An Improved Artificial Bee Colony Algorithm for the Capacitated Vehicle Routing Problem

S.Z. Zhang, C.K.M. Lee
Department of Industrial and Systems Engineering
the Hong Kong Polytechnic University, HK SAR
shuzhu.zhang@connnect.polyu.hk
ckm.lee@polyu.edu.hk

***Abstract*—The capacitated vehicle routing problem (CVRP) is one of the combinatorial optimization problems with the most widespread applications in practice. Because of the intrinsic computational complexity, the approximate algorithms are commonly employed to solve the CVRP rather than the exact algorithms. In this research, the artificial bee colony algorithm (ABC), derived from the swarm intelligence, is adapted to handle the CVRP. The application of the ABC algorithm in solving the CVRP exploited the inherent features of the swarm intelligence. More importantly, a routing directed ABC algorithm (RABC) is further proposed consisting of numerous improvements in order to enhance the capability of the diversified search and intensified search of the conventional ABC algorithm, which incorporates the useful information from the routing as well. The RABC algorithm is examined with different benchmark test instances. The experimental results show that the RABC algorithm excels the conventional ABC algorithm significantly. Moreover, the application of the RABC algorithm in solving the CVRP can provide practical insights for the implementation of swarm intelligence in solving other combinatorial optimization problems.**

Keywords—Artificial Bee Colony Algorithm, Capacitated Vehicle Routing Problem, Swarm Intelligence

I. INTRODUCTION

The vehicle routing problem (VRP) is one of the combinatorial optimization problems with widespread application background in various disciplines. The VRP was first introduced by Dantzig and Ramser [1]. The objective of the VRP is to find the optimal routes to serve a number of customers without any violation in terms of the vehicle requirements and the customer requirements. The optimal routes are commonly expressed in several ways, such as the shortest travelling distance, the shortest travelling time and the minimum travelling cost [2]. Ever since the introduction of the VRP, it has been widely studied. Various variants are derived from the VRP related to specific problems, among which the capacitated VRP (CVRP) is of greatest significance [3]. The CVRP is to further consider the vehicle capacity in addition to the VRP when designing the optimal routes. Each vehicle has its capacity, which cannot be violated by the cumulative demand of the customers it serves. More VRP variants or models can be found from the literature [4].

The VRP is well-known as the NP-hard problem, which cannot be solved within a reasonable polynomial time. Exact algorithms, e.g., Linear Programming (LP), Branch and Bound (BB), and Cutting Plane (CP) method, are gradually becoming less popular, since they are either unable to solve the VRP or consume too much computational cost, especially when the size of the VRP is large [5]. Comparatively, the approximate algorithms are becoming increasing popular for solving the VRP [6], [7]. The critical feature of the approximate algorithms is that they are capable of finding near-optimal solutions within an affordable and reasonable time period. The easy and robust applicability promotes the popularity of the approximate algorithms, even though they cannot guarantee the optimal solution. For example, Bräysy and Gendreau [8] employed a Tabu Search (TS) heuristics to solve the VRP with time windows. Gambardella, et al. [9] proposed an Ant Colony Optimization (ACO) to handle the VRP in advanced logistics systems. Prins [10] introduced a simple and effective Genetic Algorithm (GA) for VRP. Nagata and Bräysy [11] presented an edge assembly-based Memetic Algorithm (MA) for the CVRP. Kuo [12] used Simulated Annealing (SA) to minimize fuel consumption for the time-dependent VRP. Gong, et al. [13] adapted a discrete Particle Swam Optimization (PSO) approach to optimize the VRP with time windows.

In this research, a new approximate algorithm, named Artificial Bee Colony (ABC) algorithm, is adapted and further improved to handle the CVRP. The ABC algorithm is a relatively new algorithm, which was introduced by Karaboga [14] in 2005. The ABC algorithm could be classified as an exemplar of the swarm intelligence, which simulates the collective intelligent behaviors of natural species, such as ant foraging, fish schooling, birds gathering and bee swarming [15]. The swarm intelligence possesses three intrinsic features, i.e., decentralization, self-organization and collective behavior, which guarantees the swarm intelligent behaviors [16]. All these three features are explicitly exemplified in the ABC algorithm.

