

MODELAGEM E SIMULAÇÃO DO CRESCIMENTO URBANO UTILIZANDO AUTÔMATOS CELULARES

Vitor do Vale Bernardo
Universidade Federal de Goiás
vitor dovale.ufg@gmail.com

Douglas Farias Cordeiro
Universidade Federal de Goiás
cordeiro@ufg.br

Núbia Rosa Da Silva
Universidade Federal de Goiás
nubia@ufg.br

Resumo: Junto da criação dos Sistemas de Informações Geográficas (SIG), surgiram modelos para simulação da expansão de futuras áreas urbanas, onde alguns utilizam Autômatos Celulares (CA) para tal tarefa. Este artigo identifica trabalhos e aplicações na área e especifica as etapas para a idealização e desenvolvimento de um novo sistema semelhante. São coletados dados relevantes de estudos sobre as características e particularidades no crescimento das cidades em diferentes regiões, detalhando a construção das regras do autômato e a descrição dos passos para experimentação da solução desenvolvida. Por fim, os resultados obtidos são apresentados e brevemente discutidos, destacando o potencial da utilização de autômatos celulares enquanto alternativa de computação aplicada para o estudo de problemas relacionados ao crescimento urbano.

Palavras-chave: Crescimento urbano; Autômatos celulares; Computação aplicada; Modelagem.

Abstract: The development of Geographic Information Systems (GIS) enhanced the creation of models to simulate the expansion of future urban areas, where some use Cellular Automata (CA) for this task. This paper identifies works and applications in the area and specifies the steps for the idealization and development of a new similar system. Relevant data are collected from studies on the characteristics and particularities in the growth of cities in different regions, detailing the construction of the automaton rules and the description of the steps for experimenting with the developed solution. Finally, the results obtained are presented and briefly discussed, highlighting the potential of using cellular automata as an applied computing alternative for the study of problems related to urban growth.

Word Keys: Urban growth; Cellular automata; Applied computing; Modeling.

1 Introdução

De acordo com dados da Organização das Nações Unidas (ONU), a população urbana deve ir de 7.7 bilhões, registrado em 2019, para 9.7 bilhões em 2050 (ONU, 2019). Esse cenário acaba por provocar um processo de crescimento urbano irrefreável e irreversível, gerando impactos sociais e ambientais de grande escala. De acordo com Stanganini e Lollo (2018), o intenso processo de urbanização das cidades é um dos principais catalisadores dos problemas ambientais, seja pela destruição da vegetação, degradação do solo, ou mesmo pela diversificação do uso da terra.

Tal cenário demanda por soluções inovadoras, as quais possibilitem tomadas de decisão estratégicas no que se refere à garantia e promoção da cidadania sob os mais variados aspectos, desde serviços básicos à própria distribuição de áreas comerciais. Neste sentido, a construção de modelos para simulação de crescimento urbano baseados em autômatos celulares se torna uma interessante alternativa para o problema, sendo frequentemente utilizados para estes propósitos.

Desde sua idealização, nos primórdios da computação digital, os Autômatos Celulares (do inglês, *Cellular Automata - CA*) se mostram ferramentas eficazes para o trabalho em diversas áreas. Sua flexibilidade e o peso dado à atributos como espaço e tempo, tornaram essas estruturas altamente adaptáveis e bastante eficientes em variadas situações (BARRETO *et al.*, 2003).

A partir de então, essa ideia, agora já chamada de Autômatos Celulares (CA), foi se popularizando e atraindo a atenção de novos pesquisadores, por sua grande promessa de uma variada gama de aplicações, desde abordagens mais simples, chegando até trabalhos e métodos extremamente complexos, como algoritmos genéticos e redes neurais (MACRAE, 1999). Burks (1970) deu continuidade direta aos trabalhos de Von Neumann, trabalhando na Universidade de Michigan, nas décadas de 1960 e 1970.

Uma das definições de Autômato Celular os descreve como um conjunto de células dispostas em uma grade (matriz) de forma específica, que evolui através de iterações de tempo discretas, seguindo um conjunto de regras preestabelecidas e baseando-se nos estados de suas células vizinhas. As regras de transição são aplicadas iterativamente, pela quantidade ciclos de tempo desejadas (WEISSTEIN, 2018).

