An Artificial-Neural-Network-based, Constrained CA Model for Simulating Urban Growth

Qingfeng Guan  
*University of California - Santa Barbara*

Liming Wang  
*Chinese Academy of Sciences*

Keith C. Clarke  
*University of California - Santa Barbara*

Follow this and additional works at: [http://digitalcommons.unl.edu/natrespapers](http://digitalcommons.unl.edu/natrespapers)

Part of the [Natural Resources and Conservation Commons](http://digitalcommons.unl.edu/natrespapers)

[http://digitalcommons.unl.edu/natrespapers/287](http://digitalcommons.unl.edu/natrespapers/287)

This Article is brought to you for free and open access by the Natural Resources, School of at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in Papers in Natural Resources by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.
An Artificial-Neural-Network-based, Constrained CA Model for Simulating Urban Growth

Qingfeng Guan, Liming Wang, and Keith C. Clarke

ABSTRACT: Insufficient research has been done on integrating artificial-neural-network-based cellular automata (CA) models and constrained CA models, even though both types have been studied for several years. In this paper, a constrained CA model based on an artificial neural network (ANN) was developed to simulate and forecast urban growth. Neural networks can learn from available urban land-use geospatial data and thus deal with redundancy, inaccuracy, and noise during the CA parameter calibration. In the ANN-Urban-CA model we used, a two-layer Back-Propagation (BP) neural network has been integrated into a CA model to seek suitable parameter values that match the historical data. Each cell’s probability of urban transformation is determined by the neural network during simulation. A macro-scale socio-economic model was run together with the CA model to estimate demand for urban space in each period in the future. The total number of new urban cells generated by the CA model was constrained, taking such exogenous demands as population forecasts into account. Beijing urban growth between 1980 and 2000 was simulated using this model, and long-term (2001-2015) growth was forecast based on multiple socio-economic scenarios. The ANN-Urban-CA model was found capable of simulating and forecasting the complex and non-linear spatial-temporal process of urban growth in a reasonably short time, with less subjective uncertainty.

KEYWORDS: Constrained cellular automata, artificial neural network, urban growth

Introduction

A classical cellular automata (CA) model is a set of identical elements, called cells, each one of which is located in a regular, discrete space. Each cell can be associated with a state from a finite set. The model evolves in discrete time steps, changing the states of all its cells according to a transition rule, homogeneously and synchronously applied at every step. The new state of a certain cell depends on the previous states of a set of cells, which can include the cell itself, and constitutes its neighborhood.

Urban cellular automata (Urban-CA) are those cellular automata used to simulate and forecast land-use and land-cover change in urban areas. As a consequence of years of study, Urban-CA models have become much more complex than the original basic CA concepts. The neighborhood patterns, the transition rules, the linkages to other socio-economic models have all made Urban-CA advanced and intensive computing systems (Benenson and Torrens 2004).

One of the most important features of CA is that models can be used to simulate complex dynamic spatial patterns through a set of simple transition rules. However, several spatial factors, which have impact on urban development, should be considered while simulating a real geo-spatial phenomenon. Usually, in a model, these spatial factors are indicated by a set of parameters that reflect the contributions of corresponding factors to the model. Previous studies have discovered that model parameters have significant impact on the simulation results of CA models (Wu and Webster 1998). Thus, calibration processes are needed to determine the appropriate parameter values so that CA models can produce more realistic simulation results.

In practice, many variables, even hundreds of them, could be used to simulate a complex spatial system (White and Engelen 1994), most linked by non-linear relationships. After decades of study, the calibrating methods have become one of the most crucial components of geographic/urban CA models. One popular calibration method was developed by Clarke and Gaydos (1998) for a CA-based urban growth model called SLEUTH. SLEUTH is a CA model of urban growth and land use (LU) change.
which couples two CAs together and calibrates for historical time sequences using Geocomputational methods (Silva and Clarke 2002). This method uses the computer to produce simulation results using different combinations of parameter values, and then compares each result with real historical data to determine the best matching combination. However, since there could be an enormous number of combinations of parameter values, this calibration process is extremely time-consuming. Actually, even with high-performance workstations, it takes hundreds, even thousands, of hours to calibrate the model in practice. Furthermore, the computing time would extend exponentially if more variables were involved. Another method is to set the parameter values according to experts’ experiences or a hypothesis. Obviously, this method is very subjective and unreliable.

