Research Article

A Hybrid Algorithm of GA + Simplex Method in the WSN Localization

Feng Wang, Cong Wang, ZiZhong Wang, and Xue-ying Zhang

College of Information Engineering, Taiyuan University of Technology, Taiyuan 030024, China

Correspondence should be addressed to Feng Wang; wangfeng@tyut.edu.cn

Received 9 April 2015; Revised 8 July 2015; Accepted 15 July 2015

Academic Editor: Lillykutty Jacob

Copyright © 2015 Feng Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Localization provides the key support for wireless sensor networks (WSNs). In order to solve the large-error problem in the third phase and the poor position accuracy of the least square method in the weighted DV-Hop algorithm, a hybrid algorithm of GA + simplex method was proposed in this paper. The weighted DV-Hop position algorithm was applied to estimate the distance between the unknown node and the anchor node in the first and second phase. In the third phase, a hybrid genetic algorithm with the simplex method was proposed to optimize the coordinates of the unknown nodes. In the hybrid genetic algorithm, a fitness function which combined the cost function with the penalty function was built, and the simplex method was used to increase the local search ability of the algorithm. Experiments show that both of the localization accuracy and network coverage rate are improved significantly, and the hybrid algorithm of GA + simplex method is suitable for the WSN localization.

1. Introduction

In the past decades, the advances in microelectronics, computer, and wireless communication technologies stimulate the rapid development of low-power and multifunction sensors, which can integrate the information acquisition, data processing, wireless communication, and other functions in a tiny volume. The wireless sensor network (WSN) is a multiple-hops and self-organizing network system which consists of a large number of low-cost microsensor nodes deployed in monitoring area and communicating through wireless. WSN can be used to aware, collect, and process the information of perception objects in the network coverage area and then send the results to observers. The three components of a sensor network are sensor, cognitive object, and observer.

The application prospects of sensor networks are very broad and they can be widely used in military, environmental monitoring, health care, and so forth. With in-depth research and extensive application, sensor networks will gradually be applied to the various areas of human life.

In sensor networks, the location information is crucial. Monitoring messages without the location information are usually useless [1]. Therefore, the node location is one of the most basic functions in sensor networks, and it plays a key role in various applications. The node localization can be divided into two categories: range-based and range-free. The range-based localization (RBL) is obtained by calculating the actual distance between the unknown node and the anchor nodes. The range-free localization (RFL) is obtained by estimating the distance of the two nodes. RFL is less sensitive to the complex environment in signal transmission, but its hardware costs and energy consumption are low. Therefore, RFL is very suitable for WSN applications [2].

The traditional distance vector-hop (DV-Hop) algorithm was proposed by Dragos Niculescu, which uses GPS and distance vector routing. It is a range-free method [3], meaning that it does not measure the distance directly but rather utilizes the information of the distance vector and network connectivity to get the estimated distance. Therefore, DV-Hop has less demanding on the physical measuring unit, and it performs well in the isotropic networks [2].

The biological evolution law of selecting the superior and eliminating the inferior exists in natural systems. The evolution is a self-adaptive process that behaves robustly. Inspired by the evolution law, J. Holland proposed the genetic algorithm (GA), which is very suitable for solving global optimal solution in the combination and optimization
problems [4]. It can obtain the optimal solution with a high probability and is easy to implement in mathematic models as well [5].

The paper is organized as follows. In the next section, the traditional DV-Hop positioning algorithm with its error analysis is introduced. The GA + simplex method for restricted problems is discussed in Section 3. In Section 4, an improved DV-Hop algorithm, the hybrid algorithm based on the GA + simplex method, is proposed. Experimental results and related analysis are presented in Section 5, and the conclusion is followed in the last section.

2. The DV-Hop Algorithm

2.1. The Principle. The procedure of the traditional DV-Hop algorithm is shown in Figure 1.

In general, there are three stages in the algorithm [6].

The first stage is to calculate the minimum counts of hop between unknown nodes and every anchor node. Neighboring nodes receive group messages in which anchor nodes broadcast their own position. The messages include the range of hop-counts with the initial value of zero. Receiving node will ignore group messages with the maximum of hop-counts which are broadcasted by the same anchor node until the minimum of hop-counts from every anchor node is recorded and then add 1 to the recorded hop-counts and send the value to a neighboring node. With this strategy, all nodes including unknown nodes and anchor nodes in WSN can record the minimum hop-counts to every anchor node.

