

CENTRO UNIVERSITÁRIO DE ANÁPOLIS – UniEVANGÉLICA
MESTRADO EM SOCIEDADE, TECNOLOGIA E MEIO AMBIENTE

WILLIAM PEREIRA DOS SANTOS JÚNIOR

**MODELAGEM ESPACIAL DE FLUXO DE ÁGUA UTILIZANDO
AUTÔMATOS HÍBRIDOS.**

Anápolis
2009

WILLIAM PEREIRA DOS SANTOS JÚNIOR

**MODELAGEM ESPACIAL DE FLUXO DE ÁGUA UTILIZANDO
AUTÔMATOS HÍBRIDOS.**

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Sociedade, Tecnologia e Meio Ambiente do Centro Universitário de Anápolis como parte dos requisitos para a obtenção do título de Mestre em Sociedade, Tecnologia e Meio Ambiente.

Área de concentração: Tecnologia e Meio Ambiente

ORIENTADOR: Prof. Dr. Nilton Correia da Silva

Anápolis
2009

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Júnior, William Pereira dos Santos

Modelagem Espacial de fluxo de água utilizando Autômatos Híbridos. / William Pereira dos Santos Júnior. – Anápolis: PP-GEE da UniEVANGÉLICA, 2009.

77 f.: il.

Dissertação (mestrado) – Centro Universitário de Anápolis. , Anápolis, BR–GO, 2009. Orientador: Nilton Correia da Silva.

1. Modelagem Computacional. 2. Fluxo Acumulado. 3. Autômato Híbrido. 4. Escoamento Superficial. I. Correia da Silva, Nilton. II. Título.

CENTRO UNIVERSITÁRIO DE ANÁPOLIS

Reitor: Profº. Me. Carlos Hassel Mendes da Silva.

Pró-Reitor de Pós-Graduação, Pesquisa, Extensão.: Prof. Dr. Francisco Itami Campos.

Coordenadora do Programa: Profª. Drª. Genilda Darc Bernardes.

WILLIAM PEREIRA DOS SANTOS JÚNIOR

**MODELAGEM ESPACIAL DE FLUXO DE ÁGUA UTILIZANDO
AUTÔMATOS HÍBRIDOS.**

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Sociedade, Tecnologia e Meio Ambiente e aprovada em sua forma final pelo Orientador e pela Banca Examinadora.

Orientador: _____
Prof. Dr. Nilton Correia da Silva, UniEvangélica

Banca Examinadora:

Prof. Dr. Nilton Correia da Silva, UniEvangélica

Prof. Dr. Osmar Abílio de Carvalho Junior, UNB
Avaliador Externo

Prof. Dr. Clarimar José Coelho, UniEvangélica

Anápolis, 05 de Fevereiro de 2009.

Dedico este trabalho a todas as pessoas que incentivaram-me a não desistir, em especial minha esposa Tatiana por sua compreensão nos momentos que não estive presente e pelos incentivos nas horas que a indolência vencia o meu entusiasmo e a minha filhinha Sophia que veio em um momento muito especial na minha vida, aos meus pais, William e Maria que desde o início da minha vida ensinaram-me a encarar com tenacidade os obstáculos da vida, a minha irmã Patrícia que mesmo nas horas de pouca presença fazia suprir com incentivos a sua ausência.

AGRADECIMENTOS

Ao todo poderoso, senhor arquiteto do universo nosso Deus, que proporciona a cada filho, suas bênçãos nas horas certas da vida, a ele meu agradecimento e reconhecimento às graças recebidas, sem ele não somos quem somos.

Ao meu ilustríssimo orientador, o professor Dr. Nilton por suas palavras corretas e ditas em momentos decisivos, pelo seu entusiasmo contagioso diante do tema do trabalho, por sua generosidade e compreensão pelos horários em que estive ausente do trabalho e por sua amizade.

Ao Centro Universitário de Anápolis representado na pessoa do Prof. Dr. Eliseu que como gestor administrativo financeiro da instituição incentivou-me a concluir esta pesquisa, minha imensa gratidão.

Ao Prof. Wagner Oliveira de Araujo de Ciência da Computação e Sistemas de Informação, por ter cedido gentilmente o arquivo *diss_mestado.tex* para a realização do trabalho de minha pesquisa, e pelas longas discussões a respeito das famosas normas da ABNT, o meu muito obrigado.

Ao Pessoal da UNB, os Professores Osmar e Roberto que contribuíram para o amadurecimento da pesquisa, ao Sr. Eusébio que ofereceu o espaço de pesquisa da Embrapa para realização de testes, os meus sinceros agradecimentos.

À todas aquelas pessoas que por algum motivo não acreditam em si mesmas, gostaria de dizer que o sonho concretizado é melhor que o sonho distraído.

"Chegará o dia que talvez as máquinas pensem, porém elas nunca terão sonhos."
Theodor Heuss

RESUMO

A modelagem de Fenômenos Ambientais é uma área de pesquisa ampla que proporciona várias oportunidades no âmbito da modelagem computacional. No sentido de simular os processos físicos desses fenômenos para o entendimento de suas dinâmicas, essa pesquisa utiliza as técnicas e conceitos de Autômatos Celulares e Autômatos Híbridos para construção de uma ferramenta computacional capaz de simular a dinâmica espacial do fluxo d'água proveniente do escoamento superficial hortoniano causado pelas precipitações. O simulador intitulado "AutoHidro" utiliza informações sobre o tempo precipitado aliado as informações sobre a altimetria do terreno juntamente com as teorias dos Autômatos Celulares e Híbridos para discretizar o escoamento superficial no espaço em um tempo de simulação pré-determinado, procurando representar o comportamento do escoamento superficial da água e determinar o volume do acúmulo do fluxo de águas pluviais no terreno.

Palavras-chave: Modelagem Computacional, Fluxo Acumulado, Autômato Híbrido, Escoamento Superficial.

ABSTRACT

The modeling of environmental phenomena is an area of extensive research that provides several opportunities in the context of computational modeling. In order to simulate the physical processes of these phenomena for understanding their dynamics, this research uses the techniques and concepts from Cellular Automata and Hybrid Automata to build a computational tool capable of simulating the spatial dynamics of water flow from the runoff Hortonian caused by rainfall. The simulator entitled “ AutoHidro” uses information about time ally precipitated the information about the terrain altimetry along with theories of Cellular Automata and Hybrids to discretize the runoff into space in a simulation time predetermined for representing the behavior of surface water runoff and determine the volume of flow from the accumulation of rainwater on the ground.

Keywords: Computational Modeling, Accumulated flow, Hybrid automaton, Cellular automaton.

LISTA DE ILUSTRAÇÕES

Figura 1:	Modelo de escoamento superficial Hortoniano e não Hortoniano. Fonte: (FRANK, 1995)	20
Figura 2:	Matriz regular retangular - GRID.	22
Figura 3:	Matriz Irregular Triangular - GRID.	23
Figura 4:	MDT com resolução espacial de $90m^2$ do município de Anápolis - GO, - Fonte: (MIRANDA, 2005)	24
Figura 5:	Representação do método de direção de fluxo D8.	25
Figura 6:	Escoamento do Fluxo com o método de direção de fluxo D8.	26
Figura 7:	Cálculo da direção de fluxo do método Dinfinito.	27
Figura 8:	Cálculo da declividade do método Dinfinito.	28
Figura 9:	Cálculo do ângulo (r) para a direção do fluxo e da força de declividade (s).	29
Figura 10:	Ilustração das posições das facetas e as constantes para o cálculo da direção de fluxo superficial.	30
Figura 11:	Representação gráfica de um autômato com dois estados.	30
Figura 12:	Exemplo de aplicação de um autômato finito.	31
Figura 13:	AC de Neumann. Fonte (SENNÁ CARNEIRO, 2004)	33
Figura 14:	Sistema de controle de temperatura. Fonte: adaptado de (SENNÁ CAR- NEIRO, 2004)	35
Figura 15:	Fluxograma de funcionamento do simulador AUTOHIDRO.	36
Figura 16:	Diferença de altitude entre dois pixels.	37
Figura 17:	Definição do Autômato Híbrido para os componentes do Sistema AutoHi- dro.	38
Figura 18:	Definição da Classe AutoHidro.	39
Figura 19:	Módulo de controle do sistema.	42
Figura 20:	Sobreposição das Imagens MDT, Área de Contribuição e Fluxo Acumu- lado com AH.	43
Figura 21:	Evolução do Escoamento Superficial: Testes realizados sem considerar as variáveis ambientais	43
Figura 22:	Evolução do Escoamento Superficial: Testes realizados sem considerar as variáveis ambientais	44
Figura 23:	Evolução do Escoamento Superficial: Testes realizados considerando as variáveis ambientais.	46
Figura 24:	Evolução do Escoamento Superficial: Testes realizados considerando as variáveis ambientais.	47
Figura 25:	Evolução do Escoamento Superficial: Testes realizados considerando as variáveis ambientais.	49

LISTA DE TABELAS

Tabela 1:	Classe de Testes nº 1: Considerando a Precipitação e a Saturação Inicial do solo.	42
Tabela 2:	Valores inseridos no simulador para teste, considerando o Limite de Saturação do Solo, a Infiltração e a Evaporação.	46
Tabela 3:	Valores inseridos para teste, considerando todas as variáveis ambientais. . .	48

LISTA DE ABREVIATURAS

ABNT	Associação Brasileira de Normas Técnicas
AC	Autômato Celular
AF	Autômato Finito
AH	Autômato Híbrido
DEMON	Digital Elevation Model Network
D_{∞}	DInfinito
HP	Hewlett-Packard
MDT	Modelo Digital de Terreno
MFA	Modelagem de Fluxo Acumulado
MS	Multiple Directions based on slope
SRTM	Shuttle Radar Topography Mission
TIM	Triangular Irregular Network

LISTA DE SÍMBOLOS

α_n	Proporção para cálculo da direção do fluxo
m^2	Metros Quadrados
e_i	Variável do valor de altitude da Célula
d_i	Variável da distância ente as Células vizinhas
S_1 e S_2	Variáveis que representam vetores de declividade
a_c	Constante definida para cálculo da direção das facetas
a_f	Constante definida para cálculo da direção do ângulo das facetas
i	Índices de linhas de uma matriz
j	Índices de colunas de uma matriz
p	Variável representando uma posição em relação ao espaço.
t	Variável representando uma variação do tempo em relação a transição de um estado para outro.
r	Ângulo de direção do fluxo superficial.
s	Variável que representa a força de declividade.
r_g	Variável a direção de fluxo superficial através do método D_∞ .
W e Q	Estados do autômato.
S_t	Arcos do autômato indicando a mudança do estado W para o estado Q .
\sum	Somatório do conjunto de eventos inseridos como entrada em um autômato híbrido.
X	Conjunto finito de variável reais passadas a um autômato híbrido.
x	Variável de entrada para o AH controlador de temperatura.
h	Variável para os valores de altitude do pixel.
t	Variável de tempo entre as altitudes do pixel.
θ	Ângulo da rampa de inclinação.
T	Variável do tempo de simulação do simulador AutoHidro.
T_n	Variável de tempo total de simulação do simulador AutoHidro.

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Modelagem	14
1.2	Delimitação do Tema	14
1.3	Motivação e Objetivos do Trabalho	15
1.4	Justificativa	16
1.5	Metodologia	17
2	FUNDAMENTAÇÃO TEÓRICA	19
2.1	Ciclo Hidrológico	19
2.2	Modelagem Ambiental	20
2.3	Modelo Digital de Terreno - (MDT)	22
2.4	Metodologias para o cálculo da direção de fluxo da água.	24
2.5	Autômatos	30
2.5.1	Autômato Finito	30
2.5.2	Autômato Celular	31
2.5.3	Autômato Híbrido	34
3	DESCRIÇÃO DOS MÉTODOS	36
3.1	Descrição dos Métodos	36
3.1.1	Definição do Sistema	36
4	RESULTADOS	41
4.1	Análise e interpretação dos Dados	41
4.1.1	Primeira Classe de Testes	42
4.1.2	Segunda Classe de Testes	45
4.1.3	Terceira Classe de Testes	48
5	CONCLUSÃO	50
	REFERÊNCIAS	51
	GLOSSÁRIO	53
APÊNDICE A	APRESENTAÇÃO DO CÓDIGO FONTE DA CLASSE AUTOHI-	
	DRO	54

1 INTRODUÇÃO

1.1 Modelagem

A modelagem permite construir procedimentos teóricos que envolve um conjunto de técnicas com a finalidade de representar simplificadamente acontecimentos naturais, como forma de compreensão do homem, perante a complexidade do ambiente em que ele está inserido. (CHRISTOFOLETTI, 1999).

Neste sentido um dos principais desafios do homem é criar métodos e técnicas capazes de imitar os processos e ciclos de fenômenos da natureza. Por tempos, o homem vem observando os fenômenos e tentando simular a dinâmica dos processos ambientais desses fenômenos. Dessa forma, diversas pesquisas foram desenvolvidas com o intuito de tentar imitar através de modelos artificiais o comportamento de tais processos ambientais, (CLARKE; HOPPEN; GAYDOS, 1997), (GREGORIO; SERRA; VILLANI, 1999), (SENNA CARNEIRO, 2004), (TAROLLI; TARBOTON, 2006) e (PEDROSA, 2003).

A modelagem ambiental tem como função principal a representação dos fenômenos da natureza e funda esboços para a preparação de novas hipóteses no contexto das teorias ou leis físicas. Os modelos podem assumir a formulação qualitativa ou quantitativa, expressa em termos lógicos ou matemáticos, e referem-se aos objetivos descritivos ou declarativos (CHRISTOFOLETTI, 1999).

1.2 Delimitação do Tema

A pesquisa conteve-se no estudo da modelagem de sistemas ambientais mais especificamente na modelagem computacional do comportamento hortoniano do fluxo acumulado, gerado pelo excesso de água no solo em função do processo de precipitação existente naquele instante. Para aplicação do modelo computacional utilizou-se modelo digital de terreno (MDT) que representa numericamente o relevo, dados do sensor *Shuttle Radar Topography Mission* (SRTM) (RODRIGUEZ et al., 2005) do município de Anápolis no estado de Goiás da carta

SE-22-X-B, retirado da base de dados da Embrapa solos¹ e monitoramento por satélite².

1.3 Motivação e Objetivos do Trabalho

A hidrologia é uma ciência que tem a água como objeto de estudo. A água circula entre a atmosfera e a superfície da terra formando o fenômeno do ciclo hidrológico. A precipitação é um evento do ciclo hidrológico e se forma com a mudança de estados da água, quando essa na atmosfera da terra passa do estado gasoso para o estado líquido formando a chuva. As principais características das precipitações são a quantidade precipitada, duração e distribuição espacial (BERTONI; TUCCI, 1997).

Ao determinar a intensidade da precipitação, pode-se prever eventos nocivos ao meio ambiente como a erosão e as enchentes. Outro evento ligado ao ciclo hidrológico, que pode ser considerado como variável a ser analisada é a infiltração. O solo é um meio poroso, a água ao atingí-lo segue vários caminhos sendo que boa parte infiltra-se no solo enquanto esse não fica saturado. Na ocorrência de saturação do solo, pode ser verificada a existência de outro evento importante para ser analisado chamado de escoamento superficial, proveniente do relevo das regiões de bacia e da gravidade que impulsiona para cotas mais baixas do solo (SILVEIRA, 1997).

Outro caminho da água em seu ciclo é a evaporação, evento de muita relevância para o estudo do ciclo hidrológico. A precipitação, infiltração, escoamento superficial e a evaporação se tornam informações importantes, inerentes ao ciclo hidrológico que devem ser consideradas no estudo para um modelo que represente de uma forma mais realística esse fenômeno, por formarem de início a fim, de forma interativa, todo o ciclo.

A proposta de um sistema que modele os eventos causados por fenômenos naturais de maneira que tenha uma interação realística com suas variáveis é um desafio. O conhecimento a cerca do fenômeno é imprescindível na hora de sua modelagem, pois o modelo pode ser diferente, seguindo a interpretação de cada modelador ou observador do fenômeno. O fenômeno é um fato ou situação concreta de interesse científico, descritivo ou explicativo, como os fenômenos meteorológicos (SENNA CARNEIRO, 2004).

A principal motivação da pesquisa é criar um modelo computacional que simule o fluxo da água, buscando representar a dinâmica de movimentação da água em função do espaço. Para isso a proposta é utilizar o método de direção de fluxo D8 (TARBOTON, 1997) para especificar a direção do fluxo, o MDT do município de Anápolis para representação do relevo e Autômatos Híbridos (AH) para discretização espacial da chuva.

Tal modelo seria útil para compreensão do comportamento do fluxo da água em áreas de bacia para o estudo do aproveitamento de água, possibilitando a quantificação dos recur-

¹<http://www.cnps.embrapa.br> (MIRANDA, 2005)

²<http://www.relevobr.cnpm.embrapa.br> (MIRANDA, 2005)

tos hídricos. Essa quantificação poderá contribuir em projetos e planejamentos dos recursos tais como: construção de usinas hidrelétricas para geração de energia, abastecimento de água, modelagem de redes de drenagem, navegação, delimitação de áreas de risco às inundações e erosões e transportes de sedimentos.

