of above optimization problems, it is difficult to solve the objective function directly. So Lagrange multiplier method is used to obtain the Lagrange function L:

$$L = -\sum_{i=1}^{n}\sum_{j=1}^{n} T_{ij}(\ln T_{ij} - \ln T) + \sum_{k=1}^{m} l_k (V_k - \sum_{i=1}^{n}\sum_{j=1}^{n} T_{ij} P_{ij}^{k}) \quad (7)$$

The following Equation is the result of the first-Order derivative of T_{ij}

$$\frac{\partial L}{\partial T_{ij}} = -\ln\frac{T_{ij}}{T} - \sum_{k=1}^{m} l_k P_{ij}^{k} = 0 \quad (8)$$

So,

$$T_{ij} = T\exp(-\sum_{k=1}^{m} l_k P_{ij}^{k}) \quad (9)$$

T is given as $T = \exp(-l_0)$ and take it into the Equations (7) and (9) respectively. The results are as follows:

$$\begin{cases} \sum_{i=1}^{n}\sum_{j=1}^{n} \exp(-\sum_{k=1}^{m} l_k P_{ij}^{k}) = 1 \\ \sum_{i=1}^{n}\sum_{j=1}^{n} P_{ij}^{k} \exp(-l_0 - \sum_{k=1}^{m} l_k P_{ij}^{k}) = V_k \end{cases} \quad k = 1, \mathbf{L}, m \quad (10)$$

Above equations are a set of nonlinear equations. The number of variables is (m+1) which is equivalent to the number of equations. Lagrange multipliers: $l_0, l_1 \mathbf{L}\,\mathbf{L}\, l_m$ can be solved. And then the OD matrix can be obtained according to Equation (9).

3. Design of the Quantum Behaved Particle Swarm Optimization Algorithm

3.1 Particle Swarm Optimization Algorithm

Particle Swarm Optimization algorithm is a global optimization algorithm that can reproduce swarm intelligence. It is inspired from the foraging behavior of animal groups. When groups search for the optimal target, each individual searches for its own goal. At the same time, individual refers to other individuals who have achieved optimal location and then adjusts the next search. The algorithm uses the speed-location search model. The current location of No.i particle is defined as $X_i = (x_{i1}, x_{i2}, \ldots x_{id})$, experienced position is defined as $P_i = (p_{i1}, p_{i2}, \ldots p_{id})$. Fitness function determines the level of merits of the location, while fitness function is determined by the optimization goal. Where, the individual particle best position is abbreviated as $pbest$, and the best location that all particles have experienced is regarded as global best location ($gbest$). The speed of No. i particle is defined as $V_i = (v_{i1}, v_{i2}, \ldots v_{id})$ that is the distance each iterative particle moves. The Equations of conventional PSO algorithm are described as follows:

$$v_{i+1} = v_i + f_1 \cdot r_1 \cdot (pbest - x_i) + f_2 \cdot r_2 \cdot (gbest - x_i) \quad (11)$$

$$x_{i+1} = x_i + v_{i+1} \quad (12)$$

3.2 Quantum Behaved Particle Swarm Optimization Algorithm

Particle Swarm Optimization algorithm is based on the theory of swarm intelligence optimization algorithm. As in the classical system, particles achieve convergence in the form of orbit. Moreover, the speed of particles is limited, and the space of particles is also a limited region that can not cover the entire feasible space. Therefore, with the quantum mechanics' point of view that particles have quantum behavior, QPSO (Quantum Particle Swarm Optimization) algorithm is proposed [3,7–8]. Particle swarm achieves iterative update through the following four Equations:

$$\begin{aligned} mbset &= \sum_{i=1}^{M} pbest_i / M \\ &= \left(\sum_{i=1}^{M} pbest_{i1} / M, \sum_{i=1}^{M} pbest_{i2} / M, \mathbf{L}, \sum_{i=1}^{M} pbest_{id} / M \right) \end{aligned} \quad (13)$$

$$b = b_{\max} - \frac{(b_{\max} - b_{\min}) \cdot T_{\max}}{T} \quad (14)$$

$$p = (r_1 \cdot pbest + r_2 \cdot gbest) / (r_1 + r_2) \quad (15)$$

$$x(t+1) = \begin{cases} p + b \times |mbest - x(t)| \times \ln(1/u) & if\ u < 0.5 \\ p - b \times |mbest - x(t)| \times \ln(1/u) & if\ u \geq 0.5 \end{cases} \quad (16)$$

