Available online www.jocpr.com

Journal of Chemical and Pharmaceutical Research, 2014, 6(5):1706-1713

Research Article

ISSN : 0975-7384
CODEN(USA) : JCPRC5

A particle swarm optimization algorithm for bayesian network structure learning based on chain model

Xing Wang[1*], Xuewu Zhao[1] and Yihua Lan[2]

[1]School of Software, Nanyang Normal University, Nanyang, China
[2]School of Computer and Information Technology, Nanyang Normal University, Nanyang, China

ABSTRACT

Bayesian network is one of the most effective tools for knowledge representation and inference under uncertainties, the structure learning of which is currently a hot topic for research. This thesis proposes a chain-model based particle swarm optimization algorithm for the learning of Bayesian network structure. This algorithm first defines and uses a rule chain model, which contains information about the causality among the Bayesian network nodes, to enhance the quality of the topological sequences searched. It further adds a dynamic weight coefficient to the optimization algorithm for the setting of position of particles to balance the global search with the local search, and to improve the searching capacity of the algorithm at large. Experiments conducted on the general ALARM data set has shown that the new algorithm proposed herein produces better solutions with enhanced rate of convergence compared with the existing Bayesian Network Structure Learning Algorithm Based on Conditional Independence Test and the Ant Colony Optimization (I-ACO-B).

Keywords: Knowledge representation and inference under uncertainties; Structure learning of Bayesian network; Topological sequence; Particle swarm optimization algorithm; Rule chain model

Introduction

Knowledge representation and inference under uncertainties has long been a hotspot of research in the area of artificial intelligence. Bayesian Network (BN) is one of the most important theoretical modeling to deal with uncertainties. It is an effective tool and technique that applies the probability, statistics and graph theory into complex areas for inference under uncertainties and data analysis. With merits such as high observability, clear-cut meaning, flexible inference and easy decision-making, BN has become an increasingly important and fascinating theoretical model for knowledge representation and inference under uncertainties. In recent years, the structure learning of Bayesian network has aroused extensive interest among scholars at home and abroad along with the development of searching techniques and rising of data mining. So far, many algorithms have been proposed for the structure learning of Bayesian network [1～9], as represented by the newly rising genetic algorithm[1,2,3,8], particle swarm algorithm, and ant colony algorithm, all of which have already been successfully applied into the areas of medical diagnosis, financial analysis, and biological engineering informatics etc.

With the advancement of computer technology and theoretical research, the research of structure learning of Bayesian network has also achieved rapid developments. Its learning methods can be mainly categorized into the following two types: (1) Learning by way of constraint satisfaction of random variables [10,11]: This method takes the structure learning process as a process of constant satisfaction of constraints. That is, for instance, in a given data set D, to find out the Bayesian network structure that is most consistent with the independence-dependency relation as contained in D by making use of the conditional independence among variables of the knowledge testing nodes of the information theory. This method is easy to realize. But the reliability of its testing results also becomes increasingly unreliable with its dramatically expanding volume of test data resulted from the increasing augmentation of the exponent numbers. Especially when the scale of the Bayesian network is relatively large, the precision of Bayesian network structure so learned is hard to guarantee. (2) Learning based on grade search [3,12,13]: This method, for a given data set, first defines a grading standard, and then uses it to judge the matching degree between the Bayesian network structure thus produced and a sample data thereof until the network with the highest grade is located. Thanks to the scientific grading system used, the precision of the Bayesian network structure so found can be guaranteed. But this method also produces many invalid searches in searching for the parent node, which is also a waste of time. Therefore, a new way of research is proposed for based on the basis of combination of the two methods mentioned above. So far, a number of methods of learning of the Bayesian network structure have emerged, for example, the learning method based on ant colony algorithm [6,7]. In addition, the K2 algorithm is also a classical algorithm for the learning of Bayesian network structure, but it also has the defect of being sensitive to the topological sequence.