Os objetivos estabelecidos para a compreensão deste estudo foram, como objetivo geral, apresentar um modelo computacional espacial do fluxo acumulado das águas demonstrando a dinâmica do comportamento desse evento. Como objetivos específicos, fazer um levantamento do comportamento dos eventos precipitação e escoamento superficial hortoniano (HORTON, 1933. apud (MENDIONDO; TUCCI, 1997)) e da teoria dos Autômatos Híbridos de modo a simular a dinâmica do comportamento do fluxo acumulado de cursos d'água em área de bacia hidrográfica.

1.4 Justificativa

A modelagem de sistemas ambientais nasceu da preocupação de pesquisadores em compreender o comportamento dos fenômenos naturais, mas não somente a compreensão é importante, o prenúncio de seus resultados na superfície da terra também é significativo para a prevenção de acontecimentos desastrosos. Estudos da década de 60 mostram o uso de modelos aplicados à Geociência. A partir desta data apareceram diversas contribuições ressaltando o uso de modelos apresentando conceitos para a compreensão da dinâmica dos sistemas ambientais. Com o advento da computação, os modelos computacionais de simulação da dinâmica de processos ambientais se tornaram mais próximos da realidade.

A modelagem computacional para análise dos eventos relacionados com a dinâmica de processos ambientais teve seu ápice na década de 80 com diversas contribuições utilizando técnicas computacionais aplicadas à teoria das geociências. As técnicas de AH foram bem aceitas no âmbito da modelagem ambiental, por facilitar a mensuração os resultados da dinâmica ambiental em espaço e tempo, apresentando resultados similares aos da realidade.

Na década de 90 pesquisas apresentaram temas que tratavam da modelagem computacional da dinâmica urbana. Essas pesquisas apresentam a evolução de processos antrópicos, contemplando a evolução espaço temporal da construção de casas, indústrias e shoppings, no ano de 2006 estudos apresentaram modelos computacionais para demonstrar a dinâmica de prováveis pontos de deslizamentos e sua evolução no tempo. Estes estudo utilizaram a teoria de Autômatos Celulares (AC) para determinar a direção do deslizamento, imagens com informações digitais do relevo e as variáveis inerentes aos processos físicos próprios do deslizamento. Pesquisas com esses propósitos se mostram eficazes para a prevenção de problemas causados pelos processos físicos naturais.

Nos últimos meses, o Brasil inteiro acompanhou os acontecimentos causados pelas

chuvas em algumas regiões. Acompanhamos os esforços de todo o país no intuito de ajudar de alguma forma as pessoas que ali perderam tudo. Hoje, em cidades que são margeadas por rios as enchentes são muito comuns. Com base nas pesquisas relacionadas ao tema do trabalho e nos problemas hidrológicos como o mostrado acima, a proposta da presente pesquisa se justifica na apresentação de teorias que auxiliem o entendimento dos processos hidrológicos relacionados à dinâmica da água no solo e a apresentação de um modelo computacional que possa simular o escoamento superficial da água, apresentando seu comportamento espacial, podendo representar o que acontece no mundo real.

1.5 Metodologia

A pesquisa em sua natureza é considerada como pesquisa aplicada e foi feita por meio de um estudo abordando o método de pesquisa hipotético-dedutivo, pois possui um embasamento no estudo e observação dos fenômenos naturais, mais especificamente o ciclo hidrológico.

O volume de precipitação, o tempo precipitado e informações sobre a altitude do terreno apresentadas pelo Modelo Digital de Terreno (MDT), foram considerados como variáveis de entrada do sistema, pois configuram eventos quantificáveis. Outros eventos do ciclo hidrológico como infiltração e evaporação também foram abordados, entretanto não foram considerados como eventos variáveis, portanto foram inseridos como eventos constantes a todo o processo de simulação e são definidas pelo usuário do sistema no início da simulação. O fluxo acumulado foi calculado com base no modelo hortoniano de escoamento em função do volume e tempo precipitado e do relevo do solo apresentados pelo MDT.

Através da quantificação das variáveis precipitação e a declividade do terreno representada por MDT e em função do espaço em que ocorrem esses eventos, pôde-se fazer uma simulação do comportamento do fluxo das águas pluviais. Foi considerado para mensuração da precipitação os conceitos da pluviometria.

A pesquisa contou com um estudo descritivo, utilizando, para isso, modelos computacionais elaborados por pesquisa de laboratório, fundamentados por uma pesquisa bibliográfica ampla acerca do tema do trabalho. Através dos modelos computacionais foi possível verificar a eficácia das técnicas computacionais aplicadas ao estudo ambiental na simulação dos processos físicos. Foram utilizados os seguintes materiais:

- a)** Computador Hewlett-Packard (HP) com processador AMD Turion 64 x2;
- b)** Imagens do Satélite SRTM com informações de MDT;
- c)** Biblioteca de funções de David Tarboton;
- d)** Software C++ builder Versão 6.0;

e) Software Led Versão 0.526300;

f) Software Abílio Versão 2008.3.2.1;

Para a edição do trabalho científico utilizou-se editor de textos LED para compilação do texto em Latex, para a construção do modelo computacional utilizou-se linguagem de programação C++ editada e compilada com o software C++ Builder 6.0, para a análise da direção de fluxo e declividade do terreno foi utilizado as funções da biblioteca de (TARBOTON, 1997), utilizou-se para os testes, imagens de satélite do sensor SRTM com informações digitais do relevo. A metodologia científica adotada no trabalho baseou-se no livro de Marina e Eva (ANDRADE MARCONI; LAKATOS, 2001), para a formatação técnica do trabalho escrito seguiu-se as normas da Associação Brasileira de Normas Técnicas (ABNT).

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Ciclo Hidrológico

Hidrologia é a ciência que estuda a ocorrência, distribuição e movimentação da água na superfície da Terra, é uma ciência interdisciplinar que tem tido evolução significativa em face aos problemas relacionados ao uso ininterrupto da água e seu impacto sobre o meio ambiente (TUCCI, 1997a). A hidrologia então, se encarrega de estudar os fenômenos inerentes ao ciclo hidrológico. O ciclo hidrológico é o fenômeno global de circulação fechada da água entre a superfície e a atmosfera da terra, esse ciclo compreende eventos relacionados com a movimentação da água no ambiente e que se inicia com a ocorrência do fenômeno da precipitação (SILVEIRA, 1997).

A precipitação ocorre pelo acúmulo de vapor de água presente na atmosfera, que em determinadas condições atmosféricas pode condensar formando a chuva. A chuva é imprescindível para a hidrologia por sua capacidade de distribuição espacial ou o escoamento da água precipitada, a disponibilidade dessa água precipitada em uma bacia que pode determinar de forma quantitativa a necessidade de irrigação de culturas e o abastecimento de água para uso doméstico e industrial. Pela capacidade de produzir escoamento, a chuva é o tipo de precipitação mais importante para a hidrologia (BERTONI; TUCCI, 1997).

A quantificação da precipitação média em um região é mensurada através da pluviometria que mede a lâmina d'água em uma região sem considerar a infiltração, a evaporação e o escoamento da água. Essa mensuração só é possível com a utilização do pluviômetro que calcula a altura pluviométrica da precipitação. A duração da precipitação é imprescindível para determinar sua intensidade, pois representa a quantidade de chuva em função do tempo que geralmente é dado em minutos ou horas. Essa intensidade, apresenta uma variabilidade temporal, mas para a sua análise geralmente são considerados intervalos de tempo nos quais geralmente é considerada como evento constante.

Existe uma grande variação na distribuição temporal das chuvas durante as tempestades. Foram desenvolvidos alguns métodos como o método da média aritmética, o método de Thiessen, o método das isoietas entre outros, para o cálculo da distribuição das chuvas tem-

poralmente. Assim como a distribuição temporal, a distribuição espacial das precipitações é um elemento importante para o projeto de obras de drenagem e controle dos volumes escoados superficialmente (BERTONI; TUCCI, 1997).

O escoamento superficial é o relacionamento entre a vazão e o tempo, a distribuição da vazão no tempo é o resultado de todos os eventos do ciclo hidrológico que ocorre com o início da precipitação e finaliza com sua vazão na bacia hidrográfica. O solo recebe a chuva, e este leva um tempo até atingir seu nível de saturação que depende muito do tipo de solo da vegetação e do relevo da bacia hidrográfica (TUCCI, 1997b).

Dois modelos principais descrevem o comportamento da água em áreas de declive, são eles designados modelo não hortoniano e modelo hortoniano. O modelo de escoamento superficial hortoniano assume que a água infiltre nas áreas mais altas do terreno resurgindo às superfície juntando-se novamente ao fluxo superficial e evolui sem obstáculos até o leito dos cursos d'água. O modelo de escoamento superficial não hortoniano supõe que a água infiltre no solo e se move como fluxo sub-superficial para se juntar ao fluxo superficial com pouco ou nenhum escoamento superficial, a figura 1 demonstra o comportamento do escoamento superficial hortoniano pela trajetória 1 e o não hortoniano pela trajetória 2 (FRANK, 1995).

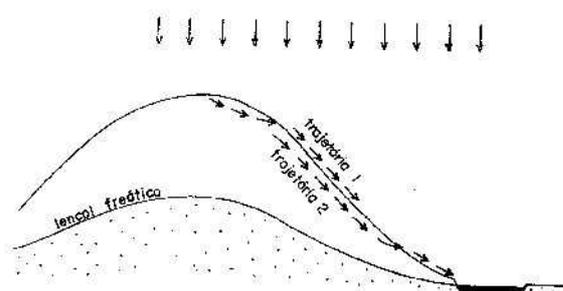


Figura 1: Modelo de escoamento superficial Hortoniano e não Hortoniano. Fonte: (FRANK, 1995)

2.2 Modelagem Ambiental

A definição de um modelo ambiental que simule a natureza em todos os seus aspectos depende exclusivamente da elaboração do modelo (SENNA CARNEIRO, 2004). Este modelo utilizará de técnicas inerentes a determinadas áreas do conhecimento que serão utilizadas para representar as propriedades físicas de um evento da natureza.

Um modelo ambiental pode ser pensado na forma de um micro-mundo que é definido por uma ontologia que consiste em um conjunto de entidades que o habitam, uma estrutura temporal, uma estrutura espacial, regras de comportamento e uma lógica (Smyth(1988) apud (SENNA CARNEIRO, 2004)).

A modelagem consiste na união dos princípios de várias teorias que são utilizadas para produzir um modelo que represente um fenômeno do mundo real (SENNA CARNEIRO, 2004). O mesmo autor ainda diz que o "modelo" é a representação de um sistema real ou imaginário utilizando-se ferramentas para a representação, um meio onde ocorre o problema e a interpretação do problema por parte do modelador.

Um modelo bem definido é aquele capaz de demonstrar a simplicidade em sua modelagem e a capacidade de representar com fidelidade as características do fenômeno. De acordo com (CUNHA TRIVELATO, 2003), modelo é a representação do conhecimento e a principal ferramenta para o estudo do comportamento de sistemas complexos.

O primeiro passo para a análise de um sistema natural é a modelagem, se o modelo for uma representação adequada do sistema natural, pode-se retirar informações importantes sobre suas particularidades. Outra definição sobre modelagem é que o processo demanda de conhecimentos prévios sobre o fenômeno.

As modelagens matemáticas e computacionais podem representar o comportamento das propriedades físicas que caracterizam um evento através de variáveis quantitativas, e essas variáveis possuem quatro elementos que são fundamentais para a definição de um modelo, são eles: um conjunto de nomes para os objetos e agentes que interagem com ele, um conjunto de variáveis descritivas que irão representar os objetos, as equações matemáticas do modelo que irão demonstrar a sua evolução no espaço e no tempo e a interpretação do modelador, que fará uma relação entre as variáveis descritivas e as propriedades do evento real (SENNA CARNEIRO, 2004).

A definição das variáveis descritivas que irão compor o modelo ambiental é um passo importante para a modelagem do sistema. Variáveis do objeto representam as propriedades inerentes a cada objeto. Por exemplo, o objeto solo tem como variáveis a declividade e rugosidade. Variáveis de Estado representam as propriedades de elementos que podem variar no espaço e no tempo, por exemplo, a posição inicial do elemento "Água" e a sua transição para um novo estado, demonstra a variação de espaço e tempo. De espaço, pois a água saiu de uma posição (p) para uma nova posição ($p + 1$), e de tempo, pois na transição do estado para um novo, partiu de um tempo (t) para um tempo ($t + 1$). Entretanto ainda deve-se levar em consideração a variável de interação, que representa a interação de algum objeto do meio externo com os objetos que estão sendo utilizados para modelar o sistema. Por exemplo, um obstáculo que se opõe entre a água e o solo pode diminuir ou aumentar a intensidade com que essa tenha variações no espaço e no tempo.

A modelagem ambiental segue princípios e teorias da modelagem. O modelador necessita conhecer o fenômeno e as técnicas de modelagem necessárias para a concepção de um modelo. Algumas teorias são descritas como pioneiras e servem de suporte para o estudo de modelagem ambiental. A teoria de sistemas pode ser considerada como a maior contribui-

ção para a unificação de várias áreas do conhecimento de cientistas para modelar os diversos fenômenos na nossa natureza. A Teoria de Sistemas objetiva a organização e relações entre as partes integrantes e interdependentes que se conectam e fazem com que essas formem um todo (SENN CARNEIRO, 2004).

2.3 Modelo Digital de Terreno - (MDT)

Um elemento preponderante na definição da direção do escoamento superficial do fluxo acumulado é o MDT. O MDT é um tipo de arquivo que é representado computacionalmente por valores de altitude a fim de simular a altimetria de um terreno, ou seja, para representação do relevo. O MDT possui algumas utilidades como, armazenamento de dados da altimetria do terreno para construção de mapas topográficos, análise de corte-aterro para projetos de construção de estradas e barragens, a elaboração de mapas de declividade para auxiliar a análise da geomorfologia e erodibilidade, análise de variáveis geofísicas e geoquímicas e a apresentação tridimensional de mapas do relevo (somados a outras variáveis). A aquisição de dados sobre o relevo para criação do MDT pode ser feita através de técnicas como digitalização do terreno, sensoriamento remoto entre outros. Esses dados são editados para geração do MDT.

Os MDT's podem ser entendidos como uma representação espacial de um terreno através dos elementos (pixel) de uma matriz. Cada elemento da matriz armazena o valor da elevação do terreno, essa representação se dá através de uma grade regular retangular, também chamada de GRID. Na figura 2 os números representam os valores de altitude do terreno, onde os valores mais altos representam a altitude média das áreas mais elevadas e os valores mais baixos representam a altitude média das áreas mais baixas. Uma desvantagem deste GRID é a dependência do tamanho da célula que deve ser único para toda a região modelada (Meijerinc et al. apud,(MÜLLER; GARCIA, 2001)).

58	53	47	41	48	49
44	57	40	49	46	50
39	43	30	37	38	48
24	33	26	22	31	34
35	41	25	21	16	12
24	47	34	11	7	9

Figura 2: Matriz regular retangular - GRID.

Outra estrutura bastante utilizada são as descritas como *Triangular Irregular Network* (TIN) mostrado na figura 3, ou matriz irregular triangular, que são representações da superfície em malhas triangulares que não são superpostas e contíguas e não são triângulos iguais ou do mesmo tamanho, dependendo da superfície do terreno a triangulação pode ficar mais ou menos complexa e mais ou menos detalhada. Com essa visão pode-se ter uma idéia de triangulação complexa, entretando demonstra mais detalhes o relevo.

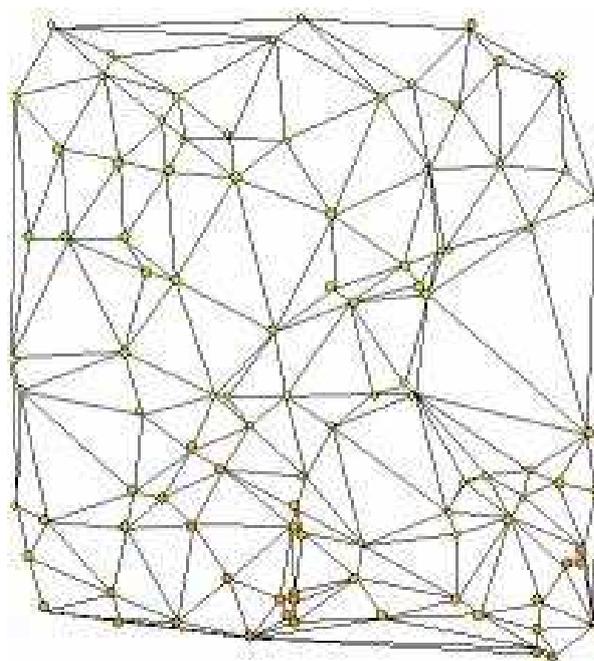


Figura 3: Matriz Irregular Triangular - GRID.

A partir de imagens com informações de MDT é possível extrair a rede de drenagem de uma região, definindo os caminhos de escoamento (direção de fluxo) e a delimitação de bacias hidrográficas. O tamanho das células analisadas é um fator determinante no processamento das informações, geralmente o tamanho das células de um MDT estão disponíveis em tamanhos de $30m^2$, $90m^2$, $500m^2$ e $1Km^2$ (MÜLLER; GARCIA, 2001).

A escolha do tamanho da imagem para estudo está diretamente relacionada com o tipo de análise a ser feita, imagens com resolução espacial muito pequenas podem ser mais susceptíveis a detalhes do que imagens com a resolução espacial de maior tamanho, pois podem esconder detalhes importantes na hora da análise.