Following the instruction of the conventional ABC algorithm, an improved ABC algorithm with the specific design for the CVRP called (RABC) is proposed in order to better balance the effect of diversification and intensification. In the RABC algorithm, the trade-off between the exploration of the search space and the exploitation of the promising area is well realized. The knowledge derived from the CVRP is integrated into the operation of the RABC algorithm effectively and efficiently. The computational performance of RABC algorithm is measured in numerical experiments in comparison with the conventional ABC algorithm.

978-1-4799-8697-2/15 $31.00
DOI 10.1109/SMC.2015.371

The remains of this research are organized as follows. Section 2 presents the mathematical model for the CVRP. In section 3, the procedures of the conventional ABC algorithm are described and the application of the RABC algorithm is presented. Section 4 demonstrates the effects of the RABC algorithm through some benchmark test instances. Finally, section 5 concludes this research.

PROBLEM FORMULATION

The CVRP is defined on a graph, denoted as $G = (V, A)$. The $V = \{0,1,2,\cdots,N\}$ is the set of nodes, while the $A = \{(i,j)|i,j \in V\}$ is the set of edges connecting the nodes. The graph possesses the characteristics, such as non-direction and full-connectivity. Within the node set, $i = 0$ indicates the depot, where the vehicles depart from and return to. The vehicles are denoted as $k = 1,2,\cdots,K$, and all the vehicles share the same capacity as q_k. The customers are indexed as $i = 1,2,\cdots,N$ with a predetermined number of demand for each customer as d_i. The travelling cost from node i to node j is denoted as c_{ij}, in terms of the travelling distance or travelling time. Therefore, the objective of CVRP is to find the optimal routes with minimum travelling cost.

Two decision variables are listed as follows.

$x_{ijk} = \{1,0\}; i \neq j; i,j \in \{0,1,\cdots,N\}$. $x_{ijk} = 1$ if the customer j is served by vehicle k directly after customer i, otherwise $x_{ijk} = 0$.

$y_{ik} = \{1,0\}$; $i \in \{1,2,\cdots,K\}$. $y_{ik} = 1$ if customer i is served by vehicle k, otherwise $y_{ik} = 0$.

With the above settings and notations, the whole mathematical model is presented as follows.

Objective functions

$$\min f = \sum_{i=0}^{N}\sum_{j=0}^{N}\sum_{k=1}^{K} c_{ij}x_{ijk} \tag{1}$$

Subject to

$$\sum_{j=1}^{N} x_{0jk} = \sum_{i=1}^{N} x_{i0k} \leq 1, k \in \{1,\cdots,K\} \tag{2}$$

$$\sum_{k=1}^{K} y_{ik} = 1, i \in \{1,\cdots,N\} \tag{3}$$

$$\sum_{i=0}^{N} x_{ijk} = y_{jk}, j \in \{0,\cdots,N\} \tag{4}$$

$$\sum_{j=0}^{N} x_{ijk} = y_{ik}, j \in \{0,\cdots,N\} \tag{5}$$

$$\sum_{k=1}^{K}\sum_{j=1}^{N} x_{0jk} \leq K \tag{6}$$

$$\sum_{i=1}^{N} d_i y_{ik} \leq q_k, k \in \{1,\cdots,K\} \tag{7}$$

$$\mathrm{x_{ijk}} = 0 \text{ or } 1, \mathrm{i,j} = 0,1,\cdots,\mathrm{N}; \mathrm{k} = 1,2,\cdots,\mathrm{K} \tag{8}$$

$$y_{ik} = 0 \; or \; 1, i = 0,1,\cdots,N;\; k = 1,2,\cdots,K \tag{9}$$

Objective (1) aims at finding the optimal routes with minimum cost. Constraint (2) indicates that all the vehicles depart from the depot at the beginning and return to the depot finally. Constraint (3) implies that one customer can only be served by vehicle. Constraint (4) and Constraint (5) guarantee that the visiting vehicle and the serving vehicle for one customer have to be the same one. Constraint (6) means that the number of vehicles cannot exceed a predetermined number. Constraint (7) means that the demand of customers in one route cannot exceed the capacity of the vehicle, which serves this route. Constraint (8) and constraint (9) are the binary constraints for the decision variables.