The second stage is to calculate the estimated distance between unknown nodes and anchor nodes. For every anchor node, we use the hop-counts between two anchor nodes recorded in the first stage and the following formula to calculate the average hop-size:

\[
\text{HopSize}_i = \frac{\sum_{j \neq i} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{j \neq i} h_{ij}}. 
\]

Here \((x_i, y_i)\) and \((x_j, y_j)\) are coordinates of the anchor nodes \(i\) and \(j\), and \(h_{ij}\) is the hop-count between the two anchor nodes \((j \neq i)\). Anchor nodes propagate the calculated average hop-size to the network; unknown nodes receive the first average hop-size which is recorded as the average hop-distance and broadcast it to neighbor nodes. This method makes sure that most nodes recorded the latest average hop-size of anchor nodes. The distance from unknown nodes to every anchor node is calculated according to hop-counts and average hop-size.

As is shown in Figure 2, \(L_1\), \(L_2\), and \(L_3\) are anchor nodes and \(A\) is unknown node. The average hop-size of \(L_2\) is \((40 + 75)/(2 + 5) = 16.42\). Assuming that the unknown node \(A\) obtains the average hop-size from \(L_2\), then the distances between \(A\) and three anchor nodes are \(L_1: 3 \times 16.42\), \(L_2: 2 \times 16.42\), and \(L_3: 3 \times 16.42\).

The third stage is to calculate the unknown node’s location with a three-sided measurement method, the maximum
With the least square method, we have [6] \( \hat{X} = (A^T A)^{-1} A^T b \).

### 2.2 Simulation Analysis

The simulation results of the traditional DV-Hop localization algorithm are shown in Table 1, in which the total number of nodes distributed randomly is 250, and the numbers of anchor nodes are 5, 10, 15, 20, 25, and 30. From the table, we can see that, with the increase of anchor nodes' ratio, the positioning error decreases and the accuracy improves. However, the decreasing rates of positioning errors are declining, and overall errors are still significant.

The position error formula is as follows:

\[
\text{error} = \sum_{i=1}^{N-M} \sqrt{\frac{(x_i-x_p)^2 + (y_i-y_p)^2}{(N-M)R}}.
\]

Here \((x_p, y_p)\) is the true position of the \(i\)th unknown node, and \((x_{ie}, y_{ie})\) is estimated coordinate. \(R\) is WSN communication radius.

### 2.3 The Error Analysis

#### 2.3.1 Error Analysis and Improved Ideas in the Second Phase

The traditional DV-Hop location algorithm is suitable for the wireless sensor network in which anchor nodes are uniformly distributed, the network is isotropic, or most of the nodes are densely distributed. When nodes are distributed unevenly, there may be a large error when unknown nodes regard the average hop-size received first as the average hop-size to all anchor nodes. The authors in [7] use the average distance of every hop among all of the anchor nodes instead of the average hop-sizes in the traditional algorithm. In this paper, we improve the traditional algorithm through methods of assigning every received average hop-size a weight, namely, using the normalized thought to obtain the average hop-size.

#### 2.3.2 Error Analysis and Improved Ideas in the Third Phase

From the third stage of DV-Hop positioning algorithm, we can see that \(d_e\) exists in every element of \(b\). Therefore, the coordinates obtained with the formula \( \hat{X} = (A^T A)^{-1} A^T b \) are affected by the accuracy of \(d_e\). To resolve this issue, we could transform the problem into the constrained optimization question [8].

Let us denote that the real distances from the anchor nodes \(A_1, A_2, \ldots, A_n\) too are \(r_1, r_2, \ldots, r_n\), and \(\epsilon_1, \epsilon_2, \ldots, \epsilon_n\) are the measuring errors of corresponding anchor nodes. We have

\[
(x_i - x)^2 + (y_i - y)^2 = r_i^2,
\]

\[
(d_1 - \epsilon_1)^2 \leq r_1^2 \leq (d_1 + \epsilon_1)^2.
\]
Table 2: The positioning errors of DV-Hop in [7] and the modified DV-Hop.