Os resultados alcançados por Von Neumann foram potencializados através do estudo desenvolvido por Ulam (1992), culminando no modelo hoje conhecido como Autômatos Celulares de duas e de três dimensões. Foi a partir dessa linha de pensamento que em 1970 o matemático John Conway sugeriu a noção do que se tornaria uma das primeiras e mais populares aplicações de CA, intitulada de “Jogo da Vida” (do inglês, *Game of Life*), apelidado pelos acadêmicos de *Life* (CONWAY, 1970). Conway conseguiu simular do modo mais simples possível as características que compõem a reprodução da vida.

Mais tarde, em 1982, Stephen Wolfram realizou uma série de estudos sobre o funcionamento dos autômatos celulares e os padrões descobertos ao observar sua execução (WOLFRAM, 1982). Os resultados desses estudos mostraram que eles poderiam ser utilizados para a criação de modelos para sistemas matemáticos físicos, biológicos e computacionais, mostrando que esses autômatos sempre se baseiam em regras simples, mas que geram resultados complexos e imprevisíveis, onde uma alteração mínima pode gerar uma mudança radical no comportamento do algoritmo. Outro exemplo de aplicação considerada relevante de CA, inclui os famosos autômatos celulares de John von Neumann, que se assemelham a arquitetura de mesmo nome utilizada nos computadores atuais.

Entrando no campo da geografia, as aplicações com CAs começaram desde a década de 1950 (BATTY *et al.*, 1997), seguindo na década subsequente com Jane Jacobs e seu estudo sobre as cidades americanas (JACOBS, 1961). Os pesquisadores perceberam que a estrutura dos CAs facilitava a representação de dados geográficos, distribuídos em forma de células de uma matriz (TOBLER, 1979). Além da representação, a análise e operações sobre esses dados também foi facilitada, por outra ferramenta computacional. Em 1968, surgiram os chamados “Sistemas de Informação Geográfica”, ou SIG (do inglês, *Geographic Information System - GIS*), primeiramente idealizado por Roger Tomlinson, apelidado de “pai do SIG” (TOMLINSON, 1968).

Não demorou para a técnica de CAs ser aplicada em conjunto à esses novos sistemas, resultando em SIGs que utilizam regras de modelos celulares para processar os dados do sistema em operações personalizadas pelo pesquisador (WHITE; ENGELEN, 1993). O sucesso deve-se ao fato dos mesmos fornecerem *frameworks* com a devida simplicidade e flexibilidade que permitem replicar resultados de modelos mais complexos. As estruturas geradas e manipuladas nesses modelos dispensam grandes quantidades de cálculos e pressupostos teóricos.

Os sistemas descritos auxiliam tarefas de prever o futuro formato de uma cidade, a maneira na qual o uso da terra se expande, a taxa de crescimento de sua população. Existem atualmente no mercado alguns sistemas já consolidados que atendem à essa demanda, mas o uso de Autômatos

Celulares para tal tarefa vem sendo bastante explorado nos últimos anos, por sua grande flexibilidade em se adaptar à novas condições de execução (UMBELINO; BARBIERI, 2010).

O principal objetivo desse trabalho é a construção de um modelo de CA para a simulação do crescimento de ambientes urbanos utilizando autômatos celulares, onde a representação do ambiente físico e espacial da cidade é feita através de células em uma matriz de tesselação e a simulação ficará a cargo das regras de transição do autômato que conduzirão as mudanças ao longo do tempo (gerações) no crescimento da cidade.

A criação do modelo demandará primeiramente o estudo e a identificação de características de outros modelos em trabalhos previamente realizados, a coleta dos dados mais relevantes e que possam ser aplicados no projeto em questão. Com o sucesso dessa etapa, o próximo passo é melhorar e incorporar essas características junto a novas que venham a ser propostas, compondo assim a geração do novo modelo. O foco especial deve ser a etapa de criação das regras de transição, que requer uma grande atenção pois é citada em muitos estudos como sendo a principal influência no sucesso ou fracasso do modelo proposto (ALMEIDA *et al.*, 2003).