A modelagem de fluxo acumulado utiliza-se de métodos que têm o MDT como base para determinação das direções e acúmulos de água. Grande parte dos métodos de extração de características topográficas existentes utiliza uma matriz regular retangular para a representação da superfície da região, onde os pixels ou pontos amostrados representam as elevações do terreno e armazenam em intervalos regulares.

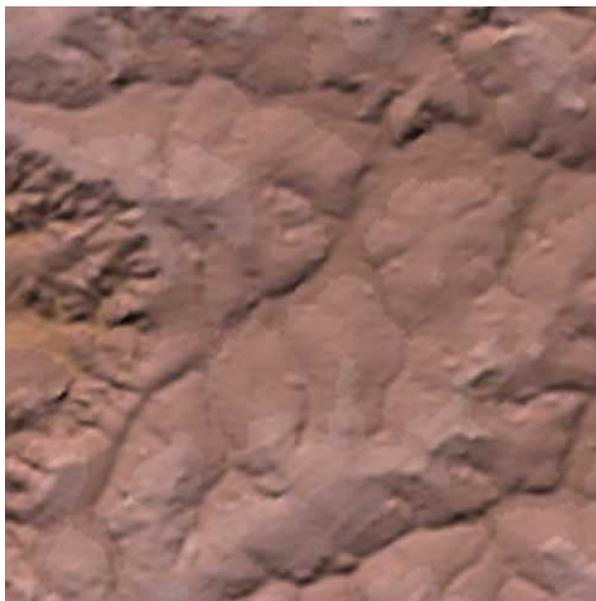


Figura 4: MDT com resolução espacial de $90m^2$ do município de Anápolis - GO, - Fonte: (MIRANDA, 2005)

2.4 Metodologias para o cálculo da direção de fluxo da água.

Vários métodos foram utilizados ao longo das últimas décadas para a obtenção de modelos que representassem satisfatoriamente os processos hidrológicos e que pudessem simular automaticamente o comportamento da rede de drenagem utilizando MDT. Alguns modelos descritos foram implementados com o objetivo de imitar fielmente as características dos fenômenos da natureza.

Na década de 80, foi proposto por (John F. O'Callaghan e David M. Mark apud (TARBOTON, 1997)) o método para extração da direção de fluxo designado D8. Tal método mais tarde seria denominado por outros pesquisadores como sendo o mais simples para extração da direção de fluxo. Por ser o mais simples, tal método é até os dias atuais o mais utilizado. No entanto, o método apresentava algumas desvantagens, e essas fizeram com que outros pesquisadores implementassem baseados no método D8 algumas funções no intuito de sanar suas deficiências.

O método D8 discretiza a direção do fluxo em somente oito direções, isso poderia causar uma informação errônea quando fosse aplicado a imagens com uma baixa resolução (TARBOTON, 1997), pois imagens com alta resolução espacial abrange regiões extensas, e podem esconder pequenos cursos d'água. Utilizando um MDT, o método analisa cada célula da matriz e seus oito vizinhos. A célula com valor inferior à de seus oito vizinhos recebe o seu fluxo.

A figura 5 demonstra os valores de MDT, onde a maior diferença entre os níveis de terreno e a declividade em relação ao pixel central determina a direção do fluxo. Esse método define uma rede de drenagem a partir de um MDT representado na forma matricial ou uma

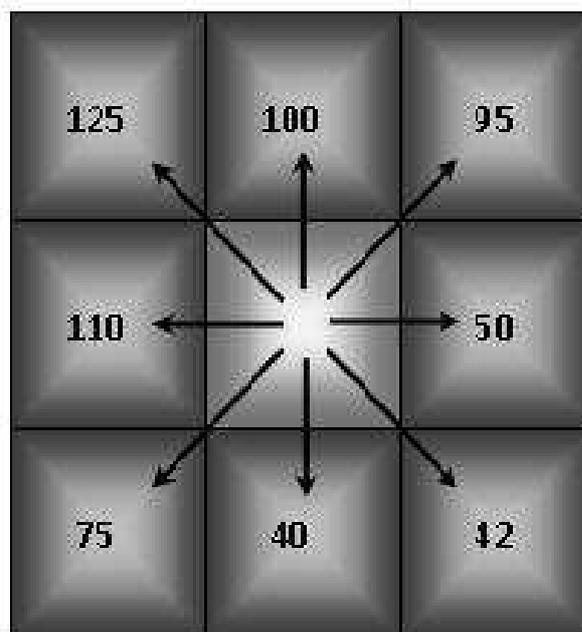


Figura 5: Representação do método de direção de fluxo D8.

matriz regular retangular.

O D8 identifica o caminho mais íngreme para o fluxo entre cada célula de um MDT e suas oito células vizinhas, por isso o seu nome D8, o ponto de escoamento representa os locais através do qual a água flui de uma célula a outra. Apesar de muito utilizado para trabalhos de modelagem de rede de drenagem o método D8 apresenta algumas limitações na extração da direção do fluxo, ele possui uma discretização do fluxo somente em uma de suas oito direções, dessa forma o D8 direciona o fluxo para a célula que possuir o menor valor numérico de elevação como na figura 6.

Em áreas planas ou depressões o método D8 tem dificuldades de representar a rede de drenagem, pois as células vizinhas apresentarem o mesmo valor altimétrico ou são análogas a um terreno em forma de bacia.

Na tentativa de minimizar ou solucionar os problemas originados no método D8, pesquisadores propuseram o método *Multiple Directions based on slope* (MS) (Quinn et al, 1991, apud (TARBOTON, 1997)), que aloca uma fração do fluxo para cada vizinho com igual ou menor elevação em relação ao declive do terreno, ele notou também que uma desvantagem desse método é o fluxo de um único pixel poder se espalhar para todos os seus vizinhos com menor elevação. Um método que viria minimizar os problemas do método D8 foi o proposto por (Lea, 1992, apud (TARBOTON, 1997)), que desenvolveu um algoritmo que utilizava aspectos inerentes a cada pixel para especificar a direção de fluxo. Uma vantagem desse método foi demonstrar a capacidade de especificar o fluxo de maneira contínua sem dispersão para vizinhos com menor elevação.

No ano de 1994 (Costa Cabral e Burges, 1994, apud (TARBOTON, 1997)) apresentou

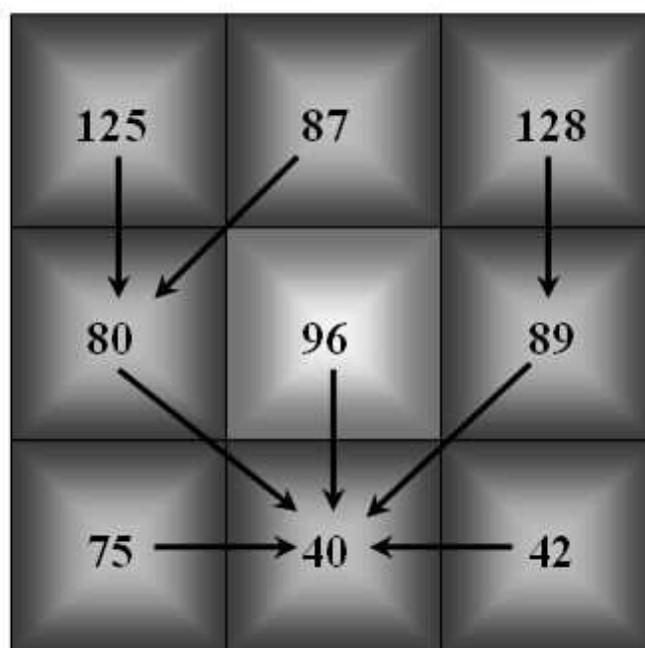


Figura 6: Escoamento do Fluxo com o método de direção de fluxo D8.

um novo procedimento com o nome de *Digital Elevation Model Network* (DEMON) que iria incorporar o algoritmo de Lea 1992, mas os problemas encontrados no método de Lea continuariam com o método de DEMON. Esses problemas serviram de motivação para (TARBOTON, 1997) propor um novo método denominado D_{∞} (lê-se “DInfinito”) que suprisse as lacunas dos métodos anteriores.

A motivação para tal estudo, estava em suprir as frustrações existentes em métodos anteriores e na necessidade do desenvolvimento de procedimentos para o estudo de direção de fluxos utilizando MDT. Notou-se que em alguns dos métodos anteriores ocorria um derramamento de fluxo para os pixels com elevação inferior. O método D8 foi o único a não apresentar dispersão, e não demonstrou habilidade quando aplicado em dados mais complexos como áreas com depressões e áreas de terrenos planos.

O modelo MS apresentou muita dispersão do fluxo, outros métodos (Lea; Costa Cabral and Burges) conseguiram determinar com precisão a direção do fluxo, porém foram susceptíveis a problemas citados nos estudos de (TARBOTON, 1997) que sugeriu que os algoritmos fossem robustos e tivessem a habilidade de resolver problemas com dados complexos de áreas planas e terrenos de vale (depressões), e ainda sugeriu que fossem avaliados cinco aspectos para se chegar a um modelo de maior precisão. Os Cinco aspectos relevantes enumerados por Tarboton foram:

- A necessidade de evitar ou minimizar a dispersão de fluxo;
- A necessidade de evitar uma grade de dispersão de fluxo;

- Resolver a direção do fluxo com precisão;
- Uma grade simples e eficiente com estrutura de armazenamento de matriz;
- Robustez, a habilidade do modelo de enfrentar dados complexos, como no caso de áreas de vale e áreas planas.

A partir desses dados, ele uniu as vantagens dos métodos de Lea e Demon para determinar um simples fluxo para cada célula da grade, seu primeiro passo para determinar um fluxo de direção simples, foi representar uma quantidade contínua desse fluxo entre os valores de 0 e 2π que é determinada em direção da declividade mais íngreme de uma grade 3×3 de oito facetas triangulares com o pixel de interesse no centro da grade.

Diferente do método D8, o fluxo não segue para somente oito direções a partir do centro, ele segue um caminho inerente ao cálculo de declividade entre dois pixel vizinhos com menor valor altimétrico e o de interesse. O armazenamento dos dados de cada pixel é minimizado, pois a única informação inerente ao mesmo que é guardada é o valor do ângulo. Todos os pixels da grade são representados por um valor de elevação. A partir do pixel de centro são formadas oito facetas triangulares com os seus vizinhos.

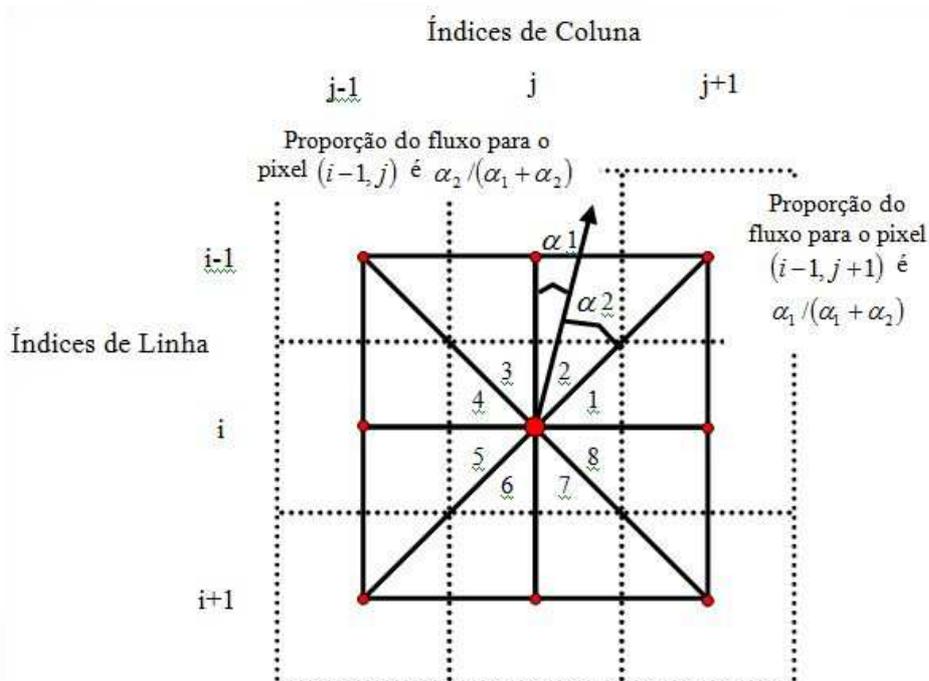


Figura 7: Cálculo da direção de fluxo do método Dinfinite.

Na figura 7 a direção do fluxo é definida por um ângulo da faceta com menor valor de declividade onde: os valores 1, 2, 3..., 8 representam as facetas triangulares definidas para o cálculo da direção entre o pixel central e seus oito vizinhos, o cálculo do valor da proporção de fluxo do ponto central representado pela posição (i, j) para o vizinho de posição $(i - 1, j)$

é representado pela equação $\alpha_2/(\alpha_1 + \alpha_2)$ que determina a quantidade de fluxo que escoará naquela direção, pode-se notar que o fluxo maior escoará para o pixel de posição $(i - 1, j + 1)$ representado pela equação $\alpha_1/(\alpha_1 + \alpha_2)$.

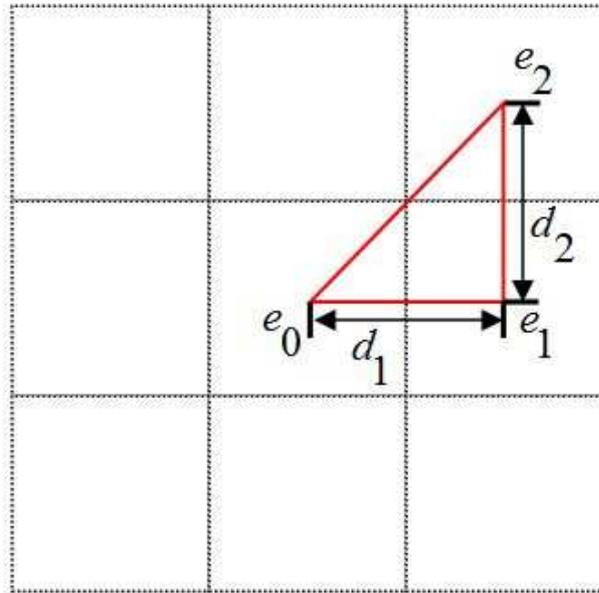


Figura 8: Cálculo da declividade do método Dinfito.

Para implementar este procedimento foi considerado primeiro facetas triangulares como ilustrado na figura 8, onde cada direção é representada por uma faceta, e essa é definida pelas variáveis, e_i que representa os valores referentes à altitude do terreno e d_i que representa a distância entre os valores de altitude de um ponto central até o ponto central do vizinho.

A figura 8 faz uma análise da faceta de número 1 mostrado na figura 7, que tem como pontos analisados os pixels e_0 de posição (i, j) , o pixel e_1 de posição $(i, j + 1)$ e o pixel e_2 de posição $(i - 1, j + 1)$, e a distância entre estes pontos definidos por d_1 e d_2 .

A distância entre o pixel analisado e seus dois vizinhos que formam a faceta para análise depende exclusivamente da resolução espacial do MDT.

$$S_1 = (e_0 - e_1)/d_1 \quad (1)$$

$$S_2 = (e_1 - e_2)/d_2 \quad (2)$$

Cada valor de S_1 e S_2 gera uma força de declividade S e uma direção de fluxo r representado por um ângulo em radianos, como mostrado na figura 9.

A representação da declividade é dada pelos vetores (S_1, S_2) de cada faceta demonstrada na figura 9, onde os valores da elevação e_i e distância d_i entre os pixels, são ilustrados como na figura 8.

Cada valor de S_1 e S_2 gera uma força de declividade s e uma direção de fluxo r representado por um ângulo em radianos, como mostrado na figura 9.

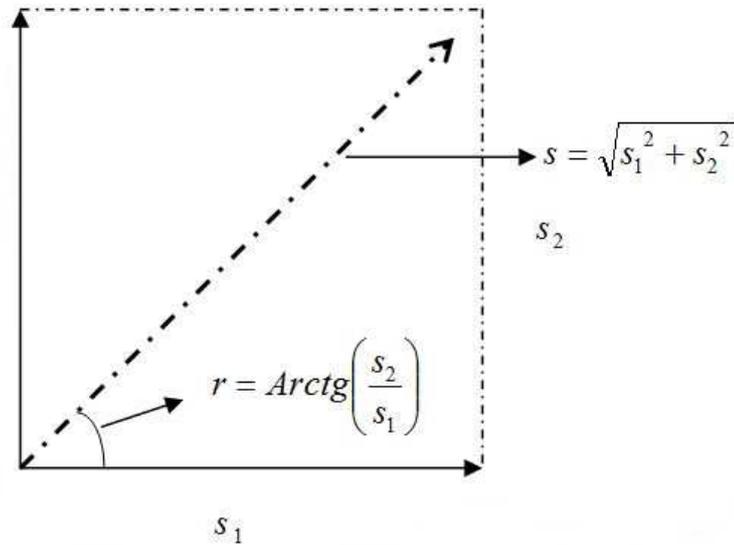


Figura 9: Cálculo do ângulo (r) para a direção do fluxo e da força de declividade (s).