METHODOLOGY

A. General Description of the conventional ABC Algorithm

As we mentioned in the introduction section, swarm intelligence has three intrinsic properties, i.e. decentralization, self-organization and collective behavior. Decentralization indicates that there is no central control mechanism. The self-organization means that an individual could determine its behavior by itself through an interaction with other individual or the environment. Moreover, the interaction rules are very simple, and the interaction result would affect the behavior of an individual. In addition to the interaction, the decision of an individual might be affected by some other random factors. Collective behavior refers to the overall behavior pattern of a swarm, in which the individual behavior may be random, however the aggregation of individual behavior turns to be globally intelligent.

The ABC algorithm is an exemplar of the swarm intelligence, which mimics the intelligent behavior of honey bee foraging. There are three types of bees in a bee swarm, namely scout bees, employed bees and onlooker bees. Each type of bees shares the same goal when searching for food sources. The implementation of the ABC algorithm consists of four phases. (1) Scout bees leave the hive and try to find food sources nearby. (2) Then, employed bees are assigned to exploit the found food sources. One employed bee corresponds to one food source. During the exploitation, employed bees also try to search other food sources around. If better food sources are found, employed bees would abandon the previous food sources and exploit the better ones. (3) After that, all the employed bees return to the hive and communicate the information of food sources to the onlooker bees. Subsequently, the onlooker bees become the employed bees aiming to exploit the corresponding food sources. (4) After some time, one food source may be gradually consumed and finally exhausted. In that case, the exhausted food source would be abandoned and replaced with another food source.

Fig. 1 illustrates the procedures of the ABC algorithm. From the methodological perspective, food sources are treated as the solutions to specific problems. The one with more nectar corresponds to the better solution, which is measured in terms of its associated fitness. In phase 1, the initial solutions are found randomly within the search space. All solutions are evaluated using the fitness function. In phase 2, the neighborhood search is implemented through the mutation of the current solution. The greedy selection mechanism is applied in the neighborhood search to make sure that new solution is employed only if it is better than the current solution. In phase 3, the preferable selection from the onlooker bees to the employed bees is conducted using the roulette wheel selection mechanism considering the chosen probability of each solution. In phase 4, one solution is abandoned if it cannot be improved through a certain number of iterations, and the new solution is found following the same procedure as the phase 1.

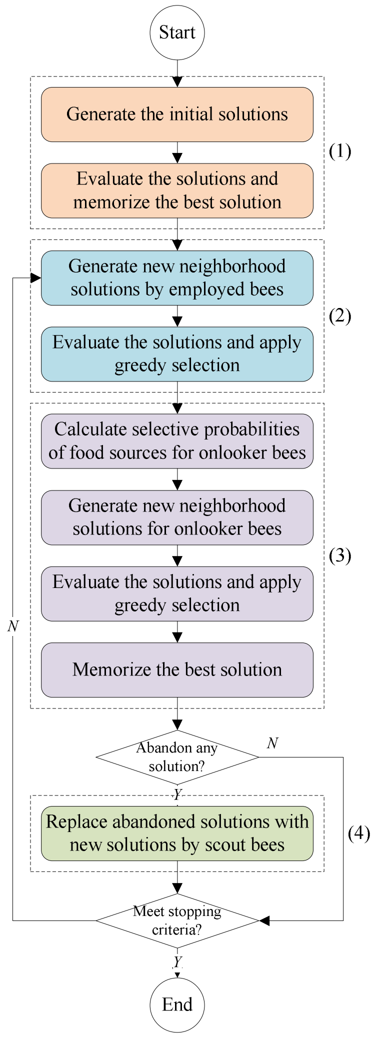

Fig. 1 The flowchart of the ABC algorithm

Application of the RABC Algorithm

Following the procedures of the conventional ABC algorithm, the application of RABC algorithm to tackle the CVRP is described in detail.

Solution scheme and initial solutions

The solution scheme is one of the critical factors which affect the algorithm performance significantly. In this research, the format of single numeric-string with multiple delimiters is employed. Each sub-string between two close delimiters is regarded as a route. Table I describes one example of this format, which use 0 as the delimiter. In this case, four routes existed, among which the first three are valid, while the fourth one is not. One vehicle serves one route, which means three vehicles are needed in this example.