2 Fundamentação Teórica

Os Autômatos Celulares representam um modelo matemático de sistemas naturais, consistindo em um conjunto de unidades, denominadas células, dispostas em uma grade ou matriz, com cada célula assumindo um valor inteiro finito e discreto. Seguindo um número também finito de etapa, chamadas de iterações, as células têm seus valores alterados de acordo com um conjunto preestabelecido de regras determinísticas. O valor escolhido para cada célula leva em consideração o valor das células na vizinhança em questão (PURNELLE, 2004).

Em seu formalismo, um CA é composto por uma quintupla, descrita no Quadro 1, sendo seus cinco campos:

- O conjunto das células;
- O conjunto de possíveis estados;
- O conjunto de vizinhos de cada célula;
- O conjunto das regras de transição do CA;
- O espaço celular, que representa a matriz completa de células, cada uma delas já contendo seus respectivos estados.

$CA = (I, Q, N, \delta, (I, Q))$, onde	
<ul style="list-style-type: none">➤ Células $\rightarrow I$➤ Estados $\rightarrow Q$➤ Vizinhos $\rightarrow N$	<ul style="list-style-type: none">➤ Regras de transição $\rightarrow \delta$➤ Espaço celular \rightarrow par (I, Q)

Quadro 1. Formalismo de um autômato celular.

Fonte: autores.

O que é chamado de vizinhança de cada célula em um CA corresponde às células próximas, podendo incluir também a própria célula. Em alguns modelos considera-se apenas as células adjacentes da mesma linha e coluna, como o modelo de Von Neumann, ou até mesmo com uma

abrangência maior de células, seguindo a mesma lógica de apenas linhas e colunas, porém considerando uma sequência maior de células, na chamada vizinhança estendida de Von Neumann.

Enquanto isso, outros modelos também consideram as células das diagonais para uma aproximação mais real, como o modelo de Moore ou o modelo radial. Esses dois modelos são mais indicados por fornecerem resultados mais próximos da realidade, por representarem melhor um ambiente tridimensional, não sendo então limitados à apenas as duas dimensões de largura e comprimento, como no modelo de Von Neumann.

Quanto à divisão de tipos de CAs, Wolfram (1982) primordialmente definiu que existem quatro classes de CAs, divididas de acordo com o comportamento do CA durante sua execução. Essa classificação foi uma das primeiras a considerar como parâmetro a complexidade das regras de transição. Outra divisão nos tipos de CAs classifica essas estruturas em dois, unidimensionais e bidimensionais. A divisão se baseia na quantidade de estados que cada célula pode assumir, assim como em quantas células vizinhas são consideradas durante a aplicação das regras de transição. Um Autômato Celular Unidimensional com dezoito estados em cada célula e interações entre vizinhos mais próximos (vizinhança de Von Neumann) foi mostrado equivalente à máquina de Turing universal mais simples conhecida (WOLFRAN, 1982).

Um tipo especial de Autômato Celular Unidimensional é o chamado Autômato Celular Elementar, onde apenas existem dois possíveis estados (0 e 1) e a única regra que determina o estado de cada célula na próxima geração depende apenas do estado atual dessa célula e dos seus dois vizinhos mais próximos, da esquerda e direita. É considerado um dos modelos de computação mais simples possíveis.

Crescendo em um formato que lembra uma “pirâmide”, este CA pode assumir $2^3 = 8$ configurações possíveis, quando se leva em consideração cada célula, em conjunto com os vizinhos da esquerda e direita. Após a aplicação de cada regra, o CA pode então assumir 256 configurações diferentes, onde Wolfram propôs um esquema para ligar cada uma dessas configurações à uma determinada regra, numeradas de 0 a 255, esquema que seria futuramente conhecido como o Código de Wolfram (WOLFRAN, 1983).

Em alguns casos, onde a representação de um CA não é mais um modelo linear de células como no Unidimensional, tendo-se uma nova representação, em forma de uma grade retangular ou hexagonal de células. Este é o chamado Autômato Celular Bidimensional, que é o tipo de CA que será explorado neste trabalho. Em um CA bidimensional, pode-se ter os casos onde serão consideradas uma célula central e as quatro células próximas na horizontal e vertical, formando a chamada “vizinhança de cinco células” (vizinhança de Von Neumann). Teremos também casos onde considera-se uma célula central e as oito células mais próximas, formando a “vizinhança de oito células” (vizinhança de Moore).

