

Genetic Algorithm Calibration of Probabilistic Cellular Automata for Modeling Mining Permit Activity

Sushil J. Louis
Genetic Algorithm Systems Laboratory
Department of Computer Science
University of Nevada, Reno, NV 89557
sushil@cs.unr.edu

Gary L. Raines
U.S. Geological Survey
MS 176 c/o Mackay School of Mines
University of Nevada, Reno, NV 89557
graines@usgs.gov

Abstract

We use a genetic algorithm to calibrate a spatially and temporally resolved cellular automata to model mining activity on public land in Idaho and western Montana. The genetic algorithm searches through a space of transition rule parameters of a two dimensional cellular automata model to find rule parameters that fit observed mining activity data. Previous work by one of the authors in calibrating the cellular automaton took weeks - the genetic algorithm takes a day and produces rules leading to about the same (or better) fit to observed data. These preliminary results indicate that genetic algorithms are a viable tool in calibrating cellular automata for this application. Experience gained during the calibration of this cellular automata suggests that mineral resource information is a critical factor in the quality of the results. With automated calibration, further refinements of how the mineral-resource information is provided to the cellular automaton will probably improve our model.

1. Introduction

We use a genetic algorithm to calibrate a cellular automaton designed to model spatially resolved mineral-related activity in Idaho and Western Montana. The model needs to account for when and where mining permits will be issued in order to support planning for resource and land management on public lands. Any technique for modeling exploration for mineral resources must account for the following properties:

- Public land is not uniformly distributed
- Exploration permits are only required for public land
- Mineral resources constitute the product of rare events that are irregularly distributed

- Most mining-industry activity occurs near previous or current exploration activity because of the mature state of knowledge about mineral resources in Idaho and western Montana.

It is not enough to model how many mining permits will be issued in a particular year; more importantly, we need to model the locations where these permits will be issued. A two dimensional cellular automaton (CA) maps well to locations in two dimensional space and thus provides a natural way to model spatially resolved events[19]. Our CA uses a modified annealed voting rule that simulates permit activity with a spatial resolution of one mile squared and a temporal resolution of one year [14].

However, calibration of this CA is a complex and time-consuming process that would benefit from an automated approach and help facilitate further research. For example, the calibration bottleneck prohibits running what-if scenario analysis for decisions support in resource land management. In contrast to the current manual approach, our genetic algorithm calibrator takes a day to calibrate the cellular automaton to fit observed data as well as a geologist using trial and error. This paper describes our genetic algorithm approach to cellular automata calibration and provides good evidence that genetic algorithms are a viable alternative to manual calibration of large cellular automata.

The next section provides an introduction to the application area (resource/land management) and the use of cellular automata in spatial modeling. We then describe our CA's rules and the issues in using a genetic algorithm to calibrate these rules. Section 4 discusses our results and the last section gives conclusions and future work.

2. Spatial Modeling with Cellular Automata for Land and Resource Management

The Federal Government has mandated the development of plans to guide natural resource management activities

for 10-15 year periods. The first round of planning was conducted in the 1980's, and the second round of planning has now started. The US Geological Survey (USGS) is providing the US Forest Service (USFS) minerals-related data and interpretations to assist in plan revision for western Montana and Idaho. The USFS is particularly interested in USGS forecasts that indicate surface disturbances - mining activity is a good indicator of surface disturbances.

Our model makes several simplifying assumptions that are sketched below. We assume that Idaho and western Montana represent a mature exploration environment. This area has been mapped and studied and as a result, many large deposits have been found in successive iterations of exploration. Next, exploring near large, discovered deposits reduces risk [11]. We expect future exploration near significant, known deposits and in areas with a history of exploration. We expect exploration near areas with positive evidence of mineralization or with known mineral resources. Finally, metal prices and mining costs will remain like the last ten years. This is a significant assumption because US mining regulations changed in 1992 and subsequently mineral activity declined and became more focused in smaller areas. Our model takes this into account by treating 1993 (when the changes took effect) as a singularity and using different CA rules for the years before 1993 and for the years after 1993. Note that a relatively quick GA calibrator allows us to analyze scenarios with different assumptions.