In this paper, an artificial neural network (ANN) is proposed as a solution to the constraints of CA model calibration. The ANN, simulating a human brain neural network, has been used to simulate complex systems, including geo-spatial dynamic phenomena (Openshaw, et al. 1998). A neural network can learn from available data, and deal with redundancy, inaccuracy, and noise. Knowledge and experience can be easily learned and stored for further simulation. Thus, integrating ANN into CA will significantly reduce the computing time of calibration. In addition, CA models will benefit from the ANN’s capacity of dealing with non-linear systems in terms of simulating and forecasting complex dynamic geo-spatial systems (Li and Yeh 2002).

Another important issue of Urban-CA research is how to model the interaction between socio-economic factors and the development of an urban area. Many studies have shown that socio-economic factors, including government, police, planning, economic development, and population growth play important roles in the process of urban development. More and more researchers try to model these relationships in their Urban-CA models. Usually there are two approaches: loose integration and internal merging. Using the loose integration approach, one or more stand-alone socio-economic models are linked with the CA model through parameter exchange or linking functions. These socio-economic models act as external factors interacting with the CA model. Using the internal merge approach, socio-economic factors are involved in the CA model as non-spatial variables. Socio-economic factors are characterized or abstracted, and they become one or more numerical indicators. These indicators, then, are merged into the CA model as variables while defining cell neighborhood or specifying transition rules.

In this paper, the first approach, loose integration, is used. We consider socio-economic factors as exogenous impacts on urban development. In addition, socio-economic systems have their own mechanisms and rules. Thus, stand-alone models might be more appropriate to model these complex systems. Another factor in favor of loose integration is that there are already several sophisticated socio-economic models developed. It is easier to employ them by linking with the CA model rather than developing new models that have not been tested sufficiently.

A macro-scale socio-economic model called the Tietenberg model is used in this study to generate the proper demand for urban space in each period in the future. In the Tietenberg model, population increase is considered as the main driving factor of urban space expanding. Using population forecasts as exogenous demands, the total number of new urban cells generated by the CA model is constrained.

**ANN-Urban-CA Model**

The CA model’s bottom-up structure helps to capture the micro-scale dynamics of geospatial entities/objects in order to simulate the macro-scale dynamics of the entire geographic space. While the socio-economic model, integrated and interacting with the CA model, simulates macro-scale socio-economic processes and forms a top-down structure to constrain the propagation of cells in the CA model.

The ANN is employed to automatically determine the transition rules and parameter values of the CA model through the training process, based on historical data. With ANN’s capacity of dealing with non-linear systems, this process would save much computing time, compared to the calibration method used in SLEUTH. Also, without interference from human beings, it would be much less subjective than calibration methods with human–machine interactions. Based on these factors, an Artificial-Neural-Network-Based constrained CA model for simulating urban growth (called ANN-Urban-CA) was developed in this study.

As Figure 1 shows, our ANN-Urban-CA model has two main components: micro-scale geospatial dynamic simulation, and macro-scale socio-economic dynamic simulation. It has two possible outputs/modes: simulation of urban growth history, and forecast of future urban growth.
Structure of the ANN

The Back-Propagation (BP) neural network, a feed-forward multi-layer network based on the Back-Propagation algorithm developed by Rumelhart and McClelland (1986), has become one of the most widely used ANNs in practice. The Activation Transfer Function (ATF) of a BP network, usually, is a differentiable Sigmoid (S-shape) function, which helps to apply non-linear mapping from inputs to outputs.

