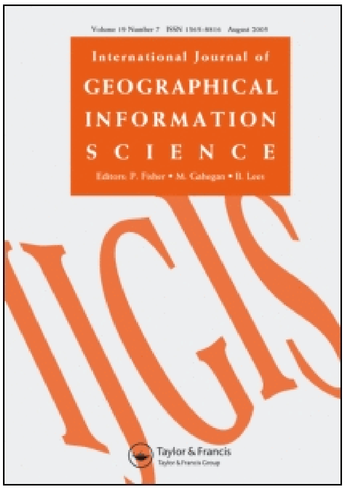


International Journal of Geographical Information Science
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713599799

A bottom-up approach to discover transition rules of cellular automata using ant intelligence

Xiaoping Liu [a]; Xia Li [a]; Lin Liu [b]; Jinqiang He [a]; Bin Ai [a]
[a] School of Geography and Planning, Sun Yat-sen University, Guangzhou 510275, Guangdong, China [b] Department of Geography, University of Cincinnati, USA

Online Publication Date: 01 January 2008

To cite this Article Liu, Xiaoping, Li, Xia, Liu, Lin, He, Jinqiang and Ai, Bin(2008)'A bottom-up approach to discover transition rules of cellular automata using ant intelligence',International Journal of Geographical Information Science,22:11,1247 — 1269

To link to this Article: DOI: 10.1080/13658810701757510

URL: http://dx.doi.org/10.1080/13658810701757510

International Journal of Geographical Information Science
Vol. 22, Nos. 11–12, November–December 2008, 1247–1269

Research Article

A bottom-up approach to discover transition rules of cellular automata using ant intelligence

XIAOPING LIU†, XIA LI*†, LIN LIU‡, JINQIANG HE† and BIN AI†
†School of Geography and Planning, Sun Yat-sen University, Guangzhou 510275, Guangdong, China
‡Department of Geography, University of Cincinnati, USA

(*Received 24 April 2007; in final form 15 October 2007*)

This paper presents a new method to discover transition rules of geographical cellular automata (CA) based on a bottom-up approach, ant colony optimization (ACO). CA are capable of simulating the evolution of complex geographical phenomena. The core of a CA model is how to define transition rules so that realistic patterns can be simulated using empirical data. Transition rules are often defined by using mathematical equations, which do not provide easily understandable explicit forms. Furthermore, it is very difficult, if not impossible, to specify equation-based transition rules for reflecting complex geographical processes. This paper presents a method of using ant intelligence to discover explicit transition rules of urban CA to overcome these limitations. This 'bottom-up' ACO approach for achieving complex task through cooperation and interaction of ants is effective for capturing complex relationships between spatial variables and urban dynamics. A discretization technique is proposed to deal with continuous spatial variables for discovering transition rules hidden in large datasets. The ACO–CA model has been used to simulate rural–urban land conversions in Guangzhou, Guangdong, China. Preliminary results suggest that this ACO–CA method can have a better performance than the decision-tree CA method.

Keywords: Ant colony optimization; CA; Urban simulation; Artificial intelligence

1. Introduction

Cellular automata (CA) were originally conceived by Ulam and Neumann in the 1940s to provide a formal framework for investigating the behaviour of complex, self-reproducible systems (White and Engelen 1993). One key feature of CA is that complex global spatial patterns can be generated by a set of simple local rules. This 'bottom-up' approach coincides with complexity theories stating that a complex system comes from the interactions of simple subsystems. CA provide an effective way of simulating and predicting the spatial-temporal evolution of complex geographical phenomena (Batty and Xie 1994, Takeyama and Couclelis 1997). At the end of the 1980s, Couclelis (1988) put forward the theoretical framework for geographically oriented CA, and applied it to the simulation of urban expansion (Couclelis 1985, 1988, 1989) and population dynamics (Couclelis 1988). CA have been used to simulate many geographical phenomena, such as wildfire diffusion

*Corresponding author. Email: lixia@mail.sysu.edu.cn

International Journal of Geographical Information Science
ISSN 1365-8816 print/ISSN 1362-3087 online © 2008 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/13658810701757510

(Clarke *et al.* 1994, Hargrove *et al.* 2000), epidemic propagation (Sirakoulis *et al.* 2000), forest dynamics (Lett *et al.* 1999), landscape changes (Wang and Zhang 2001, Soares-Filho *et al.* 2002), urban evolution (Batty and Xie 1994, Clarke *et al.* 1997, Couclelis 1997, Li and Yeh 2000, 2002, Wu 2002), and land-use changes (White and Engelen 1993, Batty and Xie 1994, Couclelis 1997, Clarke and Gaydos 1998, Wu and Webster 1998, Li and Yeh 2002, 2004). These studies have demonstrated the capability of CA for simulating and predicting complex geographical processes.

The core of CA is how to define the transition rules, which determine the state conversion of geographical processes. The determination of transition rules is a challenging task because of many spatial variables and parameters involved. Moreover, the experts' knowledge and preferences also affect the model structure of CA (Yeh and Li 2006). Some methods are based on heuristics for defining transition rules. These heuristic models are confronted with a severe computational constraint (Wu 2002). For example, Clarke *et al.* (1997) simulated development scenarios using different combinations of parameter values, and then determined the most optimal combination through visually comparing the simulated patterns with the actual ones. However, it is difficult to find the best combination from an extremely large set of permutations. Clarke and Gaydos (1998) used high-performance workstations which need at least several hundred hours to find the best combination of parameter values. Wu (1998) proposed a more structured procedure based on the hierarchical analysis process (AHP) that defines the parameter values in a heuristic way. Jenerette and Wu (2001) developed an urban cellular automaton using the genetic algorithm (GA). Wu (2002) presented an even less subjective method employing logistic regression models. However, both the AHP and the logistical regression methods are inherently linear and not good at dealing with the complex relationships between spatial variables and urban dynamics. An artificial neural-network (ANN) CA model was developed to obtain parameter values automatically for capturing a complex relationship (Li and Yeh 2002). Because of its black-box nature, ANN does not give an insight into the relations actually used in modelling, leaving the user uninformed about the possible lack of causality in the relations that are used in the model. Decision tree models were later presented to discover transition rules of CA (Li and Yeh 2004), which tends to be vulnerable to local optimization. Kernel-based learning machine has been applied to the induction of nonlinear transition rules in high-dimension feature space (Liu and Li 2006). However, this method is also constrained by the use of implicit transition rules and the need for intensive computation. Thus, we consider it academically interesting and practically valuable to explore intelligent methods for effective discovering transition rules in urban CA.

In this paper, an intelligent method (ACO) is used to discover transition rules of CA. Ant colony optimization (ACO), a computational method derived from natural biological systems, is first proposed by Dorigo *et al* (1991). ACO is a multi-agent system that simulates the natural behaviour of ants according to the mechanisms of cooperation and adaptation (Dorigo 1992). Complex tasks, such as optimizing the route for seeking food, can be effectively fulfilled by cooperation and interaction between ants. Although there is no centralized control dictating the behaviour of the artificial agents, local interactions among agents can result in the emergence of some global patterns in the simulation (Dorigo 1992). ACO is in fact a swarm intelligence-based heuristic system, using positive feedback between agents as a search mechanism. ACO has become a hot topic in artificial intelligence field (Dorigo

1992, Dorigo *et al.* 1996). Satisfactory results have been obtained in solving travelling salesman problems (TSP), data clustering, combinatorial optimization and network routing by using ACO Algorithms (Lumber and Faieta 1994, Kwang and Weng 2002). However, applying ACO to urban simulation has not been reported for solving nonlinear problems. ACO is based on the 'bottom-up' approach to accomplish complex tasks through cooperation among agents. CA are also based on a 'bottom-up' approach for simulating the behaviour of complex systems, such as urban evolution. The definitions of transition rules of CA usually require tedious jobs because of numerous spatial variable and parameters involved in urban simulation. It is appropriate to use ACO to discover transition rules of CA by capturing complex relationships.

The studies on classification rule induction using ACO are relatively unexplored. In fact, ACO is very appropriate to discover classification rules, since mining of classification rules is a search problem, and ACO is very successful in global searches. Furthermore, ACO can cope better with attribute correlation than greedy rule induction algorithms such as decision trees (Alatas and Akin 2005). Parpinelli *et al* (2002) were the first to propose ACO for discovering classification rules, with the Ant-Miner system. In their study, it is demonstrated that Ant-Miner produces better accuracy and simpler rules than decision trees.

This paper will examine the potential of using Ant-Miner for discovering transition rules of CA. The objective is to find complex relationships hidden in large datasets by using GIS and remote sensing data. It is expected that the positive feedback mechanism of ACO can produce better results in discovering optimal rules by simulating the behaviour of ants seeking food. A discretization technique is incorporated in the model to improve the performance of transition rules by using continue attributes. This ACO–CA model is applied to the simulation of rural–urban land conversions in a fast-growing city, Guangzhou in China.