A figura 9 ilustra as equações para o cálculo do ângulo de direção e a força de declividade onde: as equações 3 e 4 representam a direção do fluxo superficial e a magnitude da declividade que são dadas por r e s respectivamente.

$$r = \arctan\left(\frac{S_2}{S_1}\right) \quad (3)$$

$$s = \sqrt{(S_1^2 + S_2^2)} \quad (4)$$

Tarboton definiu que o tamanho do ângulo r deveria ser entre 0 e $\tan^{-1}\left(\frac{d_2}{d_1}\right)$, dessa forma atribuiu algumas condições para que essa restrição fosse atendida. Se $r < 0$, então esse seria igualado a 0 e o valor de $s = 1$. Se $r > \tan^{-1}\left(\frac{d_2}{d_1}\right)$, então r teria o mesmo valor da $\tan^{-1}\left(\frac{d_2}{d_1}\right)$.

A figura 10 definida por Tarboton, demonstra os valores das posições de cada faceta representados por e_0 , e_1 e e_2 e as constantes definidas para determinar precisamente qual faceta irá ser analisada, e os valores a serem atribuídos para o cálculo da direção do fluxo.

De acordo com os dados mostrados na figura 10 as equações que determinam a direção do fluxo e a magnitude de seu escoamento, são definidas para um ângulo entre 0 e 2π radianos, e até então a direção do fluxo havia sido definida para apenas uma faceta.

facetas	1	2	3	4	5	6	7	8
e_0	$e_{i,j}$							
e_1	$e_{i,j+1}$	$e_{i-1,j}$	$e_{i-1,j}$	$e_{i,j-1}$	$e_{i,j-1}$	$e_{i+1,j}$	$e_{i+1,j}$	$e_{i,j+1}$
e_2	$e_{i-1,j+1}$	$e_{i-1,j+1}$	$e_{i-1,j-1}$	$e_{i-1,j-1}$	$e_{i+1,j-1}$	$e_{i+1,j-1}$	$e_{i+1,j+1}$	$e_{i+1,j+1}$
a_c	0	1	1	2	2	3	3	4
a_f	1	-1	1	-1	1	-1	1	-1

Figura 10: Ilustração das posições das facetas e as constantes para o cálculo da direção de fluxo superficial.

(TARBOTON, 1997) atribuiu uma variável $r' = r$ cujo valor de s é máximo, ou seja, ajusta-se para um ângulo anti-horário entre 0 e 2π , partindo do ângulo 0. Para tanto, considera-se a faceta determinada por r' para o cálculo de um ângulo r_g real entre 0 e 2π .

$$r_g = a_f r' + a_c \frac{\pi}{2} \quad (5)$$

O cálculo do fluxo acumulado é gerado pela equação 5, onde: r_g é representa a direção de fluxo do método Dinfinite, a_f e a_c são as constantes definidas por Tarboton para determinar a faceta, r' para ajuste do ângulo para o cálculo da direção e $\frac{\pi}{2}$ que separa as facetas.

Através de comparações com os outros métodos implementados para determinar a direção de fluxo Tarboton demonstrou que o método proposto Dinfinite mostrou-se mais eficiente que o métodos D8 e o método de Lea.

2.5 Autômatos

2.5.1 Autômato Finito

De acordo com as definições de (HOPCROFT; ULLMAN; MONTWANI, 2002), um autômato finito ou uma máquina de estados finitos pode ser análogo a um grafo com um número finito de vértices e um conjunto de arcos que pode ser definido como transição entre os estados do autômato de acordo com a equação seguinte, $grafo_x = (W, Q, S_t)$, onde *entrada* são os valores de entrada para o autômato, W e Q são o conjunto dos estados do autômato e S_t é o numero de transições possíveis do autômato figura 11.

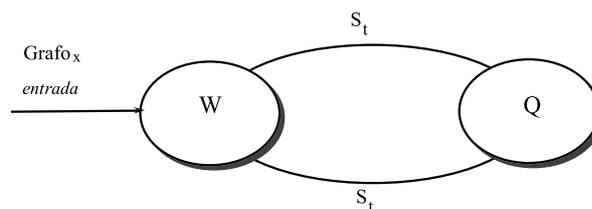


Figura 11: Representação gráfica de um autômato com dois estados.

A figura 11 ilustra o diagrama de estados do autômato, que irá reagir com as variáveis

de entradas em conjunto com as regras de transição e controles de estado em um passo de tempo discreto. (HOPCROFT; ULLMAN; MONTWANI, 2002) descreve um autômato finito simples que simula um interruptor de liga/desliga, esse autômato apresenta dois estados, o estado de desligado como estado inicial do sistema, ou seja, o estado em que se encontra o sistema inicialmente, e o estado ligado com estado final do sistema, que é acionado após o usuário pressionar um botão do interruptor e ativar a transição os estados. A figura 12 ilustra o funcionamento de um interruptor de lâmpada simulado por um autômato finito.

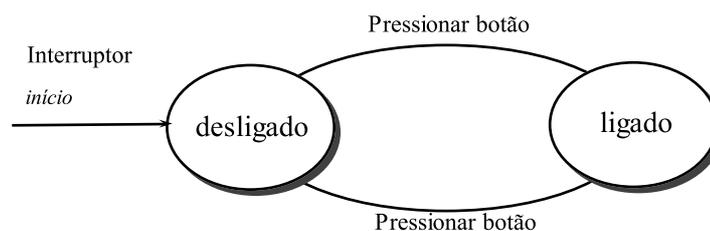


Figura 12: Exemplo de aplicação de um autômato finito.

2.5.2 Autômato Celular

Em 1940, Von Neumann desenvolveu a teoria de autômatos, com uma sistemática lógica e matemática que contribuíram fundamentalmente para a compreensão de computadores analógicos e digitais. Para tal contribuição ele escreveu cinco trabalhos que foram essenciais:

- a) *The General and Logical Theory of Automata*. Leia no Simpósio hixon em setembro de 1948, publicado em 1951, 5,288-328.
- b) *Theory and Organization of Complicated Automata*. Quinta conferência proferida na Universidade de Illinois em dezembro de 1949.
- c) *Probabilistic Logics and the Synthesis of Reliable Organisms from Unreliable Components*. Palestras dadas no Instituto de Tecnologia da Califórnia, em Janeiro, 1952, 5,329-378.
- d) *The Theory of Automata: Construction, Reproduction, Homogeneity*. Von Neumann começou este manuscrito no outono de 1952 e continuou trabalhando nele por cerca de um ano. Este trabalho é a parte 2 do volume de (NEUMANN, 1966).
- e) *The computer and the Brain*. Escrito entre os anos 1955 e 1956; publicado em 1958.

Von Neumann se especializou em autômatos complicados e tinha como fonte de inspiração o sistema nervoso humano. Sua morte em 1957 causada por câncer, talvez tenha privado o mundo da ciência de descobertas valiosas nos campos de interesse desse pesquisador. Além de contribuir com excelência no campo da matemática aplicada ele teve grande contribuição

para o desenvolvimento e programação de computadores e na construção do primeiro computador digital o ENIAC desenvolvido entre os anos de 1943 e 1946.

Dois de seus trabalhos foram inseridos em seu manuscrito (NEUMANN, 1966), sobre a teoria de auto reprodução de autômatos celulares, no primeiro Von Neumann descreve a teoria e organização de um autômato complexo, na outra ele demonstra a teoria da construção e reprodução de autômatos, trabalhos que foram amplamente utilizados nas pesquisas sobre a complexidade do sistema biológico humano.

Em seu primeiro trabalho ele demonstra procedimentos analógicos e digitais, e o uso da matemática para a construção de autômatos e suas contribuições para as máquinas computáveis, no segundo trabalho ele descreve precisamente a forma de construção de autômatos como esses se reproduzem.

As principais questões a serem analisadas no ponto de vista de Von Neumann foram a universalidade lógica utilizada na definição da classe do autômato e a evolução do autômato que tenta responder as seguintes questões: Pode um autômato ser construído a partir de outro autômato? Pode um autômato se reproduzir exatamente igual a ele? Essas questões são pontos extremamente importantes para o desenvolvimento de sistemas autômatos auto-reprodutíveis com o interesse de reproduzir o fenômeno biológico da auto-reprodução (NEUMANN, 1966).

Antes mesmo dos estudos de Von Neumann, outro pesquisador na década de 30 já havia desenvolvido uma máquina de calcular abstrata que tinha as características dos computadores atuais (HOPCROFT; ULLMAN; MONTWANI, 2002). A máquina de Turing pode ser um exemplo de autômatos finitos determinísticos, na qual sua configuração utiliza uma fita dividida em células adjacentes e um cabeçote para leitura e escrita de símbolos na fita.

Outro trabalho desenvolvido a partir da teoria dos AC foi o Jogo da Vida (LIFE, 2008) desenvolvido na década de 70 pelo matemático britânico John Horton Conway, talvez esse seja o exemplo mais conhecido de autômato celular da história.

Nas últimas décadas a área das ciências ambientais tem visto várias aplicações de modelagem da dinâmica dos sistemas ambientais utilizando a teoria de AC. A utilização de AC no campo da modelagem de sistemas ambientais deve-se a sua facilidade de implementação e a capacidade em simular as variáveis particulares aos sistemas além de poder simular dinamicamente os processos ambientais, ou seja, simular o espaço em que acontece o fenômeno e seu tempo de ocorrência.

Na década de 90 a teoria de AC foi utilizada em várias pesquisas: biorremediação de solos contaminados (GREGORIO; SERRA; VILLANI, 1999), urbanização na baía de San Francisco (CLARKE; HOPPEN; GAYDOS, 1997), dinâmica urbana através de sistemas de informações geográficas - SIG baseados em AC (BATTY; XIE; SUN, 1999), modelagem dinâmica ambiental baseada na teoria de autômatos híbridos(AH) para abstrair de um sistema dinâmico seus elementos contínuos e discretos (PEDROSA, 2003).

As pesquisas objetivando a simulação de sistemas dinâmicos ambientais ainda é recente, e que a utilização da teoria de AC para concepção dos modelos computacionais tem bastante importância e vem crescendo bastantes no campo das pesquisas ambientais.

O autômato de Von Neumann foi definido partindo de um reticulado de células bidimensionais como em uma matriz, o endereço de cada matriz é designado pelas coordenadas i que representa as linhas da matriz e j que representa as colunas. Cada célula do AC é formado por um autômato finito (AF) que se conecta com seus quatro vizinhos. Neste ambiente as regras de transição de estados são iguais para todos as células.

((NEUMANN, 1966)) descreve em seu trabalho que um AC simula o tempo como sendo uma variável de tempo discreto e simultâneo, isso quer dizer que as regras de transição entre estados são analisadas sobre todo o espaço celular, e esses acontecimentos ocorrem em intervalos de tempos fixos, ou seja, o AC inicialmente está em tempo t_0 e atualiza todo o espaço celular em um tempo $t+1$. Na figura 13 um AC é ilustrado com um conjunto de AF conectados com seus vizinhos, onde: o AC é representado pelos quadrados e os AF são representados pelas bolas. As setas direcionais entre as bolas representa a mudança de estados entre ela. As cores dos AF são os estados que este pode assumir variando de acordo com as regras de transição entre seus estados e as regras de vizinhança.

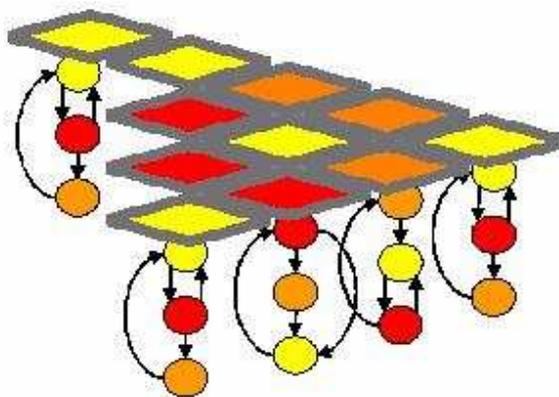


Figura 13: AC de Neumann. Fonte (SENN CARNEIRO, 2004)

Segundo (WOLFRAN, 1983) existe dois tipos de vizinhança para AC bidimensionais, (a) a vizinhança de Neumann com cinco autômatos que é o próprio autômato mais seus quatro vizinhos das diagonais e (b) a vizinhança de Moore que consiste do próprio autômato e seus oito vizinhos.

Matematicamente esse modelo de AC é adequado para o estudo de fenômenos espaciais, mas (COUCLELIS, 1997) ressalta que seu uso em modelagem de fenômenos ambientais geográficos com variáveis reais é muito restritivo, pois o espaço celular não deveria ser homogêneo e por este motivo foram propostos extensões para o modelo de AC.

Alguns fatores segundo (Couclelis, 1997; O'Sullivan, 2002. apud (PEDROSA, 2003)), foram determinantes para que as limitações de AC em modelos de simulação ambiental fossem identificados:

- a) A estacionariedade de seus vizinhos, ou seja todos os autômatos têm a mesma configuração de vizinhança.
- b) O espaço de estados é finito, ou seja apenas mudanças de estados em tempos discretos podem ser representados.
- c) Ações locais, ou seja a transição entre estados de uma célula acontece em decorrência do estado de suas vizinhas.

As extensões partiram estritamente sobre as regras de mudanças de estados do autômato, que passaram a utilizar variáveis de eventos reais, para tanto (HENZINGER, 1996) descreve em sua pesquisa sobre um autômato híbrido(AH), capaz de modelar sistemas dinâmicos com variáveis de tempo e espaço discreto e contínuo.

2.5.3 Autômato Híbrido

Um sistema híbrido é um sistema dinâmico com componentes discretos e contínuos (HENZINGER, 1996). Um sistema híbrido ou um AH é capaz de simular processos ambientais dinâmicos pois trabalha com os componentes reais desses processos. Para Henzinger um AH consiste dos seguintes componentes:

- a) Variáveis: Um conjunto finito de variáveis reais $X = \{x'_1, \dots, x'_n\}$ onde X é o conjunto de variáveis e x'_1, \dots, x'_n é um conjunto finito de estados do autômato.
- b) Grafos de controle: Um conjunto finito de grafos de controle, são os controladores de estados do autômato finito conforme figura 12.
- c) Variável inicial, invariantes e condições de fluxo: As variáveis iniciais e as invariantes são predicados pertencentes a X e as condições de fluxo são predicados pertencentes a $X \cup X'$.
- d) Condições de Mudança de estados: condições de mudança que atribui a cada grafo de controle uma condição para transição entre os estados.
- e) Eventos: Um conjunto de eventos \sum finitos e uma aresta rotulada de função de eventos $E \rightarrow \sum$ que atribui a cada grafo um evento.

Na figura 14 o funcionamento de um sistema de controle de temperatura é simulado por um AH, onde: x é uma variável de entrada com informações da temperatura ambiente. No

modo de controle *esfriando* o controlador está esfriando o ambiente de acordo com as condições de fluxo $x' = -0,1x$. No modo de controle esquentando o controlador está esquentando o ambiente conforme condição de fluxo $x = 5 - 0,1x$. No início do sistema a temperatura inicial é a mesma de entrada, ou seja 20° , conforme a condição de mudança de estado $x < 19$, se a temperatura for menor que 19° o controlador de temperatura passa para o estado esquentando. A condição de mudança de estados $x > 21$ e executada se a temperatura for maior que 21° , dessa forma ocorre a transição do estado *esquentando* para o estado *esfriando*. Conforme as condições invariantes $x \geq 18$ ou $x \leq 22$ a temperatura ambiente pode continuar caindo até que a temperatura mínima de 18° for atingida, ou a temperatura pode continuar subindo até ser atingido o valor máximo de 22° .

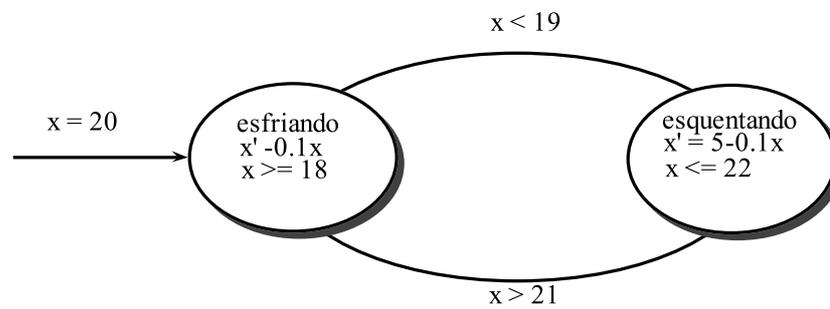


Figura 14: Sistema de controle de temperatura. Fonte: adaptado de (SENNÁ CARNEIRO, 2004)

3 DESCRIÇÃO DOS MÉTODOS

3.1 Descrição dos Métodos

O simulador de fluxo acumulado intitulado “AutoHidro”, é uma ferramenta implementada em linguagem de programação $C++$ com o intuito único e exclusivo de simular a dinâmica do fluxo acumulado da água. Para a concepção do simulador foi utilizado como arcabouço dois conceitos chaves descritos nas seções 2.5.2 sobre AC e 2.5.3 sobre AH. A figura 15 descreve o funcionamento do simulador AutoHidro, onde: os dados de volume de água inicial $i = 0$ serão passados ao simulador AutoHidro no “passo 1”, após passar pelo simulador o volume de água é calculado para $i = i + 1$, enquanto o passo de tempo da simulação naquele instante for menor que a duração total de simulação “ n ”, “passo 2”, determinado pelo usuário do sistema, o simulador registrará o frame, que é uma imagem representativa do volume de água naquele instante, “passo 3”. Esse frame será novamente calculado pelo simulador passando novamente para $i = i + 1$, “passo 4”, até que o tempo da simulação n seja atingido. Após esse processo os volumes de água são registrados e armazenados como dados do fluxo acumulado.