A two-layer BP network was used in our ANN-Urban-CA model. The first layer (Input Layer) has $s$ neurons (or nodes), and the second (Output Layer) has only one neuron. The input vector of this network has $r$ elements that indicate $r$ spatial factors, and the output vector has only one element that indicates the probability of urbanization of a specific non-urbanized cell in the cell space. The structure of a network is a crucial issue of ANN research. In this study, the structure of the BP network was varied, which means that the number of layers, the number of neurons of each layers, and the size of input vectors could be different in certain cases. The input weight matrix $IW$, which connects the input vector $P$ and the input layer, is a $s \times r$ matrix.

$$IW = \begin{bmatrix} w_{i1} & w_{i2} & \cdots & w_{ir} \\ w_{s1} & w_{s2} & \cdots & w_{sr} \\ \vdots & \vdots & \ddots & \vdots \\ w_{r1} & w_{r2} & \cdots & w_{rr} \end{bmatrix}$$  \hspace{1cm} (1)

In this matrix, the variable $iw_{ij}$ indicates the weight between the neuron $i$ in the input layer and the spatial factor $j$. The input bias matrix $b^i$ is a $s \times 1$ matrix:

$$b^i = \begin{bmatrix} b^i_1 \\ b^i_2 \\ \vdots \\ b^i_s \end{bmatrix}$$  \hspace{1cm} (2)

The activation transfer function between the input vector $P$ and the input layer is:

$$f(x) = \frac{1 - \exp(-2x)}{1 + \exp(-2x)}$$  \hspace{1cm} (3)

The function of the neuron $i$ in the input layer is:

$$net_{(i,t)} = \frac{1 - \exp(-2 \sum_{j=1}^{r} P_{(j,t)} iw_{ij} + b^i_i)}{1 + \exp(-2 \sum_{j=1}^{r} P_{(j,t)} iw_{ij} + b^i_i)} \hspace{1cm} i = (1, 2, \ldots, s)$$  \hspace{1cm} (4)

where:

$net_{(i,t)}$ = the value of the neuron $i$ in the input layer at time $t$; and

$P_{(j,t)}$ = the value of the spatial factor $j$ at time $t$.

The layer weight matrix $LW$, which connects the input layer and the output layer, is a $1 \times s$ matrix.
Let \( w_i \) indicate the weight between the neuron \( i \) in the input layer and the neuron in the output layer, and layer bias matrix \( b^2 \) be a \( 1 \times 1 \) matrix. The activation transfer function between the input layer and the output layer is then:

\[
f(x) = \frac{1}{1 + e^{-x}}
\]  

Thus, the function of the neuron in the output layer is as follows:

\[
O_i = \frac{1}{1 + e^{-\sum_{j=1}^{s} w_{ij} x_j + b^2}} \quad i = (1, 2, \ldots, s)
\]

The variable \( O_i \) is the output value of this network, which is actually the probability of urbanization of a specific non-urbanized cell, at time \( t \). The structure of this network is shown in Figure 2.

### Training the ANN

Through the training process, the neural network adjusts the weight and bias values. In the following simulation process, combinations of activation transfer functions, weights, and biases assume the roles of transition rules and parameters in the CA model. The output of the trained neural network will be the probability of urbanization of a specific non-urbanized cell.

The BP network uses the Back-Propagation training algorithm (Figure 3), which is a type of the supervised training/learning algorithm. In the training process, the outputs of the network are evaluated and compared with sample targets of the real world, e.g., historical urban growth data in this study. Based on this comparison, the weight and bias values of the network are adjusted in order to reduce the difference between network outputs and the targets. This process is repeated, until the difference between the outputs and the targets reaches a previously set goal. After training, the BP network should be sufficiently adjusted to simulate urban growth.

However, using the original BP training algorithm, the training process could be extremely time-consuming in some cases. Several fast BP training methods have thus been developed to deal with various situations. Following a number of experiments, the scaled conjugate gradient (SCG) method (Møller 1993), which is especially suitable for function fitting and pattern classification (Xu and Wu 2002), was employed in the ANN-Urban-CA model.

### CA Model Computation and Simulation

The computation of the CA model is actually the simulation process of the trained ANN. The
ANN receives a set of spatial variable values as the input vector, performs network calculation (Equations (4) and (7)) with trained weight and bias (Equations (1), (2), and (5)) values, and produces an output value which is the probability of urbanization of a specific non-urbanized cell. A specific function decides whether this cell will be urbanized according to a preset threshold or the restriction derived from the macro-scale socio-economic model. All the non-urbanized cells are evaluated with this process at each time step, so as to simulate the spatial-temporal dynamics of urban growth. This model possesses all the characteristics of a classic CA model, including:

- **Cell**: To simulate urban growth, a cell is defined as a land-use unit or a land parcel in a given urban area. To facilitate implementation, the cells are square, which can be represented in geographic information systems (GIS) using pixels.