2. Ant colony optimization

ACO is based on ants' behaviours for finding the shortest path when seeking food without the benefit of visual information (Dorigo 1992). This intriguing ability of blind ants has been extensively studied by ethologists. They have discovered that, in order to exchange information about which path should be followed, ants communicate by means of a pheromone, unique to ants. Ants deposit the pheromone along their moving path, and the pheromone will be used to guide the movement of other ants. The amount of pheromone deposited will increase when the number of ants increases in selecting a certain route. This results in a higher probability for other ants to choose this route. In this way, ants are capable of finding the shortest route from their nests to food sources without using visual cues by exploiting pheromone information. This process can be described as a loop of positive feedback, in which the probability that an ant chooses a path is proportional to the number of ants that have already passed by that path (Dorigo 1992).

The above food-seeking process based on positive feedback information indicates that ACO is self-adaptive. The process of seeking food by an ant colony is illustrated by figure 1. If there are no obstacles between ant nests and food sources, the ants will walk in a straight line to the food (figure 1(*a*)). If an obstacle cuts off the straight path at location F (figure 1(*b*)), the ants have to decide whether to turn right or left at the cut off point. The first group of ants reaching point F (or H) has no

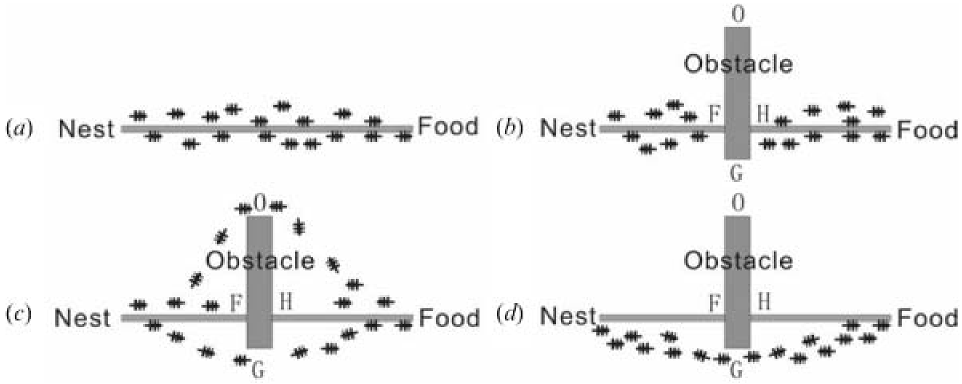


Figure 1. Route-choice behaviours of ants seeking food.

information about which way would lead to the shorter route, they have to make a random choice. It is expected that, on average, half of the ants turn left and the other half turn right (figure 1(*b*)). Because path F–G–H is shorter than F–O–H, the first half following path F–G–H reaches H before the other half following path F–O–H. Ants deposit pheromone on paths that they passed by. Intuitively, the amount of pheromone on path H–G–F is higher than that of path H–O–F. Therefore, an ant returning from food source to H finds a stronger trail on path H–G–F, which results in more ants preferring (in probability) path H–G–F to path H–O–F (figure 1(*c*)). Consequently, the number of ants following path F–G–H will be higher than the number of ants following path F–O–H. This causes the quantity of pheromone on the shorter path to grow faster than on the longer one, and ultimately all ants will choose the shorter path under this exploratory process (figure 1(*d*)).

3. ACO-based geographical CA

ACO is basically a multi-agent system in which low-level interactions between simple agents result in a complex behaviour of the whole ant colony (Dorigo 1992, Dorigo *et al.* 1996). ACO belongs to complex adaptive system with some plausible features, such as robustness, versatility, and nonlinearity. This allows ACO to solve complex and nonlinear problems through cooperation of ants (Bonabeau *et al.* 1999). The use of ACO for discovering classification rules, in the context of data mining, is a new research area with very limited studies. The method of ACO-based rule discovery was first proposed by Parpinelli *et al.* (2002). In this paper, a new method is proposed by using an Ant-Miner program to discover transition rules of CA.

This Ant-Miner program can discover optimized classification rules by simulating the behaviour of ants seeking foods. In the simulation, artificial arts can find the best links between attribute nodes and class nodes (figure 2). The attribute node can only be selected once and must be associated with a class node. Each route corresponds to a classification rule, and data-mining for a classification rule can be regarded as searching for the optimal route. The rule can be represented as follows:

$$\text{IF}\langle \textit{conditions} \rangle \text{THEN} \langle \textit{class} \rangle \tag{1}$$

where <*conditions*> contains a logical combination of predictor attributes, in the

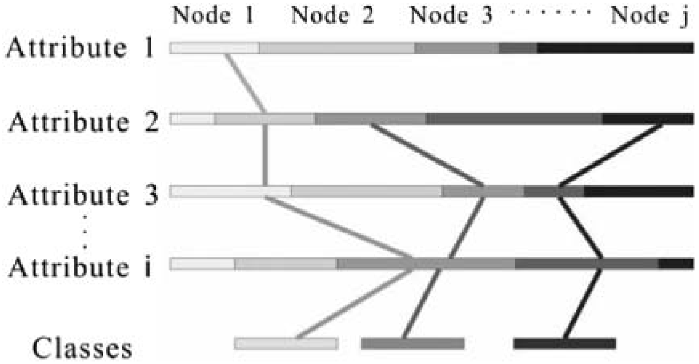


Figure 2. Route corresponding to classification rule derived from Ant-Miner.

form: *term1* AND *term2* AND …, where each item expressed as the triple <*attribute*, *operator*, *value*>, where *value* is a value belonging to the domain of *attribute*. The operator element in the triple is a relational operator. <*class*> contains the class predicted for cases whose predictor attributes satisfy <*conditions*>.

Each ant starts with an empty rule. The choice of a term to be added to the current rule depends on both heuristic function and the amount of pheromone associated with each term, which will be discussed in detail in section 3.2.

It should be noted that the original continuous values must be discretized in preprocessing. If the original values are V_1, V_2, …, V_n for the attributes of A_1, A_2, …, A_n, these values should be discretized as V_{11}, V_{12}, …, V_{21}, V_{22}, …, V_{nm}. The following sections will provide a detailed procedure for applying ACO to discover transition rules of CA.

3.1 *Discretization of data*

Ant-Miner is capable of finding better rules than other types of rule induction algorithm, such as decision trees (Parpinelli *et al.* 2002). However, Ant-Miner only copes directly with discrete data and cannot work with continuous data. Therefore, Ant-Miner should use a discretization process to deal with continuous attributes. Discretization is an effective technique in dealing with continuous attributes for rule generation (Su and Hsu 2005). This procedure can increase the speed and accuracy of machine learning (Liu and Wang 2005). In general, results obtained through decision trees or induction rules using discretized data are usually more efficient and accurate than those using continuous values (Liu *et al.* 2002).

Continuous values of spatial variables should be discretized for the discovery of rules by ACO. Selecting proper methods for this transformation is very important because it determines the overall quality for generating rules. In this paper, an entropy-based method is adopted to measure the importance of breakpoint for discretization of spatial variables (Xie *et al.* 2005).

A decision table is defined as a table of information comprising a four-element set (U, R, V, f), among which U refers to a set of objects, i.e. domain; $R=C\cup D$, C refers to a set of spatial variables, and D a Boolean variable indicating if an area is urbanized or not. V represents the value range of each spatial variable, and f is the information function.