The Particle Swarm Optimization Algorithm (PSO), as put forward by Dr. Kennedy and Dr. Eberhart in 1995, is a swarm intelligence algorithm based on the simulation of bird foraging behaviors. With only a few parameters, this algorithm can be easily realized and widely applied. Ever since its formation, it has been attractive to many scholars. Many domestic and overseas scholars have conducted researches on it and also have made improvements on it. Literature No [15] in the attached Reference presents a weighted particle swarm optimization algorithm that strengthens the importance of primary speed. This algorithm enhances the searching capacity of PSO. The results of experiments conducted for cases where the weight coefficient is a fixed value and cases where the weight coefficient is a decreasing linear function have shown that PSO performs better in the latter cases. This is because the decreasing linear function well embodies the varied characteristics of searching at different stages, thus can better balance the global search with the local search. Literature No [16] in the Reference introduces a new multi-step particle swarm optimization algorithm with updateable position for participles. The updating formula is divided into three steps so as to widen the search length and improve the searching efficiency. However, this literature did not consider the function of the dynamic weight coefficient to PSO with respect to searching.

In view of this, this thesis proposes a particle swarm optimization algorithm for Bayesian structure learning based on chain model (PSO-BN). First, it introduces a dynamic weight coefficient into the new multi-step particle swarm optimization algorithm with updateable position for participles to enhance the searching capacity of PSO. Then, it defines a rule chain model and uses it to evaluate the topological sequences of the Bayesian network structure to be learned. Finally, it learns the highest quality Bayesian network structure on the basis of the best topological sequence obtained from conditional independence tests and K2 algorithm. The validity of PSO-BN has already been proven in experiments conducted on the ALARM dataset.

2 Background

2.1 Bayesian Network

A Bayesian network can be represented by a 2-element tuple <G, Θ >, where G is a directed acyclic graph (DAG), and $\Theta = \{\theta_i\}$ is a parameter set of probability distribution of local conditions of the network. G is the graphical representation of the conditional independence relations in the modeling. It indicates the structural property of nodes in BN. As a random variable is abstracted as a node of a question in modeling, so the directed arc between nodes in Grapy G represents the direct dependency relationship between related variables. Θ represents the conditional probability distribution of node X_i in the given parent node set. It illustrates the quantity property of the distribution of the random variable set under such conditional independence relations. Thus, by using its network structure and network parameters, a complete Bayesian network <G, Θ > uniquely specifies a joint distribution on the given random variable set X $= \{X_1, X_2, \cdots, X_n\}$ as follows:

$$P(X_1, X_2, \cdots, X_n) = \prod_{i=1}^{n} P(X_i \mid \prod (X_i)) \quad (1)$$

where $\prod(X_i)$ represents the parent node set of X_i.

Informally speaking, Bayesian network structure learning is a process like this: for a given sample dataset D, search and find out a DAG G which best matches the sample D. This graph G indirectly represents parts of the conditional independence between node variables.

2.2 Particle Swarm Optimization Algorithm

Particle swarm optimization algorithm is a search algorithm based on swarm intelligence. Its core concept is to use the information sharing mechanism of colonial organism to intelligently work out the solution to a question. In PSO, a particle can be formally represented as a 2-element tuple <x,v>, where x means the position of the particle, and v the speed of the particle. Each particle is updated via two values: one is the optimal solution the particle itself finds called "individual extremum", which is also called "personal experience"; the other is the optimal solution the swam as a whole finds called "global extremum", which is also named "social experience". The iteration formula of PSO is as follows:

$$v_i^{(k+1)} = v_i^{(k)} + c_1 \cdot r_1 \cdot (pbest_i^k - x_i^k) + c_2 \cdot r_2 \cdot (gbest_i^k - x_i^k) \quad (2)$$

$$x_i^{(k+1)} = x_i^k + v_i^{(k+1)} \quad (3)$$

where c_1, c_2 are both positive numbers, which respectively represent the self -learning factor and social learning factor. They decide the effect of the personal and social experience of a particle on the motion trajectory of the particle, and reflect the information exchange between particles. r_1, r_2 are two random numbers valued with the range [0,1]. $pbest_i^k$ denotes the optimal position of participle i at generation k, e.g. the "individual extremum". $gbest_i^k$ represents the optimal position of the whole participle swam at generation k, namely, the "global extremum". x_i^k means the position of participle i at generation k . v_i^k stands for the speed of participle i at generation k . And, $x_i^{(k+1)}$ is the position the particle i "flied" to at the speed of $v_i^{(k+1)}$.