In order to avoid algorithm premature, Mean Best Position (mbest) is regarded as the barycenter of all particles. Where, $pbest_i$ is the best position of the particle. b is the contraction and expansion coefficient that impacts the convergence speed and performance of algorithm. In this paper, deal with b by adaptive changes in accordance with Equation (14). T is the number of current iteration. T_{max} is the maximum number of iterations. $r_1, r_2 \in$

$rand(0,1)$, $gbest$ is the global optimal solution, $u \in rand(0,1)$.

3.3 Optimal Design of Origin—Destination Matrix Calculation Model Algorithm

Algorithm design, including as follows:

1) *the determination of objective function*

In order to obtain better accuracy, this problem will be translated into the following optimization problem. The optimizing goal is that minimizes the mean square deviation of the calculated value of the left from the true value of the right in Equation (13). That is：

$$\min J(l)=\sqrt{\frac{\sum_{k=0}^{m}(v_k-V_k)\cdot(v_k-V_k)}{m+1}} \qquad (17)$$

Where, $v_0=\sum_{i=1}^{n}\sum_{j=1}^{n}P_{ij}^{k}\exp(-\sum_{k=1}^{m}l_k P_{ij}^{k}) \qquad V_0=1$

$$v_k=\sum_{i=1}^{n}\sum_{j=1}^{n}P_{ij}^{k}\exp(-l_0-\sum_{k=1}^{m}l_k P_{ij}^{k}), \qquad k=1,\ldots m$$

Lagrange multipliers $l_0, l_1 \mathbf{L}\,\mathbf{L}\, l_m$ are unknown. Solution space is the range of Lagrange multipliers. Because the Lagrange multipliers are not given in this model explicitly, it is necessary to estimate conservatively according to the specific issues and enlarge the range appropriately.

2) *detailed implementation steps are:*

Step1: Initialize particle swarm.

Step2: Calculate the value of particle objective function.

Step3: Update $pbest$ and $gbest$ according to particles' fitness.

Step4: Calculate *mbest* according to Equation (13).

Step5: Calculate random point p of each particle according to Equation (15).

Step6: Calculate new location of each particle according to Equation (16).

Step7: Double counting, until meet the number of iterations.

3.4 Simulation of a Typical Function Optimization

Now, QPSO algorithm is illustrated that can be applied to the circuit performance equation to solve the global minimum feasibility and effectively, by solving the Schaffer's f6 function. The value of the global minimum is 0. The Schaffer's f6 function is:

$$f(x,y)=0.5+\frac{\sin^2\sqrt{x^2+y^2}-0.5}{(1+0.001\times(x^2+y^2)^2)^2} \qquad (18)$$

$$x\in(-10,10)\,;\quad y\in(-10,10)$$

Use QPSO algorithm and PSO algorithm to solve the above equation respectively.

The selected parameters are as follows: Particle number is 10; the maximum number of iterations is 1000; the accuracy is set to 1e-25. Figure 1 and Figure 2 show the obtained convergence curves.

As can be seen from Figure 1, when the iteration number reached 210, the curve tends to converge. Whereas, when the iteration number reached 310, the curve of Figure 2 tends to converge; the solution of QPSO algorithm from Figure 1 is 0, while, the solution of PSO algorithm from Figure 2 is 0.000114686. Therefore, this example shows that the convergence speed and accuracy of QPSO algorithm are far better than the PSO algorithm for optimization problems.

4. Simulation Examples

4.1 Simulation Results

This paper uses MATLAB programming language to carry out the simulation test. The intersection is shown in Figure 3. The traffic of every import is regarded as x_1, x_2, x_3, x_4 respectively. The traffic of every export is regarded as y_1, y_2, y_3, y_4 respectively. The traffic at intersection can be obtained by the detector. Count the total number of observed vehicles from 8:00 a.m. to 8:00 p.m.