If $X \subseteq U$ is considered as a training subset consisting of $|X|$ samples, among which k_j samples are noted with decision attribute $j(j=1, 2, ..., r)$, then the information entropy for the training set is (Theil 1967, Xie *et al.* 2005):

$$H(X) = -\sum_{j=1}^{r} P_j \log_2 P_j,\ P_j = \frac{k_j}{|X|} \tag{2}$$

where a small value of the entropy indicates that the set X is determined by several dominant values of decision attributes and a small degree of disorder. With c_i^a being the breakpoint i selected from spatial variable a, samples of decision attribute $j(j=1, 2, ..., r)$ belonging to the set X can be divided into two types: those with an attribute value smaller than c_i^a are recorded as $l_j^X\left(c_i^a\right)$, while those greater than c_i^a are recorded as $r_j^X\left(c_i^a\right)$, which can be represented as:

$$l^X\left(c_i^a\right) = \sum_{j}^{r} l_j^X\left(c_i^a\right) \tag{3}$$

$$r^X\left(c_i^a\right) = \sum_{j}^{r} r_j^X\left(c_i^a\right) \tag{4}$$

Consequently, the set X is divided into X_l and X_r, whose information entropies are respectively calculated as follows (Theil 1967, Xie *et al.* 2005):

$$H(X_l) = -\sum_{j=1}^{r} P_j \log_2 P_j,\ P_j = \frac{l_j^X\left(c_i^a\right)}{l^X\left(c_i^a\right)} \tag{5}$$

$$H(X_r) = -\sum_{j=1}^{r} q_j \log_2 q_j,\ q_j = \frac{r_j^X\left(c_i^a\right)}{r^X\left(c_i^a\right)} \tag{6}$$

Additionally, the information entropy of breakpoint c_i^a relative to the set X is defined as:

$$H^X\left(c_i^a\right) = \frac{|X_l|}{|U|} H(X_l) + \frac{|X_r|}{|U|} H(X_r) \tag{7}$$

Suppose $L=\{Y_1, Y_2, ..., Y_m\}$ is the equivalent samples derived from division of the set P with breakpoints selected from the decision table; then the new information entropy after addition of breakpoint $c \notin P$ becomes:

$$H(c, L) = H^{Y_1}(c) + H^{Y_2}(c) + \cdots + H^{Ym}(c) \tag{8}$$

where a small $H(c, L)$ indicates that the decision attribute value of the new equivalent subset divided tends to be more monotonous after adding the breakpoint, which will be more important.

If P is defined as the set of breakpoints, L as the set of equivalent samples divided by the breakpoint set P, B as the set of breakpoints to be selected, and H as the information entropy of the decision table, the process of discretizing continuous attribute values according to information entropy can be described as follows:

1. For each $c \in B$, calculate $H(c, L)$;
2. If $H \leqslant \min H(c, L)$, then end;
3. Select and add breakpoint $c_{\min}$ which can make $H(c, L)$ minimum into the set P;
4. For all $X \in L$, if equivalent X can be divided into X_1 and X_2 with $c_{\min}$, then, X can be removed from L, while the equivalent class X_1 and X_2 can be added into L;
5. If each equivalent samples among L shows the same decision, terminate the circulation. If not, return to step 1.

3.2 *Rule construction*

ACO is applied to the derivation of classification rules by using these discretized data. These rules are retrieved according to an approach similar to the collective process of seeking food by ants. The procedure selects terms repeatedly until a complete route is constructed.

A heuristic function is designed to guide the search so that the computation time is significantly reduced. For each term to be added to the current rule, Ant-Miner computes a heuristic function to estimate the quality of this term, with respect to its ability to improve the predictive accuracy of the rule. The information entropy is used to define this function, in which the heuristic value for each attribute node is proportional to its classification capability (Parpinell *et al.* 2002). In this paper, a heuristic function based on the statistical attribute of the data (frequency) is designed, in which the heuristic value η_{ij} of the condition term (term$_{ij}$) is defined as follows (Duan 2005):

$$\eta_{ij} = \frac{\max\left(\sum_n \text{freq}\,T_{ij}^1,\ \sum_n \text{freq}\,T_{ij}^2,\ \cdots \sum_n \text{freq}\,T_{ij}^k\right)}{\sum_n T_{ij}} \tag{9}$$

where η_{ij} is denoted as the density-based heuristic value of the condition term (term$_{ij}$), T_{ij} refers to the number of cases fitting to the condition term (term$_{ij}$), and freqT_{ij}^w is the frequency of class w in T_{ij}.

The record that satisfies the condition part of the rule should be removed after a final rule has been obtained. Therefore, the values for $\max\left(\sum_n \text{freq}\,T_{ij}^1,\ \sum_n \text{freq}\,T_{ij}^2,\ \cdots \sum_n \text{freq}\,T_{ij}^k\right)$ and $\sum_n T_{ij}$ is updated after a final rule has been found.

The other two parameters, the amount of pheromone and the probability for the attribute nodes to be selected, are also important to the generation of rules. The initial amount of pheromone deposited at each path position is inversely proportional to the number of values of all attributes, which is given by the following equation:

$$\tau_{ij}(t=0) = \frac{1}{\sum_{i=1}^{a} b_i} \tag{10}$$

where τ_{ij} is the amount of pheromone for the condition term (term$_{ij}$), a is the total number of attributes, and b_i is the number of possible values that can be taken on by attribute i.

The roulette wheel selection technique is adopted to decide which attribute node will be included for constructing a path according to the heuristic value (η_{ij}) and the thickness of pheromone (τ_{ij}). The probability of term$_{ij}$ being selected for inclusion in the current rule is defined as follows:

$$P_{ij}(t) = \frac{\tau_{ij}(t)\cdot\eta_{ij}(t)}{\sum_{i=1}^{a}\sum_{j=1}^{b_i}\tau_{ij}(t)\cdot\eta_{ij}(t)} \tag{11}$$

The selected attribute nodes will be continuously added to the route until all attributes (including class attributes) are selected to form a complete route (a classification rule).

The validity of this rule can be assessed by using the following equation (Parpinell *et al.* 2002):

$$Q = \left(\frac{\text{TruePos}}{\text{TruePos}+\text{FalseNeg}}\right)\cdot\left(\frac{\text{TrueNeg}}{\text{FalsePos}+\text{TrueNeg}}\right) \tag{12}$$

where TruePos (true positives) is the total number of cases covered by the rule that have the class predicted by the rule; FalsePos (false positives) is the total number of cases covered by the rule that have a class different from the class predicted by the rule; FalseNeg (false negatives) is the total number of cases that are not covered by the rule but that have the class predicted by the rule; TrueNeg (true negatives) is the total number of cases that are not covered by the rule and that do not have the class predicted by the rule. The larger the value of Q is, the higher the quality of the rule becomes.

3.3 *Rule pruning*

The next step is to prune the discovered rules for improving classification performance. Rule pruning is a typical technique in data mining. The goal of rule pruning is to remove irrelevant terms and improve the quality of rules, since a shorter rule is in general more comprehensible by users. Another motivation for rule pruning is to improve the predicative accuracy of rules. Furthermore, rule pruning prevents the rules from overfitting the training data (Brewlow and Aha 1997).

The basic idea of rule pruning is to remove one term from the rule once at a time if this removal can significantly improve the quality of the rule. One starts with the full rule consisting of all the terms then removes a term such that the overall quality of the rule is improved the most. It should be noted that this step might involve replacing the class in the consequent rule, since the majority class in the cases covered by the pruned rule can be different from the majority class in the cases covered by the original rule (Parpinell *et al.* 2002). In the next iteration, another term whose removal can most improve the quality of the rule is eliminated from the rule. The removal process was repeated until there was only one term left in the rule antecedent or no increase in rule quality was observed. The rule quality is defined by equation (11).

3.4 *Pheromone updating*

At the start of Ant-Miner, the pheromones of all terms are given the same value based on equation (10). The amount of pheromone at each node of all paths will be updated after a rule has been accepted. The amount of pheromone associated with each term will increase if this term is included in a rule, but it will decrease if this term is excluded from this rule. Therefore, the evaporation coefficient ρ is used to represent this process. The amount of pheromone at each attribute node is updated according to the following formula:

$$\tau_{ij}(t+1)=(1-\rho)\cdot\tau_{ij}(t)+\Delta\tau_{ij}(t) \tag{13}$$

$$\Delta\tau_{ij}(t)=\sum_k \Delta\tau_{ij}^k(t) \tag{14}$$

$$\Delta\tau_{ij}^k(t)=\begin{cases}\frac{Q_k}{1+Q_k} & \text{if ant } k \text{ passes over node term}_{ij}\\ 0 & \text{else}\end{cases} \tag{15}$$

where ρ is the pheromone evaporation coefficient, Q is the quality of a classification rule, and $\Delta\tau_{ij}^k(t)$ is the pheromone amount remaining on the node term$_{ij}$ by ant k.

When the amount of pheromone for all term nodes has been updated, the next ant starts a new round of search. This search becomes convergent when the majority of ants locate the same route for seeking food. The iteration continues until all ants complete their search. At each iteration, these ants may construct many rules, but only the rule of the best quality is preserved, and the others are discarded. This process is repeated until the number of remaining training classes is less than the predefined number of cases.

The detailed procedures of discovering transition rules of CA by using Ant-Miner are as follows:

1. obtaining the discretized values for the spatial variables;
2. starting from an empty route for an ant, adding nodes to this route to find a complete route according to the amount of pheromone at each node;
3. when an ant passes a route, it releases an amount of pheromone at the nodes according to the travel time (cost). the amount of pheromone will affect the probability of selecting this route by other ants;
4. pruning redundant rules;
5. updating the amount of pheromones on all the nodes of each route—this provides feedback for the next round of search;
6. go to step 2 until all the ants have been examined in selecting routes;
7. choose the best rule as a final rule by evaluating all the rules constructed by these ants;
8. remove the set of cases correctly covered by the final rule discovered by the step 7;
9. go to step 2 until the number of remaining cases is less than a threshold value.