TABLE I. EXAMPLE OF THE SOLUTION SCHEME

Customers	1	2	3	4	5	6	7	
Solution	$X = \{0,1,2,0,4,5,3,0,6,7,0,0\}$							Route 1: {0,1,2,0} Route 2: {0,4,5,3,0} Route 3: {0,6,7,0} Route 4: {0,0}

The dimension of such a scheme is $N + K + 1$. N is the number of customers, while K is the number of vehicles. However, one possible issue is that the number of vehicles is not known before. Actually one of the potential objectives is to find the minimum number of vehicles in the CVRP. For simplicity, the number of vehicles is frequently set as the number of the customers. However, in this research, the proper number of vehicles is calculated through the equation $K^* = \alpha \frac{\sum_{i=1}^{N} d_i}{q_k}, 1 \leq \alpha \leq 3$. Compare with the conventional one, the dimension size could be largely reduced, which could promote the algorithm performance in return.

The initial population is generated as table II. First, customers are permutated randomly in terms of their indices. Then delimiters are inserted into this permutation considering the capacity of vehicles to construct different routes. The number of the delimiters is decided by the number of vehicles. Such a generation mechanism could guarantee the feasibility of the initial solutions, which can facilitate the algorithm performance. However, during the iteration, different operators are applied to the solutions, which lead to the high possibility of emerging infeasible solutions due to the violation of the capacity of vehicles. One of the improvements herein is that the parallel handling of both feasible and infeasible solutions. In case of the infeasible solutions, the measurement of the infeasible solution is calculated as $f(x) = c(x) + \beta * p(x)$, while $p(x) = \sum_{i=1}^{N} d_i y_{ik} - q_k$. The parameter β is self-adjusted as $\beta = \delta * iteration\ index$. The value of the parameter δ refers the maximum number of the iteration. When the iteration is small, the infeasible solutions are allowed in order to assist the diversified search. With the increase of the iteration, all the infeasible solutions are gradually excluded from the population.

TABLE II. THE GENERATION OF AN INITIAL SOLUTION

Customer indices	1	2	3	4	5	6	7
Random permutation	2	3	7	6	1	4	5
Insert delimiter	0,2,3,7,0,6,1,0,4,5,0,…,0						

Employed bee phase

After the initialization phase, the employed bees are assigned to exploit the found food sources. Meanwhile, the employed bees also attempt to explore better food sources with more nectar. From the perspective of algorithm, a neighborhood search is conducted for each given solution. Once a new solution is found in the neighborhood, the greedy selection mechanism is applied, which means, the current solution is replaced with the new found one only if the new found solution is better; otherwise the current solution is kept. The most frequently used operator is the swap operator, which means to swap two random customers as illustrated in table III.

TABLE III. THE SWAP OPERATOR

Before swap	0	2	3	7	0	6	1	0	4	5	0
After swap	0	4	5	3	7	0	6	1	0	2	0

In this research, in order to utilize the existing value of the given solution, a new operator, named best route mapping crossover (BMX), is introduced as shown in table IV. The procedure of BMX operator is as follows. Given the current solution as solution 1, the best sub route within this solution, e.g.

{0, 6, 1, 0}, is found and kept. Then the customer indices within this best sub route are removed from the referenced solution. Finally the new solution is generated using the best sub route from solution 1 and the remains from solution 2. Such a new solution could utilize the routing information from both two solutions, which benefit the algorithm convergence considerably.

TABLE IV. THE BMX OPERATOR

Solution 1	0	2	3	7	0	6	1	0	4	5	0
Solution 2	0	~~1~~	2	5	~~0~~	7	4	~~0~~	3	~~6~~	0
Keep the best sub route					0	6	1	0			
Supplement the rest	0	2	5	7					4	3	0
New solution	0	2	5	7	0	6	1	0	4	3	0

Onlooker bee phase

After that, all the employed bees communicate their associated food source information to the onlooker bee waiting in the hive. The onlooker bees may decide to trace certain employed bees preferably. Such a preference is proportional to the nectar amount of food sources. In the algorithm, the nectar amount is represented as the fitness of one solution. The roulette wheel selection is applied in this research and the associative chosen probability for solution i is calculated through $p_i = \frac{fit_i}{\sum_{i=1}^{SN} fit_i}$, SN is the number of solutions. Once an onlooker bee determines to exploit one food source, it becomes an employed bee and repeats the procedures as the employed bee phase.