- **State**: Only two possible states are available for a cell at a certain time step: non-urbanized land (0) and urbanized land (1). Each cell has an urbanization probability ranging within [0, 1].

- **Neighborhood**: Our model used an extended (11×11) Moore neighborhood. The number of urbanized cells in this neighborhood is counted, as it is one of the spatial variables in the ANN.

- **Transition Rules**: The ANN’s ATFs, trained weights, and biases play the roles of transition rules and parameters in the CA model. That is to say, the behavior of the CA is determined by the ATFs, weights, and biases of the ANN. One issue of importance is that the output of the ANN is the urbanization probability of a cell, but not the final state of the cell.

- **Cell Space**: Cell space is the whole urban area being studied.

- **Time**: In this study, the time step was set to one year, which means an iteration in the model represents one year of dynamics in the real world.

**Structure and Simulation of the Tietenberg Model**

The Tietenberg model, a macro-scale socio-economic model integrated in the ANN-Urban-CA model, was used to generate appropriate demand for urban space in a given period (e.g., a year) in the future. The total number of new urban cells generated by the CA model at each iteration is calculated using the Tietenberg model.

The Tietenberg model is actually a resource economics model, which is usually used to solve the problem of “how to consume resources in the future according to the principle of sustainable development” (Tietenberg 1992). In this study, lands are treated as non-regenerative resources, while the urbanization process consumes land sources. The amount of land consumed each year can be calculated using the following equations:

\[
\frac{a-bq_t/IP_{tu} - c}{(1+r)^{t-1}} - \lambda = 0
\]

\[
Q - \sum_{t=1}^{n} q_t = 0 \quad t = (1,2,\cdots,n)
\]

where:

- \(Q\) = total amount of land resources;
- \(q_t\) = the amount of consumed land in year \(t\) (number of new urban cells at the iteration \(t\));
- \(P_{tu}\) = increased urban population in year \(t\);
- \(r\) = discounting rate of urbanization;
- \(a\) = the intercept of the marginal benefit curve, which is set to 1000 in this study;
- \(b\) = the slope of the marginal benefit curve, which is set to 1; and
- \(c\) = the constant of the marginal benefit curve, which is usually set to \(a/2\) (Yeh and Li 1998).

To execute the Tietenberg Model, increased population in the future is needed. By using the Logistic regression, based on the population data in the history, the increasing curve of population can be calculated as follows (Yeh and Li 1998):

\[
X_t = \frac{X_m}{1 + (\frac{X_m}{X_0} - 1)e^{-rt}}
\]

where:

- \(X_m\) = population carrying capacity;
- \(r\) = constant population increase rate; and
- \(X_t\) = population in year \(t\).

**Case Study: Forecasting Urban Growth for Beijing City**

Urban growth in Beijing City during 1980-2000 was simulated, and multiple, possible long-term (2001-2015) growth options were forecasted based on various socio-economic scenarios.
Data Collection and Pre-processing

Digital elevation models (DEM) and data about political and administrative regions (provinces, cities, and counties), transportation lines, and water system were utilized so as to identify those spatial factors that impact urban development. To obtain target samples for the ANN training process, historical land-use data for Beijing City in 1980, 1995, and 2000 were collected and re-classified into urbanized area (value 1) and non-urbanized area (value 0). All the data were converted into 100m×100m grids. We selected eight factors which potentially impact the urban growth:

- Elevation (p1), derived from the DEM;
- Slope (p2), derived from the DEM;
- Distance to the center of the city (p3), derived from the map of city center;
- Distance to centers of towns (p4), derived from the map of towns;
- Distance to railways (p5), derived from the map of railways;
- Distance to highways (p6), derived from the map of highways;
- Distance to the national highways (p7), derived from the map of national highways; and
- Number of urbanized cells (p8) in the 11×11 neighborhood of cells, derived from the re-classified land-use map.