We incorporate these assumptions into the transition rules of a two dimensional cellular automaton [14, 19]. A 2D CA represents space as a uniform grid of cells where each cell contains a small amount of information (transition rules). Application of these transition rules represents time. Cellular automata were popularized by Conway's game of life and have since been applied in many areas [4, 14, 18]. More recently, cellular automata based spatio-temporal models have been used for urban planning [2, 3], forest fire management, modeling lava flow, environmental modeling, and complex systems modeling [9, 12, 17, 19]. Takeyama and Couclelis provide a formal presentation of the mathematics of cellular automata in a GIS context [13].

Wolfram's recent book shows how simple transition rules can lead to complex behavior [19]. Our problem is the inverse - to find a set of rules that will lead to observed spatially and temporally resolved behavior. In our CA, rules operate over a 3x3 neighborhood of nine cells including the center as its own neighbor. A cell can be in one of four states: stayed active, stayed inactive, newly active, and newly inactive. Many rules can be defined over this 3x3 neighborhood, for example, a simple rule may be: if more than 5 cells in the neighborhood are active, then the center cell state has a 0.5 probability of being set to active.

No. of active neighbors	Current state	
	Active	Inactive
N > HIGH	V. LIKELY	LIKELY
LOW < N < HIGH	LIKELY	S. LIKELY
N < LOW	V. S. LIKELY	UNLIKELY

Table 1. The probabilistic annealed voting rule

We constrained the problem by restricting our search to a modified annealed voting rule which has worked well in the past [10, 14]. Table 1 depicts the transition rule set.

To interpret the table as a set of rules, consider the first row. This defines two rules: (1) **If** N, the number of active neighbors, is above *high* **and** the current state of the center cell is **active**; **Then** set the next state of the center cell to active with a probability corresponding to Very Likely. (2) **If** N, the number of active neighbors, is above *high* **and** the current state of the center cell is **inactive**; **Then** set the next state of the center cell to active with a probability corresponding to Likely.

In the table, *High* and *Low* are upper and lower bounds of the anneal window, V means "Very" and S means "Somewhat." These imprecise concepts are made more precise by assigning a fuzzy logic interpretation of their meanings [20, 15]. Thus the *VeryLikely* probability is calculated as the square root of the *Likely* probability. In the same way, *VerySomewhatLikely* is calculated as the square root of *SomewhatLikely*.

We have distilled and incorporated domain knowledge from USGS databases on mineralization, past permit activity, exploration activity, and activity supported by the Defense Mineral Exploration Administration into our model through the use of a resource grid. There is a one to one correspondence between cells in our CA and cells in the resource grid. Resource grid cell values indicate the possibility of permit activity and CA cells can be active only if the corresponding resource grid threshold value is over a parameter that we call the **resource threshold**.

Finally for cells with fewer than nine (9) neighbors (border cells), we scaled the number of neighbors in a particular state linearly. Thus if a cell had only 5 neighbors and 2 were active we would scale the number of active cells to be $9/5 \times 2 = 3.6$ which rounds off to 4 active neighbors.

From this discussion and the table above (Table 1) we need values for the following parameters (and their ranges) to completely specify the state transition rule for our CA. Likely [0..1], Somewhat Likely [0..1], Unlikely [0..1], Upper bound on anneal window [0..9], Lower bound on anneal window [0..9], and the Resource threshold [0..1]. Note that we essentially create one CA model (set of transition rules) for the years 1988 - 1993 and a different CA model for the

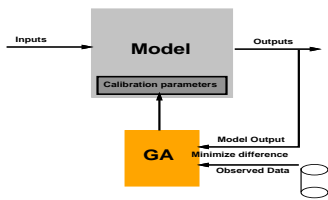


Figure 1. The GA searches through the space of parameter values to calibrate the model to observed data.

years from 1993 - 1998.

Finding good values for these two sets of parameters took one of the authors two weeks through trial and error. We next describe the genetic algorithm used for searching the parameter space for good parameter values. The GA provides equivalent (to human expert) or better parameter values within a day.

