The Black Hole Clustering Algorithm Based on Membrane Computing

Qian Li[1, a], Zheng Pei[2, b]

[1]*Center for Radio Administration & Technology Development, Xihua University,*

Chengdu, Sichuan, 610039, China

[2]*Center for Radio Administration & Technology Development, Xihua University,*

Chengdu, Sichuan, 610039, China

[a]*li_qian111@163.com,* [b]*pqyz@263.net*

Abstract.

Over the last few decades, nature has stimulated many successful heuristic optimization methods and computational tools for dealing with optimization problems. In this paper, we investigate the black hole clustering algorithm based on membrane computing. Formally, we embed the black hole clustering algorithm in the membrane structure, provide the three parts of the membrane system to perform black hole clustering algorithm, which are the membrane structure based on membrane computing, the objects in elementary membrane and the evolution rules in the elementary membrane, and analyze the performance of the black hole clustering algorithm based on membrane computing. The proposed algorithm in the paper is compared with some other classical algorithms, and the simulation results indicate the algorithm in the paper can achieve better performance in clustering.

Keywords: Heuristic algorithm; Black hole algorithm; Membrane computing

Introduction

In recent decades, various heuristic algorithms have been developed [1] [2]. Most of the algorithms are inspired by the behaviors of natural phenomena [3]. For example, genetic algorithms (GAs) are a kind of searching optimal solution which was inspired by the process of the natural evolution, using natural genetic variation and natural selection operators [4]. The simulated annealing (SA) is a stochastic optimization algorithm developed by modeling the steel annealing process [5]. The ant colony optimization (ACO) was inspired from the behavior of the ant colony, which is able to find the shortest path to the food source [6]. The particle swarm optimization (PSO) was inspired from the swarm behavior,

such as fish and bird schooling in nature [7]. The gravitational search algorithm (GSA) was developed based on the Newtonian gravity and the laws of motion [8]. The black hole algorithm (BH) was proposed as a new heuristic optimization approach for data clustering inspired from the black hole phenomena.

In this paper, the black hole algorithm based on membrane computing is proposed. Membrane computing was proposed by G. Păun inspired from the structure of the biological cells, aimed at abstracting a calculation model from the way of treating compound in the hierarchy of the cells [9]. Actually, there has been a growing interest in computer model abstracted from the viewpoint of biology [10] in the past few decades. For instance, swarm intelligence algorithms are evolutionary computation technique abstracted from biological communities [7]. The artificial life inspired from the individual life is a simulation of the life system by artificial [11]. As the optimization approach, the immune algorithms were developed based on immune system [12]. The Artificial neural network is a kind of mathematical model which was inspired from the neural network of the human body [13].

Membrane computing has been a new branch of natural computation and received extensive attention of many scholars since it was put forward. So far, there have been many papers expound the membrane computing from various subjects, such as computer science [14], biology [15], linguistics [16], automatic control [17], etc, and applied in many fields including computer graphics [11], approximate optimization [18], economics [19], etc.

The rest of the paper is organized as follows. Section 2 provides a brief review of the black hole algorithm and the membrane computing. In Section 3, the black hole based on membrane computing is described. The experimental results are demonstrated by comparing with other algorithms in Section 4. Finally, the summary and the conclusion are presented in Section 5.

Preliminaries

In this section, we briefly review the black hole algorithm and the membrane computing.

Firstly, we briefly describe the origin, process and the performance of the black hole algorithm, respectively.

The black hole algorithm (BH) proposed in [20] is a new optimization method for data clustering inspired by the black hole phenomenon in the universe. Actually, John Michell, the British natural philosopher, has put forward that there is a kind of object that runs faster than the light in the eighteens-century. However, this kind of object was not known as a black hole during that period and it was only in 1967 that John Wheeler first named the phenomenon of mass collapsing as the black hole. As we all known, black hole in universe is what forms when a star of massive size collapses. The gravitational power of the black hole is so high that there is no way for the nearby objects to escape from its gravitational pull. Even for the light, once it falls into a black hole, it will also be swallowed and gone forever.