Scout bee phase

During the exploitation process, one food source is gradually consumed. When the consumption reaches certain level, the food source will be abandoned and replaced with another food source. In the algorithm, one solution is assigned a label of trial. In each neighborhood search, if one solution is improved, then its trial *number* is reset as 0; otherwise its trial number accumulates. Furthermore, if one solution cannot be improved after a predefined criterion, it is abandoned. Such a criterion is denoted as $limit$, which is commonly set as $limit = SN * D$. SN is the number of solutions, while D is the number of dimensions. In this research, the number of dimension is replaced by the number vehicles (K) as $limit = SN * K$, as the latter one could provide a more diversified search. The abandoned solution is replaced by another randomly generated solution following the same procedures of the initialization phase.

NUMERICAL EXPERIMENT

The ABC algorithm consists of a rather simple mechanism, in which only two algorithm specific parameters existed, i.e. the number of the colony size (CS) and the setting of the abandonment criterion ($limit$). In this research, the CS is set as 20, which indicates that there are a swarm of 10 solutions evolving during the searching process. The abandonment criterion is set as $limit = 0.5 * CS * K$. Compare with the conventional ABC algorithm, the RABC algorithm contains the following improvements. (1) More appropriate setting of K reduces the dimensional size of the solution scheme. (2) The utilization and dynamic adjustment of the infeasible solutions assists both the diversified search and the intensive search. (3) The introduced BMX operator helps to better utilize the existing value from current solutions. (4) The abandonment criterion is changed to enhance a more diversified search. The test instances are adopted from the literature [3]. The instance name contains the number of the nodes in the graph and the known optimal number of vehicles required. For instance, the instance "A-n80-k10" means that there are 80 customer nodes in this graph, and the known optimal solution consists of 10 vehicles.

The experiments are implemented using Java programming with an Eclipse IDE on a PC with 3.60GHz CPU and 16.0 GB RAM. For each instance, the program is executed 10 times, and the best result, average value and the standard deviation are provided in table V. From the results, it can be concluded that, the best results and the average results are improved 17.89% and 18.67%. Moreover, such a difference is gradually enlarged with the increase of the nodes in the graph, which suggests the promising applicability of the RABC algorithm to tackle the problems with high complexity of CVRP.

Table V. THE COMPUTATIONAL RESULTS

Instances	The conventional ABC			The RABC		
	best	avg	s.d.	best	avg	s.d.
A-n32-k5	821.7	917.5	36.2	772.5	800.6	31.4
A-n33-k5	709.8	759.7	26.2	660.8	679.4	12.3
A-n34-k5	808.4	888.9	32.3	787.4	807.8	11.4
A-n36-k5	907.9	961.0	28.7	802.1	837.4	22.1
A-n37-k5	750.1	815.8	34.8	691.4	722.6	20.2
A-n38-k5	866.3	889.6	20.1	739.8	756.8	20.1
A-n39-k5	930.3	1003.8	38.3	851.4	886.7	31.9
A-n44-k7	1151.7	1207.4	36.9	962.6	1019.7	45.0
A-n45-k6	1180.7	1222.8	35.2	941.6	1003.5	34.3
A-n46-k7	1128.6	1179.4	36.1	918.8	998.8	45.0
A-n48-k7	1341.2	1409.1	36.8	1130.2	1189.9	49.3
A-n53-k7	1326.8	1417.8	69.6	1063.7	1129.0	32.7
A-n54-k7	1493.1	1563.4	50.8	1278.2	1313.9	28.1
A-n55-k9	1413.1	1464.0	33.7	1131.5	1159.6	21.9
A-n60-k9	1666.2	1932.4	95.2	1465.6	1533.1	43.2
A-n61-k9	1432.8	1502.1	55.4	1096.3	1182.0	43.5
A-n62-k8	1741.8	1864.3	67.5	1418.4	1498.1	50.7
A-n63-k10	1854.7	1913.7	50.3	1403.4	1521.1	66.2
A-n64-k9	2015.7	2083.8	63.4	1530.9	1610.3	38.7
A-n65-k9	1738.9	1813.0	73.6	1274.5	1362.4	60.6
A-n69-k9	1728.5	1808.2	49.7	1259.3	1335.7	47.3
A-n80-k10	2525.7	2836.5	157.7	2070.2	2231.7	104.8
Average	**1342.4**	**1429.7**	**51.3**	**1102.3**	**1162.7**	**39.1**

