CHANGE DETECTION - CELLULAR AUTOMATA METHOD FOR URBAN GROWTH MODELING

Sharaf Alkheder, **Jun Wang** and **Jie Shan**

Geomatics Engineering, School of Civil Engineering, Purdue University
550 Stadium Mall Drive, West Lafayette, IN-47907, U.S.A
{salkhede, wang31, jshan}@ecn.purdue.edu

Commission VII, WG VI/4

KEY WORDS: Urban Growth, Cellular Automata, Genetic Algorithm, Dynamic System, Change Detection

ABSTRACT:

This paper describes an on-going work on the development and implementation of cellular automata based urban growth modeling using multitemporal satellite imagery. The algorithm is designed to simulate the historical growth as a function of local neighbourhood structure of the input data. Transition rules in the algorithm drive the urban growth over time. Calibration is introduced in the cellular automata model. Spatial and temporal calibration schemes are used to improve the prediction accuracy. Spatially, the model is calibrated on a township basis to take into account the effect of site specific features, while the temporal calibration is set up to adapt the model to the changes in the growth pattern over time. Calibration provides the optimal values for the transition rules to achieve accurate urban growth modeling. The paper discusses at the end a proposed automatic rule calibration method using genetic algorithm. The aim is to optimize the transition rule values. Prediction accuracy is selected as the fitness function. A set of strings are used as initial population over which the genetic algorithm runs till convergence. The cellular automata model is tested over city Indianapolis, IN, USA to model its urban growth over a period of 30 years. Besides the land use data derived from the satellite imagery, population density is used. Results indicate good accuracy on a township basis for short term (5 years) and long term (11 years) prediction. The model succeeds in adapting to the dynamic growth pattern. Genetic algorithm shows promising potential in the calibration process.

1. INTRODUCTION

Urban growth modeling is getting more attention as an emerging research area in many disciplines. This comes as a result of the recent dramatic increase in urban population that increases the pressure on the infrastructure services. Among all developed urban growth models, cellular automata (CA) urban growth models have better performance in simulating urban development than conventional mathematical models (Batty and Xie, 1994a). CA simplifies the simulation of complex systems (Waldrop, 1992). Its appropriateness in urban modeling is due to the fact that the process of urban spread is entirely local in nature (Clarke and Gaydos, 1998). Models based on cellular automata are impressive in terms of their technological evolution in connection to urban applications (Yang and Lo, 2003). Development of a CA model involves rule definition and calibration to produce results consistent with historical data, and future prediction with the same rules (Clarke *et al*, 1997).

Many CA-based urban growth models are reported in the literatures. White and Engelen (1992a; 1992b) CA model involves reduction of space into square grids. They implement the defined transition rules in recursive form to match the spatial pattern. One of the earliest and most well-known models in the literature is Clarke's *et al* (1997) CA-based model "SLEUTH" that has four major types of data: land cover, slope, transportation, and protected lands. This is rooted in the work of von Neumann (1966), Hagerstrand (1967), Tobler (1979) and Wolfram (1994). A set of initial conditions in "SLEUTH" is defined by `seed' cells which were determined by locating and dating the extent of various settlements identified from historical maps, atlases, and other sources. These seed cells represent the initial distribution of urban areas. A set of complex behaviour rules is developed that involves selecting a location randomly, investigating the spatial properties of the neighboring cells, and urbanizing the cell based on a set of probabilities.

Despite all the achievements in CA urban growth modeling, the selection of the CA transition rules remains a research topic (Batty, 1998). CA models are usually designed based on individual preference and application requirements with transition rules being defined in an ad hoc manner (Li and Yeh, 2003). Furthermore, calibration of CA models is still a challenge. Most of the developed CA models need intensive computation to select the best parameter values for accurate modeling.

The motivation behind this work is to develop an effective CA-based urban growth model. Also, developing a calibration algorithm that takes into consideration spatial and temporal dynamics of urban growth is another objective of this study. Genetic algorithm (GA) is introduced as a heuristic optimization technique for selecting optimal model parameters.