Based on the concept of particle swarm optimization, Literature [16] of the Reference proposes a new multi-step particle swarm optimization algorithm with updateable position for participles in combination with the above formula (2) and (3). In the new algorithm, the updating formula is split. For the generation of i , the operation is as follows:

$$v_1^{(k+1)} = v_i^k \quad (4)$$

$$x_1^{(k+1)} = x_i^k + v_1^{(k+1)} \quad (5)$$

$$v_2^{(k+1)} = v_1^{(k+1)} + c_1 \cdot r_1 \cdot (pbest_i^k - x_i^{(k)}) \quad (6)$$

$$x_2^{(k+1)} = x_1^{(k+1)} + c_1 \cdot r_1 \cdot (pbest_i^k - x_i^{(k)}) \quad (7)$$

$$v_3^{(k+1)} = v_2^{(k+1)} + c_2 \cdot r_2 \cdot (gbest_i^{(k+1)} - x_i^{(k)}) \quad (8)$$

$$x_3^{(k+1)} = x_2^{(k+1)} + c_2 \cdot r_2 \cdot (gbest_i^{(k)} - x_i^{(k)}) \quad (9)$$

In the above steps, every subsequent step is "flied" to from the preceding step. The advantage of this method is that both the intermediate position and the last position are considered. In the process of each iteration, every participle generates three positions at the aforementioned three speeds, out of which the optimal position and its corresponding speed are chosen. By this method, the overall performance of the algorithm can be greatly improved as a result of its refined global search, increased search intensity, and accelerated convergence rate.

2.3 Topological Sequence

Topological sequencing is a process of working out the total order of a set via its partial order, the result of which is a linear sequence called topological sequence. A topological sequence determines the order of nodes and represents part of their inter-relations. For example, a project consists of four sub-activities as ABCD. The order of precedence of these activities is illustrated in Figure 1, a directed acyclic graph. Its topological sequence is ABCD or ACBD. These two sequences contain an order of precedence as this：A starts ahead of all the other activities, and D is the last one to complete; B and C finish between A and D, and they two do not necessarily commence in chronological order. This illustrates part of the order relation of the four activities.

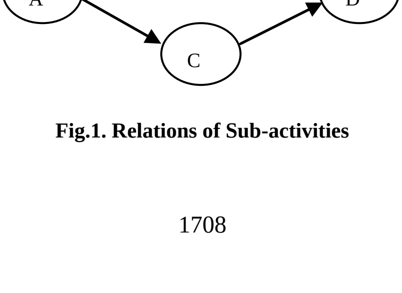
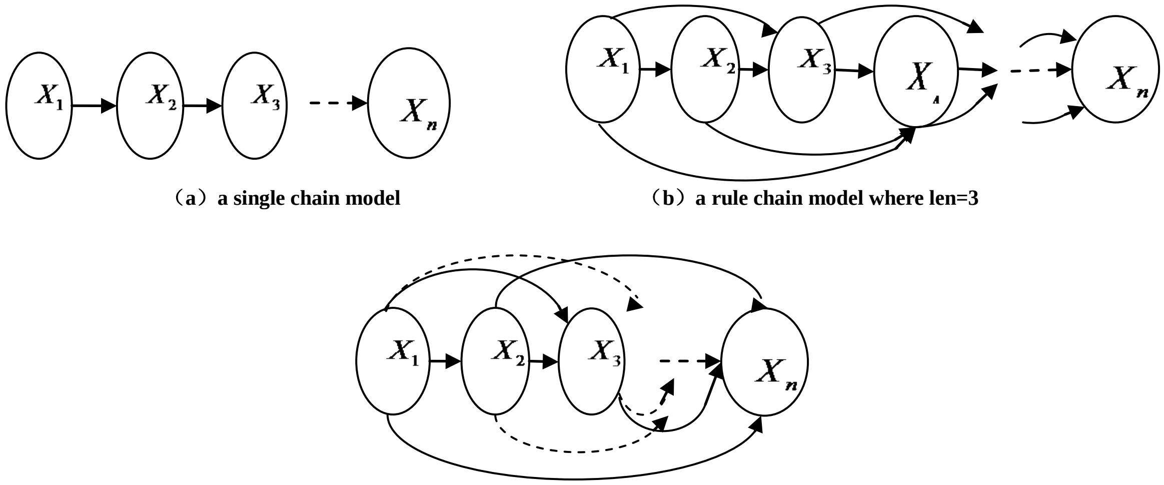