3.5 *Discovering transition rules of CA by using ACO*

One main characteristic of ACO is the indirect communication by pheromone laying also known as stigmergy, and another characteristic is the positive feedback mechanism which accounts for rapid discovery of good solutions. As a result, ACO can be used to solve nonlinear problems effectively (Dorigo 1996). Especially, ACO have potentials for mining transition rules of CA efficiently from large spatial datasets because of the use of the 'bottom-up' approach. In this paper, transition rules of CA are automatically derived from training data set using Ant-Miner. The structure of the ACO-based geographical CA model is shown as figure 3.

First, land-use data were acquired through classification of Landsat TM images. Then, a GIS package, ARCGIS, was used to derive a series of spatial variables that are related to land development. For example, the distance variables were calculated by using the *Eucdistance* function of ArcGIS. The number of developed cells in the 3 × 3 neighbourhood was counted using the *Focal* function of ArcGIS. The agricultural suitability was calculated using the *raster calculator* function of ArcGIS. The future state (land-use types) at $T+1$ is determined by the existing state at T and a number of spatial variables. The ACO–CA model consists of two parts: discovery of transition rules using empirical data and the simulation based on these transition rules. Remote sensing data will be utilized to monitor the growth of the city, and transition rules are mainly discovered using Ant-Miner, which is implemented through Visual Basic 6.0 programming. The pseudo-code for discovering transition rules of CA is as follows:

```
    ALGORITHM: A High-level Description of the Ant-Miner for discovering transition rules of CA.
      The original trainingSet
      Discretization of the original TrainingSet
      DiscoveredRuleList=[]/* rule list is initialized with an emptylist */
      WHILE (TrainingSet>Max_uncovered_cases)
  Initialize all nodes with the same amount of pheromone
        calculation the $\eta_{ij}$ of the training data for all nodes
      i=1/* ant index */
  WHILE(i<No_of_ants and m<No_rules_converg)
        FOR j=1 to No_of_attributes
        Select a node of the attribute
              NEXT j
              Obtaining Rule$_i$
```

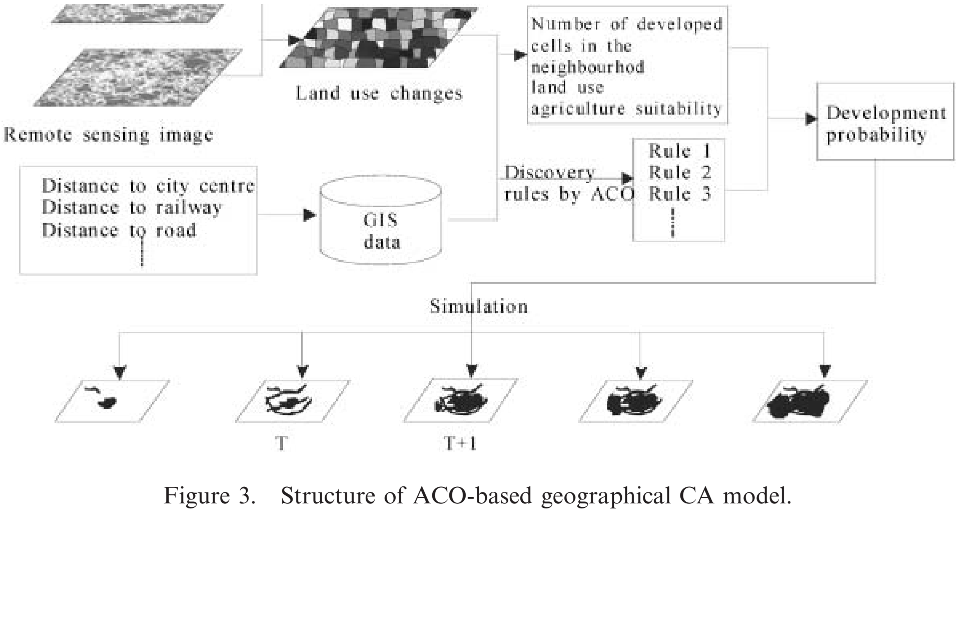


Figure 3. Structure of ACO-based geographical CA model.

```
                rules pruning
                IF (Rule$_i$ is equal to Rule$_{i-1}$ THEN
                  m=m+1
                ELSE
                  m=1
                END IF
                pheromone update
                i=i+1
          LOOP
            Select the best rule R$_{best}$ among all rules constructed by
all the ants;
            Add rule R$_{best}$ to DiscoveredRuleList
  TrainingSet=TrainingSet-{set of cases covered by R$_{best}$}
    LOOP
```

Unlike equation-based methods, the derived transition rules are easy for understanding because of using explicit forms. The following is an example of these explicit transition rules:

- Rule 1:
IF Distance to urban centres $<$8 km,
Distance to trunk roads $<$0.5 km,
The number of developed cells in the neighbourhood $>$5,
Land-use types=farmland.
THEN Development is allowed (confidence=0.98).

- Rule 2:
IF Distance to urban centres $>$50 km,
Number of developed cells in the neighbourhood $<$2,
Land-use types=forestland.
THEN Development is prohibited (confidence=0.90).

The simulation program is developed by integration of ArcObjects with Visual Basic 6.0. During the simulation, the state conversion of a central cell is determined by these transition rules. However, it is noted that the observation interval (ΔT) between remote sensing images is generally far greater than the iteration interval (Δt) of CA simulation. It may be ideal if the observation interval (ΔT) is equal or close to the iteration interval (Δt) so that derived transition rules can be used directly in urban simulation (Li and Yeh 2004). As a result, it is necessary to determine the amount of land-use conversion in the iteration interval (Δt) in the CA model.

The number of iterations (K) of CA model during the period of iteration is represented as follows:

$$K = \Delta T / \Delta t \tag{16}$$

The amount of land-use conversion (ΔQ_0) is determined from remote sensing for the larger observation interval (ΔT). If $\Delta T > \Delta t$, only a portion of land-use conversion took place in the iterationΔt. The amount of land-use conversion between t and $t+1$ is calculated as follows:

$$\Delta q_0 = \Delta Q_0 / K \tag{17}$$

where Δq_0 is the amount of land-use conversion for the iteration interval Δt.

Urban evolution is affected by some degree of uncertainties. As a common practice in urban simulation, a random variable is often incorporated to generate realistic patterns (White and Engelen 1993, Wu and Webster 1998). There is a concern if such probabilistic element can affect simulation accuracies. Fortunately, stable patterns can be generated using these probabilistic CA models because the uncertain parts are in small proportions, usually located in the fringe of urbanized clusters (Yeh and Li 2006). In this paper, a random variable (γ) is then used to determine the locations of land-use conversion at a smaller interval. The following additional rule is used to obtain the smaller portion of land use conversion:

IF $x(i, j)$ should be converted according to the original transition rules, and $x(i, j)$ have not developed at $t-1$ and $\gamma \leqslant \beta_0$ THEN $x(i, j)$ will be developed at t

$$\beta_0 = \frac{\Delta q_0}{\Delta Q_0} = \frac{1}{K} \tag{18}$$

where $x(i, j)$ is the cell at location (i, j), ΔQ_0 is the amount of land-use conversion retrieved from the two images, and Δq_0 is the amount of land-use conversion for its iteration interval.

4. Application and simulation results

4.1 *Study area and spatial data*

The ACO-based CA model was applied to the simulation of urban development of a fast-growing city, Guangzhou, in the Pearl River Delta of China. TM satellite images in 1988, 1993, and 2002 are used to provide actual urban areas, which are divided into the training data (TM data in 1988 and 1993) and the test data (TM data in 2002). The training data are used to calibrate the model for obtaining transition rules, and the test data are used to confirm the predictability of the calibrated model. A series of spatial variables were chosen for the simulation of urban development. They include various distance-based variables, neighbourhood functions, and physical properties (table 1). Studies have shown that these variables are closely related to urban development and land-use changes (White and Engelen 1993, Wu and Webster 1998). Therefore, the probability of urban development can be estimated using these spatial variables. Some physical constraints can be incorporated in the model by using GIS data. For example, slope and land-use type are physical constraints for land development. There are usually low development probabilities in the areas of rivers, steep slopes, and agricultural protection zones. These constraints play an important role in the estimation of development probability (White *et al.* 1997, Li and Yeh 2000). These spatial variables and constraints were derived from remote sensing and GIS data. These spatial variables

Table 1. Spatial variables required for derivation of transition rules using Ant-Miner.