3. Calibrating CAs with GAs

Genetic algorithms are randomized search methods that search from a population of points and use simple operators modeled on natural selection to generate subsequent populations and make progress [5, 8]. These simple operators work on string representations and individuals in the genetic algorithm's population are usually binary strings. Although randomized, genetic search is guided by the relative differences in (application-dependent) fitness of members of the population. Each member of the population represents a possible solution to the problem. Genetic algorithms maximize a non-negative fitness and for our problem, fitness is defined from the difference between modeled and observed permit activity data.

Genetic algorithms have been used extensively for model tuning or calibration, and there is much empirical evidence in support of their use in this kind of problem [6, 7, 16]. Figure 1 depicts how the GA is used to calibrate the cellular automata model. Specifically, the GA searches for a set of values of the six parameters that define the CA's transition rules and each individual in the population thus encodes for possible values of these six parameters. Figure 2 shows how we encode the six parameters into a binary string.

Evaluating an individual results in a measure of fitness and proceeds as follows: Extract the encoded parameter values for the individual, then, run the cellular automaton model with these parameter values. For each year of activity simulated by the CA, compare the number of cells in state X in the CA model with the number of cells in state X given by USGS observations. The closer the two numbers, the better the CA model agrees with observed data and the higher the fitness of this individual. If we think of

the GA as an optimization procedure, then the GA is minimizing the following objective function:

$$\text{Minimize } g = \sum_{j=0}^{\text{nyears}} \sum_{i=0}^{\text{nstates}} \left(100 \times \frac{|M_{ij} - O_{ij}|}{M_{ij} + O_{ij}} \right)$$

where M_{ij} is the number of cells in state i in year j as predicted by the model and O_{ij} is the number of cells in state i in year j as given by USGS observations. Thus the objective function is a simple error minimization function normalized over the number of cells in each state. Since the transition rules are probabilistic, we ran the CA model three times and took the average over these three runs as the value of g .

Any minimization problem can be turned into an equivalent maximization problem by the principle of duality. Thus, our genetic algorithm maximizes a fitness given below

$$\text{Maximize fitness} = f(g) = C_{max} - g$$

where C_{max} is a constant chosen large enough to ensure non-negative fitnesses.

We used the three-operator genetic algorithm as described in Goldberg's book [5] but modified the selection operator. In our elitist selection strategy, if the population size is N then, the offspring produced by crossover and mutation initially double the population. The new generation consists of the best N individuals from the combined $2N$ parents and offspring. Individuals are chosen for crossover and mutation using fitness proportional selection with linear fitness scaling [5]. This selection strategy induces strong convergence and needs to be balanced by high crossover and mutation rates.

4. Results

We compare the performance of the CA model with parameter values found by the genetic algorithm against the performance of the CA model with the best parameter values found by one of the expert authors. We considered modeling the years from 1988 - 1993 as a problem that was separate from modeling the years from 1993 - 1998. Thus we find two sets of CA transition rule parameter values. Since

High	Low	Likely	S.Likely	Unlikely	RT
4	4	7	7	7	7

Figure 2. Encoding the six parameters for the GA. High and low define the anneal window and require 4 bits. The probabilities: *Likely*, *SomewhatLikely*, and *Unlikely*, as well as the Resource Threshold (RT) require seven bits each to be accurate to two decimal places.

we did not change anything in the genetic algorithm for either of the two problems, the discussion below applies to both.

We ran the genetic algorithm 10 times with different random seeds and used the best parameter values found out of these 10 runs. The genetic algorithm went through 75 iterations for each run. We used a population size of 50; two point crossover with a crossover rate of 1.0; point mutation with a mutation rate of 0.05.

A population size of 50 iterating 75 times requires $50 \times 75 = 3750$ fitness evaluations. Our cellular automaton covering Idaho and western Montana at a 1 mile resolution contained 249,488 grid cells. Executing the transition rule over this many cells 3 times per evaluation, 3750 times for each random seed, for each year of the simulation, took significant computational time - between 12 - 15 hours on a ten processor parallel cluster.