3 Metodologia

O algoritmo para simulação do CA, construído sobre a linguagem de programação Python (versão 3.6.4) em conjunto com a biblioteca gráfica nativa *TkInter*, representa como um sistema que simula a expansão de um perímetro urbano se comporta. A cidade é representada em forma de matriz, com células de tamanhos regulares. Cada célula contém um valor que representa qual é o uso predominante do solo naquela área.

Na matriz, é considerado que o uso do solo é composto de oito categorias diferentes: vegetação, água, rodovia, rua, residências, comércios, indústrias e parques. Uma célula é considerada vazia quando seu uso corresponde à vegetação (valor 0), enquanto outros estados indicam que a célula está ocupada. O crescimento só é realizado em células consideradas vazias. Uma vez que a célula assume o estado de ocupada, ou em outras palavras, diferente de vegetação

(valor 0), seu valor deve permanecer fixo, não podendo mais ser alterado em futuras gerações do CA.

A representação na matriz de cada uma dessas áreas no algoritmo é feita por um número associado ao estado de cada célula (variando de 0 a 7 no código):

- Vegetação - célula vazia, sendo esse o único estado que futuramente poderá assumir outro valor/uso de terra;
- Água - representa mares, rios, lagos, lagoas. Este estado é fixado no início da execução do CA, sendo assim não é permitido o crescimento ou destruição de células com este estado;
- Rodovia - representa ruas de alta densidade;
- Rua - representa ruas de baixa densidade;
- Residência - indica qualquer tipo de uso residencial do solo, seja de baixa ou alta densidade;
- Comércio - corresponde a todos os tipos de comércios: lojas, pequenas conveniências, grandes centros comerciais, etc;
- Indústria - este estado também é fixado no início da execução do CA, por isso as áreas industriais também não sofrerão alterações com o passar das gerações;
- Parque - representa áreas de lazer, reservas, zoológicos, ou qualquer outro tipo de área protegida, que também não sofrerão alterações entre as gerações do CA.

Já a representação gráfica resultante, representada por uma matriz de 120 linhas por 120 colunas (totalizando 14.400 células), conta com diferentes cores para cada estado das células: Vegetação - verde; Água - azul claro; Rodovia - cinza claro; Rua - cinza escuro; Residência - azul escuro; Comércio - vermelho; Indústria - marrom; Parque - rosa. Durante a simulação, é aplicada a regra de vizinhança de Moore, ou "vizinhança de oito células", onde as células consideradas vizinhas são todas as oito células ao redor da célula atualmente analisada, considerando então células na horizontal, vertical e diagonal.

Como a entrada dos dados requer como parâmetro uma matriz de números inteiros, optou-se inicialmente pelo CSV (do inglês, *comma separated values*), formato de arquivo comumente utilizado para salvar tabelas em editores de planilhas. Neste formato, cada linha do arquivo de texto CSV representa uma linha na matriz, sendo a quebra de linha o delimitador para as linhas da matriz. As células são separadas em colunas por outro delimitador, onde a escolha padrão normalmente é uma vírgula. As funções utilizadas para leitura e escrita do arquivo CSV com os dados utilizados no programa, são funções padrões do próprio módulo CSV para Python, criado para uma melhor manipulação desse formato de arquivos na linguagem. São elas as funções `csv.reader()` para a leitura dos dados e `csv.writer()` para a escrita no arquivo CSV.

A segunda forma de entrada dos dados escolhida para o programa requer uma quantidade maior de procedimentos, pois trata-se da leitura de uma imagem em cores representando o mapa de entrada para o programa. Antes de tudo é necessário um pré-processamento, ou seja, essa imagem precisa ser previamente colorida em programas de edição de imagens, para que as cores correspondam ao espectro de cores reconhecido pelo autômato.

Quanto ao código, para a função que realiza essa leitura das imagens, foi escolhida a biblioteca PIL (*Python Imaging Library*), uma biblioteca de manipulação de imagens para Python. A função `get_pixel()`, dentro de um laço de repetição para percorrer toda a imagem, realiza a extração dos valores de cores da imagem, retornando três números inteiros, que variam de 0 a 255, que representam a quantidade de vermelho, verde e azul (RGB) em cada pixel lido.