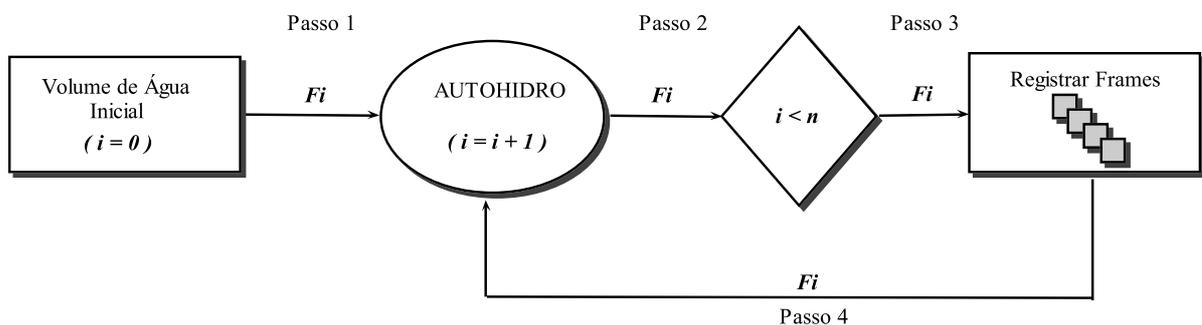


Figura 15: Fluxograma de funcionamento do simulador AUTOHIDRO.

3.1.1 Definição do Sistema

O sistema considera como ponto de partida uma série de precipitação informada pelo usuário. Esses dados são informados ao sistema representando o volume precipitado que consi-

dera como unidade de medida “milímetros por minutos”, outras informações como infiltração do solo, Evaporação e o tempo total de precipitação são informados no início da simulação.

Como parâmetros de entrada o sistema utiliza um MDT com dados da altimetria do terreno, o cálculo da direção e declividade se fundamentou nas definições da biblioteca de funções de (TARBOTON, 1997) e em funções trigonométricas. Dados sintéticos sobre precipitação, infiltração, evaporação, limite de saturação do solo e o tempo de precipitação são passados ao sistema como parâmetros de entrada.

Os cálculos da declividade entre um pixel e o seu vizinho com menor valor de altitude foram realizados considerando a diferença de altitude e a rampa provocada por essa diferença, levando em consideração o tamanho do pixel do MDT, que é o mesmo para todas as células.

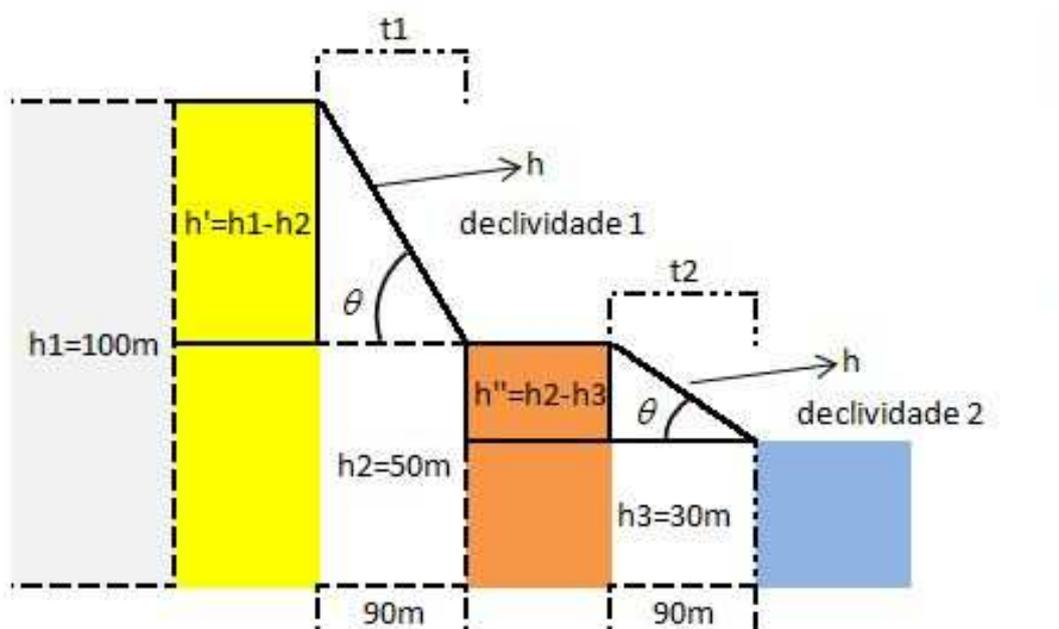


Figura 16: Diferença de altitude entre dois pixels.

A figura 16 ilustra três pixels com diferentes altitudes, essa diferença define a rampa de inclinação para o escoamento superficial, onde h_1 , h_2 e h_3 são os valores de altitude retiradas do MDT, os valores de h' e h'' representam o cálculo da diferença entre as altitudes de h_1 e h_2 , h_2 e h_3 , a distância entre os pontos centrais das células é dada pelo o valor da resolução espacial do pixel estudado (90m). O tempo discreto t_1 e t_2 do processo de escoamento e o ângulo da rampa de inclinação, dado por θ , é demonstrada pela *declividade1* e *declividade2* obtida pelo cálculo da função arctan do ângulo θ .

Para calcular o tempo de escoamento de um pixel para outro foi utilizado a equação 6 onde t representa o tempo de escoamento, h é o comprimento da rampa de inclinação entre

um pixel e outro, a proporção da força de gravidade que influencia na velocidade e no tempo de decida do escoamento é dada por $(g \cdot \text{sen}\theta)$, onde g é a aceleração da gravidade.

$$t = \sqrt{2h/g} \quad (6)$$

Com esses cálculos foi possível determinar o tempo de escoamento superficial hortoniano da água, de um pixel até seu vizinho com valor altimétrico inferior. O sistema determinará a dinâmica do escoamento de acordo com a transição de estados, levando em consideração a altitude, a distância entre seus vizinhos e a declividade.

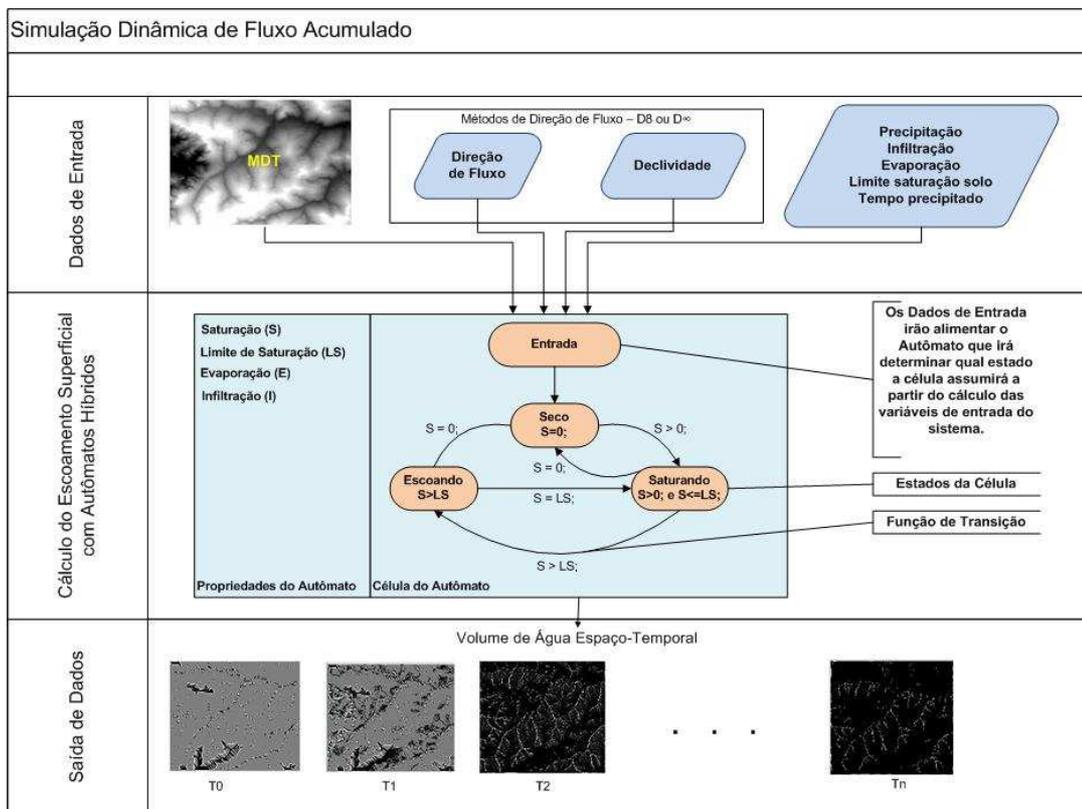


Figura 17: Definição do Autômato Híbrido para os componentes do Sistema AutoHidro.

Uma condição de intervalo de precipitação foi inserida com o intuito de determinar em qual intervalo da simulação a precipitação é iniciada, ou seja, em que instante da simulação começa e termina a chuva. A figura 17 ilustra os principais componentes do sistema. Como dados de entrada o sistema recebe informações a altitude do terreno por meio do MDT, recebe o método de direção de fluxo D8 definido para esse sistema e as variáveis de ambiente que serão inseridas ao simulador pelo usuário, na tela de simulação do sistema AutoHidro.

Para o cálculo do fluxo acumulado o AutoHidro recebe os dados de entradas definindo os estados que o autômato híbrido pode assumir durante a simulação. Os dados de saída o sistema são resultados do cálculo entre o MDT, o método de direção de fluxo D8 e as variáveis de ambiente informadas pelo usuário.

Quando ocorrer a precipitação, o AH sairá do estado *Seco* que significa ausência total de água na célula, onde $S = Sat - (Limsat - Evap - Infil)$ e $S = 0$, assumindo então o estado de *Saturado*. Este estado significa que a propriedade *Saturado* é igual ao limite de saturação do solo (*Limsat*), ou seja, a célula é dita saturada quando a propriedade *Saturado* for igual à propriedade *Limsat*, onde $Sat = Limsat$, e irá se manter neste estado até que a quantidade precipitada em uma célula ultrapassar seu limite, considerando ou não a infiltração e evaporação. Dessa forma o sistema calcula o excesso de água a ser escoado. Esse excesso será transportado para a célula vizinha com menor valor altimétrico, e a célula analisada assumirá o estado de *Escoando* que se caracteriza pelo transporte do excesso de água no solo, onde a transição entre os estados *Saturado* e *Escoando* ocorre por meio do método *Provoca_Escoamento* figura 18. Esse processo de atualização de estados do AH será contínuo até finalizar o tempo de simulação.

AutoHidro
-Precipitacao : void
-Infiltracao : void
-Evaporacao : void
-DuracaoPrecipitacao : void
+Atualiza_Precipitacao() : void
+Atualiza_Infiltracao() : void
+Atualiza_Evaporacao() : void
+Atualiza_Escoamento() : void
+Provoca_Escoamento() : float
+Get_Excesso() : float
+Get_Limite() : float
+Analisa_ET() : AutoHidro

Figura 18: Definição da Classe AutoHidro.

Para determinar a dinâmica do sistema foi definido a classe *AutoHidro*, figura 18, onde todas as funções principais do sistemas estão definidas. Como propriedades da classe, foram estabelecidas para receberem os valores de entrada inseridos pelo usuário do sistema, as funções *Set_Precipitacao()*, *Set_Infiltracao()*, *Set_Evaporacao()*. Os métodos da classe para a atualização do sistema são os seguintes:

- a) *Atualiza_Precipitacao*: método que atualiza para todas as células o valor da precipitação digitado pelo usuários.

- b)** *Atualiza_infiltracao*: método que atualiza a infiltração para todas as células o valor da precipitação digitado pelo usuários.
- c)** *Atualiza_evaporacao*: método que atualiza a evaporação para todas as células o valor da precipitação digitado pelo usuários.
- d)** *Provoca_Escoamento*: método responsável por calcular o escoamento superficial transportando o excesso de fluxo de uma célula para sua vizinha.
- e)** *Atualiza_Escoamento*: método que atualiza o fluxo acumulado para todas as células da matriz.
- f)** *Analise_ET*: método que define o fluxo acumulado em espaço para todo o sistema.
- g)** *Set_Saturacao*: método que atualiza a saturação inicial quando houver, em todo o espaço da matriz.

Para o cálculo escoamento superficial foi necessário calcular o excesso de água no solo, para isso o método *Get_Excesso()* foi definido e considera o valor precipitado menos o limite de saturação do solo e passa esse valor como parâmetro para o método *Provoca_Escoamento()*.

4 RESULTADOS

4.1 Análise e interpretação dos Dados

O processo de simulação do fluxo acumulado realizado pelo software “AUTOHIDRO” calcula o volume de água precipitada em função do espaço em determinada área. O cálculo é realizado para a água que, em contato com o solo escoar sem interferências ou ocorra interferências da evaporação e(ou) infiltração.

Toda a simulação compreende 3(três) etapas específicas informadas pelo usuário do sistema: (1ª Etapa) duração do processo de simulação, (2ª Etapa) informação sobre o volume precipitado e (3ª Etapa) intervalo de ocorrência da precipitação em função da duração do processo de simulação. A última etapa assegura que durante o processo de simulação possa ocorrer ou não a precipitação, ou ainda, determinar se a precipitação ocorre em toda a simulação ou em partes.

A figura 19 ilustra o módulo de controle do simulador onde os dados sobre a duração da simulação, o intervalo em que serão coletadas as informações sobre o escoamento, os dados sobre a quantidade precipitada e o intervalo da precipitação em função da duração do processo de simulação são passados para o sistema.

Para análise do simulador e validação de seus resultados, foram estabelecidas 3(três) classes de testes que serão aplicadas às funcionalidades do simulador com o intuito de verificar se as variáveis do sistema estão interagindo espacialmente. Em todos os testes será utilizado um recorte de MDT feito na imagem extraída do site da Embrapa monitoramento por satélites (MIRANDA, 2005) da área do município de Anápolis. A duração da simulação será a mesma em todos os testes. Os valores inseridos para as variáveis serão atribuídos com o intuito de avaliar a simulação do comportamento do fluxo de água em função do espaço para diferentes cenários.

Como saída do sistema, o simulador irá demonstrar uma sequência de imagens com a evolução do escoamento superficial considerando as informações inseridas como valor inicial para as variáveis do sistema. Os testes estão descritos a seguir:

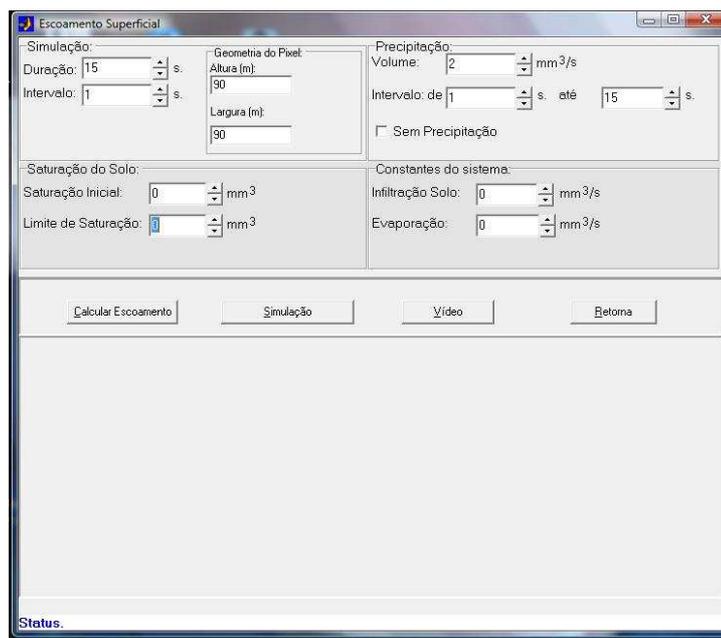


Figura 19: Módulo de controle do sistema.

4.1.1 Primeira Classe de Testes

A primeira classe de testes apresenta 5(cinco) simulações, cada uma com o objetivo de simular o comportamento da água da precipitação que cai e esco, sem que partes de seu volume infiltre no solo e(ou) evapore. Os testes (T1.1 ... T1.4) simulam um cenário em que ocorre a precipitação em diferentes intervalos, considerando que o solo esteja seco, sem água nenhuma. No teste (T1.5) é considerado um cenário onde não ocorre uma precipitação, mas existe uma saturação inicial do solo, conforme dados mostrados na tabela 1.

Tabela 1: Classe de Testes nº 1: Considerando a Precipitação e a Saturação Inicial do solo.

Variáveis	Número de testes				
	T 1.1	T 1.2	T 1.3	T 1.4	T 1.5
Tempo de Simulação	300s	300s	300s	300s	300s
Quantidade Precipitada	2mm/s	2mm/s	2mm/s	2mm/s	2mm/s
Intervalo Precipitado	1 – 240s	1 – 180s	1 – 120s	1 – 60s	0s
Saturação Inicial	0	0	0	0	1000
Limite de Saturação	0	0	0	0	0
Infiltração	0	0	0	0	0
Evaporação	0	0	0	0	0

Pretende-se demonstrar com esse teste o comportamento da evolução do escoamento superficial, desconsiderando o limite de saturação do solo, a infiltração da água no solo e a evaporação. A verificação dos resultados será realizado com sobreposição das imagens figura

20, onde teremos como referência a imagem do MDT e da área de contribuição do método D8.