| The number of anchor nodes | 5   | 10  | 15  | 20  | 25  | 30  |
|---------------------------|-----|-----|-----|-----|-----|-----|
| DV-Hop in [7]             | 74.35% | 72.21% | 68.32% | 65.14% | 63.02% | 61.25% |
| Modified DV-Hop           | 70.11% | 68.24% | 66.28% | 63.49% | 61.53% | 57.14% |

So the average distance of every hop is

\[ \text{HopSize}' = \sum_{i=1}^{n} \omega_i \times \text{HopSize}_i, \quad (13) \]

The above algorithm is based on the weighted average distance of each hop, and the simulation results are shown in Table 2, in which the total number of nodes distributed randomly is 250, and the number of anchor nodes is 5, 10, 15, 20, 25, and 30. The results show positioning errors comparing the above modified algorithm with [7].

As can be seen from the table, with the proportion of anchor nodes increasing, the positioning errors decrease, and the modified algorithm performs better. Although the positioning accuracies of both algorithms are increasing, the errors are still relatively large.

With the least squares weighted DV-Hop algorithm, the third phase error is still very large for nonlinear constraint problems. A hybrid genetic algorithm of GA + simplex method is presented to optimize the weighted DV-Hop algorithm.

3.2. The GA + Simplex Method for Restricted Problems in the Third Stage

3.2.1. The Principle. The basic procedure of GA is as follows. With constant iterations, the algorithm updates the assumption groups, and a fitness function is used to evaluate all individuals in each iteration, and then individuals with higher fitness values are picked with a higher probability. A new population is created through selection, crossover, and mutation genetic methods [14].

The defects of the genetic algorithm are time-consuming in computations and poor local search ability [15]. Therefore, the improved GA adopts the principle of saving the best individual in order to shorten the computation process.

The simplex method converts a variety of nonstandard forms into standard forms through series of transformations, and it is effective to solve the linear problem which contains many constraints of nonstandard forms. Therefore, it behaves strong apriority in local search ability.

3.2.2. Solving Restricted Problems. To satisfy the constraint condition in (7) which contains many constraint conditions [16], the penalty function could be used to transform the restricted problems into the unconstrained optimization problem:

\[
\begin{align*}
\text{Min} & \quad f(x, y) \\
\text{s.t} & \quad (d_i - \varepsilon_i)^2 \leq g_i^2(x, y) \leq (d_i + \varepsilon_i)^2, \\
& \quad (x_i, y_i) \in R \times R.
\end{align*}
\]
Here \((x, y) \in S\), and \(S\) is the searching space of the objective function \(f(x, y)\):

\[
g_i(x, y) = \sqrt{(x_i - x)^2 + (y_i - y)^2}.
\]  

(15)

To construct the punishment function, let

\[
h_i(x, y) = \max \left\{ \max \left( 0, g_i^2(x, y) - (d_i + \epsilon_i)^2 \right), \right.
\]

\[
\max \left( 0, (d_i - \epsilon_i)^2 - g_i^2(x, y) \right) \right\}.
\]  

(16)

Then the punishment function is \(\sigma(x, y) = \sum_{i=1}^{n} |h_i(x, y)|\).

One has \((d_i - \epsilon_i)^2 \leq g_i^2(x, y) \leq (d_i + \epsilon_i)^2\).

If \((x, y) \in \) feasible region, \(h_i(x, y) = 0\).

If \((x, y) \notin \) feasible region, we have the following two cases:

(i) \(g_i^2(x, y) > (d_i + \epsilon_i)^2 \Rightarrow h_i(x, y) = \max (0, g_i^2(x, y) - (d_i + \epsilon_i)^2 = g_i^2(x, y) - (d_i + \epsilon_i)^2;\)

(ii) \(g_i^2(x, y) < (d_i - \epsilon_i)^2 \Rightarrow h_i(x, y) = \max (0, (d_i - \epsilon_i)^2 - g_i^2(x, y)) = (d_i - \epsilon_i)^2 - g_i^2(x, y).\)

From the above two cases, we have \(\sigma(x, y) > 0\).