Os valores RGB, retornados para cada posição da matriz principal do programa, passam então em um último teste, para saber qual o uso do solo naquela região. Um intervalo de cores é associado a cada um dos oito estados presentes do CA. Caso exista alguma cor que não corresponda à nenhum dos intervalos de cores presentes no CA, o programa então associa àquela região o estado padrão de *vegetação*. O retorno então será uma matriz contendo apenas números inteiros, num intervalo de 0 a 7, contendo os usos do solo presentes na imagem processada.

Na construção das regras de transição do CA, foi necessário levar em consideração uma série de fatores geográficos. Para construir um modelo para ter uma visão de como seria o padrão da cidade no futuro, era necessário realizar o caminho inverso, analisando informações do passado, sobre como foi o crescimento da região estudada. De acordo com estudos Yang *et al.* (2013), e White e Engelen (1993), após a realização de uma série de análises em dados existentes sobre as cidades (em sua grande maioria mapas de diversos anos), foi possível notar o aparecimento de alguns padrões nesses dados. Padrões esses que afetam especialmente o crescimento do uso de solo residencial e comercial nas cidades.

Além disso, é importante pontuar que Mavroudi (2007) realizou a identificação de alguns desses parâmetros. São padrões que puderam ser identificados em mapas históricos de cidades da região, em especial na cidade de Salônica, no litoral da Grécia. Os parâmetros identificados são:

- Grande índice de imigração nas últimas duas décadas (1987 - 2007), resultando em um aumento na população urbana;
- Deslocamento populacional para áreas periféricas, com menores valores de terrenos e melhor ambiente, com menos problemas relacionados ao trânsito;
- População no centro da cidade permanece estável, pois o número de imigrantes que chegam à essa área compensa o número de pessoas que deixam a região, rumo às periferias;
- Grande aumento no uso comercial, principalmente nas regiões das principais estradas;
- Área industrial permanece fixa, com um pequeno desenvolvimento, porém um crescimento quase irrelevante em relação a outras áreas;
- Grande presença de áreas de cultivo e pastagem em áreas periféricas, além também de conter algumas áreas protegidas.

Após esse levantamento de dados, é possível então iniciar a definir quais serão as regras que definirão o CA. O primeiro ponto a ser analisado é o que diz a respeito ao uso industrial do solo. Nos trabalhos citados anteriormente, o uso industrial é desconsiderado devido ao fato de que, uma vez que as indústrias se estabelecem na cidade elas permanecem estáveis, não sofrendo alterações consideráveis nas gerações seguintes. Em outras palavras, será considerado que as áreas industriais não alteram com o tempo.

Outro aspecto a ser observado diz respeito às áreas onde o crescimento de novas células é restrito. No CA desenvolvido, essas áreas são as células de água (rios, oceanos, lagos, etc.), indústrias (que são fixas pelo motivo citado anteriormente) e áreas de parques e reservas florestais, que são consideradas áreas protegidas e não permitem o desenvolvimento de quaisquer construções na região. Também estão excluídas do crescimento as células de ruas e rodovias, pois estas serão predefinidas na entrada dos dados.

Além então das células de vegetação, que serão consideradas como células “vazias” quanto ao uso do solo, as células dinâmicas escolhidas serão as de uso residencial e comercial do solo. Estas são as regiões onde estão concentradas a grande maioria da massa de pessoas em qualquer cidade, seja grande ou pequena. São essas áreas que comportam o maior número de pessoas durante o dia, no expediente de trabalho ou em centros comerciais, como também durante o

período da noite, onde se encontram em suas residências, justificando assim a sua escolha para as células que terão crescimento permitido ao longo da execução do CA.