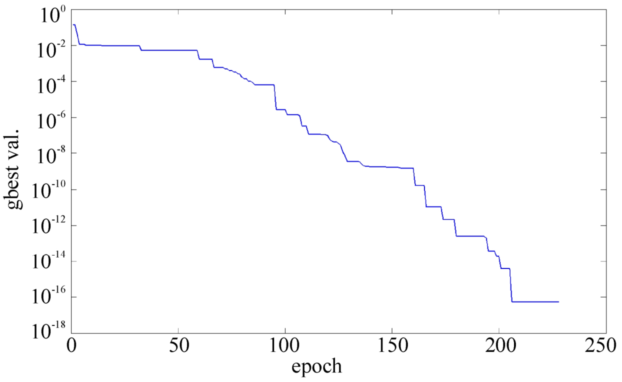

Figure 1. Convergence curve of QPSO algorithm

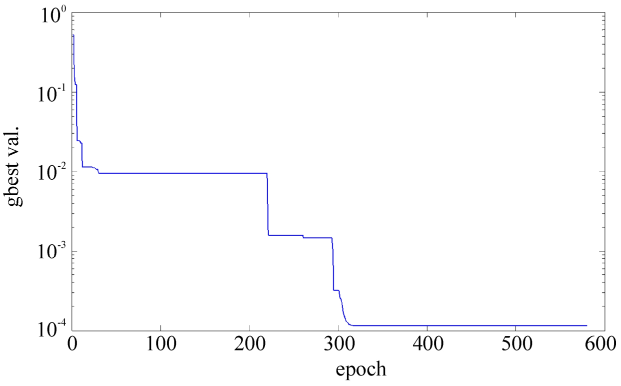

Figure 2. Convergence curve of PSO algorithm

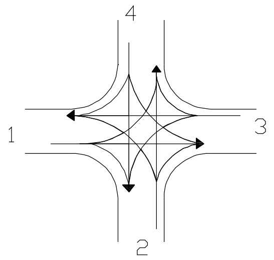

Figure 3. A intersection plot of Tianjin

$$V = (x_1, x_2, x_3, x_4, y_1, y_2, y_3, y_4)^T$$
$$= (5321, 5069, 4986, 5035, 5109, 5220, 4823, 5259)^T$$

Traffic distribution matrix is as follows:

$$P = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

4.2 Simulation Steps

1) *Determine the solution space*

The value of the OD matrix T can be calculated. It is equivalent to the total of all the exports (or imports): $T = \sum_{i=1}^{4} x_i = \sum_{j=1}^{4} y_j = 20411 = \exp(9.92)$. We can obtain that l_0 = -9.92, according to $T = \exp(-l_0)$.

In addition, various components of the OD matrix should meet $1 \le T_{ij} \le T$, so we can obtain $0\,\mathbf{p} -l_0 - \sum_{k=1}^{m} l_k P_{ij}^k\ \mathbf{p} -l_0$. Or $0\,\mathbf{p} \sum_{k=1}^{m} l_k P_{ij}^k\ \mathbf{p} -l_0\ \mathbf{p}\ 10$. According to $T_{ij} = T\exp(-\sum_{k=1}^{m} l_k P_{ij}^k)$, the element of P matrix is 0 or 1, so $l_k (k = 1, \mathbf{L}\ m)$ can not be all negative. There are two situations: The whole elements are nonnegative; one part of elements is positive, and the other parts are zero or negative. For the situation that the whole elements are nonnegative, the value of l_k should be within the range [0,10]. For the second situation, the range can not be defined easily. As result of mutual influence between several unknown quantities, its value may be very large or may be very small. Therefore, we can obtain that there is a solution set and the values are within the range [0,10].

2) *Determine the conditions of termination*

Set the maximum number of iterations as 300. When the average mean square error (MSE) of the objective function $J(l)$ is less than 0.1, we regard the obtained l_k (k=0,1… 8) as the approximate optimal solution of Equations (10) and terminate iterations.

3) *Determine the number of particles*

The optimal goal is that the objective function (mean square error MSE) close to zero as possible. Selected parameters are as follows: The initial velocity of particles is generated randomly; the number of particles is 20.