2. STUDY AREA

Indianapolis, Indiana, USA is selected as a case study over which the CA model is designed and tested. Indianapolis is located in Marion County at latitude 39° 44' N and longitude of 86° 17' W as shown in Figure 1. Representing the main city in the state of Indiana, Indianapolis encounters recognizable accelerated growth in population and urban infrastructure over the last few decades. The necessity arises to model the urban growth over time for better planning of infrastructure services.

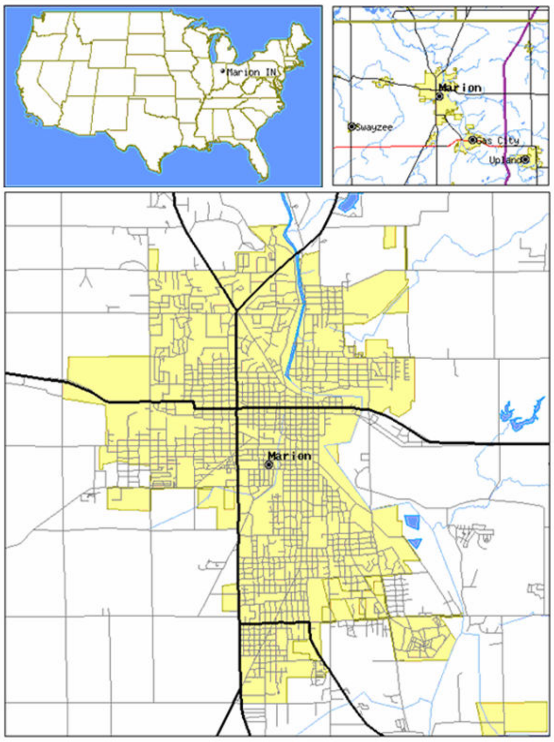


Figure 1. City of Indianapolis, Indiana, USA (US Census Bureau)

3. DATA PREPARATION

This section describes the input data processing scheme. Five historical TM images (1973, 1982, 1987, 1992 and 2003) are collected over the study area (Figure 2). These images are rectified and registered to the same projection of UTM NAD1983 to fit each other spatially. Seven classes of interest are identified in the images: water, road, commercial, forest area, residential areas, pasture and row crops. Residential and commercial classes represent urban class. The overall classification accuracy for all classified images is above 93%.

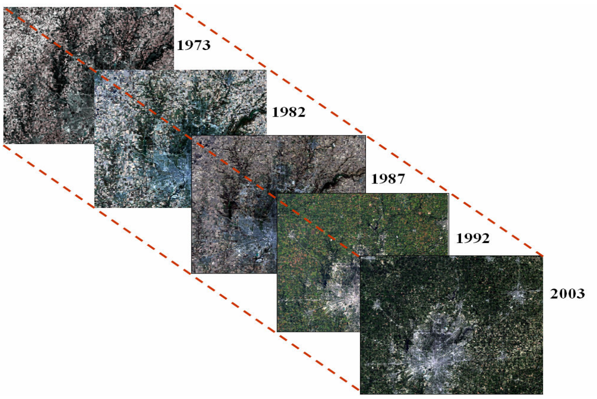
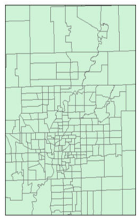


Figure 2. Historical TM images over Indianapolis

Population density is used as another input for the CA model algorithm. Population census tract maps for year 1990 and 2000 over Indianapolis are collected. The population densities are computed for all census tracts by dividing their populations by the tract areas. Figure 3 shows the census tract map (left) and the calculated population density for each census tract (right). To model the population, the centroid (Xc,Yc) for each census tract is calculated.