Após todas as observações e restrições analisadas, algumas das regras finais para a alteração das células no decorrer das gerações, durante a execução do CA, serão:

1. SE vizinhança = residência ≥ 3 & vizinhança \neq água, indústria \rightarrow residência;
2. SE vizinhança = rua ≥ 1 & vizinhança = residência $\geq 1 \rightarrow$ residência;
3. SE vizinhança = rua OU rodovia ≥ 1 \& vizinhança = comércio $\geq 1 \rightarrow$ comércio;
4. SE vizinhança = parque ≥ 3 & vizinhança \neq água, indústria \rightarrow residência;
5. SE célula = residência & vizinhança = comércio $>$ residência \rightarrow comércio;
6. SE vizinhança = residência ≥ 5 & vizinhança \neq água, indústria, parque \rightarrow comércio.

Para a aplicação das regras de transição, a escolha foi de criar uma série de condicionais, utilizando as estruturas condicionais (*if e else*), onde testes sobre qual o uso de cada célula e de suas oito células vizinhas são executados linearmente. Na forma como foi programado o CA, na linguagem Python, existe uma matriz principal **m** que contém todos os estados atuais das células e é onde os testes são realizados, uma segunda matriz **n** que é cópia dessa matriz principal, onde as alterações são realizadas. O programa também conta com matrizes auxiliares que guardam o número de vizinhos das células, contendo a quantidade de cada estado nessa vizinhança, para indicar quantas células estão ocupadas e quais são os seus respectivos estados.

Assim que a matriz principal é criada, e ao final da execução de cada geração, é realizado um teste em cada posição da matriz, para saber qual o uso atual de cada uma das células do CA. Como cada célula contém apenas o número inteiro identificador, de 0 a 7, de seu uso do solo, é realizado um teste condicional onde cada valor é associado à uma cor, para a representação gráfica da matriz do CA.

Quanto aos testes para aplicação das regras de transição, este é composto por uma sequência de testes. Inicialmente, realiza-se um teste para identificar se a célula em questão contém um estado que pode ou não ser alterado, evitando então futuros testes desnecessários e uma possível sobrecarga no código, em células que não poderão sofrer alteração em seu estado.

O segundo é um teste de igualdade, onde a célula selecionada e/ou sua vizinhança são iguais, maiores ou menores que um determinado estado escolhido. Em algumas regras, está presente também um terceiro e último teste, sendo este de negação, onde existe uma restrição que é aplicada à vizinhança da célula a ser testada. Caso os testes tenham um resultado falso, a configuração da região em questão não respeita as regras e então a célula analisada não sofre alterações naquela geração. Em casos contrários, as alterações serão realizadas nessas células, podendo também uma célula sofrer mais de uma alteração em uma mesma geração.

4 Experimentos e Resultados

Para a realização de alguns testes de experimentação do CA, alguns parâmetros foram definidos. Entre esses parâmetros está a quantidade de gerações ou iterações que serão executadas no CA, para gerar resultados satisfatórios. Essa quantidade foi definida de forma separada nas diferentes imagens de bases de dados, de forma a evitar situações em que a configuração inicial dos estados do CA não geraram mudanças significativas no estado final. Como citado por Wolfram (1982), essa situação ocorre principalmente em CAs de Classe 3, onde os estados evoluem de maneira descoordenada, de forma indefinida e imprevista, gerando em alguns casos demasiadas

alterações, e em outros mantendo praticamente a mesma configuração, com algumas mudanças locais.

As imagens dos mapas escolhidos das bases de dados em questão precisaram passar por adaptações para funcionarem como entrada para o programa. Foram realizados cortes em partes menores, desconsiderando regiões que pouco forneciam informações relevantes, que não permitiriam a geração de resultados satisfatórios para o trabalho. Essas regiões incluíam alguns cortes onde a maior parte (mais de 80%) da imagem correspondiam ao mesmo uso de célula e que não teriam nenhum resultado relevante, como por exemplo nas áreas mais laterais das imagens, contendo grandes concentrações de água ou de florestas. Também foi feita uma adaptação nas cores dos mapas, para corresponder às mesmas cores dos estados no programa.

Após os cortes necessários nas imagens e as adaptações na leitura de cores, ajustadas no programa, as imagens finais escolhidas para os testes foram duas de regiões diferentes de Detroit e Saskatoon e uma imagem de St. Thomas, totalizando dez imagens de entrada. Assim que as imagens são carregadas para o programa, a matriz que representa o CA é então preenchida com os respectivos resultados da função de leitura das cores.