Distance-based variables						Neighbourhood	Physical properties	
Distance to city proper	Distance to town centres	Distance to national highways	Distance to roads	Distance to railways	Distance to expressways	Number of developed cells in the neighbourhood	Slope	Land-use type
(PropD)	(TownD)	(NatD)	(RoadD)	(RailD)	(ExprD)	(Nsum)		

were automatically discretized into a number of classes ranging from 5 to 10 based on equations (2)–(8).

Stratified random sampling method is used to extract the samples from the training data for deriving transition rules (Li and Yeh 2002). A total of 6500 samples are randomly selected in the remote sensing images. The sample data set is further divided into two groups—3500 as the training data set, and 3000 as the test data set. The total amount of urban areas classified from these satellite images is used as the global constraint for urban simulation.

4.2 *Data-mining of transition rules and urban simulation*

The spatial variables are treated as the attribute nodes of ant route, and the cell state is treated as the class node of ant route. Prior to rule discovery, continuous spatial variables should be discretized using Ant-Miner. Ant-Miner requires the users to provide the following four parameters:

1. No_of_ants (number of ants): this is the maximum number of candidate rules constructed during an iteration.
2. Min_cases_per_rule (minimum number of cases per rule): the minimum number of cases that each rule must cover, which helps avoid overfitting the training data.
3. Max_uncovered_cases (maximum number of uncovered cases in the training set): the process of discovering rules is iteratively performed until the number of uncovered cases is smaller than this threshold.
4. Max_iterations (maximum number of iterations): the program stops when the number of iterations is more than this threshold.

The default parameter settings for the Ant-Miner are as follows: No_of_ants=220, Min_cases_per_rule=8, Max_uncovered_cases=50, and Max_iterations=200. The experiment indicates, among these four parameters, that the number of ants (No_of_ants) and the minimum number of case per rule (Min_cases_per_rule) are the two most sensitive factors in determining classification results. The sensitivity of these two parameters is shown in figures 4 and 5. Classification accuracies improve with the increase in No_of_ants. This

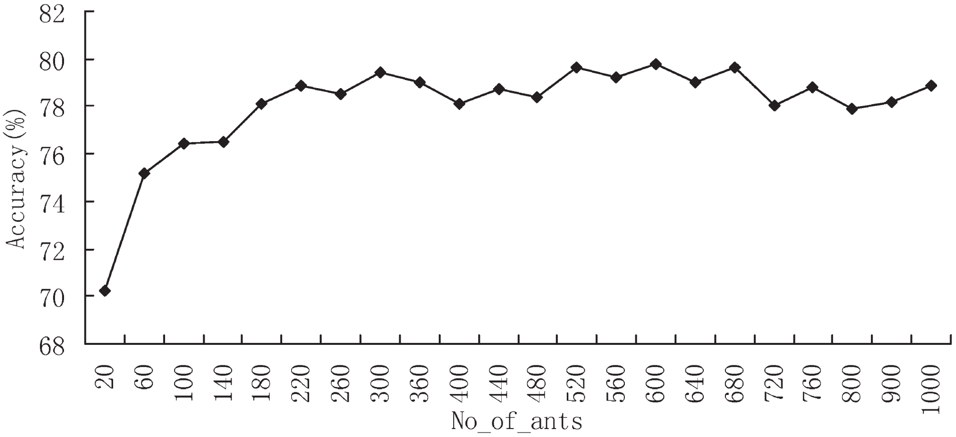


Figure 4. Influence of No_of_ants on the performance of Ant-Miner.

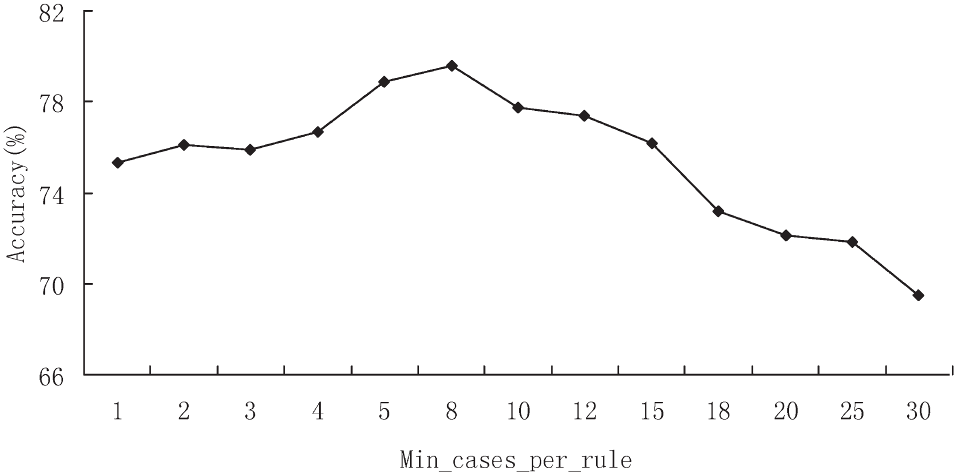


Figure 5. Influence of Min_cases_per_rule on the performance of Ant-Miner.

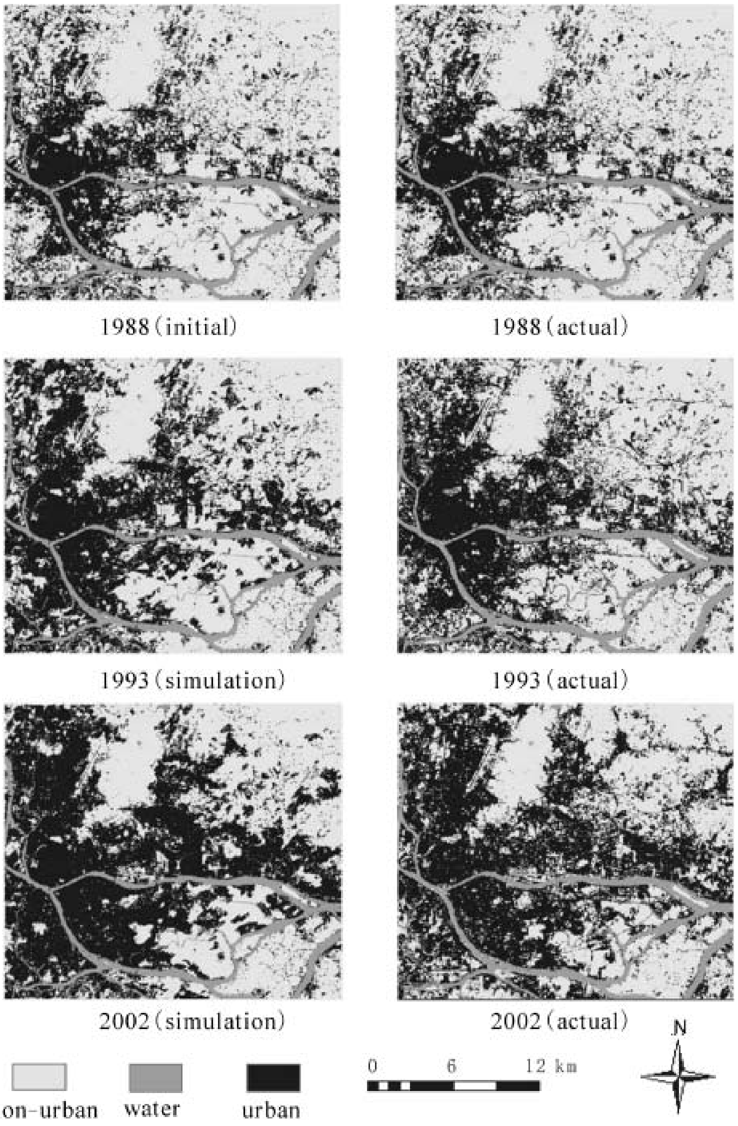
improvement stabilizes after No_of_ants reaches 220 (figure 4). The classification accuracy is obtained by comparing the predicted class obtained from the rules and the actual class from remote sensing. As shown in figure 5, the classification accuracy of Ant-Miner improves when Min_cases_per_rule increases from 1 to 8, but the predictive accuracy decreases after Min_cases_per_rule is greater than 8.

In this study, a total of 78 rules are generated from the training data by Ant-Miner. It takes 4 minutes to complete the rule induction. A selected set of the transition rules is listed in table 2.

The simulation of urban dynamics of Guangzhou during the period of 1988–1993 and 1993–2002 was implemented using the transition rules obtained from Ant-Miner. The simulation starts from the initial land use, which is obtained by the classification of TM data in 1988. The land use in 1993 and 2002 is then simulated by running this model with 200 iterations and 400 iterations, respectively (figure 6).

Table 2. Part of the transition rules derived by using Ant-Miner.