Fig.1. Relations of Sub-activities

3 Particle Swarm Optimization Algorithm for Bayesian Structure Learning Based on Chain Model

3.1 Chain Model Based on Topological Sequence

A directed acyclic graph can be topologically ordered. The result of such ordering is called a topological sequence of the graph. Since a Bayesian network is structurally a directed acyclic graph, it can be easily topologically ordered. This means that we can learn the structure of a Bayesian network by using a topological sequence of higher quality, if we can find one. For this, we define a chain model and a rule chain model as shown in Figure 2.



（a）a single chain model **（b）a rule chain model where len=3**

(c) a complete rule chain model

Fig.2. chain model

Definition 1: Within a topological sequence, a parent node can only come from its preceding nodes. Such paternity is represented by a directed arc. Except for the first node of the sequence, the parent node set of no other nodes is a nonempty set. A chain structure satisfying the above constraints is called a chain model.

Definition 2: Within a chain model, the number of parent nodes of a node is called the search length of the node.

Definition 3: For a chain model, the search length of the node with the longest search length is called the search length of the chain model (len).

Definition 4: In a chain model with a search length of len, where the number of nodes preceding a node is larger than len, the parent node of the node is the len node immediately preceding the node; where the number of nodes preceding a node is less than or equates len, all the nodes preceding the node are its parent nodes. A chain model meeting the above conditions is called a rule chain model.

Especially, for a rule chain model, where len=1, it is called a single chain model as shown by Figure 2(a); where len=n-1, it is called a complete rule chain model as represented by Figure 2(c); where 1<len<n-1, it is called a partial rule chain model as illustrated by Figure 2(b) where len=3.

According to the hypothesis proposed by Literature [7] in the Reference, chain model can well ascertain the topological sequence of a Bayesian network to be learned in that the grade of the Bayesian network is correspondingly predominated by the topological sequence. Therefore, the merits of a topological sequence can be judged through its corresponding rule chain model. Actually, a chain model is a transitional model from topological sequence to Bayesian network with important information of the Bayesian network, which can generate sequences on topology. In PSO-BN, for a given dataset D, it first obtains the topological sequences of a better Bayesian network through rule chain model learning. With much more information, the topological sequences so obtained are closer to reality. To a certain extent, the grade of len reflects the complexity of the network. Therefore, we can search, via len, the topological sequence that more genuinely represents the Bayesian network structure that actually exists on a finer participle level.

3.2 Improvement on Weight Coefficient of Particle Swarm Optimization Algorithm

In particle swarm optimization algorithm, w, the inertia coefficient, functions as a balance between the global search and local search. Therefore, the design of the inertia coefficient has an important impact on the performance of the search. Considering that the search algorithm has different characteristics at varied search stages, this thesis uses a dynamic inertia coefficient as shown in Formula (10).

$$w = w_{\max} - k \times \frac{w_{\max} - w_{\min}}{it_{\max}} \quad (10)$$

where $w_{\max}$ is the maximum value of w, and $w_{\min}$ the minimum value. $it_{\max}$ represents the maximum iteration, while k the current iteration. It can be seen from the formula above that, at the early stage of the iteration, with the value of k being relatively small and the value of w relatively large, the algorithm focuses on the global search and has a comparatively strong "searching" capacity; and that, at the late stage of the iteration, , with the value of k increasing upwards and the value of w decreasing downwards, the algorithm focuses on the local search and has a comparatively strong "using" capacity. Because such design and algorithm are in correspondence with the characteristics of the searching at different stages, the overall search performance is enhanced.