CONCLUSION

The ABC algorithm is a relatively new swarm intelligent algorithm, which exemplifies the potential and the power of the swarm intelligence. The ABC algorithm was firstly introduced to solve the continuous numerical problems. Various modifications and improvements are needed in order to adapt the ABC algorithm to the discrete combinatorial optimization problems. In this research, an improvement of the conventional ABC algorithm, named as RABC algorithm, is introduced to solve the CVRP. Both the conventional ABC algorithm and the RABC algorithm are examined through benchmark test instances. The experimental results demonstrate the effectiveness and efficiency of the RABC algorithm.

ACKNOWLEDGMENT

This work is supported by the Hong Kong Polytechnic University. Great gratitude is also extended to the research committee and the Department of Industrial and Systems Engineering of the Hong Kong Polytechnic University for support in this project (#4-RTY0) and (H-ZD88)

REFERENCES

[1] G. B. Dantzig and J. H. Ramser, "The Truck Dispatching Problem," *Management* Science*,* vol. 6, pp. 80-91, 1959.

[2] B. L. Golden, S. Raghavan, and E. A. Wasil, *The vehicle routing problem: latest advances and new challenges* vol. 43: Springer, 2008.

[3] P. Augerat, J. Belenguer, E. Benavent, A. Corberán, D. Naddef, and G. Rinaldi, "Computational results with a branch and cut code for the capacitated vehicle routing problem," *Rapport de recherche- IMAG,* 1995.

[4] P. Toth and D. Vigo, "An overview of vehicle routing problems," *The vehicle routing problem,* vol. 9, pp. 1-26, 2002.

[5] G. Laporte, M. Gendreau, J. Y. Potvin, and F. Semet, "Classical and modern heuristics for the vehicle routing problem," *International Transactions in Operational Research,* vol. 7, pp. 285-300, 2000.

[6] J.-F. Cordeau, M. Gendreau, G. Laporte, J.-Y. Potvin, and F. Semet, "A guide to vehicle routing heuristics," *Journal of the Operational Research* Society*,* vol. 53, pp. 512-522, 2002.

[7] K. C. Tan, L. H. Lee, Q. L. Zhu, and K. Ou, "Heuristic methods for vehicle routing problem with time windows," *Artificial Intelligence in Engineering,* vol. 15, pp. 281-295, 2001.

[8] O. Bräysy and M. Gendreau, "Tabu Search heuristics for the Vehicle Routing Problem with Time Windows," *Top,* vol. 10, pp. 211-237, 2002/12/01 2002.

[9] L. M. Gambardella, A. E. Rizzoli, F. Oliverio, N. Casagrande, A. V. Donati, R. Montemanni, and E. Lucibello, "Ant Colony Optimization for vehicle routing in advanced logistics systems," in *Proceedings of the International Workshop on Modelling and Applied Simulation (MAS 2003)*, 2003, pp. 3-9.

[10] C. Prins, "A simple and effective evolutionary algorithm for the vehicle routing problem," *Computers & Operations Research,* vol. 31, pp. 1985-2002, 2004.

[11] Y. Nagata and O. Bräysy, "Edge assembly-based memetic algorithm for the capacitated vehicle routing problem," *Networks,* vol. 54, pp. 205-215, 2009.

[12] Y. Kuo, "Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem," *Computers & Industrial Engineering,* vol. 59, pp. 157-165, 2010.

[13] Y. J. Gong, J. Zhang, O. Liu, R. Z. Huang, H. S. H. Chung, and Y. H. Shi, "Optimizing the Vehicle Routing Problem With Time Windows: A Discrete Particle Swarm Optimization Approach," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on,* vol. 42, pp. 254-267, 2012.

[14] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," 2005.

[15] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization,* vol. 39, pp. 459-471, 2007/11/01 2007.

[16] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm intelligence: from* natural *to artificial systems* vol. 4: Oxford university press New York, 1999.