Therefore, the new objective function is

\[
f'(x, y) = f(x, y) + \sigma(x, y).
\]  

(17)

To solve \(f'(x, y)\) with the hybrid genetic algorithm of GA + simplex method, we need to design an objective function and adopt the maximum coefficient method to obtain the minimum of \(f'(x, y)\):

\[
F(x, y) = C_{\text{max}} - f'(x, y).
\]  

(18)

Here \(C_{\text{max}}\) is the maximum coefficient. From (18), we see that when \(f'(x, y)\) is the minimum, \(F(x, y)\) reaches the maximum.

After the penalty term is added, all solutions in the 2-dimensional space can be regarded as feasible ones, and it can obviously avoid taking the local optimal solution as the global optimal solution, but the corresponding amount of computation is increased [17]. We adopt the principle of saving the best individual, which can shorten the operation time. And the new algorithm can improve the poor performance from adding the penalty function.

3.2.3. The Improved DV-Hop Algorithm Based on the GA + Simplex Method.

We assume that there are total \(N\) nodes in a WSN which contains \(M\) anchor nodes, and we want to locate the \((N - M)\) unknown nodes. During the positioning process, the unknown nodes obtained coordinates are converted into new “anchor nodes.” The distance between any two nodes cannot be more than the radius of the WSN distance. Here we assume that there are not isolated points in the WSN [8].

The procedure of the proposed hybrid algorithm of GA + simplex method is as follows.

**Step 1.** Initialize the number \(t\) of anchor nodes \(M\).

**Step 2.** From \(M_x\) \((M_x = N - t)\) unknown nodes, find the unknown node \(N_x\) with the largest number of neighboring anchor nodes.

**Step 3.** Randomly generate the initial population of \(n\) groups.

**Step 4.** Put the fitness function \(F(x, y)\) in the process of index calibration, and create a new fitness function \(e^{kF(x, y)}\). Use the new fitness function to calculate the fitness of all the members of the group.

**Step 5.** Arrange all individuals according to the fitness level, and adopt the principle of saving the best individual. For the top five individuals with the highest fitness values, go to Step 7, and for the rest of \(n - 5\) individuals that are named group \(p\), continue to the next step.

**Step 6.** Generate a new group \(P_i\) with the genetic algorithm.

1. Selection (copy): here we use the roulette method.
2. Crossing (restructuring): two-point crossover method is used.
3. Mutations: select one or more genes, and then do the corresponding change according to the mutation probability \(P_m\).
4. Update \(P_i\) with \(P\).

**Step 7.** Combine the top five individuals with the highest fitness from Step 5 with the group \(P_i\), with the individual number of \(n - 5\) from Step 6 and form a new group.

**Step 8.** Adopt the simplex method. Select individuals with a certain probability and perform the simplex method in the new group, and then replace the original individual with the new individual.

**Step 9.** If the number of iterations reaches the maximum, continue to Step 10. Otherwise, go back to Step 4.

**Step 10.** Obtain the position coordinates of anchor nodes \(N_x\) according to the last step. Let \(t \leftarrow t + 1\); when \(t < N\), go back to Step 2; when \(t = N\), all the unknown nodes in the WSN are positioned, and the algorithm ends.

In Step 4, the fitness function \(F(x, y)\) is dealing with indexation, constructing a new fitness function which can promote the competition but do not violate the fundamental principle of fitness function. In Step 5, we implement the principle of saving the best individuals. That is, the best individuals do not go through genetic operator process and directly go to the next step. As a result, operation time is reduced for less iteration and meets the goal of effectively iterative optimization. In Step 8, the simplex method is used as the local search operator of GA, so that you can take advantage of the strong local searching ability.

4. Experimental Results and Analysis

4.1. The Simulation Scenario. In the experiment, MATLAB was employed to perform the simulation of the weighted DV-Hop localization algorithm, the GA DV-Hop positioning algorithm [10], and the hybrid GA + simplex algorithm, respectively. In order to ensure the accuracy of simulation
results, before the start of each simulation, the network topology was constructed according to the parameters related to WSN connectivity and the density of anchor nodes. Anchor nodes are randomly generated and uniformly distributed in the region for better performance [1]. The experiment was repeated many times with the same parameters, and the average of results for each experiment was recorded. Only the isotropic WSN is discussed here.

Position performance is evaluated according to the average position error [16] and the network coverage rate.