Existem algumas observações a serem feitas em relação a função de leitura das cores dos mapas. Essa função, por considerar a cor mais predominante em cada área, trata-se de uma aproximação do resultado real. Antes mesmo da execução da simulação do CA já era possível notar, nas matrizes correspondentes geradas que as matrizes contavam com algumas diferenças entre o que foi lido dos mapas e o resultado transferido ao algoritmo, o que consequentemente impactou no resultado final da simulação.

Para a realização dos testes, foram executadas dez gerações para cada entrada de dados. Os resultados apresentados correspondem às gerações de número 1, 3, 7 e 10 de cada entrada. Um teste realizado em uma imagem da cidade de Detroit, EUA, gerou como resultado a matriz na Figura 1. Neste teste, havia um grande número de ruas e rodovias na imagem original, que não eram detectadas com exatidão pela função de extração das cores, gerando falhas na representação dessas estradas, que interferiram no resultado final.

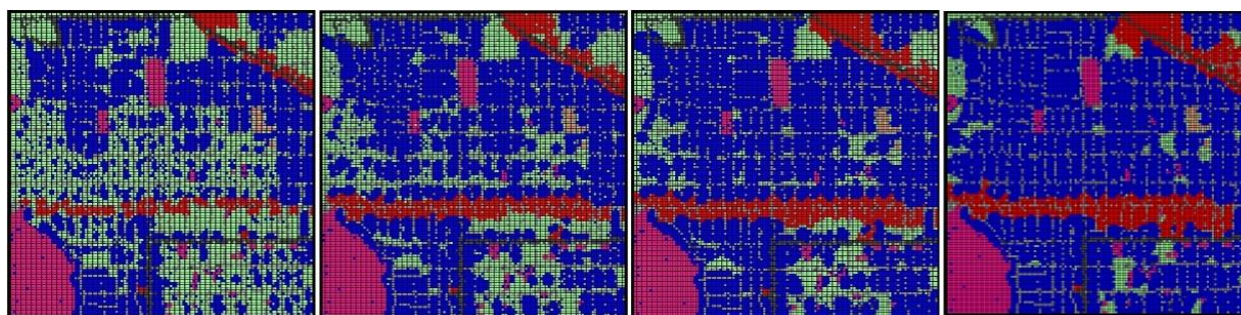


Figura 1. Resultados das gerações 1, 3, 4 e 7, da simulação no mapa de Detroit.
Fonte: autores.

5 Conclusões

Durante todo o decorrer do projeto e análise de trabalhos relacionados, é possível destacar a evidente e grande relevância que é dada à etapa de construção das regras de transição. Esta se inicia ainda durante o levantamento de dados e pesquisas sobre informações geográficas relevantes, que são analisados à procura de identificar os padrões que darão forma à essas regras, e permitirão replicar e simular o crescimento das cidades ao longo dos anos.

Durante a fase de testes, as simulações foram realizadas em diferentes cenários de cidades, que foram de grandes metrópoles e seus grandes centros industriais altamente desenvolvidos, até

a pequenos cidades, com um foco em uma indústria local de pequeno porte e grande presença da agricultura. Esta, porém, não é uma escolha comumente observada em trabalhos semelhantes, onde existe um foco maior em criar sistemas que tenham um funcionamento o menos genérico possível, para assim garantir uma maior taxa de acerto, focando apenas no cenário em que foram propostos mas, em contrapartida, perdendo desempenho quando aplicados em outras regiões.

O trabalho também mostrou que, mesmo partindo de um dos mais simples e primordiais exemplos, como o “Jogo da Vida”, foi possível a construção de um CA com regras mais variadas e complexas, através das devidas análises e coleta de dados no campo de estudo corresponde, de forma a aglomerar uma quantidade de informações suficientes para a criação dessas regras.

Como possíveis melhorias em um futuro projeto, pode-se citar as alterações necessárias na função de leitura das cores das imagens dos mapas, ou um pré-tratamento nas imagens para a remoção de ruídos deixados na retirada de textos e legendas dos mapas. Outra mudança que traria benefícios ao algoritmo, aprimorando o seu desempenho, seria a aplicação de *threads* para a execução dos testes das regras de transição de modo a acelerar essa etapa, responsável por grande parte da sobrecarga durante a execução do programa.