Similar to other population-based algorithms, the black hole algorithm starts with an initial population of candidate (called star in [20]) solutions and an objective function that is calculated for them. In every iteration of BH, the best candidate is selected to be the black hole according to the objective function. Then the black hole starts pulling other stars around it. In other words, all the stars start moving towards the black hole. If a star gets too close to the black hole, it will be swallowed by the black hole and gone forever and a new star will be randomly generated and placed in the search space and starts a new search. Briefly, the BH algorithm can be summarized as the following steps:

a) Initialize a population of stars randomly in the search space.
b) For each star, evaluate the objective function and select the best star which has the best fitness value as the black hole.
c) Change the location of each star.
d) If a star reaches a location with lower cost than the black hole, exchange their locations.
e) If a star crosses the event horizon of the black hole, replace it with a new star in a random location in the search space.
f) Repeat steps c) to e) until the stop criteria is reached.
g) End

From the process of the BH algorithm, we can see that it has a simple structure and it is easy to implement. The results in [20] show that the BH algorithm has the greater advantages than other algorithms in solving clustering problems.

Secondly, the development and the main structure of the membrane system are discussed below.

Membrane computing is also known as P system. It is indicated from [21] that P system consists of three parts: membrane structure, multisets of objects and evolution rules. The membrane structure is a hierarchical rooted tree of compartments that delimit regions, where the root is called skin. Multisets of objects are chemical substances present inside the compartments (membranes) of a cell and the evolution rules are chemical reactions that can take place inside the cell.

In the P system, the multisets of objects are placed in compartments surrounded by membranes, and evolved by some given rules. In this way, every compartment can exchange information with other compartments according to these evolution rules. P system is a model with active membrane, and the model is formed as follows:

$$\prod = (O, \mu, \omega_1, \omega_2, \cdots, \omega_q, R)$$

(1)

where $q \geq 1$ is the initial degree of the P system. O is the alphabet of objects. μ represents the membrane structure (a rooted tree). $\omega_1, \omega_2, \cdots, \omega_q$ are strings over O, describing the multisets of objects placed in the q regions of μ, and R is a finite set of rules.

Generally, there are three main types of P system: Cell-like P systems, Tissue-like P systems and Neural-like P systems [22]. Cell-like P systems are developed mainly according to the life cycle of biological cells and intercellular substance exchange [23]. Tissue-like P systems are important expansion of cell membrane system which can exchange information between various cells [24]. Neural-like P systems are another type of P systems which were inspired by the biological neural system and are also the current hot topics in the study of membrane computing [25].

To sum up, P systems are a class of distributed parallel computing models which have synchronous and non-deterministic properties and maximally parallel computing ability [13]. Because of the parallel character, not only NP complete problems can be solved in linear time in an easy way by P systems with active membranes [26], even the limitations of Turing machine can be exceeded by the simple models of the P systems [15].

The black hole algorithm based on MC (BH-MC)

Although the black hole algorithm mentioned above outperforms other traditional heuristic algorithms described in [20], the rate of convergence in it is so slow that we should use more iterations to achieve the effect. In order to improve the algorithm performance, we introduce the membrane computing into the black hole algorithm (BH-MC, for short).

As mentioned above, the membrane system consists of three parts: membrane structure, multisets of objects and the evolution rules. Combined with the model described by Eq.(1), we give the model of membrane system which has been concretized in BH-MC as follows:

$$\prod_{BH} = (O_{BH}, \mu_{BH}, \omega_{BH}^1, \omega_{BH}^2, \cdots, \omega_{BH}^i, \omega_{BH}^q, R_{BH})$$

(2)

where $q \geq 1$ is the initial degree of the P system in BH-MC. O_{BH} is the alphabet of objects and μ_{BH} represents the membrane structure in BH-MC. ω_{BH}^i is the finite set of O_{BH} and R_{BH} is the rule in BH-MC algorithm.

The three parts of the membrane system combined with the BH algorithm will be introduced respectively as follows. The first is the membrane structure of the BH-MC. We adopt the Cell-like P system with a two-layer membrane structure which consists of a skin membrane and q elementary membrane in BH-MC. Fig.1 shows the membrane structure of the BH-MC briefly. As shown in Fig.1, $1, 2, \cdots, q$ are elementary membranes and each of them contains one or more objects and some evolution rules. In the designed cell-like P system, each elementary membrane contains n objects and evolution rules while the skin membrane without any evolution rules contains only one object which represents the global best object.