Given that the searches at varied search stages have different characteristics, a dynamic weight coefficient complies more appropriately with the algorithm and actual circumstances. Therefore, this paper adds a dynamic weight to the Formula (5) upon integrating the dynamic weight coefficient with Literature [16]. As a result, we have a dynamic weight coefficient to guarantee that the search focuses on global search at the early stage and on local search at the late stage; meanwhile, thanks to the updateable position of the participles, the global search is also refined with increased search intensity, accelerated convergence rate, and varied solutions to avoid local optimum.

3.3 PSO Coding for Bayesian Structure Learning Based on Rule Chian Model

As PSO-BN is to use the particle swarm optimization algorithm where the position of participles with dynamic weight can be updated to search on the discrete topological sequence based on rule chain model, it needs to be re-coded. In accordance with the PSO concept and characteristics of topological sequence, we design a coding scheme as follows:

Particle: in particle swarm optimization algorithm, the position of each particle represents a solution to the question. As PSO-BN uses an optimized PSO to search the optimal topological sequence, the position of each participle means a topological sequence. To put it another way, an ordering arrangement of a topological sequence is called a "position" of a corresponding participle. For instance, for a particle $p_i = (x_i, v_i)$, $x_i = BDCEAFG$.

Interchange operator: for a topological sequence with (i, j) regarded as its interchange operator, it means that the node at position i and the node at position j exchange their positions as follows:

$$BDCEAFG \xrightarrow{(2,4)} BECDAFG$$

Speed: speed is a set of interchange operators used to update position of participles. For example: $\{(1,3),(4,7)\}$

Sum of speeds: the sum of one speed with another is still a speed, and since speed is a set of interchange operators, so, from the perspective of a set, the sum of one speed with another is a union of different sets. Such as:

$$\{(1,3),(2,5)\} + \{(2,5),(4,7)\} = \{(1,3),(2,5),(4,7)\}$$

Subtraction of positions: the subtraction of one position from another produces speed, while speed is the set of interchange operators, so the subtraction result is also a set of interchange operators. Subtraction of positions is to compare the nodes at relevant positions, so, if the nodes are the same, no interchange operator will be produced, otherwise, an interchange operator will be generated. Such a so generated interchange operator, when used against the subtrahend participle, can change the node at the position to the same node of the minuend participle at its corresponding position, such as $BEDFAG - DBEFAG = \{(1,2),(2,3)\}$. The speed so produced reflects the difference of positions (ordering arrangements) of the subtrahend participle and the minuend participle.

Scalar speed: where 0<a<1, $a \cdot v$ means a speed, and$\|a \cdot v\|$ means the largest integer less than a$\|v\|$（$\|v\|$ means the cardinal number of v , and v speed）. Where a>1, $\|a \cdot v\|$I means the largest integer less than (2-a)$\|v\|$.

4 Experiment Result and Analysis

To test the performance of PSO-BN, We use the well known Benchmark ALARM dataset (dataset sourced from http://www.cs.huji.ac.il/labs/compbil/Repository/) to conduct the following experiment, and then compare the results with that of I-ACO-B for further analysis. The experiment is operated in Windows XP2 with a Pentium4 CPU and an EMS memory of 512MB. JAVA is used to realize the algorithm. The parameters used by the algorithm are listed in Table 1.