Area	Density	cX	cY	distance
91.086045	54.871194	636781.472632	4564759.26662	174.894957
166.097149	31.066156	621815.123851	4566659.01416	171.759662
230.032768	23.340153	615808.843737	4551339.44611	155.364124
39.476346	149.405924	627795.058689	4559502.83159	166.816338
80.816454	53.899915	625480.477611	4553682.73295	160.557824
86.707483	20.724855	635260.686086	4554999.90133	165.277779
180.577767	17.327715	630148.408284	4545074.41109	154.155671
169.652027	14.553319	526512.521152	4524079.97479	130.163493
160.051919	18.212840	509282.386791	4524355.24879	137.412420
444.439337	5.307811	499741.412349	4507388.17329	127.739865
139.397549	16.327403	513363.124636	4507761.46725	120.858300

Figure 3. Year 2000 census tract map and population density

The Euclidean distance from each census tract centroid to the city center is computed. This process is repeated for all tracts so that a table of population densities versus distance is prepared. Population densities for census tracts within specified distance from city center are averaged to reduce the variability in data. For example, an average population density for all census tracts within 2 km is calculated then another average density is calculated for tracts within 2-4 km and so forth. An exponential function is fitted representing population density as a function of distance from the city centre:

$$POPULATION-DENSITY = Ae^{-(B*DISTANCE)} \qquad (1)$$

Model parameters A and B are calculated for year 2000 as shown in Figure 4. This exponential model is used to calculate the population density for each pixel in the imagery based on its distance form city center for year 2000.

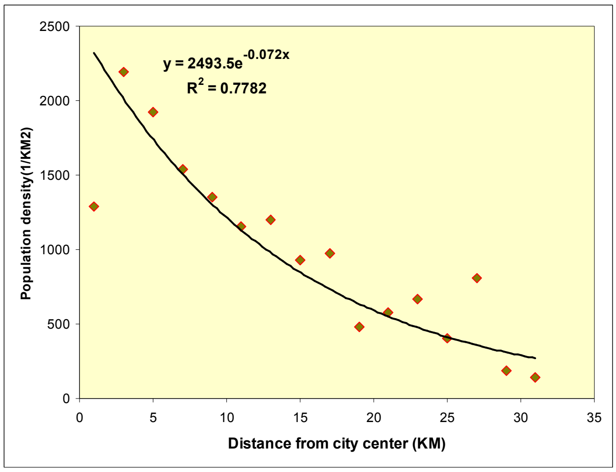


Figure 4. Year 2000 population density

The same procedure is repeated for year 1990 to find its exponential population model with A and B parameters. The change in model parameters over the 10 years (from 1990 to 2000) is used to calculate the yearly rate of change in A and B parameters. The updated parameters (A and B values that changed year by year) are used to calculate population density grids for each year from 1973 to 2003 matching the same size of the input imagery. These grids are used as CA data inputs for the purpose of running the model over historical growth period.

4. CELLULAR AUTMATA URBAN MODELING

This section discusses in detail the design of the CA urban growth modeling. The design phases include: transition rule definition, calibration method and evaluation strategy for the model. Calibration modules for accurate modeling over the historical satellite imagery to adapt the urban pattern are discussed in details.

4.1 CA Algorithm Design

The design of the CA algorithm consists of defining the transition rules that control the urban growth, calibrating these rules, and evaluating the results for prediction purpose. Figure 5 presents a flowchart that describes the CA algorithm structure.