The network coverage rate is the ratio of the unknown nodes that obtain the coordinates and the total unknown nodes in the network. It is one of the targets of location algorithm to achieve the precise positioning of unknown nodes as much as possible. In the case of meeting a certain positioning accuracy, the higher network coverage rate represents the better network performance.

The GA + simplex method needs to set the parameters, and we use the binary coding with the code length \( l = 14 \), population size \( M = 50 \), crossover probability \( p_c = 0.9 \), mutation probability \( p_m = 0.02 \), and the number of iterations \( t_m = 100 \). We use the Nelder-Mead simplex method in which the parameters need to be reset, including the reflection factor \( \alpha = 1 \), the expansion factor \( \beta = 0.5 \), and the shrinkage factor \( \gamma = 2 \). The simulation experiment would be discussed in the next section.

4.2. Results and Analysis

4.2.1. The Impact of Network Connectivity on the Positioning Accuracy. We arranged a total of 250 nodes in the region, and the number of anchor nodes was 10. Figure 3 shows the simulation diagram of the weighted DV-Hop algorithm, the GA DV-Hop algorithm, and the hybrid DV-Hop algorithm based on the GA + simplex method. From the figure, we can see that when the connectivity is less than 10, errors of three algorithms are reduced, and margins become large. The GA positioning algorithm performs better than the weighted DV-Hop positioning algorithm. The error of the hybrid of GA + simplex method is less than that of GA positioning algorithm. The simplex method can make up for some drawbacks of GA. When the connectivity is larger than 10, errors for all of the three algorithms are deteriorated. But the hybrid algorithm of GA + simplex method performs better than the other two. The reason is that when the network connectivity is large and over a certain value, hop-counts between nodes are inaccurate.

4.2.2. The Impact of the Number of Anchor Nodes on the Positioning Accuracy. There are 250 nodes in the region, and the anchor node ratio varies from 0.02 to 0.12. Figure 4 shows the impact of anchor node ratio on the three algorithms in the isotropic network. From the figure, we see that, with the increase of anchor node ratio, performances of the three positioning algorithms are all improved. This is because, with the increase of the anchor node ratio, the number of available positioning nodes is increased, and the knowing ability of the network is also increased. The hybrid algorithm of GA + simplex method performs better than the other two.

4.2.3. The Influence of Anchor Nodes on the Network Coverage. Figure 5 shows that, with the increase of the proportion of anchor nodes, the network coverage rates of the three positioning algorithms are increased. The positioning coverage of GA DV-Hop is larger than that of the weighted DV-Hop algorithm, and the hybrid algorithm of GA + simplex method performs significantly better than the GA DV-Hop. When anchor nodes proportion is small, the network coverage
rate increases by a big margin, and the coverage gaps of the three algorithms are large. As the proportion of anchor nodes increases to a certain degree, the network coverage rate increases more slowly and stably.

4.2.4. The Impact of Number of Nodes on the Positioning Accuracy. Figure 6 shows the simulation results with the number of nodes being 50–250. The figure shows that, for a certain proportion of anchor nodes, when the number of nodes increases, the average positioning errors of the three algorithms are reduced, but the accuracy of the new algorithm is better than the other two algorithms, especially when the number of nodes is relatively small.

4.2.5. Positioning Stability. The confidence interval of the average positioning error reflects the stability of an algorithm. The general formula is defined as follows:

$$\bar{x} - t_{a/2,n-1} \frac{s}{\sqrt{n}} \leq \mu \leq \bar{x} + t_{a/2,n-1} \frac{s}{\sqrt{n}}$$  \hspace{1cm} (19)$$

Here $n$ is the number of samples, $\bar{x}$ is the average positioning error of each algorithm in different node numbers.
4.2.6. Algorithm Convergence. From Figure 8, it can be found that $\varepsilon$ can influence the convergence of GA + simplex method. More concretely, when $\varepsilon < 0.5$, the speed of convergence becomes faster and the stability becomes better with the increase of $\varepsilon$. Conversely, when $\varepsilon > 0.5$, the speed slows down and stability becomes worse with the increase of $\varepsilon$. So, the value of $\varepsilon$ is set at 0.5 in the GA + simplex method.

4.2.7. Time Complexity. The GA + simplex method saves the best individuals when updating the population to reduce the total execution time. From Figure 9, the runtime increases along with population size but decreases after saving the best individuals. Therefore, the new algorithm can reduce the time complexity.