Rule1:
IF
RoadD <=3 km and ExprD <=5 km and 18 km< TownD <=21 km and land use='cropland' and Nsum >=3
THEN
Converted to urban development (confidence=0.92)
Rule 2:
IF
NatD <=10 km and RoadD <=3 km and 5 km< ExprD <=20 km and 15 km< RailD <=18 km and TownD <=26 km and land use='orchard' and Nsum >=4
THEN
Converted to urban development (confidence=0.86)
Rule 3:
IF
41 km< PropD <=45 km and 15 km< RoadD <=18 km and RailD >23 km and TownD >24 km and Nsum <=2
THEN
Not converted to urban development (confidence=0.83)



1988 (initial)

1988 (actual)

1993 (simulation)

1993 (actual)

2002 (simulation)

2002 (actual)

Figure 6. Simulated and actual urban development of Guangzhou in 1988, 1993, and 2002.

5. Model validation and comparison

Validation is usually required if urban CA are applied to the simulation of real cities (Li and Liu 2006). A simple method to assess the goodness-of-fit is to compare the simulated patterns with the actual ones visually for validating CA (Clarke *et al.* 1997, White *et al.* 1997, Ward *et al.* 2000). The visual comparison indicates that the simulated patterns fit well with the actual patterns classified from remote sensing images (figure 6).

Visual comparison is a rather preliminary method for validating the simulation accuracy. A further quantitative analysis is to produce a confusion matrix about the concordance between the simulated development and the actual development. This is based on the spatial overlay of these two patterns cell by cell. The simulation of

Table 3. Simulation accuracies of the ACO-based CA for Guangzhou.

	1988–1993 (cells)		
	Simulated 1993 non-urban	Simulated 1993 urban	Accuracy
Actual 1993 non-urban	107 347	16 618	86.6%
Actual 1993 urban	16 150	55 821	77.6%
Total accuracy			83.3%
Kappa coefficient			0.64
	1988–2002 (cells)		
	Simulated 2002 non-urban	Simulated 2002 urban	Accuracy
Actual 2002 non-urban	76 815	23 293	76.7%
Actual 2002 urban	22 253	73 575	76.8%
Total accuracy			76.8%
Kappa coefficient			0.53

the 1993 patterns can be compared with the actual 1993 patterns. This comparison is a common practice for validating CA models. For example, Wu (2002) used a logistic regression to calibrate development probability using the empirical land-use data of 1973 and 1993. In the analysis of model performance, the land use of 1993 generated by simulation was compared with the actual land use of 1993.

Table 3 lists the comparison of these two patterns in 1993 and 2002 for this ACO-based CA model. The total accuracies are computed from cross-tabulation, which compares the goodness of fit on a cell-by-cell basis. The total accuracies are 83.3% and 76.8%, and the kappa coefficients are 0.64 and 0.53 for the simulation of urban development in 1993 and 2002 respectively. The simulated results in 1993 have a better accuracy because the model is calibrated using the transition from 1988 to 1993 (table 3).

Structural conformity is also important in the assessment of simulation results (Wu 2002). The indicator of Moran I can be used as the spatial statistics for measuring land-use patterns. Moran I is a useful spatial indicator that can reveal the degree of spatial autocorrelation (Goodchild 1986). The indicator is able to estimate how close the simulated land-use pattern is to the actual urban development (Wu 2002). Table 4 shows the structural conformity using the indicator of Moran I. The Moran I values are 0.627 and 0.687 for the simulation of land development in 1993 and 2002, respectively, using this ACO–CA model. They are 0.626 and 0.684 for the actual land development in 1993 and 2002, respectively. This indicates that there is a good conformity between the simulated and actual land development according to the measurement of using Moran I.

The indicator of Moran I can only provide the aggregate information about urban morphology. More characteristics of urban morphology should be measured for the validation, including connectivity, shape, and fragmentation. Landscape

Table 4. Assessment of the goodness of fit for the CA model using the Moran I index.

Time	1988	1993	2002
Actual	0.633	0.626	0.684
Simulated (ACO)	0.633	0.627	0.687
Simulated (See5.0)	0.633	0.621	0.680

indices should be able to describe the characteristics of spatial patterns and provide useful insights about urban morphology. We selected a set of metrics for measuring the entire landscape, including number of patches (NP), the largest patch index (LPI), edge density (ED), landscape shape index (LSI), and contagion (a measure of landscape configuration). These metrics were computed from the rasterized land-use maps using the Fragstats software package (McGarigal and Marks 1995). These spatial pattern metrics capture ecologically relevant aspects of spatial pattern such as fragmentation (NP, LPI, ED, and contagion), patch shape (LSI), and amount of edges between contrasting patch types (ED and contagion) (Jenerette and Wu 2001).

The validation is carried out by examining the differences between the simulated patterns and the actual ones classified from remote sensing (table 5). The comparison indicates that there is a good conformity between the simulated and the actual because each of these indices yields very close values for these two patterns. For example, the differences are only 5.2% and 1.7% for the NP and LPI, respectively.

This bottom-up approach is compared with a rule-based approach, See5.0 decision tree, which is a top-down algorithm. The See5.0 system is based on the 'information gain ratio' to determine the splits at each internal node of the decision tree (Quinlan 1993). Transition rules derived from the See5.0 decision tree model are used to simulate the urban development in the study area. The accuracy of this simulation is calculated based on the cell-by-cell comparison (table 6). It shows that the total accuracies of this simulation are 81.5% and 73.2%, and the kappa coefficients are 0.60 and 0.46 for the simulation of urban development in 1993 and 2002, respectively. Comparison of tables 3 and 6 indicates that the ACO-based CA model has a better simulation performance than the See5.0-based CA model. As shown in table 4, The Moran I values are 0.621 and 0.680 for the simulation of land development in 1993 and 2002, respectively, for the See5.0 model. The See5.0 model has larger differences for the Moran I values than the ACO model. Therefore, the proposed model shows a better performance in structural conformity than the See5.0 model. Table 5 also shows that the ACO model also has better conformity between the simulated and actual patterns in terms of landscape structure than the See5.0 model. This could be attributed to the fact that the See 5.0 model gains intelligence through heuristic evaluations of entropy at each decision point, but their evaluations are typically static in nature. ACO, on the other hand, updates

Table 5. Landscape indices of the simulated patterns of CA compared with those of the actual patterns derived from the TM data.

	NP	LPI	ED	LSI	Contagion
1988~1993					
Actual	3335	55.74	62.14	36.57	19.98
Simulated (ACO)	3160	56.72	59.39	33.72	21.44
Simulated (See5.0)	3084	57.13	58.42	32.47	21.85
% deviation (ACO)	*5.2*	*1.7*	*4.4*	*7.8*	*7.3*
% deviation (See5.0)	*7.5*	*2.4*	*6.0*	*11.2*	*9.4*
1993~2002					
Actual	3119	43.88	61.36	34.99	19.26
Simulated (ACO)	2921	45.53	56.85	29.56	21.86
Simulated (See5.0)	2875	46.12	56.41	29.03	22.14
% deviation (ACO)	*6.3*	*3.8*	*7.4*	*15.5*	*13.5*
% deviation (See5.0)	*7.8*	*5.1*	*8.0*	*17.0*	*14.9*

Table 6. Simulation accuracies of the decision-tree CA model for Guangzhou.

	1988–1993 (cells)		
	Simulated 1993 non-urban	Simulated 1993 urban	Accuracy
Actual 1993 non-urban	105 893	18 072	85.4%
Actual 1993 urban	18 100	53 871	74.9%
Total accuracy			81.5%
Kappa coefficient			0.60
	1988–2002 (cells)		
	Simulated 2002 non-urban	Simulated 2002 urban	Accuracy
Actual 2002 non-urban	73 692	26 416	73.6%
Actual 2002 urban	26 079	69 749	72.8%
Total accuracy			73.2%
Kappa coefficient			0.46

pheromone in time by using the bottom-up approach. The continuously updated pheromone guides dynamic heuristic evaluations, thus minimizing the likelihood of being trapped in local optima.

Recently, Pontius and Malanson (2005) pointed out that a predictive model should be compared with a Null model of pure persistence (no change) for model validation. A Null model is a kind of model that predicts nothing, as nothing would change. The baseline is that a predictive model should have better performances than a null model (Pontius and Malanson 2005). For instance, if urban land use changes 15% in a period, a Null model of pure persistence would be 85% correct based on the standard overall accuracy, while the overall accuracy of a predictive model should be higher than 85%. However, the overall accuracy has a bias because of the difference between the actual agreement and chance agreement (Congalton 1991, Li and Liu 2006), which can be effectively explained with Kappa coefficient. Kappa coefficient is calculated as follows (Congalton 1991):

$$\text{Kappa} = \frac{M \sum_{i=1}^{r} x_{ii} - \sum_{i=1}^{r} (x_{i+} \cdot x_{+i})}{M^2 - \sum_{i=1}^{r} (x_{i+} \cdot x_{+i})}, \tag{19}$$

where x_{ii} are the elements on the main diagonal of the error matrix, x_{i+} is the sum of the ith row of the error matrix, and x_{+i} is the sum of the ith column of the error matrix.