The second is the objects in elementary membrane. Like other population-based methods, we generate the objects randomly in every

elementary membrane. The multiset of the objects in each elementary membrane of the form

$$\omega_i = \{O_1, O_2, \cdots, O_n\}$$

$$(3)$$

where $i = 1, 2, \cdots, q$. ω_i is the set of the objects in the i th elementary membrane and n is the number of the objects in elementary membrane. Actually, each object of elementary membrane represents the cluster centers of the candidate. Assume that $D = \{x_i \in R^d \mid i = 1, 2, \cdots, m\}$ is the dataset to be clustered, which is divided into k clusters. $z_1, z_2, \cdots, z_k$ are the k clusters centers respectively, where $z_i \in R^d$ $(i = 1, 2, \cdots, k)$. Thus, the object is an $(k \times d)$ -dimensional vector of the form

$$O_j = \{(z_{11}, z_{12}, \cdots, z_{1d}), \cdots, (z_{i1}, z_{i2}, \cdots, z_{id}), \cdots, (z_{k1}, z_{k2}, \cdots, z_{kd})\}$$

$$(4)$$

where O_j is the j th object in elementary membrane and $(z_{i1}, z_{i2}, \cdots, z_{id})$ is the i th cluster center z_i .

By the way, the objects in different elementary membrane may be exactly the same in theoretically because of the randomness in the process of objects generation. Therefore, we adopt the following way in order to solve this problem:

$$\left| \omega_1 \cap \omega_2 \cap \cdots \cap \omega_i \cdots \cap \omega_q \right| \leq \delta$$

$$(5)$$

where ω_i is the set of objects in the i th elementary membrane. δ is the threshold in the interval [0, 1]. In this way, we can say that the objects in each elementary membrane are not exactly the same and the diversity of the objects is increased by it to a certain extent.

The third is the evolution rules in the elementary membrane. There are three evolution rules in BH-MC, which consists of selection, move and absorb rules. The three evolution rules are described as follows:

i. Selection rule. For each of the n objects (called stars) in elementary membrane, evaluate the fitness values and select the best star which has the best fitness value as the black hole. In the BH-MC, the fitness values of the objects are assessed by the following equation:

$$F(x, z) = \sum_{j=1}^{k} \sum_{i=1}^{m} w_{ij} \left\| x_i - z_j \right\|$$

<div align="right">(6)</div>

where $\left\| x_i - z_j \right\|$ is the Euclidean distance between data point x_i and the cluster center z_j. m is the number of the data point in D and k is the number of the clusters. $w_{i,j}$ is the association weight of data point x_i with cluster j , which will be either 1 or 0 (if data point x_i is assigned to cluster j : $w_{i,j}$ is 1, otherwise 0).

ii. Move rule. After a star was selected as the black hole by the selection rule, the rest stars start moving towards the black hole according to the following equation:

$$y_i(t+1) = y_i(t) + rand * (y_{BH} - y_i(t))$$

<div align="right">(7)</div>

where $i = 1, 2, \cdots, n$. $y_i(t)$ and $y_i(t+1)$ are the locations of the i th star at iterations t and $t+1$. y_{BH} is the location of the black hole in the search space. *rand* is a random number in the interval [0, 1]. While moving towards the black hole, a star may reach a location with better effect on the data clustering than the black hole. In such a case, the star will replace the black hole and become the new black hole for the next iteration.

iii. Absorb rule. During the process of moving towards the black hole, some stars may cross the event horizon of the black hole. In this situation, these stars will be sucked by the black hole and the same number of the stars will be generated randomly at the same time to keep the number of candidate solutions constant. The radius of the event horizon is formulated as follows:

$$R = \frac{f_{BH}}{\sum_{i=1}^{n} f_i}$$

<div align="right">(8)</div>

where f_{BH} is the fitness value of the black hole and f_i is the fitness value of the i th star. n is the number of stars. If the distance between a star and the black hole is less than R, that candidate will be sucked by the black hole and a new candidate is created randomly in the search space.
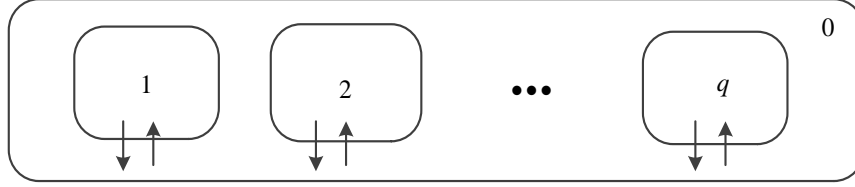
Fig.1.The membrane structure of the BH-MC algorithm