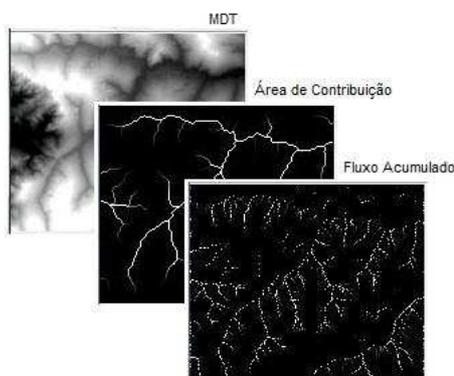


Figura 20: Sobreposição das Imagens MDT, Área de Contribuição e Fluxo Acumulado com AH.

O MDT representa a altitude do terreno de forma que nas áreas mais claras predomina as altitudes mais elevadas e nas áreas mais escuras as altitudes mais baixas. A imagem contendo a área de contribuição representa as áreas onde ocorre os cursos d'água de forma que as cores claras representam o curso d'água e o fluxo acumulado é representado pelos pontos mais claros na figura de nome *FluxoAcumulado* como ilustrado na figura 20.

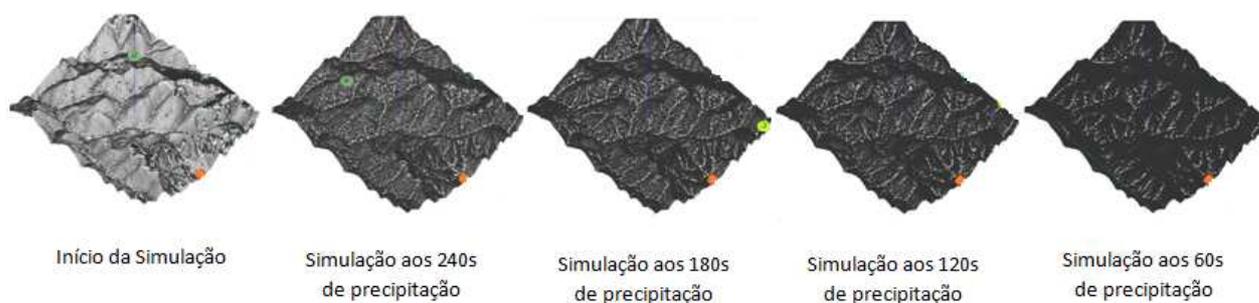


Figura 21: Evolução do Escoamento Superficial: Testes realizados sem considerar as variáveis ambientais

A figura 21 ilustra a evolução do escoamento superficial como resultado da simulação da primeira classe de testes conforme descrito na tabela 1. Para visualização do fluxo d'água escoado, foi necessário realizar uma composição em tons de cinza com ajuda do software "Abílio". Preferiu-se visualizar as imagens em tons de cinza, pois demonstraram melhor capacidade de visualização, podendo assim demonstrar os diferentes comportamentos do escoamento superficial. Utilizou-se como referência em todos os testes a imagem do início do processo de simulação, que representa um cenário onde, o solo está todo coberto pela água da precipitação, na figura 21 esse cenário é representado pela primeira imagem da sequência. Em todos os testes o tempo de duração da simulação é igual, o que diferencia é o tempo de duração da

precipitação, ver tabela 1.

Observa-se uma diferença considerável entre as imagens, essa diferença deve-se ao fato de a duração da simulação ser igual para todos os testes realizados na primeira classe de testes e a duração da precipitação entre as simulações destes testes serem diferente, o que configura cenários diferentes entre as simulações. Com essas informações pode-se notar que, ao passo que a duração da precipitação diminui, o excesso de fluxo superficial acumulado de água é escoado aos pontos mais baixos do relevo.

A figura 22 mostra um gráfico do comportamento do escoamento superficial para as imagens que representam os testes realizados na primeira classe de testes. No gráfico os testes são apresentados como (T1.1, T1.2, T1.3 e T1.4), que correspondem às imagens (240s, 180s, 120s e 60s) respectivamente. Ainda no gráfico, é apresentado o teste (T1.5). O objetivo dessa simulação é testar a capacidade do sistema de simular um cenário onde, o solo está todo coberto por água, sem a ocorrência da precipitação e a interferência das variáveis “Infiltração e Evaporação”, demonstrando o comportamento do fluxo acumulado de água nessa situação.

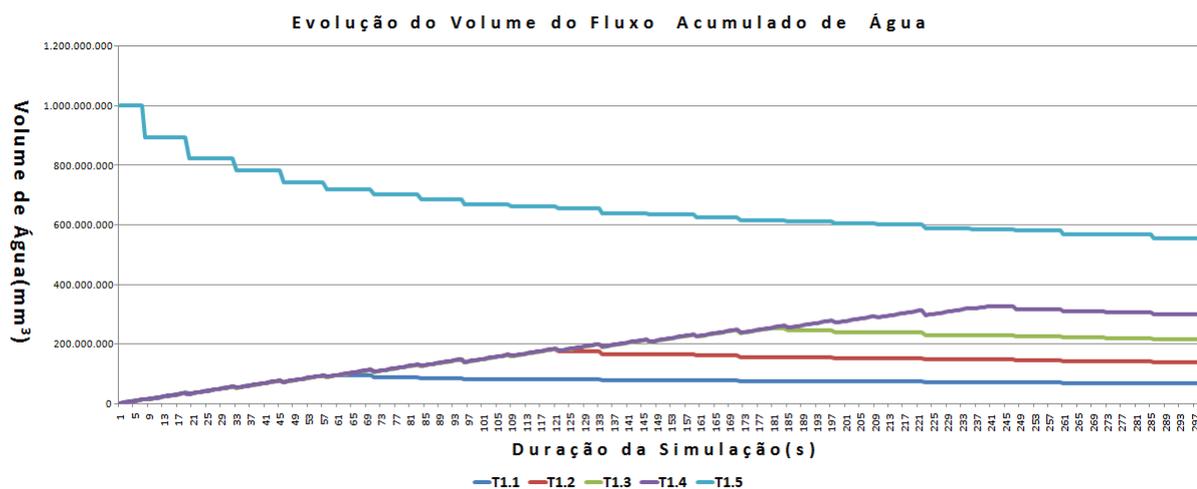


Figura 22: Evolução do Escoamento Superficial: Testes realizados sem considerar as variáveis ambientais

O gráfico da figura 22 demonstra que houve um acúmulo no volume de água até o encerramento da precipitação onde, a partir deste ponto, o acúmulo do fluxo do escoamento superficial começa a diminuir e a convergir para as áreas mais baixas do relevo, o que pode ser visualizado na figura 21, com a imagem que representa a simulação aos 60s de precipitação. Nessa imagem o intervalo entre o término da precipitação e o término da simulação é maior, proporcionando maior duração do escoamento para as áreas mais baixas, o que também resulta em uma diminuição no volume do fluxo de água nas áreas mais altas do relevo. O oposto ocorre na imagem que representa a simulação aos 240s de precipitação, neste caso o intervalo entre o término da precipitação e término da simulação é menor, proporcionando maior valor no volume do fluxo de água nas áreas mais altas do relevo e um volume inferior de fluxo de

água nas áreas mais baixas, resultando em uma área maior de solo encharcado no solo mais elevado.

O comportamento do gráfico em todos os testes realizados na primeira classe de testes demonstrou um aumento crescente no volume de água, até o ponto em que a duração da precipitação se encerra. A partir desse ponto nota-se que o valor no volume de água decresce em função da duração da simulação. Nota-se que os valores de volume de água não chegam à zero(0), o que configura um cenário onde o solo está seco, pois a duração da simulação não foi suficiente. Essa é uma deficiência do sistema, pois se limita a uma duração de simulação de 300s.

A simulação (T1.5) demonstra um comportamento pouco diferente, pois a simulação inicia com um volume de água no solo alto e demonstra uma forte queda, que é atribuído ao alto volume de água no solo, percebe-se que a curva do gráfico no início é muito brusca tornando-se suave em função da duração da simulação, isso é atribuído às bordas da imagem, quanto maior o volume escoado até as bordas da imagem, maior será a queda na curva do gráfico.

4.1.2 Segunda Classe de Testes

A segunda classe de testes apresenta 4(quatro) simulações, cada uma com o objetivo de simular os diferentes comportamentos do fluxo acumulado da água da precipitação quando essa toca o solo e escoar. Para tanto nestes testes serão consideradas as seguintes propriedades do solo, “limite de saturação” e “infiltração”. Será analisado o comportamento da variável “evaporação” considerando que parte da água se evapora.

Estes testes simulam um cenário em que ocorre a precipitação em diferentes intervalos, considerando um limite de saturação do solo, onde, à medida que a água infiltra o limite de saturação é preenchido, tornando o solo “saturado”. Uma vez “saturado”, o excesso de água escoar para um solo de menor altitude, perfazendo todo o ciclo novamente. Parte da água do excesso ainda sofre o fenômeno da evaporação que reduz o volume de água escoada no solo. O objetivo dessa classe de testes é simular cenários reais do comportamento da água precipitada no solo considerando suas propriedades. O teste (T2.1) não considera a variável evaporação conforme tabela 2.

Para esta classe de testes a mesma imagem será considerada. Serão realizadas simulações com intervalos de duração da precipitação de 240s 180s, 120s e 60s conforme tabela 2. O volume de precipitação inicial será de 4 mm/s, e será considerado para duração de todo o processo de simulação o mesmo intervalo da primeira classe de testes 300s. Nestes testes será considerado um valor para limite de saturação e valores para as variáveis “infiltração” e “evaporação” do solo.

Nos testes da tabela 2 foi considerado a duração da precipitação diferente para cada simulação. Pretende-se com este teste, verificar o comportamento do fluxo após o término da

precipitação, considerando que as variáveis de ambiente estarão agindo em todo o processo de simulação. Também foi inserido na simulação valores iguais para a variável “Lim_Saturação”, e diferentes valores para as variáveis “infiltração” e “evaporação”, dessa forma pretende-se verificar o comportamento da dinâmica do escoamento superficial considerando as variáveis anteriormente mencionadas.

Tabela 2: Valores inseridos no simulador para teste, considerando o Limite de Saturação do Solo, a Infiltração e a Evaporação.

Variáveis	Número de testes			
	T 2.1	T 2.2	T 2.3	T 2.4
Tempo de Simulação	300s	300s	300s	300s
Quantidade Precipitada	4mm/s	4mm/s	4mm/s	4mm/s
Intervalo Precipitado	1 – 240s	1 – 180s	1 – 120s	1 – 60s
Saturação Inicial	0	0	0	0
Limite de Saturação	2	2	2	2
Infiltração	2	2	1	1
Evaporação	1	1	1	0

A figura 23 ilustra a evolução do escoamento superficial como resultado da simulação da segunda classe de testes conforme descrito na tabela 2. Utilizou-se como referência em todos os testes a imagem do início do processo de simulação, que representa um cenário onde, o solo está todo coberto pela água da precipitação, na figura 23 esse cenário é representado pela primeira imagem da sequência. A duração da simulação para os testes realizados nesta classe é igual ao da primeira classe de testes, o que diferencia é o tempo de duração da precipitação e os valores para as variáveis ambientais, ver tabela 2.



Figura 23: Evolução do Escoamento Superficial: Testes realizados considerando as variáveis ambientais.

O objetivo da primeira imagem da sequência é demonstrar exatamente a diferença de um cenário inicial, onde o solo está todo coberto pela água sem a interferência das variáveis

ambientais e um cenário demonstrado pelas próximas imagens da sequência, as simulações de 240s, 180s, 120s e 60s, que demonstram cenários onde ocorre o escoamento superficial para as áreas mais baixas do solo com a interferência das variáveis ambientais. Dessa forma percebe-se na imagem que, a quantidade das áreas embranquecidas diminui em função da duração do processo de simulação.

Na imagem da simulação com 240s de duração, o limite de saturação do solo é considerado inicialmente, o que demonstra que, enquanto o valor de saturação não for atingido, a água continua infiltrando até este solo ficar saturado e o excesso de água escoar para outras áreas. Dessa forma percebe-se na imagem uma pequena diferença em relação à primeira imagem da sequência, que é atribuída aos diferentes valores das variáveis.

Essa diferença é visualizada também nas imagens das simulações de 180s, 120s e 60s. Percebe-se que ao passo que a duração da precipitação é reduzida e os valores das variáveis ambientais elevados, o volume do fluxo de água acumulado é reduzido em função do tempo de duração da simulação, tendendo a um volume de fluxo de água zero, ou seja, um cenário onde, não há a ocorrência de precipitação e toda a água presente no solo tenha sido escoada, infiltrada ou evaporada. O gráfico da figura 24 demonstra exatamente a discrepância entre o comportamento das simulações.

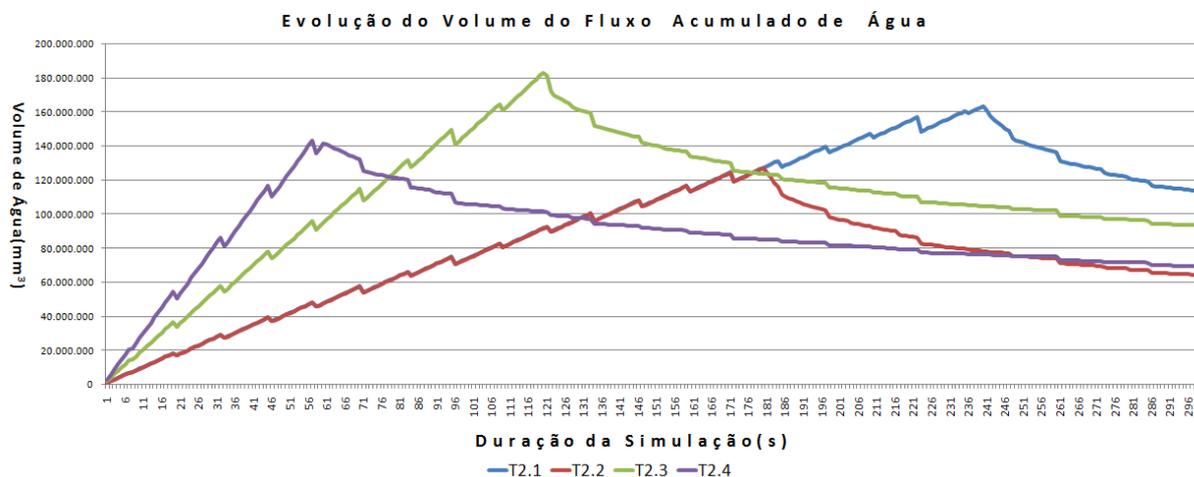


Figura 24: Evolução do Escoamento Superficial: Testes realizados considerando as variáveis ambientais.

Nota-se no gráfico da simulação “T2.1”, que o acúmulo no volume de água é menor em relação ao da simulação “T2.3”, isso ocorre pelo fato de a simulação “T2.1” ter uma duração da precipitação maior que a simulação “T2.3” e o valor para a variável ambiental “infiltração” ser maior, o que significa que ocorreu maior infiltração para essa simulação. Essa configuração de simulação demonstra que o volume de água acumulado em “T2.3” é maior devido à menor quantidade de água infiltrada, o que significa que ocorreu maior acúmulo no volume de água, uma vez que os valores para a variável “evaporação” são iguais em ambas as simulações.

Analisando o final da precipitação, percebe-se nesses testes “T2.1 e T2.3”, que o decréscimo nos valores de acúmulo do fluxo de água é atribuído à diferença entre os valores na variável “infiltração”, pois quanto maior o valor dessa variável maior será a infiltração de água no solo e conseqüentemente será menor o acúmulo no volume de água. Percebe-se ainda, que se a duração da simulação fosse mais extensa, o valor no volume final de água acumulada entre as simulações “T2.1 e T2.3”, seria menor para “T2.1”, pois para essa simulação o valor da variável ambiental “infiltração” é maior, ver tabela 2.

O comportamento demonstrado no gráfico da figura 24 para as simulações “T2.2 e T2.4” é basicamente semelhante ao comportamento das simulações analisadas anteriormente. Percebe-se na simulação “T2.2” que o comportamento segue o mesmo padrão da simulação “T2.1”, pois os valores são praticamente os mesmos, sendo diferente somente a duração da precipitação. A simulação “T2.4” apresenta um comportamento análogo à simulação “T2.3”, onde o acúmulo no volume de água demonstra-se mais verticalizado, o que é atribuído ao baixo valor da variável ambiental “infiltração”, e a variável “evaporação” não ter valor. Nota-se ainda, no final da simulação que o valor no volume de água acumulado é mais baixo para a simulação de “T2.2”, pois nessa os valores das variáveis “infiltração e evaporação” são maiores.

4.1.3 Terceira Classe de Testes

A terceira classe de testes apresenta 4 (Quatro) simulações onde, em cada simulação será considerado valores diferentes para as variáveis ambientais, conforme tabela 3. Essa classe de testes tem como objetivo simular ambientes onde a duração da precipitação é reduzida e a duração do escoamento do fluxo acumulado de água é maior. Espera-se com essas simulações que o valor do volume de água acumulado chegue a um valor 0zero, que demonstra um cenário onde o solo molhado, sofrendo a interferência das variáveis ambientais, após o escoamento de todo fluxo acumulado se torne seco. A tabela 3 demonstra os valores considerados nessa classe de testes.