Table 1 Parameters of PSO-BN

NC	c1	c2	maxintia	minintia	p1	p2	p3
100	2.0	2.0	0.9	0.2	7	13	0.5

4.1 Impact of Search Length on PSO-BN

As PSO-BN uses the rule chain model to measure the quality of topological sequences, so the search length within the rule chain model will exert an important impact on the search performance of the algorithm. We have conducted experiments for different search lengths. Table 2 shows the experiments results of PSO-BN with data capacity of 3000 for cases of len of different values respectively running for 10 times. K2 means the K2 grade, and It. represents the iteration times.

Table 2 Comparison of results where len takes different values

len	K2	It.
1	-14391.78±0.34	64.53±5.94
9	-14390.63±0.39	46.67±7.94
18	-14389.45±0.41	34.90±6.26
24	-14390.21±0.27	31.69±6.45
27	-14390.57±0.73	24.30±6.57
36	-14391.63±1.13	13.16±10.98

It can been seen from Table 2, that where len takes a large value, this algorithm is prone to a local optimum and the grading and convergence rate fluctuate wildly; and that where len takes a small value, there are more iteration times, and the convergence rate is slower, and the grading fluctuates narrowly. Moreover, the selection of len is greatly related to the complexity of the Bayesian network structure to be learnt.

4.2 Comparison of Experiments Results of the Two Algorithms

To test the overall performance of PSO-BN, experiments of I-ACO-B and PSO-BN are conducted on the ALARM with different data capacity. Table 3 shows the results of 10-times respective running of the two algorithms. K2 is the grade of the Bayesian structure learned from the running of the two algorithms and the grade given in the parentheses is the optimal grade of such running. S. abbreviated from Statistic, indicates the parameters for comparison. Ad. represents the number of additional arcs, De is the number of arcs that are left out accidentally, and In. is the number of wrongly directed arcs. It. stands for the minimum times of iterations needed to get the optimal solution. It is represented as $\mu \pm \sigma$ in this paper.

It can be seen from Table 3 that the new algorithm as proposed by this thesis excels the I-ACO-B over both the convergence rate and the quality of solution. This is because the optimized particle swarm algorithm adapts updateable positions for particles with refined global search, widened search range, and accelerated convergence rate. The topological sequences gained from this new algorithm are closer to reality with much more information and much less wrongly directed arcs. The dynamic weight coefficient well reflects the different characteristics of the search at different stages, thus improves the search capacity of the algorithm.

Table 3 comparison of experiments results of two algorithms in ALARM

Sample Capacity	S.	Algorithm	
		I-ACO-B[9]	PSO-BN
5000	K2	-23782.61.±0.42 (-23782.02)	-23782..23±0.57 (-23781.86)
	Ad.	1.7±0.13(2)	1.7±0.11(2)
	De.	1.0±0.00(1)	1.0±0.00(1)
	In.	1.8±0.13(2)	1.3±0.09(1)
	It.	58.00±5.49(51)	45.00±5.26(37)
6000	K2	-28347.69.±0.31 (-28347.11)	-28346.68±0.57 (-28345.97)
	Ad.	1.8±0.27(2)	1.6±0.29(2)
	De.	1.0±0.00(1)	1.0±0.00(1)
	In.	1.9±0.16(2)	1.5±0.42(1)
	It.	52.00±7.01(48)	43.00±6.79(35)

4.3 Comparison of the Convergence Algebras of the Two Algorithms

To verify and compare the convergence performance of the I-ACO-B and PSO-BN algorithms, we compared the average iteration times needed for each algorithm to obtain their respective optimal grade in 10-times independent running on the basis of 10 data capacities. The results are shown in Figure 3, in which the abscissa stands for the capacity of the data set, and the ordinate the iteration times.