The simulation results are as follows:

4.3 Analysis the Simulation Results

Compared to the results of conventional PSO algorithm, we can see that the results of the conventional PSO algorithm and QPSO algorithm all meet the accuracy requirements (MSE<0.1). However, when the conventional PSO algorithm is used to solve individual variables, a local optimal solution is obtained that led to a larger absolute error of individual variables. As can be seen from Table 1, the absolute error between actual traffic and calculated traffic by QPSO algorithm is too small (≤0.103) to meet the accurate requirements fully. Compared to Table 3, from Table 4 we can see the effect of the OD matrix calculation is ideal.

Table 1. Absolute error of calculating road traffic

Absolute Error	V0-v0	V1-v1	V2-v2	V3-v3	V4-v4	V5-v5	V6-v6	V7-v7	V8-v8
QPSO	**0.0002**	**-0.026**	**0.038**	**-0.103**	**0.063**	**0.025**	**-0.054**	**0.019**	**-0.0237**
PSO	**0.0003**	**-0.042**	**0.056**	**-0.120**	**0.084**	**0.023**	**-0.076**	**0.032**	**-0.0438**

Table 2. Comparison of MSE

MSE	QPSO	PSO
MSE= J（λ）	0.0485	0.0630

Table 3. Factual OD matrix of intersection

O \ D	1	2	3	4
1	0	1936	2351	1034
2	1241	0	1392	2436
3	1808	1389	0	1789
4	2060	1895	1080	0

Table 4. Calculated OD matrix with QPSO

O \ D	1	2	3	4
1	0	1936.01	2351.02	1034
2	1240.99	0	1391.98	2435.99
3	1808.04	1389.03	0	1789.03
4	2059.95	1895.01	1079.98	0

5. Conclusions

This paper calculates the OD matrix calculation model using QPSO algorithm, and determines the fitness function, according to the maximum entropy model. We use this model to simplify the linear constraints between the traffic and the OD matrix, and solve the optimal solution of nonlinear equation using QPSO algorithm. It can be seen from the above simulation that OD matrix calculation proposed in this paper is effective. It proves that application QPSO algorithm in the field of OD matrix calculation is feasible. It can considerably reduce the iterative number that objective function can reach convergence. Moreover, QPSO algorithm can improve the accuracy of the calculation and solve the problem of no convergent and insufficient accuracy. We will further study on how to apply QPSO algorithm in OD matrix calculation of a large and complex network.

6. Acknowledgments

The paper is supported by Tianjin Science and Technology Project Fund.

REFERENCES

[1] W. Wang and J. Q. Xu, "Urban transportation planning theories and methods," China Communications Press, 1992.

[2] S. N. Omkar, R. Khandelwal, T. V. S. Ananth, G. N. Naik, and S. Gopalakrishnan, "Quantum behaved Particle Swarm Optimization (QPSO) for multi-objective design optimization of composite structures," Expert Systems with Applications, Vol. 36, No. 8, pp. 11312–11322, October 2009.

[3] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," Proceedings IEEE International Conference NeuralNetworks [C]. Piscataway, NJ: IEEE Press, pp. 1942–1948, 1995.

[4] C. Arndt, S. Robinson, and F. Tarp, "Parameter estimation for a computable general equilibrium model: a maximum entropy approach," Economic Modelling, Vol. 19, No. 3, pp. 375–398, May 2002.

[5] Z. J. Gong, "Estimating the urban OD matrix: A neural network approach," European Journal of Operational Research, Vol. 106, No. 1, pp. 108–115, April 1998.

[6] M. Brenninger-Göthe, K. O. Jörnsten, and J. T. Lundgren, "Estimation of origin-destination matrices from traffic counts using multiobjective programming formulations," Transportation Research Part B: Methodological, Vol. 23, No. 4, pp. 257–269, August 1989.

[7] G. R. Widom, "Data guides enabling query formulation and optimization in semistructured databases," The 23rd VLDB[C], Athens, Greece, pp. 436–445, 1997.

[8] T. Milo and D. Suciu, "Index Structures for path expressions," The 7th ICDT[C], pp. 277–295, 1999.