More meaningful results are produced by comparing the Kappa coefficient between the Null model and a predictive model (table 7). As shown in tables 3 and 7, in the period 1988–1993 the total accuracy of the ACO–CA model is only 1.1% higher than that of the Null model, but the Kappa coefficient is 5% higher than that of the Null model; in the period 1988–2002, the total accuracy of ACO–CA model is 9% higher than that of the Null model, while the Kappa coefficient is 18% higher than that of the Null model. These results indicate that the ACO–CA model is rather more accurate in simulating urban development.

Figure 7 further displays the spatial distribution of agreement and disagreement between the simulated development and the actual in 1993–2002. Greyish blue,

Table 7. Accuracies of the null model of pure persistence for Guangzhou.

	1988–1993 (cells)		
	1988 non-urban (no change)	1988 urban (no change)	Accuracy
Actual 1993 non-urban	115 240	8725	93.0%
Actual 1993 urban	26 065	45 906	63.8%
Total accuracy			82.2%
Kappa coefficient			0.59
	1988–2002 (cells)		
	1988 non-urban (no change)	1988 urban (no change)	Accuracy
Actual 2002 non-urban	789 193	10 915	89.1%
Actual 2002 urban	252 112	43 716	45.6%
Total accuracy			67.8%
Kappa coefficient			0.35

black, and red colours are used to indicate these three correct simulated categories, ‘non-urban to non-urban’, ‘urban to urban’, and ‘non-urban to urban’, from the simulation model, respectively. White, green, and yellow colours are used to represent these three incorrect simulated categories, ‘non-urban to urban’, ‘non-urban to non-urban’, and ‘urban to non-urban’, from the simulation model, respectively. Greyish blue, black, and green colours are used to indicate these three

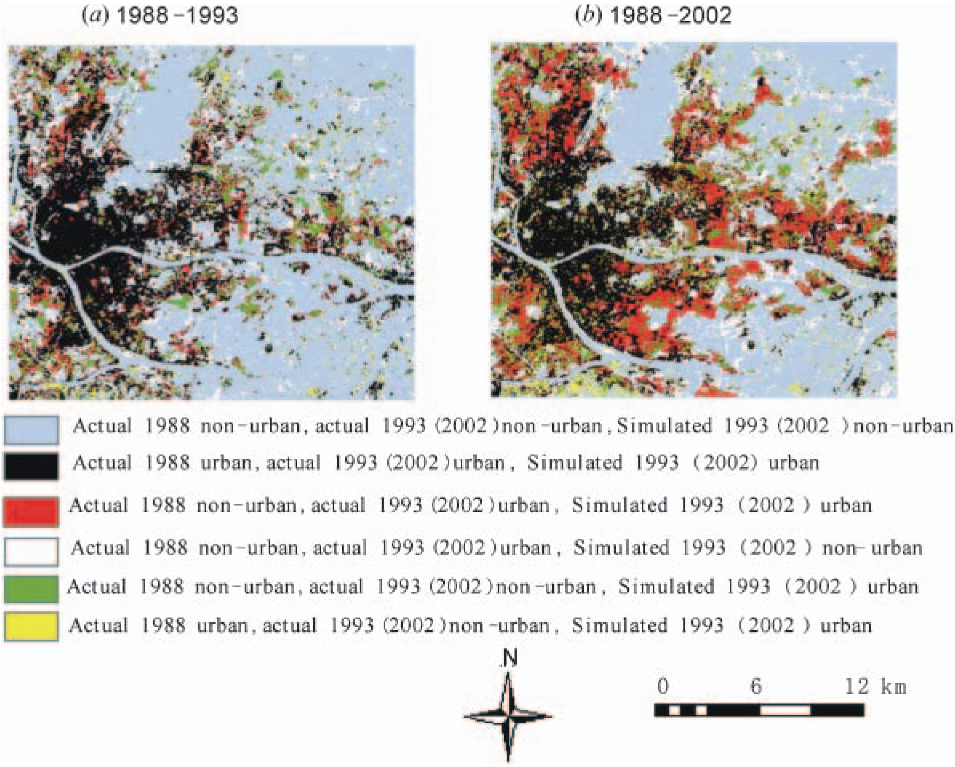

(a) 1988 -1993
(b) 1988 -2002
🟦 Actual 1988 non-urban, actual 1993 (2002)non-urban, Simulated 1993 (2002)non-urban
⬛ Actual 1988 urban, actual 1993 (2002)urban, Simulated 1993 (2002) urban
🟥 Actual 1988 non-urban, actual 1993 (2002)urban, Simulated 1993 (2002) urban
⬜ Actual 1988 non-urban, actual 1993 (2002)urban, Simulated 1993 (2002) non- urban
🟩 Actual 1988 non-urban, actual 1993 (2002)non-urban, Simulated 1993 (2002) urban
🟨 Actual 1988 urban, actual 1993 (2002)non-urban, Simulated 1993 (2002) urban
N
0 6 12 km

Figure 7. Distribution of agreement and disagreement of the simulated patterns or urban development of Guangzhou in 1993 and 2002.

correct categories, ‘non-urban to non-urban’, ‘urban to urban’, and ‘non-urban to non-urban’, from the Null model, respectively. The ACO–CA model has a higher percentage correct than the null model because there is more red than green.

6. Conclusions

CA models have become a useful tool for exploring the evolutionary processes of complex systems with a wide range of applications. Simulation of complex urban systems is not only theoretically important but also of great practical value for urban planning. The use of explicit transition rules can provide a useful insight for understanding urban evolution. Mathematical equation-based transition rules are not convenient to calibrate their parameters and interpret their meanings for large complex regions.

This paper presents a new method of using ACO for acquiring explicit transition rules of geographical cellular automata (CA), based on Parpinelli *et al.*’s (2002) work. ACO is actually a complex multi-agent system, composed of a large number of artificial ants with simple intelligence. Complex tasks, such as finding an optimal route for seeking food, can be effectively fulfilled by the mutual cooperation between ants. Our study has successfully applied ACO to construct CA transition rules expressed as logical statements. Compared with Parpinelli *et al.*’s (2002) work, our paper contributes three aspects:

1. A discretization technique is used to slice continuous spatial variables into a number of intervals so that ACO can effectively discover CA transition rules hidden in large datasets.
2. Parpinell *et al.* (2002) use information entropy to define the local heuristic function. In this paper, a heuristic function based on the statistical attribute of the data (frequency) is designed.
3. Satisfactory results have been obtained in solving travelling salesman problems (TSP), data clustering, combinatorial optimization, and network routing by using ACO algorithms. However, applying ACO to geo-simulation has not been reported. Since ACO has strong swarm intelligence, it is appropriate to ACO to discover transition rules of CA.

The ACO–CA model has been applied to the simulation of urban dynamics of Guangzhou. Remote sensing images of different years are used as the observation data for model calibration and validation. The stratified random sampling method is used to extract a training data set, which is used by ACO to mine CA transition rules automatically. The urban development of Guangzhou in the period of 1993–2002 is simulated using this model. The comparison between the ACO method and the See5.0 decision-tree method indicates that the ACO–CA model yields a higher accuracy in simulating urban development. In the period 1988–1993, the total accuracy of the ACO–CA model is 1.5% higher than that of the decision tree CA model, and the Kappa coefficient is 4% higher than that of the decision tree CA model. In the period 1993–2002, the total accuracy of ACO–CA model is 3.6% higher than that of the decision tree CA model, and the Kappa coefficient is 7% higher than that of the decision tree CA model. Furthermore, the ACO-based CA model also has better conformity between the simulated and actual patterns in terms of landscape structure than the decision tree CA model. With its strongly robustness, self-adaptation, and positive-feedback mechanism, ACO is a powerful method that has great potential in discovering reliable transition rules for modelling complex geographical phenomena.

Acknowledgements

This study was supported by the National Outstanding Youth Foundation of China (Grant No. 40525002), the National Natural Science Foundation of China (Grant No. 40471105), and the Hi-tech Research and Development Program of China (863 Program) (Grant No. 2006AA12Z206).

References

ALATAS, B. and AKIN, E., 2005, fcaco: fuzzy classification rules mining algorithm with ant colony optimization. *Lecture Notes in Computer Science*, **3612**, pp. 789–797.

BATTY, M. and XIE, Y., 1994, From cells to cities. *Environment and Planning B: Planning and Design*, **21**, pp. 531–548.