Tabela 3: Valores inseridos para teste, considerando todas as variáveis ambientais.

Variáveis	Número de testes			
	T 3.1	T 3.2	T 3.3	T 3.4
Tempo de Simulação	300s	300s	300s	300s
Quantidade Precipitada	4mm/s	4mm/s	4mm/s	4mm/s
Intervalo Precipitado	1 – 90s	1 – 60s	1 – 50s	1 – 40s
Saturação Inicial	0	0	0	3
Limite de Saturação	2	2	2	2
Infiltração	2	2	2	2
Evaporação	1	1	1	1

Pretende-se demonstrar com as simulações dessa classe de testes, o comportamento do escoamento superficial da água após o término da precipitação. Nota-se nas simulações “T3.1, T3.2 e T3.3” um comportamento semelhante, isso ocorre, pois os valores das variáveis ambientais “infiltração e evaporação” são os mesmos. A diferença entre essas simulações é exatamente a duração da precipitação, que proporciona um acúmulo de fluxo maior para a simulação “T3.1” devido a uma duração da precipitação maior. Após o término da precipitação, o valor do volume de água acumulado reduz consideravelmente, demonstrando um cenário em que a água presente no solo escoou até que esse esteja totalmente sem água.

Esse cenário é demonstrado nas simulações “T3.3 e T3.4”, pelo fato de a duração da precipitação ser menor e haver maior duração do escoamento superficial da água, levando a um cenário onde o solo está com a ausência de água, ou seja, toda a água precipitada foi escoada. O gráfico da figura 25 demonstra o comportamento das simulações.

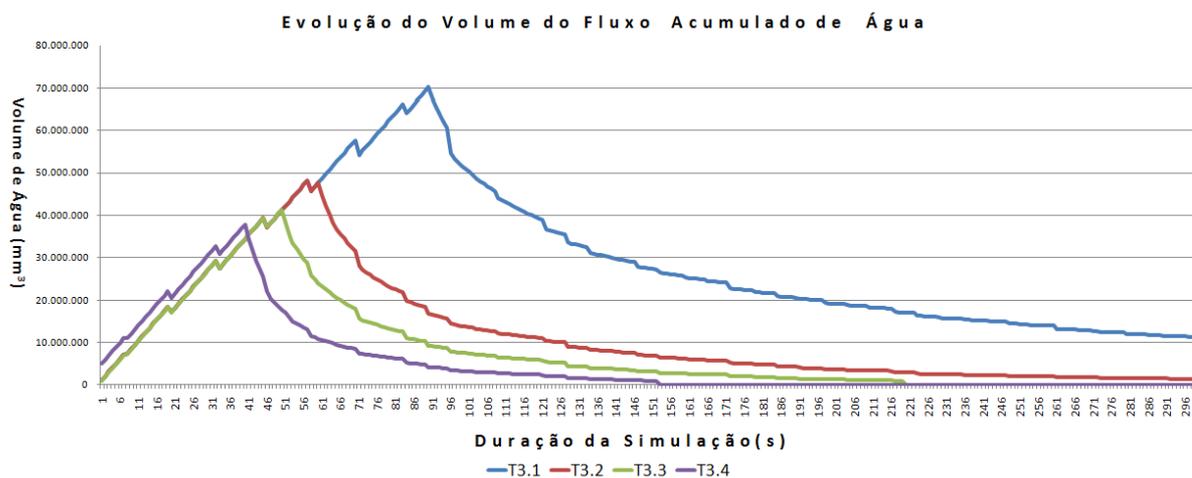


Figura 25: Evolução do Escoamento Superficial: Testes realizados considerando as variáveis ambientais.

Um comportamento pouco diferente pode ser visto na simulação “T3.4”, pois a simulação foi iniciada com uma saturação inicial do solo diferente em relação às outras simulações. Essa simulação procura demonstrar um cenário onde ocorre a precipitação em um solo já encharcado, ou seja, em um solo com a presença de água. Nas simulações “T3.3 e T3.4” percebe-se que o valor de volume de água acumulado converge para zero e permanece com esse valor até o final da simulação.

5 CONCLUSÃO

Pode-se notar através dos testes realizados, que o sistema consegue discretizar a dinâmica do escoamento superficial hortoniano espacialmente. Pode-se ainda visualizar através das imagens geradas dos tempos de escoamento que o sistema consegue direcionar o fluxo acumulado de água até as áreas mais baixas do relevo. As simulações mostraram que a duração de escoamento é proporcional à distância de um pixel para outro e também à inclinação da reta que os tangem. O comportamento do sistema alterou-se consideravelmente quando valores foram inseridos às variáveis infiltração, evaporação, limite de saturação do solo e intervalo de duração da precipitação, pois a interação dessas variáveis apresentou características mais próximas ao fenômeno real. Dessa forma pode-se concluir que as teorias apresentadas na pesquisa foram de extrema relevância para o entendimento do fenômeno e sua implementação no sistema. A dinâmica da interação dos processos que circundam o fenômeno pode ser visualizada com os resultados obtidos. Dessa forma acredita-se que o simulador está apto a simular o escoamento superficial hortoniano, objetivando principalmente auxiliar projetos que trabalham em pesquisas que envolvem o uso dos recursos hídricos. Ainda pode-se concluir que a classe “AUTOHIDRO” escrita em linguagem de programação C++, disponibiliza recursos para futuras implementações das variáveis ambientais, possibilitando a inserção das equações específicas da “Infiltração e Evaporação” permitindo que o simulador possa não somente simular, mas modelar esses fenômenos. Notou-se que o simulador é limitado no que diz respeito à modelagem de sistemas ambientais, pois atende somente ao objetivo principal do trabalho, restringindo-se apenas na concepção do simulador e na simulação do acúmulo do escoamento superficial hortoniano. Existe a necessidade de ferramentas computacionais completas que consigam modelar todos os eventos que compõem o ciclo, para tanto é sugerido como trabalhos futuros a implementação das equações para as variáveis ambientais inseridas no ciclo hidrológico e também a implementação do método de direção de fluxo D_{∞} , que supre todas as deficiências do método D_8 . Assim o simulador “AutoHidro” poderá representar com mais fidelidade o fenômeno do escoamento superficial.

REFERÊNCIAS

- ANDRADE MARCONI, M. de; LAKATOS, E. M. **Fundamentos de metodologia científica**. 4.ed. São Paulo: Atlas, 2001. 283p.
- BATTY, M.; XIE, Y.; SUN, Z. Modeling urban dynamics through GIS-based cellular automata. **Computers, Environment and Urban Systems**, St. Louis, MO, USA, v.23, n.3, p.205–233, May 1999.
- BERTONI, J. C.; TUCCI, C. E. M. Precipitação. , Universidade Estadual do Rio Grande do Sul RS, p.177–241, 1997.
- CHRISTOFOLETTI, A. **Modelagem de Sistemas Ambientais**. 1.ed. São Paulo: Edgard Blücher, LTDA, 1999. 233p.
- CLARKE, K. C.; HOPPEN, S.; GAYDOS, L. A self-modifying cellular automaton model of historical urbanization in the San Francisco Bay area. **Environment and Planning B: Planning and Design**, [S.l.], v.24, n.2, p.247–261, 1997.
- COUCLELIS, H. From cellular automata models to urban models: new principles for model development and implementation. **Environment and Planning B: Planning and Design**, [S.l.], v.24, n.2, p.165–174, 1997.
- CUNHA TRIVELATO, G. da. Técnicas de Modelagem de Sistemas Dinâmicos. **Instituto de Pesquisas Espaciais - INPE**, São José dos Campos SP, n.INPE-9665-NTC/358, p.1–19, 2003.
- FRANK, B. UMA ABORDAGEM PARA O GERENCIAMENTO AMBIENTAL DA BACIA HIDROGRÁFICA DO RIO ITAJAÍ, COM ÊNFASE NO PROBLEMA DAS ENCHENTES. , Universidade Federal de Santa Catarina, SC, 1995. Disponível em: <<http://www.eps.ufsc.br/teses/beate/indice/index.htm#index>>. Acesso em: 10 Fevereiro. 2009.
- GREGORIO, S. D.; SERRA, R.; VILLANI, M. Applying cellular automata to complex environmental problems: the simulation of the bioremediation of contaminated soils. **Theoretical Computer Science**, [S.l.], v.217, n.1, p.131–156, March 1999.
- HENZINGER, T. A. The Theory of Hibrid Automata. **Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science**, New Brunswick, NJ, USA, p.278–292, July 1996.
- HOPCROFT, J. E.; ULLMAN, J. D.; MONTWANI, R. **Introdução à Teoria de Autômatos, Linguagens e Computação**. 2.ed. Rio de Janeiro - RJ: Elsevier, 2002. Tradução de Introduction to automata theory.
- LIFE, C. G. of. **Math.com - The World of Math Online**. [S.l.: s.n.], 2008. Disponível em: <<http://www.math.com/students/wonders/life/life.html>>. Acesso em: 25 out. 2008.

MENDIONDO, E. M.; TUCCI, C. E. M. ESCALAS HIDROLÓGICAS. II: diversidade de processos na bacia vertente. **Revista Brasileira de Recursos Hídricos, ABRH**, Porto Alegre, (RS), v.2, n.1, p.59–122, 1997.

MIRANDA, E. E. de. Brasil em Relevo. **Embrapa Monitoramento por Satélite**, Campinas SP, 2005. Disponível em: <<http://www.relevobr.cnpm.embrapa.br>>. Acesso em: 12 abr. 2008.

MÜLLER, M.; GARCIA, K. S. Modelos Digitais de Terreno em Escalas Globais e sua Utilização em Recursos Hídricos. **Biblioteca do Instituto Lactec**, [S.l.], 2001. Disponível em: <http://www.lactec.org.br/OInstituto/downloads/Biblioteca/2001/036_2001.pdf>. Acesso em: 12 fev. 2008.

NEUMANN, J. V. **Theory of Self-Reproducing Automata**. University of Illinois - USA: Arthur W. Burks, 1966. Edited and completed by Arthur W. Burks.

PEDROSA, B. M. **Ambiente Computacional para Modelagem Dinâmica Espacial**. 2003. 70p. Tese (Doutorado em Computação Aplicada) — Instituto Nacional de Pesquisas Espaciais - INPE, São José dos Campos.

RODRIGUEZ, E.; MORRIS, C.; BELZ, J.; CHAPIN, E.; MARTIN, J.; W.DAFFER; HENSLEY, S. **An assessment of the SRTM topographic products**. Pasadena, Califórnia USA: Jet Propulsion Laboratory, 2005. Technical Report JPL D-31639, Disponível em: <<http://www2.jpl.nasa.gov/srtm/index.html>>. Acesso em: 27 set. 2008.

SENNA CARNEIRO, T. G. de. **Uma Arquitetura para modelagem Ambiental Empírica e Baseada nas Teorias dos Autômatos Celulares, Híbridos e Situados**. 2004. 52p. Tese (Doutorado em Computação Aplicada) — Instituto Nacional de Pesquisas Espaciais - INPE, São José dos Campos.

SILVEIRA, A. L. L. da. Ciclo Hidrológico e Bacia Hidrográfica. , Universidade Estadual do Rio Grande do Sul RS, p.35–51, 1997.

TARBOTON, D. G. A New method for the determination of flow directions and upslope areas in grid digital elevation models. **Water Resources Research**, Logan, UT, v.33, n.2, p.309–319, February 1997.

TAROLLI, P.; TARBOTON, D. G. A New method for determination of most likely landslide initiation point and the evaluation of digital terrain model scale in terrain stability mapping. **Hydrology and Earth System Sciences**, New York, NY, v.10, n.5, p.663–677, September 2006.

TUCCI, C. E. M. Hidrologia: ciência e aplicação. **Hidrologia: Ciência e Aplicação**, Universidade Estadual do Rio Grande do Sul RS, v.2, p.25–33, 1997.

TUCCI, C. E. M. Escoamento Superficial. **Hidrologia: Ciência e Aplicação**, Universidade Estadual do Rio Grande do Sul RS, v.2, p.391–441, 1997.

WOLFRAN, S. Cellular Automata. **Los Alamos Technical Reports and Publications**, Los Alamos, NM, USA, 1983. Disponível em: <<http://www.fas.org/sgp/othergov/doe/lanl/pubs/00285910.pdf>>. Acesso em: 21 Novembro. 2008.

GLOSSÁRIO

Fluxo Acumulado Acúmulo superficial de água no solo

GRID Matriz Regular Retangular para MDT

Hidrograma Gráfico da relação da Vazão Hidrogáfica em função do Tempo

LATEX Compilador de Textos

LED Editor de Textos para LATEX

PIXEL Menor ponto de uma imagem digital

Pluviometria Espessura da lâmina d'água (média) precipitada em determinada região

SRTM Projeto internacional liderado pela National Geospatial-Intelligence Agency (NGA) e a National Aeronautics and Space Administration (NASA), consiste em um sistema de radar especialmente modificado que voou a bordo do ônibus espacial Endeavour durante uma missão de 11 dias de fevereiro de 2000

APÊNDICE A APRESENTAÇÃO DO CÓDIGO FONTE DA CLASSE AUTOHIDRO

```
1  //-----
2  //
3  //          CLASSE AUTOHIDRO
4  //
5  //-----
6  #ifndef TAutoHidroH
7  #define TAutoHidroH
8  //-----
9
10 #include <Classes.hpp>
11 #include "TImagem.h"
12
13 //-----
14
15 struct TCamada
16 {
17     TImagem *Img;
18     int Banda;
19 };
20
21 class TAutoHidro
22 {
23     private:        // User declarations
24         int D1[8], D2[8];
25
26     // Propriedades da direção de fluxo
27         TCamada Direcao;
28     // Propriedades da Altura do Pixel
```

```

29         TCamada Altura;
30 // Propriedades do Ângulo de inclinação
31         TCamada Angulo;
32 // Propriedades da Água no Solo
33         float **Saturacao;
34 // Propriedades de limite de saturação de Água no Solo
35         float **LimSaturacao;
36 // Tempo de escoamento
37         float **TempoEscoa, **TempoAux, TempoPasso;
38 // Diferença de altitude
39         float **DifAlt, **Angulos;
40 // Linhas do tempo
41         bool *LTEIP, *LTEscoamento, *LTFrames;
42         float *LinhaTempo;
43         int LT_Tam; //Tamanho das linhas do tempo
44         int LT_QTframes; //Tamanho das linhas do tempo
45
46
47         int Tipo; //0 - D8, 1 - D infinito
48 //Propriedades da simulação:
49         int Duracao; //Quantidade unidades de tempo simulação
50         int TPrecI, TPrecF; //Intervalo tempo precipitação
51         int Intervalo; //Intervalo tempo simulação
52 //Geometria do espacial da modelagem
53         float Tam_Alt, Tam_Larg;
54         float CalcArea;
55 //Propriedades do ambiente:
56         float Precipitacao;
57         float Infiltracao;
58         float Evaporacao;
59 //         float Provoca_Escoamento(int plin0, int pcol0,
60                                     int plinD, int pcolD);
61         void Atualiza_Infiltracao();
62         void Atualiza_Evaporacao();
63         void Atualiza_Precipitacao(int ptempo);
64         void Atualiza_Escoamento();
65         void Calc_TempoEscoamento();
66         void Atualiza_Tempo();
67         void Atualiza_LinhasTempo();
68 //         void Atualiza_Velocidade();

```

```
69     public:          // User declarations
70
71 //Métodos GETs e SETs:
72     float Get_Direcao(int plin, int pcol);
73     void Set_Direcao(TImagem *pimg, int piband, int ptipo);
74
75     float Get_Altura(int plin, int pcol);
76     void Set_Altura(TImagem *pimg, int piband, int ptipo);
77
78     float Get_Angulo(int plin, int pcol);
79     void Set_Angulo(TImagem *pimg, int piband, int ptipo);
80
81     float Get_Saturacao();
82     float Get_Saturacao(int plin, int pcol);
83     void Set_Saturacao(int plin, int pcol, float pvalor);
84     void Set_Saturacao(float pvalor);
85     void Set_Saturacao(TImagem *pimg, int pbanda);
86
87     float Get_Excesso(int plin, int pcol);
88
89     float Get_Precipitacao();
90     void Set_Precipitacao(float pvalor);
91
92     float Get_Evaporacao();
93     void Set_Evaporacao(float pvalor);
94
95     float Get_Limite(int plin, int pcol);
96     void Set_Limite(int plin, int pcol, float pvalor);
97     void Set_Limite(float pvalor);
98
99     float Get_Infiltracao();
100    void Set_Infiltracao(float pvalor);
101
102    __fastcall TAutoHidro();
103 //    bool Recebe_Direcao(TImagem *pimg, int piband, int ptipo);
104    TImagem *Analise_ET();
105
106 //Atualiza o autômato da posição plin, pcol.
107 //    void Atualiza(int plin, int pcol);
108 //Cancela Excesso do escoamento dos Autômatos
```

```

109     void Set_Parametros(int pTempo, int pInterv,
110                         int pPrecInic, int pPrecFim);
111     float Get_PixelLarg();
112     void Set_PixelTam(float peixoy, float peiox);
113
114 };
115
116 //-----
117 #endif