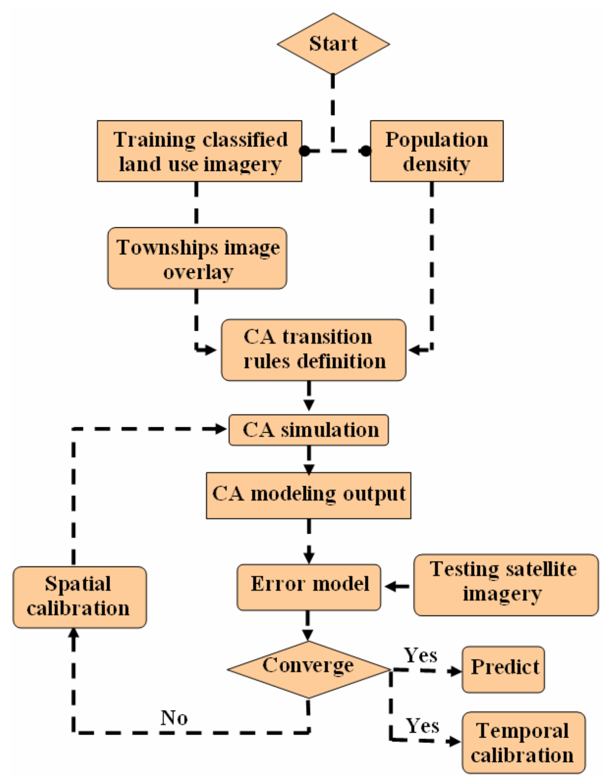


Figure 5. Flowchart of CA algorithm design

Transition rules definition is the most important phase in CA model design since they translate the effect of input data on the urban process simulation. So an accurate and realistic definition of the rules is needed. The CA algorithm design starts with defining the transition rules that drive the urban growth over time. They are designed as a function of land use effect on urban process, growth constraints and population density. The transition rules are defined over the 3x3 neighbourhood of a pixel to minimize the number of input variables to the model. The rules identify the neighbourhood needed for the tested cell to urbanize. The growth constraints should reflect the conservation strategy adopted in the study area for certain land uses. For example, conservation of certain species of natural resources can be taken into consideration through rules definition stage. Water resources protection through discouraging urban growth nearby these sites to reserve them over time is another example of constrained rules design. The future state of a pixel (Equation 2) at time (t+1) from starting time (t) depends on three factors:

- Current state of the test pixel.
- Current states of its neighbourhood pixels
- Transition rules that drive the urban growth over time.

$$S^{t+1}(\alpha) = f(S^{t}(\alpha), S^{t}(\tau), transition\_rules) \qquad (2)$$

where $S^{t+1}(\alpha)$ = test pixel future state at time epoch t+1.

$S^{t}(\alpha)$ = test pixel current state at time epoch t.

$S^{t}(\tau)$ = neighbourhood pixels states' set.

CA transition rules driving the urban growth are programmed in ArcGIS through Visual Basic for Applications (VBA). The oldest historical classified TM image (1973) is used as input to the CA model over which the transition rules are applied to model the urban growth starting from this time epoch. A polygon shapefile representing the townships in the study area is overlaid over the input image (Figure 6). A total of 24 townships in the area are identified.

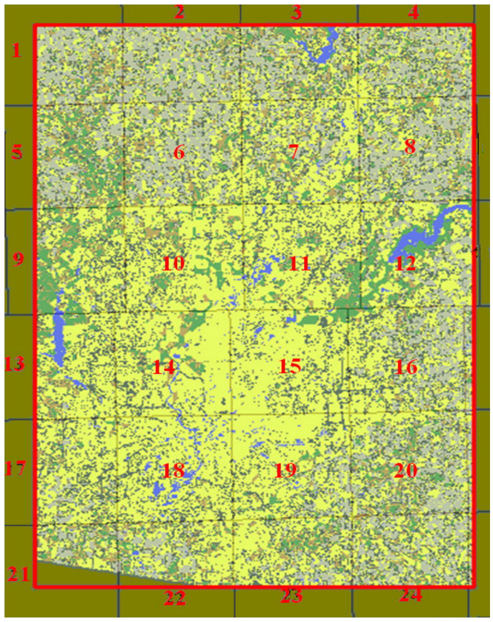


Figure 6. Township map of Indianapolis

Dividing the study area on a township basis will take into consideration the effect of site specific features in each township on the urban growth. The same CA transition rules are defined for each township, however, with different rule values. Calibration in the next section will discuss finding the optimal rule values for each township.

4.2 CA Model Calibration

Once the CA transition rules are identified and initialized for each township, the model runs from 1973 till 1982. The year 1982 image represents the first ground truth being used for calibration. For each township, the modeling accuracy is calculated as a ratio between the simulated and real urban growth data (Equation 3) for simulated 1982 image. Over/underestimation concept is introduced to represent how comparable is the simulated result to the real one. This indicates how transition rules defined on a township basis succeed in modeling the real amount of urban growth given the predefined conditions.