Referências

- ALMEIDA, C. M.; BATTY, M.; MONTEIRO, A. M. V.; CÂMARA, G.; SOARES-FILHO, B. S.; CERQUEIRA, G. C.; PENNACHIN, C. L. Stochastic cellular automata modeling of urban land use dynamics: empirical development and estimation. **Computers, Environment and Urban Systems**, v. 27, n. 5, p. 481-509, 2003.
- BARREDO, J. I.; KASANKO, M.; MCCORMICK, N.; LAVALLE, C. Modelling dynamic spatial processes: simulation of urban future scenarios through cellular automata. **Landscape and Urban Planning**, v. 64, n. 3, p. 145-160, 2003.
- BATTY, M.; COUCLELIS, H.; EICHEN, M. Urban systems as cellular automata. **Environment and Planning B: Planning and Design**, v. 24, p. 159-164, 1997.
- BURKS, A. W. **Essays on cellular automata**. Illinois: University of Illinois Press, 1997.
- CONWAY, J. The game of life. **Scientific American**, v. 223, n. 4, p. 120-123.
- JACOBS, J. **The Death and Life of Great American Cities**. New York: Vintage Books, 1961
- MACRAE, N. **John Von Neumann: The Scientific Genius who Pioneered the Modern Computer, Game Theory, Nuclear Deterrence, and Much More**. Providence, Rhode Island, USA: American Mathematical Society, 1999.
- MAVROUDI, T. **Simulating city growth by using the Cellular Automata Algorithm**. Phd Thesis, University College London, London, 2007.
- ONU. **World Population Prospects 2019: Highlights**. United Nations, 2019. Disponível em: <https://www.un.org/development/desa/publications/world-population-prospects-2019-highlights.html>. Acesso em 27 nov 2020.
- POURNELLE, J. **1001 Computer Words You Need to Know**. Oxonia, UK: Oxford University Press, 2004.
- TOMLINSON, R. F. A geographic information system for regional planning. In: G. A. Stewart (ed.), **Land Evaluation**. Melbourne, Australia: Macmillan, 1968. p. 200-210.
- STANGANINI, F. N.; Lollo, J. A. O crescimento da área urbana da cidade de São Carlos/SP entre os anos de 2010 e 2015: o avanço da degradação ambiental. **Revista Brasileira de Gestão Urbana**, v. 10, n. 1, p. 118-128, 2018.

TOBLER, W. R. Cellular geography. In: GALE, S.; OLSSON, G. (eds.) **Philosophy in Geography**. Theory and Decision Library (An International Series in the Philosophy and Methodology of the Social and Behavioral Sciences), Dordrecht: Springer, 1979. p. 379-386.

ULAM, S. **Adventures of a mathematician**. Berkeley, USA: University of California Press, 1992.

UMBELINO, G.; BARBIERI, A. Uso de autômatos celulares em estudos de população, espaço e ambiente. In: Encontro Nacional de Estudos Populacionais, n. 17, 2010, Caxambu-MG. **Anais...** Caxambu-MG: ABEP, 2010. p. 1–20.

WHITE, R.; ENGELEN, G. Cellular Automata and Fractal Urban Form: A Cellular Modelling Approach to the Evolution of Urban Land-Use Patterns. **Environment and Planning A**, v. 25, n. 8, p. 1175-1199, 1993.

WOLFRAM, S. **Cellular automata as simple self-organizing systems**. Tech Report. Pasadena: California Institute of Technology, 1982.

WOLFRAM, S. Statistical mechanics of cellular automata. **Reviews of Modern Physics**, v. 55, p. 601-644.

WEISSTEIN, W. **Cellular automaton**. Wolfram MathWorld, [sd]. Disponível em: <https://mathworld.wolfram.com/CellularAutomaton.html>. Acessado em 20 nov. 2020.

YANG, J.; ZHENG, X.; WEI, H. Modelling evolution of urban agglomerations based on cellular automata. International Conference on Natural Computation (ICNC), 9, Shenyang, 2013. **Proceedings...** Shenyang: ICNC, 2013, p. 691-696.