Based on the above description, the main steps of the proposed algorithm are shown as follows:

    a)   Determine the number of the elementary membrane.

    b)   For each elementary membrane, initialize the population of stars and select the best star which has the best fitness value as the black hole.

    c)   Change the location of each star according to Eq.(7) in every elementary membrane.

    d)   For each elementary membrane, evaluate the fitness of all stars again. If a star has the better fitness than the black hole, exchange their locations.

    e)   In every elementary membrane, if a star crosses the event horizon of the black hole, it will be swallowed by the black hole and replaced by a new star in the search space.

    f)   Repeat steps c) to e) until the stop criteria is reached.

    g)   End.

Test results

In this paper, we use two datasets which are Iris and Wine to evaluate the performance of the proposed approach. The evaluation criterion of the algorithm is the sum of intra-cluster distance called M value in this paper, which was defined in Eq. (6). Clearly, the smaller the M value, the higher the quality of the clustering.

On one hand, the performance of the BH-MC algorithm is compared against some classical algorithms, including K-means, PSO, GSA and BH. On the other hand, 2, 4 and 8 elementary membranes are used respectively in the membrane structure of the BH-MC to compare with BH algorithm. Theoretically, the bigger the number of the elementary membrane which means the more the diversity of the objects in our algorithm, the better the effect will be.

We describe the process of the BH-MC algorithm with 2 elementary membranes in details again, taking the Iris for example. The process of the BH-MC algorithm is described as follows:

Firstly, Iris has 150 data points which composed of 3 clusters and each data point has 4 features. Therefore, $m = 150$ and $k = 3$ in Eq. (6) for Iris. Assume that we set the number of the objects (stars) in each elementary membrane is 100. According to Eq. (3), the multisets of objects in elementary membrane can be described as follows:

$$\omega_1 = \{O_1, O_2 \cdots, O_{100}\}$$

$$\omega_2 = \{O_1^{'}, O_2^{'}, \cdots, O_{100}^{'}\} \tag{9}$$

$$\tag{10}$$

where ω_1 is the set of the objects in the first elementary membrane and ω_2 is the set of the objects in the second elementary membrane. O_j represents the object in the set according to Eq. (4) of the form

$$O_j = \{ z_{11}, z_{12}, z_{13}, z_{14}, z_{21}, z_{22}, z_{23}, z_{24}, z_{31}, z_{32}, z_{33}, z_{34} \}$$

$$\tag{11}$$

where $j = 1, 2, \cdots, 100$. O_j is a (3×4)-dimensional vector and $(z_{11}, z_{12}, z_{13}, z_{14})$ is the first cluster center z_1.

Secondly, calculate the M value of these objects in elementary membrane respectively according to Eq. (6). In this step, we need to select the best star as the black hole which has the minimum value of the 100 M values in each elementary membrane. We might assume that O_i is selected as the black hole in the first elementary membrane and $O_j^{'}$ is selected as the black hole in the second elementary membrane.

Thirdly, change the location of stars in each elementary membrane according to Eq. (7). The rest of stars in the first elementary membrane will move towards O_i and stars in the second elementary membrane will move towards $O_j^{'}$.

Fourthly, calculate the M value again for all the stars in each elementary membrane and select the best star which has the best fitness as the new black hole. In this step, we suppose O_p and O_q are selected in the 2 elementary membranes respectively and then we save the better one between of the two objects as the global best object. Then, the global best object will replace the old black hole in each elementary membrane as the new black hole in the next iteration.

Finally, calculate the radius of the event horizon in each elementary membrane according to Eq. (8).

Similarly, the process of the experiment on Wine is the same as Iris. We will not repeat here.

A summary of the intra-cluster distances (M values) obtained by the classical algorithm is shown in Table 1. As seen from the results in Table 1, the BH-MC algorithm achieves the best results among all the algorithms. For the Iris dataset, the best, average and worst M values obtained by BH-MC are 96.65569, 96.65636, and 96.65902 respectively. The worst M value of BH-MC is much better than the best values of other algorithms. For the Wine

dataset, the BH-MC algorithm achieves the optimum value of 16293.41896 which is much better than other algorithms.