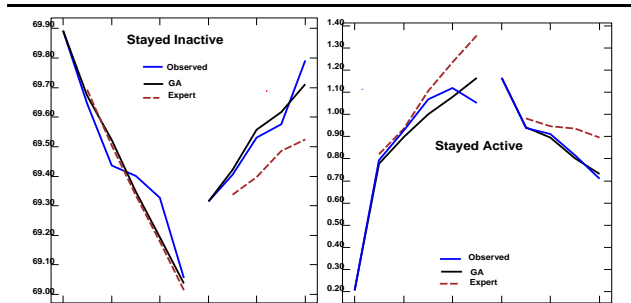


Figure 3. Left: Comparing the number of cells that stayed inactive. The Y axis is in thousands (10^3). Right: Comparing the number of cells that stayed active. The Y axis is in hundreds (10^2).

Figures 3 and 4 compare the performance of CA models calibrated by the expert and by the GA against the ground truth (USGS observations) in terms of the number of cells in a particular state. The x-axis is the year, starting with 1988 and ending with 1998 while the Y-axis measures the number of cells in a particular state.

Figure 3 (left) shows that both the expert and GA do about the same in modeling the number of cells that stayed inactive in mining's expansive years (1988 through 1993). However, the GA calibrated CA better models the rise in inactive cells during mining's contractive period. As the number of inactive cells rises we have fewer active cells which means fewer permits thus implying that mining activity is contracting. Note that the vast majority of cells (in the tens of thousands) remain inactive. Figure 3 (right) compares modeled and observed numbers of cells that stayed active. The genetic algorithm calibrated CA does better during both expansive and contractive years. It is interesting to note that

the GA calibrated CA does a very good job of modeling the cells that remain active during contractive years. Looking at

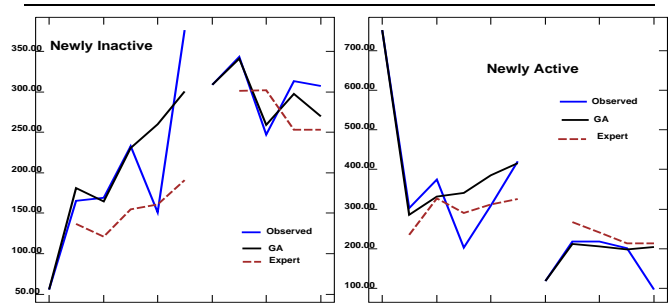


Figure 4. Left: Comparing the number of cells that became newly inactive. Right: Comparing the number of cells that became newly active.

Figure 4 (left) the GA calibrated CA does better in four of the five expansive years and all of the contractive years. Finally, the number of newly active cells is also better modeled by the GA calibrated CA which improves upon the expert in all but one year (the graph on the right of Figure 4). These figures are representative of the results from many

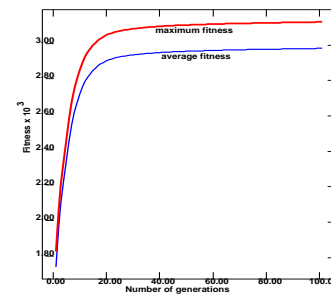


Figure 5. The genetic algorithm's typical convergence behavior.

runs of the GA and indicate that the GA is as good as or better than our expert at calibrating the cellular automaton. The calibrated CA (manually or by the GA) reproduces permit activity from 1989 - 1998 with an agreement of 94% - increasing to 98% for within one year. We find that analysing the confusion matrix and Kappa correlation statistics indicates that the CA underestimates high activity and overestimates low activity. Spatially, the major differences between the actual and calculated activity are that the calculated activity occurs in a slightly larger number of patches and is slightly more uneven than the actual activity.

The genetic algorithm converges as expected and Figure 5 displays this typical convergence behavior. The figure plots the average maximum fitness in each generation and the average of the average fitness of the population in each generation over the 10 runs with different random seeds.

5. Conclusions and Future Work

These results indicate that the genetic algorithm is a viable tool for calibrating a cellular automaton for modeling mining activity in Idaho and western Montana. To use a modified annealed voting rule, we used a genetic algorithm to find six parameter values; three probability parameters, the anneal window (two parameters), and the domain information derived resource threshold parameter. Good parameter values define the cellular automaton transition rules that lead to good models of permit activity in the region of interest. The genetic algorithm derived models do as well as, or better than, an expert at modeling permit activity in the region. These results agree well with other previous work that establish genetic algorithms as a good tool for calibrating models, and in particular cellular automaton models [6, 2, 3, 1].