Calibration in this work is meant to find the best set of rule values specific to each township for realistic urban growth modeling.

$$Accurracy = \frac{Simulated\ urban\ class}{real\ urban\ class} \times 100\% \qquad (3)$$

Calibration aims to define the best set of CA rules based on which the model run to match as close as possible the simulated results with the ground truth images. To achieve this purpose, two calibration schemes are introduced in this algorithm: spatial and temporal calibrations. In spatial calibration module, the CA transition rules at a given time t (t=1982 in our case) are modified spatially over the 2D grid space. This is done through tuning the values of each rule set on a township basis to match the urban dynamics for each township with its site specific features. This allows the model to take the variability in the spatial urban growth pattern into accounts for realistic modeling. If the township' rules result in higher growth levels (overestimated), they are modified to reduce the urban growth at this township. For the underestimation case, the rule values of the township under consideration are tuned to increase the amount of urban growth to match the real one. So, the spatial calibration aims to find the best set of rule values that fit a given township *k* according to its geographical location: *X(k)*, input data parameters: *C(k)* and its over/underestimation case: *OAE (k)* as shown in Equation 4. Townships close to each other in term of geographical location and having similar urban growth characteristics (e.g. same development level over time) are associated with the same rule values. The overall calibration over the entire test image at time epoch *t* is obtained through performing the same spatial calibration by taking the average of all townships as shown in Equation 5. Once the rule values are modified spatially, the model runs again from 1973 to 1982, the modeling accuracy is evaluated for each township and rules are calibrated once again as illustrated above. This loop continues till the defined convergence criterion is met (Equation 6). In our work convergence criterion of 100%±10% accuracy is used. The rule values after convergence will be used for the next time period modeling. The model will be calibrated at the next ground truth image to take into consideration the temporal effect.

$$SC_{epoch,t} = f(X(k), C(k), OAE(k)) \quad (4)$$

$$OC_{epoch,t} = \sum_{l=1}^{24} SC_{epoch,t}(l) \quad (5)$$

$$FC_{epoch,t} = \left\{Converge\left(Loop(OC_{epoch,t}\right)\right\} \quad (6)$$

where $SC_{epoch,t}$ = spatial calibration at time epoch t.
$OC_{epoch,t}$ = over all calibration at time epoch t.
$FC_{epoch,t}$ = final calibration at time epoch t.

The final calibrated rules at 1982 are used to run the model till the next ground truth image 1987 to perform temporal calibration. The goal behind the temporal calibration is to recalibrate the model so that the model can adapt the urban growth pattern over time (Equation 7). This way any growth variation over time related to new policy or new infrastructure plans can be learned by the model and the modeling results become more realistic. The temporal calibration module is a function of changes in spatial calibration results between two epochs *t* and *t+t1*, growth change, and accuracy.

$$TC_{(t,\, t+t1)} = f(\Delta FC_{(t,\, t+t1)},\ \Delta UG,\ Accuracy) \quad (7)$$

where $TC_{(t,\, t+t1)}$ = temporal calibration between epochs t and t+t1.
$\Delta FC_{(t,\, t+t1)}$ = changes in spatial calibration results between epochs t and t+t1.
ΔUG = urban growth change between t and t+t1.

The same spatial calibration process at 1982 is repeated at 1987 ground truth till convergence. The final calibrated rule values at 1987 are used to predict the urban growth at 1992 for short term prediction of 5 years (Figure 7a). The first test image at 1992 is used to validate the model through evaluating the prediction results at 1992 on a township basis. Table 1 summarizes the prediction results accuracies for year 1992. The rules are calibrated spatially again at 1992 to perform long term prediction of 11 years from 1992 till 2003 (Figure 7b) that are validated using the test image at 2003. Table 1 shows the prediction accuracy for predicted image at 2003.