Table 1.The summary of the intra-cluster distances obtained by different algorithms

| Dataset	Criteria	K-means	PSO	GSA	BH-MC
	Best	97.32592	96.87935	96.68794	96.65569
	Average	105.72902	98.14236	96.73105	96.65636
Iris	Worst	128.40420	99.76952	96.82463	96.65902
	Std	12.38759	0.84207	0.02761	0.00167
wine	Best	16555.67942	16304.48576	16313.87620	16293.41896
Wine	Average	16963.04499	16316.27450	16374.30912	16294.67863
	Worst	23755.04949	16342.78109	16428.86494	16296.57306
	Std	1180.69420	12.60275	34.67122	1.34279
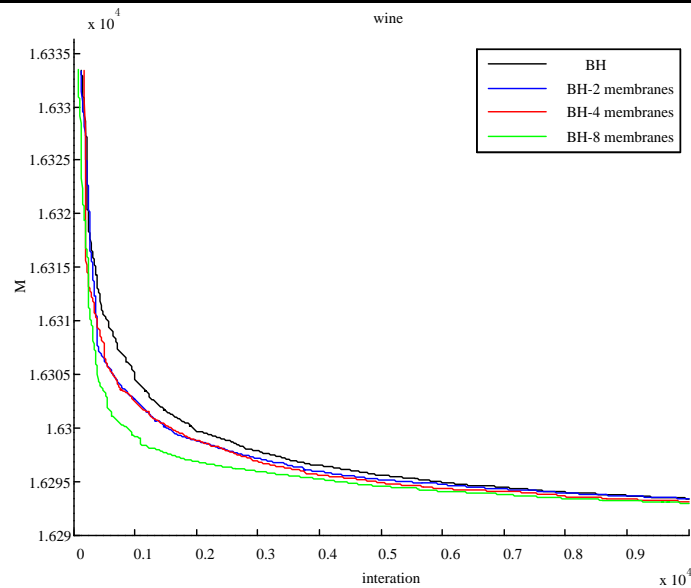
Fig.2. The comparison on Wine dataset between BH and BH-MC

Actually, the BH algorithm has achieved the same effect as our algorithm presented in Table 1. However, the rate of convergence is so slow that the BH algorithm needs to sacrifice more iteration to achieve the best results. Fig.2 above shows the comparison on Wine dataset between BH algorithm and the BH-MC algorithm which used 2, 4 and 8 membranes respectively. As seen from Fig.2 clearly, the BH-MC algorithm achieves the better effect of the rate of convergence compared with the BH algorithm. Fig.3 below shows the

experiment results for Iris dataset. It can be indicted by the Fig.3 that the results become better and better as the increase of the number of the membrane.
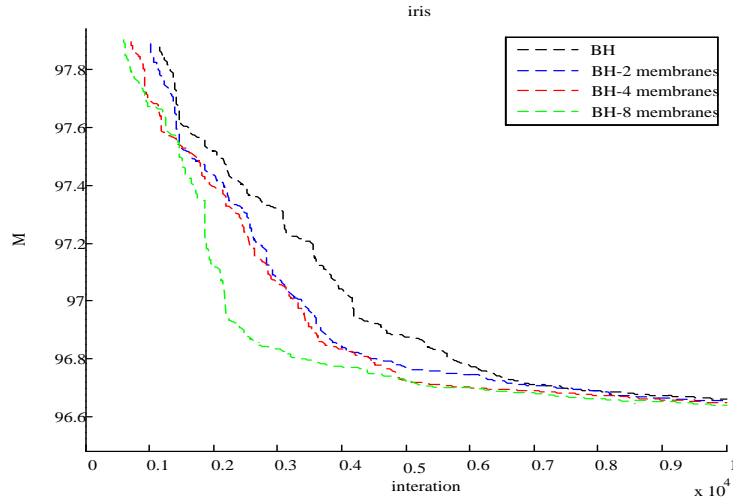

Fig.3. The comparison on Iris dataset between BH and BH-MC

Conclusion

In recent years, modelling and simulating the natural phenomena for solving complex problems has been a hot research area. In this paper, the BH-MC algorithm is proposed which combined with the advantage of the black hole algorithm and the membrane computing. The results of the experiment show that the BH-MC algorithm outperforms other classical algorithms. In future research, the BH-MC algorithm will find more and more different areas of applications.