We are currently moving towards developing a modeling tool for the USGS and making it accessible over the web. A simple visualization interface has been built and is available at <http://www.cs.unr.edu/~sushil>.

Condensing information from a number of sources into a single value in the resource grid simplifies the problem formulation. However, we believe that allowing the genetic algorithm to decide how to combine these information sources allows for more flexibility. Although the genetic algorithm model does a good job in matching the number and location of cells in a particular state, we would like to let the GA make use of spatial distribution to better match the locations of cells. In the long run, we expect this work to lead to better decision support tools for long term land management.

Acknowledgments This material is based in part upon work supported by contract number N00014-03-1-0104 from the Office of Naval Research.

References

- [1] P. W. Box. Garage band science and dynamic spatial models. *Journal of Geographical Systems*, 2(1):49–54, 2000.
- [2] K. C. Clarke, S. Hoppen, and L. Gaydon. A self-modifying cellular automaton model of historical urbanization in the San Francisco Bay area. *Environment and Planning B: Planning and Design*, 24:247–261, 1997.
- [3] Engelen, Guy, White, Roger, Inge, and Uljee. Integrating constrained cellular automata models, GIS and decision support tools for urban planning and policy making. In Timmermans, editor, *Decision support systems in urban planning*, pages 125–155. E&FN Spon, London, 1997.
- [4] M. Gardner. Mathematical games - the fantastic combinations of John Conway's new solitaire game: life. *Scientific American*, 223(4):120–123, 1970.
- [5] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
- [6] I. Golovkin, R. Mancini, S. Louis, K. Fujita, H. Nishimura, H. Shirga, H. Azechi, R. Butzbach, I. Uschmann, J. Deletre, J. Koch, R. W. Lee, and L. Klein. Spectroscopic determination of dynamic plasma gradients in implosion core. *Physical Review Letters*, 88(4), 2002.
- [7] I. Golovkin, R. Mancini, and S. J. Louis. Analysis of x-ray spectral data with genetic algorithms. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 75:625–636, 2002.
- [8] J. Holland. *Adaptation In Natural and Artificial Systems*. The University of Michigan Press, Ann Arbor, 1975.
- [9] Miyamoto, Hideaki, and S. Sasaki. Simulating lava flows by improved cellular automata method. *Computers and Geosciences*, 23:283–292, 1997.
- [10] G. L. Raines, M. L. Zientek, J. D. Causey, and D. E. Boleneus. Preliminary cellular-automata forecast of permit activity from 1998 - 2010, Idaho and western Montana. *Natural Resources Research*, to appear.
- [11] D. A. Singer and R. Kouda. Examining risk in mineral exploration. *Natural Resources Research*, 8(2):111–122, 1999.
- [12] L. T. Steyaert. A perspective on the state of environmental simulations. In M. Goodchild, B. O. Parks, and L. T. Steyaert, editors, *Environmental modeling with GIS*, pages 16–30. Oxford University Press, New York, 1993.
- [13] M. Takeyama and H. Couclelis. Map dynamics: integrating cellular automata and GIS through geo-algebra. *International Journal of Geographical Information Science*, 11(1):73–91, 1997.
- [14] T. Toffoli and N. Margolus. *Cellular automata machines - new environment for modeling*. MIT Press, Cambridge, MA, 1987.
- [15] L. H. Tsoukalas and R. E. Uhrig. *Fuzzy and Neural approaches in engineering*. John Wiley and Sons, Inc. New York, 1997.
- [16] J. Wallace and S. J. Louis. Taming a flood with a t-cup - designing flood control structures with a genetic algorithm. In *Proceedings of the 2003 Genetic and Evolutionary Computation Conference*, page To appear. AAAI Press, 2003.
- [17] J. Wilson. Special issue on the integration of GIS and environmental modeling. *Geographical Information Systems*, 9(4), 1995.
- [18] S. Wolfram. Cellular automata as models of complexity. *Nature*, 311:419–424, 1984.
- [19] S. Wolfram. *A New Kind of Science*. Wolfram Media Inc., 2002.
- [20] L. Zadeh. From computing with numbers to computing with words - from manipulation of measurements to manipulation of perceptions. *IEEE Transactions on Circuits and Systems*, 45:105–119, 1999.