BONABEAU, E., DORIGO, M. and THERAULAZ, G., 1999, *Swarm Intelligence: From Natural to Artificial Systems* (New York: Oxford University Press).

BREWLOW, L.A. and AHA, D.W., 1997, Simplifying decision trees: a survey. *The Knowledge Engineering Review*, **12**, pp. 1–40.

CLARKE, K.C., BRASS, J.A. and RIGGAN, P.J., 1994, A cellular automata model of wildfire propagation and extinction. *Photogrammetric Engineering & Remote Sensing*, **60**, pp. 1355–1367.

CLARKE, K.C. and GAYDOS, L.J., 1998, Loose-coupling a cellular automata model and GIS: long-term urban growth prediction for San Francisco and Washington/Baltimore. *International Journal of Geographical Information Science*, **12**, pp. 699–714.

CLARKE, K.C., HOPPEN, S. and GAYDOS, L.J., 1997, A self-modifying cellular automaton model of historical urbanization in the San Francisco Bay area. *Environment and Planning B: Planning and Design*, **24**, pp. 247–261.

COLORNI, A., DORIGO, M. and MANIEZZO, V., et al. 1991, Distributed optimization by ant colonies. In *Proceedings of the 1st European Conference on Artificial Life*, pp. 134–142.

CONGALTON, R.G., 1991, A review of assessing the accuracy of classification of remotely sensed data. *Remote Sensing of Environment*, **37**, pp. 35–46.

COUCLELIS, H., 1985, Cellular worlds: a framework for modeling micro–macro dynamics. *Environment and Planning A*, **17**, pp. 585–596.

COUCLELIS, H., 1997, From cellular automata to urban models: New principles for model development and implementation. *Environment and Planning B: Planning and Design*, **24**, pp. 165–174.

COUCLELIS, H., 1989, Macrostructure and Microbehavior in a MetropolitanArea. *Environment and Planning B: Planning and Design*, **16**, pp. 141–154.

COUCLELIS, H., 1988, Of mice and men: what rodent population scan teach us about complex spatial dynamics. *Environment and Planning A*, **20**, pp. 99–109.

DORIGO, M., 1992, Optimization, learning and natural algorithms. PhD thesis, Politecnico di Milano.

DORIGO, M., MANIEZZO, V. and COLORNI, A., 1996, Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics—Part B*, **26**, pp. 29–41.

DUAN, H.B., 2005, *Ant Colony Algorithms: Theory and Applications* (Beijing: Science Press).

GOODCHILD, M., 1986, *Spatial Autocorrelation, CATMOG, 47* (Norwich, UK: GeoBooks).

HARGROVE, W.W., GARDNER, R.H., TURNER, M.G., ROMME, W.H. and DESPAIN, D.G., 2000, Simulating fire patterns in heterogeneous landscapes. *Ecological Modelling*, **135**, pp. 243–263.

JENERETTE, G.D. and WU, J.G., 2001, Analysis and simulation of land-use change in the central Arizona—Phoenix region, USA. *Landscape Ecology*, **16**, pp. 611–626.

KWANG, M.S. and WENG, H.S., 2002, Multiple ant-colony optimization for network routing. In *Proceedings of the First International Symposium on Cyber Worlds*, pp. 277–281.

LETT, C., SILBER, C. and BARRET, N., 1999, Comparison of a cellular automata network and an individual-based model for the simulation of forest dynamics. *Ecological Modelling*, **121**, pp. 277–293.

LI, R.P. and WANG, Z.O., 2002, An entropy-based discretization method for classification rules with inconsistency checking. In *Proceedings of the First International Conference on Machine Learning Cybernetics*, pp. 243–246.

LI, X. and LIU, X.P., 2006, An extended cellular automaton using case-based reasoning for simulating urban. *International Journal of Geographical Information Science*, **20**, pp. 1109–1136.

LI, X. and YEH, A.G.O., 2000, Modelling sustainable urban development by the integration of constrained cellular automata and GIS. *International Journal of Geographical Information Science*, **14**, pp. 131–152.

LI, X. and YEH, A.G.O., 2002, Neural-network-based cellular automata for simulating multiple land use changes using GIS. *International Journal of Geographical Information Science*, **16**, pp. 323–343.

LI, X. and YEH, A.G.O., 2004, Data mining of cellular automata's transition rules. *International Journal of Geographical Information Science*, **18**, pp. 723–744.

LIU, H., HUSSAIN, F., TAN, C.L. and DASH, M., 2002, Discretization: An Enabling Technique. *Data Mining and Knowledge Discovery*, **6**, pp. 393–423.

LIU, X. and WANG, H., 2005, A discretization algorithm based on a heterogeneity criterion. *IEEE Transaction on Knowledge and Data Engineer*, **17**, pp. 1166–41173.

LIU, X.P. and LI, X., 2006, Retrieving CA nonlinear transition rule from high-dimensional feature space. *Acta Geographical Sinica*, **61**, pp. 663–672 (in Chinese).

LUMBER, E. and FAIETA, B., 1994, Diversity and adaptation in populations of clustering ants. In *Proceedings of the Third International Conference on Simulation of Adaptive Behavior: from Animals to Animates*, pp. 501–508 (Cambridge, MA: MIT Press/ Bradford Books).

MCGARIGAL, K. and MARKS, B., 1995, Fragstats: spatial pattern analysis program for quantifying landscape structure. USDA Forest Service–General Technical Report PNW-GTR-361.

PARPINELL, R.S., LOPES, H.S. and FREITAS, A.A., 2002, Data mining with an ant colony optimization algorithm. *IEEE Transaction on Evolutionary Computation*, **6**, pp. 321–332.

PONTIUS, G.R. and MALANSON, J., 2005, Comparison of the structure and accuracy of two land change models. *International Journal of Geographical Information Science*, **19**, pp. 243–265.

QUINLAN, J.R., 1993, C4.5: *Programs for Machine Learning* (Dordrecht, Netherlands: Kluwer Academic).

SIRAKOULIS, G.C., KARAFYLLIDIS, I. and THANAILAKIS, A., 2000, A cellular automaton model for the effects of population movement and vaccination on epidemic propagation. *Ecological Modelling*, **133**, pp. 209–223.

SOARES-FILHO, B.S., CERQUEIRA, G.C. and PENNACHIN, C.L., 2002, DINAMICA—a stochastic cellular automata model designed to simulate the landscape dynamics in an Amazonian colonization frontier. *Ecological Modelling*, **154**, pp. 217–235.

SU, C.T. and HSU, J.H., 2005, An extended chi2 algorithm for discretization of real value attributes. *IEEE Transactions on Knowledge and Data Engineer*, **17**, pp. 437–441.

TAKEYAMA, M. and COUCLELIS, H., 1997, Map dynamics: integrating cellular automata and GIS through geo-algebra. *International Journal Geographical Information Sciences*, **11**, pp. 73–91.

THEIL, H., 1967, *Economics and Information Theory* (Amsterdam: NorthHolland).

WANG, Y. and ZHANG, X., 2001, A dynamic modeling approach to simulating socioeconomic effects on landscape changes. *Ecological Modelling*, **140**, pp. 141–162.

WARD, D.P., MURRAY, A.T. and PHINN, S.R., 2000, A stochastically constrained cellular model of urban growth. *Computers, Environment and Urban Systems*, **24**, pp. 539–558.

WHITE, R. and ENGELEN, G., 1993, Cellular automata and fractal urban form: a cellular modelling approach to the evolution of urban land-use patterns. *Environment and Planning A*, **25**, pp. 1175–1199.

WHITE, R., ENGELEN, G. and UIJEE, I., 1997, The use of constrained cellular automata for high-resolution modelling of urban land use dynamics. *Environment and Planning B*, **24**, pp. 323–343.

WU, F., 1998, SimLand: a prototype to simulate land conversion through the integrated GIS and CA with AHP-derived transition rules. *International Journal of Geographical Information Science*, **12**, pp. 63–82.

WU, F., 2002, Calibration of stochastic cellular automata: the application to rural–urban land conversions. *International Journal of Geographical Information Science*, **16**, pp. 795–818.

WU, F. and WEBSTER, C.J., 1998, Simulation of land development through the integration of cellular automata and multicriteria evaluation. *Environment and Planning B: Planning and Design*, **25**, pp. 103–126.

XIE, H., CHENG, H.Z. and NIU, D.X., 2005, Discretization of continuous attributes in rough set theory based on information entropy. *Chinese Journal of Computer*, **28**, pp. 1570–1574.

YEH, A.G.O. and LI, X., 2006, Errors and uncertainties in urban cellular automata. *Computers Environment and Urban Systems*, **30**, pp. 10–28.