4.3 Results and Discussion

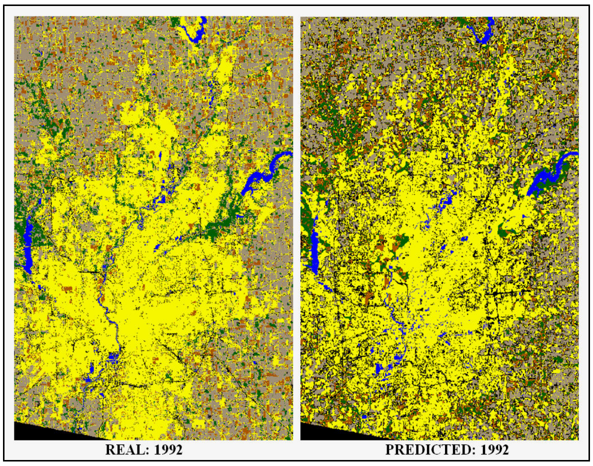
The prediction results of 1992 and 2003 show good accuracy on the township basis and on the average scale. Accuracy for short term prediction (1992) is higher as comparing with long term prediction interval (2003). The improvement in the accuracy over space for each township is noticeable as a result of the spatial calibration module. Temporal calibration also helps improve the prediction accuracy results over time. The spatial calibration at specific time epoch succeeds in reducing the variability in modeling accuracy between townships through matching each township transition rules with its site specific features. Spatial and temporal calibrations succeed in capturing the urban growth pattern over space and time based on real growth factors. Historical satellite imagery fits the spatial and temporal nature of the developed CA urban growth model. It provides important information input to the model.

4.4 Evaluation and Analysis

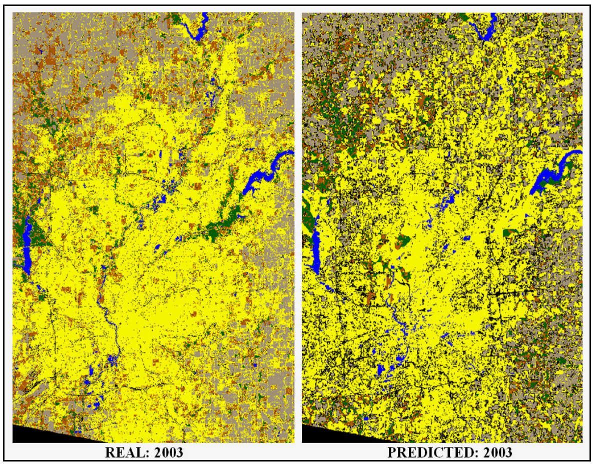
Prediction accuracy for each township is used as a basis for rule calibration. Over/under estimation principle is implemented. If a set of rules for a particular township produces underestimated results, this mean the growth rate is small and hence the rules are modified to increase the urban growth. For overestimation, the rules are modified to reduce the urban growth amount. The transition rules for a township are repeatedly calibrated till the convergence criterion is met. The ground truth imagery provides the reference for calibration process. Table 1 results indicate good spatial prediction accuracy ranging for year 1992 between 78.32% (underestimate) for township 17 up to 128.98% (overestimate) for township 1. The range for year 2003 is between 67.78% and 149.29%. The average accuracies for 1992 and 2003 are 98.35% and 94.57%, respectively. Higher accuracies achieved for short term prediction (1992) as compared to long term (2003). The spatial variability between townships' prediction results after calibration is small for both (1992 and 2003) predictions. This indicates the effect of spatial calibration in matching each township with its realistic urban growth pattern through calibrating its rules to fit such pattern. Visually, calibration on a township basis succeeds in preserving the urban pattern over space where temporal calibration preserves its dynamical changes over time. Rule values' results at the end of the calibration process indicate some similarity between townships. These townships are close to each other geographically and with similar urban growth characteristics.