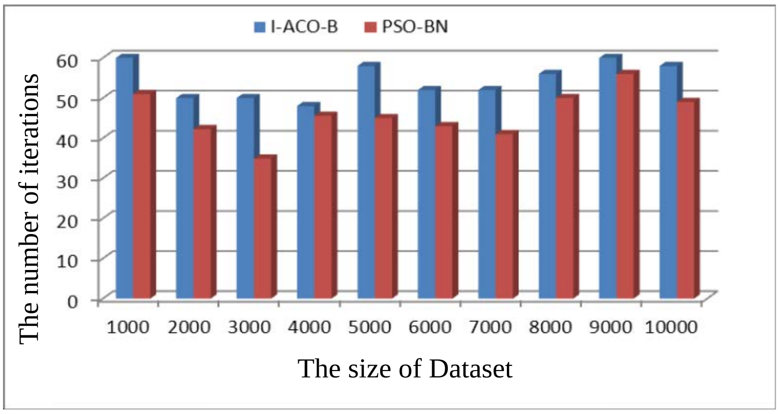


Fig.3 comparison of convergence algebra of the two algorithms

Fig.3 shows that the iteration times of convergence has no particular relationship to the size of data capacity. This proves that the ant colony algorithm, the particle swarm algorithm, and the genetic algorithm are random in nature, and that the iteration times of PSO-BN is better than that of I-ACO-B within the whole sample data set. Again, this is because PSO-BN uses a dynamic weight coefficient that well reflects the different characteristics of the search at different stages, and uses the updateable positions for particles that refines the global search and widens the search range. As a result, the search capacity of the algorithm is improved with accelerated convergence rate.

CONCLUSION

This thesis proposes a new Bayesian structure learning algorithm based on topological sequences, an algorithm that uses the rule chain model that can better reflect the causal relationship among nodes to measure the quality of the topological sequences. Through the algorithm, the topological sequences so learned are of a higher quality and the quality of solution to the Bayesian network is so improved. Furthermore, by adding a dynamic weight coefficient to the particle swarm optimization algorithm with updateable position for participles, the searching capacity of the algorithm is also enhanced with accelerated convergence rate. The experiments show that PSO-BN is better than I-ACO-B over both the precision of solution and the convergence rate. For further study, we will use this algorithm to explore Bayesian structure learning in more complex circumstances, such as where there is no sufficient data or there are hidden variables.

Acknowledgments
This paper is supported by Henan Research plan for Basic and frontier Research (No. 132300410439, 142300410183, 142300410182, 132300410438).

REFERENCES

[1]Larranaga P, Poza M, Yurramindi Y, et al. *IEEE Trans on Pattern Analysis and Machine Intelligence*,**1996**,18(9):912-925.

[2]Larranage P,Poza M,Uurramendi Y,et al. *IEEE Trans on Pattern Analysis and Machine Intelligence*,**1996**,18(9):912-925.

[3]J.cowie, L.Oteniya, R.Coles. Particle swarm Optimization for learning Bayesian Networks. proceedings of the world Congress on Engineering 2007 Vol.1,WCE,**2007** July,2-4, London,U.K.

[4]R.Kabli, F.Hermann, J.McCall. A Chain-Model Genetic Algorithm for Bayesian Network Structure Learning Proceedings of the 9th annual conference on Genetic and evolutionary computation. In IEEE world congress on computational intelligence **2007**, London, England,july,07-11,1264-1271.

[5] Luis M., Campos D., Huete J. *International Journal of Approximate reasoning*, *2000*, 24(1): 11-37.

[6] J. Cheng, R. Greiner, J. Kelly, et al. *Artificial Intelligence*, **2002**, 137: 43-90.

[7] M. L. Wong, K. S. Leung. *IEEE Trans on Pattern Analysis and Machine Intelligence*, **1999**, 21 (2) :174-178.

[8] Jose. A. G, Jose.M. P. *Pattern Recognition Letters*, **2002**, 23 (1–3): 261–277.

[9]Eberhart R C,Kennedy J.A new optimizer using paritcle swarm theory[A].Proceedings of the Sixth Internatioal Symposium on Micro Machine and Human Science[C].Nagoya,Japan,**1995**: 39-43.

[10] Shi Y, Eberhart R C. Parameter selection in particle swarm optimization[A]. Evolutionary Programming Ⅶ,

Lecture Notes in Computer Science[C]. Berlin: Springer, **1998**.591-600.