5. Conclusion

The genetic algorithm is suitable for correction of the weighted DV-Hop localization due to its own characteristics. This paper proposes a hybrid algorithm based on GA + simplex method to optimize the weighted DV-Hop localization in the third phase. Without large calculation, the algorithm preserves the advantage of the GA algorithm, while its simplex method makes up some inherent flaws of the GA. The experiment shows that the improved algorithm

$t_{\alpha/2n-1}$ is the value of $t$ distribution at $\alpha/2$ when the degree of freedom is $n - 1$. $s$ is the standard deviation.

Confidence intervals of the three algorithms in different node numbers are shown in Figure 7. From Figure 7, the GA + simplex method has the least positioning error and interval size. The GA + simplex method performs the best positioning performance with a minimum width of the confidence interval. So the new method is most stable in the three algorithms.
has higher positioning accuracy and network coverage in isotropic WSN than other localization algorithms, and it is suitable for WSN positioning.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

The authors are thankful for the financial support from the key disciplines construction of universities of Shanxi Province Foundation (80010302010053).

References

[1] X. Zhang, S. Zhou, and L. Chen, Wireless Sensor Network Technology and Its Military Application, National Defence Industry Press, 2010.

[2] B. Jing, J. Zhan, and Y. Sun, Intelligent Network Sensor and Wireless Sensor Networks, National Defence Industry Press, 2011.

[3] X. Xie and L. Cheng, “Localization algorithm for wireless sensor networks based on mobile anchor dynamic selection,” Transducer and Microsystem Technologies, vol. 30, no. 1, pp. 123–130, 2011.

[4] Y. Chai and K. Zhang, Artificial Intelligence, Mechanical Industry Press, 2012.

[5] Q. Zhang, Artificial Intelligence Technology and Application, China Petrochemical Press, 2011.

[6] L. Sun, J. Li, and Y. Chen, Wireless Sensor Network, Tsinghua University Press, 2005.

[7] G. Peng, Y. Gao, and L. Sun, “Study of localization schemes for wireless sensor networks,” Computer Engineering and Applications, vol. 35, no. 1, pp. 27–29, 2004.

[8] D. Ouyang, J. He, and H. Bai, “Constraint particle swarm optimization algorithm for wireless sensor networks localization,” Computer Science, vol. 38, no. 7, pp. 46–49, 2011.

[9] M. Li, W. Xiong, and L. Guo, “Improvement of DV-Hop localization based on artificial bee colony algorithm,” Computer Science, vol. 40, no. 1, pp. 33–36, 2013.

[10] K. Chen, L. Li, and J. Li, “Wireless sensor network nodes localization method of underground coal mines based on genetic algorithm,” Coal Engineering, no. 10, pp. 120–122, 2010.

[11] L. Deng, “Wireless sensor network based on genetic algorithm of localization algorithms,” Computer Simulation, vol. 28, no. 9, pp. 161–164, 2011.

[12] S. Zhao, M. Sun, and Y. Tang, “Gasa-based localisation algorithm for wireless sensor networks,” Computer Applications and Software, vol. 26, no. 10, pp. 189–192, 2009.

[13] Y. Fan, H. Peng, and C. Zhu, “A localization algorithm based on GASA and RSSI for wireless sensor network,” Journal of Xi’ an University, vol. 29, no. 6, pp. 51–54, 2010.

[14] A. Chehri, P. Fortier, and P. M. Tardif, “Geo-location with wireless sensor networks using non-linear optimization,” International Journal of Computer Science and Network Security, vol. 8, no. 1, pp. 145–154, 2008.

[15] S. Guo and R. Tan, “The application of genetic algorithm in wireless sensor network,” Sensors and Instrumentation, vol. 25, no. 1–4, pp. 125–127, 2009.

[16] Y. Hu and Y.-P. Wang, “New penalty model for constrained optimization problems,” Computer Science, vol. 36, no. 7, pp. 240–242, 2009.

[17] S. Yang, X. Fan, and S. Liu, “An improved algorithm of DV-Hop in wireless sensor networks,” Computer Measurement and Control, vol. 16, no. 9, pp. 1356–1357, 2008.
Submit your manuscripts at
http://www.hindawi.com