5. TRANSITION RULE CALIBRATION

This section introduces briefly an ongoing study on using genetics algorithm (GA) to automate the spatial and temporal rule calibrations. GA as a heuristic optimization technique can work over the search space to find the most suitable solution. GA improves the efficiency of rule calibration to select the best set of rule values for accurate modeling. GA is first introduced by Holland (1975) as computer programs to mimic the evolutionary processes in nature. GA manipulates a set of feasible solutions to find an optimal solution.



a. Result of year 1992 prediction (5 years)



b. Result of year 2003 prediction (11 years)

Figure 7. Cellular automata prediction results

GA's is able to find the global optimum solution. The following steps describe the design of the proposed GA-based transition rule calibration.

Step 1: Initial GA population generation
In this step, 30 sets of rule values are randomly generated as an initial population for each township over which GA module will work. Each rule value set is coded as a binary string. A string is designed as a combination of the rule values. Three rules are identified to be optimized using GA:

Rule1: The number of neighbourhood residential pixels, in the possible range of [0-8] integer values or in corresponding binary coding [0000 to 1000].
Rule2: The number of neighbourhood commercial pixels, in the possible range of [0-8] integer values or in corresponding binary coding [0000 to 1000].
Rule3: The population density threshold, continuous values representing the cut-off of population density at a pixel. This rule is scaled by multiplying its value by 10 in the range of [0-20] possible values or in binary coding [00000 to 10100].
All the rules are combined to form one binary string.

Step 2: Fitness function identification
Fitness function evaluates the performance of each string. The prediction accuracy is used as the fitness function.

Table 1. Year 1992 and 2003 prediction results

Township#	Accuracy(%), 1992	Accuracy(%), 2003
1	128.98	149.29
2	118.64	100.05
3	78.75	125.94
4	95.80	120.53
5	93.10	83.51
6	111.16	89.74
7	119.51	109.33
8	92.81	91.08
9	115.17	100.96
10	108.17	99.53
11	97.62	94.74
12	104.82	109.83
13	99.11	98.46
14	88.59	90.00
15	95.59	93.23
16	91.83	67.78
17	78.32	98.41
18	90.06	90.01
19	102.78	95.97
20	100.70	69.29
21	88.27	93.81
22	106.79	96.23
23	85.79	95.91
24	119.85	82.91
Average	98.35	94.57
Std. Dev.	13.44	17.05

Step 3: GA selection operation
Rank selection procedure is used in this work. All the strings are ordered based on their fitness values in descending order and the string with highest fitness value is given rank 30 then the second one 29 till lowest fitness value with rank 1. Rank is divided by the summation of all the ranks and the probability of selection for each string in next generation is identified.

Step 4: GA crossover and mutation parameters design
The crossover probability is selected to be 80%, 24 strings are selected for crossover, while the other 6 (the best 6 in terms of fitness values) are copied directly to the new generation (this process is known in GA as Elitism). Elitism can rapidly increase the performance of GA, because it prevents a loss of the best solution. A mutation rate of 1% is used. Once the crossover and mutation is done, the new generation of 30 strings is already produced. The next step is to run the CA model using the new strings to evaluate their new fitness values.

Step 5: Running the GA-CA model
All the steps from 1 to 4 are repeated till convergence. GA model works again over the newly created 30 strings and a new generation of 30 strings is produced and the loop continues. This continues until the convergence criterion is met. The final output is the optimized CA rule values for each township that model the temporal urban growth. The model is run over the images from 1973 till 1982 to calibrate its transition rules. The final results (Figure 8 and Table 2) indicate satisfactory results as compared to the crisp CA method.

6. CONCLUDING REMARKS

This work explores the potential of implementing the cellular automata to model the historical urban growth over Indianapolis. The main goal is to design the model as a function of local neighbourhood structure to minimize the input data to the model. Satellite imagery represents the medium over which the model works. One important issue our model takes into account is the calibration process. Two modules are introduced namely, spatial and temporal calibrations. Spatial calibration fits the model on a township basis to its site specific feature while the temporal calibration adapts it to the urban growth dynamic change over time. This shows a noticeable effect on producing a good spatial match between the real and simulated image data. On the other hand, genetic algorithm is introduced to enhance the CA calibration process. GA makes the calibration process more efficient through manipulating a set of feasible solutions in the search space to find an optimal solution. This will reduce the search space for the optimal rules' values on a township basis.