Acknowledgement

References

[1]O. Castillo, R. Martinez-Marroquin, P. Melin, et al, Comparative study of bio-inspired algorithms applied to the optimization of type-1 and type-2 fuzzy controllers for an autonomous mobile robot, Information Sciences192(2012)19-38.

[2]F. Kang, J. Li, Z. Ma, Rosenbrock artificial bee colony algorithm for accurate global optimization of numerical functions, Information Sciences181(2011)3508-3531.

[3] D. Kundu, K. Suresh, S. Ghosh, et al, Multi-objective optimization with artificial weed colonies, Information Sciences 181(2011)2441-2454.

[4]K.S. Tang, K.F. Man, S. Kwong, et al, Genetic algorithms and their applications. IEEE Signal Processing Magazine, 1996, 13(6): 22-37.

[5]G. J. Deboeck, [Ed.], "Trading On The Edge", Wiley, 1994.

[6]A. Badr, A. Fahmy, A proof of convergence for ant algorithms, Information Sciences160 (2004) 267-279.

[7]B.Y. Qu, J.J. Liang, P.N. Suganthan, Niching particle swarm optimization with local search for multi-modal optimization, Information Sciences197(2012)131-143.

[8]A. Hatamlou, S. Abdullah, H. Nezamabadi-pour, Application of gravitational search algorithm on data clustering, in: Rough Sets and Knowledge Technology, Springer, Berlin/Heidelberg, 2011, pp.337-346.

[9]G. Paun, Y. Suzuki, H. Tanaka, et al, On the power of membrane division in P systems, Theoretical Computer Science, 2004, 324(1): 61-85.

[10] M. Gheorghe, N. Krasnogor, M. Camara. P systems applications to systems biology, Biosystems , 2008,91(3): 435-437.

[11]G. Alexandros, M. Gheorghe, F. Bernardini, Membrane-based devices used in computer graphics, Applications of Membrane Computing, Springer Berlin Heidelberg, 2006. 253-281.

[12]D.E. Goldberg., Genetic algorithms in search, optimization and machine learning, Addison-Wesley, Reading, MA, 1989.

[13]G. Păun, et al, On the power of membrane division in P systems, Theoretical Computer Science 324.1 (2004): 61-85.

[14]M.A. Gutierrez-Naranjo, M.J. Perez-Jimenez, D.A. Ramirez-Martinez software tool for verification of spiking neural P systems, Natural Computing, 2008, 7(4):85-497.

[15] G. Păun, Computing with membranes, Journal of Computer and System Sciences, 61(1) (2000): 108-143.

[16] F. Bernardini, M. Gheorghe. Languages generated by P systems with active membranes, New Generation Computing, 2004, 22(4): 311-329.

[17]L. Huang, I.H. Suh, Controller design for a marine diesel engine using membrane computing, International Journal of Innovative Computing Information and Control, 2009, 5(4): 899-912, 2009.

[18]L. Huang, L. Sun, N. Wang, et al, Multiobjective Optimization of Simulated Moving Bed by Tissue P System., Chinese Journal of Chemical Engineering, 2007, 15(5): 683-690.

[19]G. Păun, R. Păun, Membrane computing and economics: Numerical P systems, Fundamenta Informaticae, 2006, 73(1-2): 213-227.

[20]A. Hatamlou, Black hole: A new heuristic optimization approach for data clustering, Information Sciences 222(2013)175-184.

[21]A. Adl, A. Badr, I. Farag, A Note on the Membrane Computer, arXiv preprint arXiv:1009.0923 (2010).

[22]G. Păun, G. Rozenberg, A. Salomaa, Handbook of membrane computing Oxford: Oxford University Press, 2009.

[23]G. Păun, Computing with membranes, Journal of Computer System Sciences, vol.61, no.1, pp.108-143, 2000.

[24]R. Freund, G. Păun, M.J. P?erez-Jim?enez, Tissue-like P systems with channel-states, Theoretical Computer Science, vol.330, no.1, pp.101-116, 2005.

[25] H. Peng, et al, Fuzzy reasoning spiking neural P system for fault diagnosis, Information Sciences 235 (2013): 106-116.

[26]G. Păun, Computing with membranes: Attacking NP-complete problems, Unconventional models of Computation, UMC'2K, Springer London, 2001. 94 -115.