The model will work with any number of spatial factors; our selection was limited by the availability of geospatial data. Had there been more data available, we would have selected more factors. The value of these factors was standardized into [0, 1] by using the following equation:

\[
P_{ij} = \frac{p_{ij} - \text{\scriptsize{Min}}(p_i)}{\text{\scriptsize{Max}}(p_i) - \text{\scriptsize{Min}}(p_i)} \quad i = (1, 2, \ldots, 8) \quad j = (1, 2, \ldots, u)
\]

where:
- \( P_{ij} \) = the value of the factor \( i \) of the cell \( j \) in the cell space;
- \( \text{\scriptsize{Min}}(p_i) \) = the minimum value of the factor \( i \) in the cell space; and
- \( \text{\scriptsize{Max}}(p_i) \) = the maximum value of the factor \( i \) in the cell space.

As mentioned above, sample data are needed for the training process of the ANN. To reduce the spatial correlations among the sample data, a stratified random sampling method was applied to select sample points in the re-classified land-use maps (Figure 4). In this way, the quotient between the number of sample points in the urban area and that of sample points in the non-urbanized area is equal to the quotient between the number of urbanized cells and that of non-urbanized cells. Values p1- p8, i.e., the input vector of the ANN, were incorporated in table P (the first eight columns of Table 1). While the urban code values (1 for urbanized and 0 for non-urbanized) of the corresponding cell, i.e., the target vector of the ANN, was inserted in table T (the last column of Table 1).

ANN Training

Since eight factors have been selected, the size of the input vector of the ANN was specified as eight. The size of the output vector would be one [size=1] that is the probability of urbaniza-
Table 1. Values of factors and urban code of sample points (partial).

<table>
<thead>
<tr>
<th>p1</th>
<th>p2</th>
<th>p3</th>
<th>p4</th>
<th>p5</th>
<th>p6</th>
<th>p7</th>
<th>p8</th>
<th>Urban code</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0063</td>
<td>0.0000</td>
<td>0.1727</td>
<td>0.0253</td>
<td>0.0026</td>
<td>0.1157</td>
<td>0.0317</td>
<td>0.86</td>
<td>1</td>
</tr>
<tr>
<td>0.0188</td>
<td>0.0051</td>
<td>0.3258</td>
<td>0.1462</td>
<td>0.0163</td>
<td>0.1699</td>
<td>0.1165</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.0184</td>
<td>0.0000</td>
<td>0.0109</td>
<td>0.2600</td>
<td>0.0249</td>
<td>0.0509</td>
<td>0.0712</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 5. Network error occurring during the training process.

Figure 6. Urbanization probabilities of non-urbanized cells.

CA Computing and Urban Growth History Simulation

An Arc Macro Language (AML) program was developed to implement the CA computation in the ArcGIS environment. Its output is the map of urbanization probabilities (range in [0, 1]) of non-urbanized cells (Figure 6).

We set 0.8 as the urbanization probability threshold and took the urban map of Beijing in 1995 as the start map of the urban growth simulation. In the simulation, one iteration corresponded to one year in the real world. At a specific iteration, all those non-urbanized cells (value 0) whose urbanization probabilities were over 0.8 were re-classified as urbanized cells (value 1). The number of neighboring urbanized cells of each cell was then re-calculated, and the new number served as the one of the spatial factors used in the next iteration. We obtained the simulated urban map of Beijing in 2000 (Figure 7) after five iterations.
To evaluate the simulation results, we used two simple methods previously used by Zhou et al. (1999):

1. The Lee-Sallee index (Clarke and Gaydos 1998) calculated by:

\[ L = \frac{A_o \cap A_i}{A_o \cup A_i} \]

where:

- \( A_o \) = urban area in the simulation map;
- \( A_i \) = urban area in the real map.

The Lee-Sallee index will be 1.0 if the simulation map matches the real map perfectly.