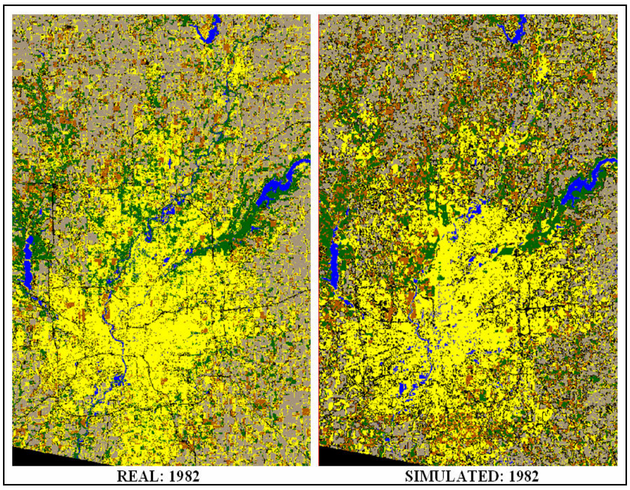


Figure 8. GA-CA calibrated image for year 1982

Table 2. GA-CA calibrated results for year 1982

Township#	Urban Pixel #	Accuracy(%)
1	2221	82.23%
2	3522	96.86%
3	3551	75.07%
4	4380	99.03%
5	5206	122.64%
6	4243	67.77%
7	8116	97.29%
8	4810	78.76%
9	4486	95.26%
10	12128	97.81%
11	10967	88.25%
12	4035	72.99%
13	5787	83.06%
14	13256	80.27%
15	19409	99.88%
16	12193	109.67%
17	8235	93.06%
18	19297	99.79%
19	17060	93.26%
20	6305	71.37%
21	1973	78.61%
22	4594	66.17%
23	6620	83.28%
24	4858	82.65%
Average		87.63%
Std. Dev		13.73%

REFERENCES

Batty, M., and Xie, Y., 1994a. From cells to cities. *Environment and Planning*, B21, pp. 531-548.

Batty, M., 1998. Urban evolution on the desktop: simulation with the use of extended cellular automata. *Environment and Planning A*, 30, pp. 1943–1967.

Clarke, K. C., Hoppen, S., and Gaydos, L., 1997. A self-modifying cellular automaton model of historical urbanization in the San Francisco Bay area. *Environment and planning B*, 24, pp. 247-261.

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

Hagerstrand, T., 1967. *Innovation Diffusion as a Spatial Process* (University of Chicago Press, Chicago, IL).

Holland, J.H., 1975. *Adaptation in Natural and Artificial Systems*, Univ. of Michigan Press, Ann Arbor, MI.

Li, X., and Yeh, A. G. O., 2003. Error propagation and model uncertainties of cellular automata in urban simulation with GIS. In: *7th International Conference on GeoComputation*, 8- 10 September 2003, University of Southampton, Southampton, UK (GeoComputation CD-ROM).

Tobler, W., 1979. Cellular geography, in Philosophy. In: *Geography Eds S Gale, G Olsson* (D Reidel, Dordrecht), pp. 379-386.

von Neumann, J., 1966. *Theory of Self-Reproducing Automata*. University of Illinois Press, Illinois. Edited and completed by A. W. Burks.

Waldrop, M., 1992. Complexity: The Emerging Science at the Edge of Order and Chaos (New York: Simon and Schuster).

Wolfram, S., 1994. Cellular automata. In: *Cellular Automata and Complexity: Collected Papers* (Addison Wesley, Steven Wolfram, Reading, MA).

Yang, X., and Lo, C. P., 2003. Modelling urban growth and landscape changes in the Atlanta metropolitan area. *International Journal of Geographical Information Science*, 17, pp. 463-488.