2. The cross correlation between the simulation map and the real map, computed by:

\[ c = \frac{\sum_{i,j} c_{ij}}{\sqrt{\sum_{i} (z_i - \bar{z}_i)^2} \sqrt{\sum_{j} (z_j - \bar{z}_j)^2}} \]

where:

- \( n \) = the total number of cells in a grid;
- \( i \) = specific cell on the first input grid;
- \( j \) = the cell on the second input grid that is located at the same position as the cell \( i \);
- \( z_i \) = the attribute value of the cell \( i \);
- \( z_j \) = the attribute value of the cell \( j \); and
- \( c_{ij} \) = the similarity of \( i \)'s and \( j \)'s attribute value:

\[ (z_i - \bar{z}_i)(z_j - \bar{z}_j) \]

When cross correlation is computed on the same attribute (e.g., urban code in this case) on two grids (e.g., simulation map and real map), it represents the similarity level of the spatial distribution of this attribute on these grids. In this study, it shows how good the simulation map matches the real map. It will be 1.0 if they match perfectly. In our study, the Lee-Sallee index of the year 2000 was 0.8318, and the correlation between the simulation map (2000) (Figure 7) and real map (2000) (Figure 8) was 0.9018.

Constrained CA Model and Urban Growth Forecasting

The Tietenberg model requires that future increases in the population be taken into account. We used the logistic non-linear regression method on Beijing’s population data in 1978–2001 to forecast the increased populations of Beijing in 2001–2015 (Figure 9).

Based on Beijing’s urban growth history, three scenarios of total urban land were developed.
Setting the discounting rate \( r \) to be 0 or 0.01, six sub-scenarios of urbanization (Table 2) were computed by the Tietenberg Model.

An AML program was developed to forecast urban growth in the future. This involved calculating the urbanization probabilities of non-urbanized cells at a specific iteration, and then re-classifying non-urbanized cells into urbanized cells in a descending order of urbanized probability until the number of new urban cells reached the number appearing in the scenario setting (Figure 10).

### Conclusion

ANN-based CA models (Li and Yeh 2002) and constrained CA models (Engelen et. al 1995; Yeh and Li 1998; White and Engelen 2000) are not new concepts. However, much less research has been done on integrating these two. This study developed a constrained cellular automata (CA) model based on an artificial neural network (ANN) in order to simulate and forecast urban growth. We hope this paper will draw some attention to this field.

Taking advantage of the ANN’s capacity of dealing with nonlinear systems, our ANN-Urban-CA model can be calibrated without heavy computing overhead and subjective human interference. Furthermore, constrained by the macro-scale, socio-economic Tietenberg Model, the forecasting enabled by the ANN-Urban-CA model merges the optimal total amount of urban areas and optimal spatial locations of urban areas to obtain the best benefits from urban land use. The pattern of urban development forecast by the model could be used to advantage in planning sustainable urban development. Thus, it is

<table>
<thead>
<tr>
<th>Years</th>
<th>Increased Pop (10,000)</th>
<th>Total Scenario 1 (hm²)</th>
<th>Total Scenario 2 (hm²)</th>
<th>Total Scenario 3 (hm²)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>( r=0 )</td>
<td>( r=0.01 )</td>
<td>( r=0 )</td>
</tr>
<tr>
<td>2001~2005</td>
<td>123,956</td>
<td>7522.9</td>
<td>8201.1</td>
<td>15186.8</td>
</tr>
<tr>
<td>2006~2010</td>
<td>141,766</td>
<td>8687.9</td>
<td>8758.7</td>
<td>17538.6</td>
</tr>
<tr>
<td>2011~2015</td>
<td>162,134</td>
<td>10033.2</td>
<td>9284.2</td>
<td>20254.6</td>
</tr>
</tbody>
</table>

Table 2. Beijing urbanization scenarios, 2001-2015.
more of a prescriptive planning tool for urban planners, rather than a tool used to validate our understanding of real-world urbanization. The ANN-Urban-CA model we describe in this paper has two limitations:

- Because the ANN-Urban-CA model inherits the black-box properties from the ANN, users may find it difficult to investigate the physical meanings of spatial factors and their contribution to urban development. This is the main reason that this model cannot be used to validate our understandings of real-world urbanization processes.

- The urban development process interacts with multi-source (political, social, economic, cultural) and multi-scale (global, national, regional, City-level, Sub-city-level) socio-economic factors. The relationships among these factors are non-linear and complex. Only population factors are considered in the model described here, which is obviously insufficient.

REFERENCES


Figure 13. Zoom-in of Figure 12.


---

380 Cartography and Geographic Information Science