```



```

1 //-----
2 //
3 //     autohidro: construtor da classe
4 //
5 //-----
6 #pragma hdrstop
7 #include "tautohidro.h"
8 #include "FluxoForm.h"
9
10 #pragma package(smart_init)
11
12
13 //-----
14 //Construtor
15 __fastcall TAutoHidro::TAutoHidro()
16 {
17
18     Direcao.Img = NULL;
19     Saturacao   = NULL;
20
21 //Usado para o cálculo da vizinhança D8
22     D1[1]=0; D1[2]= -1; D1[3]= -1; D1[4]= -1;
23     D1[5]=0; D1[6]=1; D1[7]=1; D1[8]=1;
24
25     D2[1]=1; D2[2]=1; D2[3]=0; D2[4]= -1;
26     D2[5]= -1; D2[6]= -1; D2[7]=0; D2[8]=1;
27 }
28
29 //-----
30

```

```

31 //-----
32 //Definir a direção de fluxo
33 //      para a modelagem espaço-temporal
34
35 TImagem *TAutoHidro::Analise_ET()
36 {
37     TImagem *psaida;
38     psaida = new TImagem();
39     AnsiString *pnombandas;
40     int ppos, plin, pcol;
41     float pvalor;
42
43     //Calcular o tempo de escoamento de cada autômato:
44     FluxoF->Label5->Caption ="Calculando tempo de escoamento";
45     FluxoF->Refresh();
46     Calc_TempoEscoamento();
47
48     //Atualiza linhas do tempo
49     FluxoF->Label5->Caption ="Gerando linhas do tempo...";
50     FluxoF->Refresh();
51     Atualiza_LinhasTempo();
52
53     //Vetor para armazenar o nome dos frames;
54     pnombandas = new AnsiString[LT_QTframes];
55     ppos = 0;
56     for(plin=0; plin<LT_Tam; plin++)
57     {
58         if(LTFrames[plin])
59         {
60             pnombandas[ppos] = FloatToStr(LinhaTempo[plin])
61                                     + " Segundo(s)";
62             ppos++;
63         }
64     }
65
66     FluxoF->Label5->Caption = "Gerando tempo zero...";
67     FluxoF->Refresh();
68
69     psaida->novo_img("AnaliseET.img", "", LT_QTframes,
70                   Direcao.Img->qtcol, Direcao.Img->qtlin,

```

```

71             0, 4, pnombandas, NULL, NULL,
72             Direcao.Img->georef);
73
74 //Executar as linhas do tempo:
75     for(ppos=0;ppos<LT_Tam;ppos++)
76     {
77 //A infiltração, evaporação e precipitação
78 //             acontecem de 1 em 1 segundo
79         if(LTEIP[ppos])
80         {
81 //Atualizar a infiltração em todos os
82 //             autômatos para o tempo "ptempo"
83         FluxoF->Label5->Caption ="Tempo "+FloatToStr(LinhaTempo[ppos])
84             + "s - Atualizando infiltração...";
85         FluxoF->Refresh();
86         Atualiza_Infiltracao();
87 //Atualizar evaporação em todos os autômatos para tempo "ptempo"
88         FluxoF->Label5->Caption = "Tempo " + FloatToStr(LinhaTempo[ppos])
89             + "s - Atualizando evaporação...";
90         FluxoF->Refresh();
91         Atualiza_Evaporacao();
92 //Atualizar Precipitacao em todos os autômatos para tempo "ptempo"
93         FluxoF->Label5->Caption = "Tempo " + FloatToStr(LinhaTempo[ppos])
94             + "s - Atualizando precipitação...";
95         FluxoF->Refresh();
96         Atualiza_Precipitacao((int)LinhaTempo[ppos]);
97     }
98
99     if(LTEscoamento[ppos])
100    {
101 //Atualizar Escoamento em todos os autômatos para tempo "ptempo"
102     FluxoF->Label5->Caption = "Tempo "+ FloatToStr(LinhaTempo[ppos])
103         + "s - Atualizando escoamento...";
104     FluxoF->Refresh();
105     Atualiza_Escoamento();
106    }
107
108     if(LTFrames[ppos])
109     {
110 //Gerar novo frame com a Saturação atual

```

```

111     for(plin=0;plin<psaida->qtlin;plin++)
112     {
113         for(pcol=0;pcol<psaida->qtcol;pcol++)
114         {
115             pvalor = Get_Saturacao(plin,pcol);
116             psaida->setvalor(&pvalor);
117         }
118     }
119
120 }
121
122 }
123
124 //Fechar arquivo gerado
125 psaida->Fecha();
126 //Abrir no módulo de leitura
127 psaida->carrega_imagem("AnaliseET.img","img");
128
129 //Fechar arquivo gerado
130 // psaida->Fecha();
131 // return psaida;
132 return psaida;
133 };
134
135 //-----
136 //Atualização do autômato em um intervalo de tempo
137 /*
138 void TAutoHidro::Atualiza(int plin, int pcol)
139 {
140     float psatura, pescoamento, plimsaturacao, pexcesso;
141     psatura = 0;
142     int ppos;
143
144     //Considera a saturação atual (água no solo):
145     psatura = Get_Saturacao(plin, pcol);
146     //Considerar a pluviometria (chuva):
147     psatura += Get_Precipitacao();
148     //Considerar a infiltração do solo:
149     psatura -= Get_Infiltracao();
150     //Considerar a evaporação do solo:

```

```

151     psatura -= Get_Evaporacao();
152 //Calcular o escoamento dos vizinhos:
153     pescoamento = 0;
154
155     for (ppos=1; ppos<=8; ppos++)
156         pescoamento += Provoca_Escoamento(plin+d1[ppos],
157                                             pcol+d2[ppos], plin, pcol);
158
159 //Considerar o escoamento dos vizinhos:
160     psatura += pescoamento;
161
162 //Atualizar as propriedades Saturação e Excesso:
163     if (psatura < 0)
164         psatura = 0;
165
166
167     plimsaturacao = Get_Limite(plin, pcol);
168     if (psatura > plimsaturacao)
169     {
170         pexcesso = psatura - plimsaturacao;
171         psatura = plimsaturacao;
172     }
173
174     Set_Saturacao(plin, pcol, psatura);
175
176 }
177 */
178 //-----
179 //GETs e SETs:
180 //-----
181 //Recebendo valores digitados e setando variáveis correspondentes -
182 //-----
183
184 //Limite de saturação
185 float TAutoHidro::Get_Limite(int plin, int pcol)
186 {
187     return LimSaturacao[plin][pcol];
188 }
189
190 void TAutoHidro::Set_Limite(float pvalor)

```

```

191 {
192     int plin, pcol;
193
194     if(Direcao.Img == NULL)
195     {
196         ShowMessage("Carregar a direção de fluxo antes!!!");
197         return;
198     }
199
200     LimSaturacao = criar_mat(Direcao.Img->qtlin, Direcao.Img->qtcol);
201     for(plin=0;plin<Direcao.Img->qtlin;plin++)
202         for(pcol=0;pcol<Direcao.Img->qtcol;pcol++)
203             LimSaturacao[plin][pcol] = pvalor;
204
205 }
206
207 void TAutoHidro::Set_Limite(int plin, int pcol, float pvalor)
208 {
209     LimSaturacao[plin][pcol] = pvalor;
210 }
211
212 //-----
213 //Precipitacao
214 float TAutoHidro::Get_Precipitacao()
215 {
216     return Precipitacao;
217 }
218
219 void TAutoHidro::Set_Precipitacao(float pvalor)
220 {
221     Precipitacao = pvalor;
222 }
223
224 //-----
225 //Evaporação
226 float TAutoHidro::Get_Evaporacao()
227 {
228     return Evaporacao;
229 }
230

```

```
231 void TAutoHidro::Set_Evaporacao(float pvalor)
232 {
233     Evaporacao = pvalor;
234 }
235
236 //-----
237 //Infiltração
238 float TAutoHidro::Get_Infiltracao()
239 {
240     return Infiltracao;
241 }
242
243 void TAutoHidro::Set_Infiltracao(float pvalor)
244 {
245     Infiltracao = pvalor;
246 }
247
248 //Direção de Fluxo
249 float TAutoHidro::Get_Direcao(int plin, int pcol)
250 {
251     float pvalor;
252     pvalor = Direcao.Img->pixel2(&(Direcao.Banda), &plin, &pcol);
253     return pvalor;
254 }
255
256 void TAutoHidro::Set_Direcao(TImagem *pimg, int piband, int ptipo)
257 {
258     Direcao.Img = pimg;
259     Direcao.Banda = piband;
260     Tipo = ptipo;
261 }
262
263 //Altura do Pixel ou valor numérico MDT
264 float TAutoHidro::Get_Altura(int plin, int pcol)
265 {
266     float pvalor;
267     pvalor = Altura.Img->pixel2(&(Altura.Banda), &plin, &pcol);
268     return pvalor;
269
270 }
```

```

271
272 void TAutoHidro::Set_Altura(TImagem *pimg, int piband, int ptipo)
273 {
274     Altura.Img    = pimg;
275     Altura.Banda = piband;
276     Tipo        = ptipo;
277
278 }
279
280 //Declividade
281 float TAutoHidro::Get_Angulo(int plin, int pcol)
282 {
283     float pvalor;
284     pvalor = Angulo.Img->pixel2(&(Angulo.Banda), &plin, &pcol);
285     return pvalor;
286
287 }
288
289 void TAutoHidro::Set_Angulo(TImagem * pimg, int piband, int ptipo)
290 {
291     Angulo.Img    = pimg;
292     Angulo.Banda = piband;
293     Tipo        = ptipo;
294
295 }
296 //-----
297 //Saturação
298 float TAutoHidro::Get_Saturacao(int plin, int pcol)
299 {
300     return Saturacao[plin][pcol];
301 }
302
303 void TAutoHidro::Set_Saturacao(int plin, int pcol, float pvalor)
304 {
305     Saturacao[plin][pcol] = pvalor;
306 }
307
308 void TAutoHidro::Set_Saturacao(float pvalor)
309 {
310     int plin, pcol;

```

```

311
312     if(Direcao.Img == NULL)
313     {
314         ShowMessage("Carregar a direção de fluxo antes!!!");
315         return;
316     }
317
318     Saturacao = criar_mat(Direcao.Img->qtlin, Direcao.Img->qtcol);
319     for(plin=0;plin<Direcao.Img->qtlin;plin++)
320         for(pcol=0;pcol<Direcao.Img->qtcol;pcol++)
321             Saturacao[plin][pcol] = pvalor;
322
323 }
324
325 //-----
326 //Excesso:
327 float TAutoHidro::Get_Excesso(int plin, int pcol)
328 {
329     float pexcesso=Get_Saturacao(plin, pcol)-Get_Limite(plin, pcol);
330     if(pexcesso < 0)
331         pexcesso = 0;
332     return pexcesso;
333 }
334
335 //-----
336 /*
337 //Provocar escoamento
338
339 float TAutoHidro::Provoca_Escoamento(int plin0, int pcol0,
340                                     int plinD, int pcolD)
341 {
342     float pescoamento;
343     int pdir0, plD, pcD;
344     int pqtlin = Direcao.Img->qtlin;
345     int pqtcol = Direcao.Img->qtcol;
346     pescoamento = 0;
347
348 //Posição fora da imagem
349     if((plin0<0) | (pcol0<0) | (plin0>pqtlin-1) | (pcol0>pqtcol-1))
350         return pescoamento;

```

```

351
352 //Direção de destino apontada pelo autômato de origem
353     pdirO = (int)Get_Direcao(plinO,pcolO);
354 //Posição de borda da imagem (Algoritmo D8 não considera bordas)
355     if(pdirO == -1)
356         return pescoamento;
357
358 //Posição do destino apontada pelo autômato de origem
359 // Calcula o destino do escoamento com base na
360 // direção pdirO da posição plinO, pcolO:
361 //4   3   2
362 //5     1
363 //6   7   8
364
365     plD = plinO + d1[pdirO];
366     pcD = pcolO + d2[pdirO];
367
368 //Se o destino do escoamento for o ponto de destino,
369 //                               provocar o escoamento:
370     if((plD == plinD) & (pcD == pcolD))
371     {
372         pescoamento = Get_Excesso(plinO, pcolO);
373         Cancela_Excesso(plinO, pcolO);
374     }
375     return pescoamento;
376 }
377 */
378
379 //-----
380 //Atualizar a infiltração em todos os autômatos
381 void TAutoHidro::Atualiza_Infiltracao()
382 {
383     float pnovovalor;
384     int pqtlin = Direcao.Img->qtlin;
385     int pqtcol = Direcao.Img->qtcol;
386     int plin, pcol;
387
388     for(plin=0;plin<pqtlin;plin++)
389     {
390         for(pcol=0;pcol<pqtcol;pcol++)

```

```

391     {
392         pnovovalor = Get_Saturacao(plin,pcol);
393         pnovovalor -= Get_Infiltracao();
394         Set_Saturacao(plin,pcol,pnovovalor);
395     }
396 }
397 }
398 //-----
399 //Atualizar a evaporação em todos os autômatos
400 void TAutoHidro::Atualiza_Evaporacao()
401 {
402     float pnovovalor;
403     int pqtlin = Direcao.Img->qtlin;
404     int pqtcol = Direcao.Img->qtcol;
405     int plin, pcol;
406
407     for(plin=0;plin<pqtlin;plin++)
408     {
409         for(pcol=0;pcol<pqtcol;pcol++)
410         {
411             pnovovalor = Get_Saturacao(plin,pcol);
412             pnovovalor -= Get_Evaporacao();
413             Set_Saturacao(plin,pcol,pnovovalor);
414         }
415     }
416
417 }
418 //-----
419 //Atualizar a precipitação em todos os autômatos
420 void TAutoHidro::Atualiza_Precipitacao(int ptempo)
421 {
422     float pnovovalor;
423     int pqtlin = Direcao.Img->qtlin;
424     int pqtcol = Direcao.Img->qtcol;
425     int plin, pcol;
426
427     //Verificar o intervalo de tempo da precipitação:
428     if((ptempo<(TPrecI-1))|(ptempo>(TPrecF-1)))
429     {
430         return;

```

```

431     }
432
433     for(plin=0;plin<pqtlin;plin++)
434     {
435         for(pcol=0;pcol<pqtcop;pcol++)
436         {
437             pnovovalor = Get_Saturacao(plin,pcol);
438             pnovovalor += Get_Precipitacao();
439             Set_Saturacao(plin,pcol,pnovovalor);
440         }
441     }
442
443 }
444 //-----
445 //Atualizar a escoamento em todos os autômatos
446 void TAutoHidro::Atualiza_Escoamento()
447 {
448     int plin, pcol, ppos, pdir, plD, pcD;
449     int pqtlin = Direcao.Img->qtlin, pqtcop = Direcao.Img->qtcop;
450     float pnovovalor, pexcesso, plimite;
451
452     //Posição do destino apontada pelo autômato de origem
453     // Calcula o destino do escoamento com base na
454     // direção pdir0 da posição plin0, pcol0:
455     //4   3   2
456     //5       1
457     //6   7   8
458
459     //Atualizar matriz controle do tempo. Decrementa o passo atual
460     Atualiza_Tempo();
461
462     for(plin=0;plin<pqtlin;plin++)
463     {
464         for(pcol=0;pcol<pqtcop;pcol++)
465         {
466
467         //Só atualizar os autômatos com contador de tempo zerado
468             if(TempoAux[plin][pcol] == 0)
469             {
470         //Direção de destino apontada pelo autômato plin,pcol

```



```

511 Duracao = pTempo; //Quantidade unidades tempo da simulação
512 Intervalo = pInterv; //Intervalo de tempo da simulação
513 TPrecI = pPrecInic; //Momento de início da precipitação
514 TPrecF = pPrecFim; //Momento do final da precipitação
515 }
516
517 //-----
518 /*
519 void TAutoHidro::Atualiza_Velocidade()
520 {
521     int pqtlin = Direcao.Img->qtlin;
522     int pqtcol = Direcao.Img->qtcol;
523     int plin, pcol;
524     float pG, pvelocidadef, ptqueda;
525     float pmdt, pdeclividade;
526
527     // Atualizar a velocidade e a aceleração do pixel
528     for(plin=0;plin<pqtlin;plin++)
529     {
530         for(pcol=0;pcol<pqtcol;pcol++)
531         {
532             pmdt = Get_Altura(plin,pcol);
533             pdeclividade = Get_Angulo(plin,pcol);
534             if ((pdeclividade!= -1) & (pdeclividade != 0))
535             {
536
537                 // valor da gravidade = 9,80665;
538                 // calculando a gravidade para a declividade no ponto;
539                 // gravidade =(gravidade x ângulo de inclinação);
540                 pG = (9,80665*pdeclividade);
541                 ptqueda = sqrt((2*pmdt)/pG);
542                 pvelocidadef = (9,80665*ptqueda);
543             }
544         }
545     }
546 }
547 */
548 //Calculando Area Pixel do Autômato;-----
549 float TAutoHidro::Get_PixelLarg()
550 {

```

```

551     return Tam_Larg;
552 }
553
554 void TAutoHidro::Set_PixelTam(float peixoy, float peiox)
555 {
556     Tam_Larg = peiox;
557     if(Tam_Larg <= 0.0)
558         Tam_Larg = MINFLOAT;
559     Tam_Alt  = peixoy;
560     if(Tam_Alt <= 0.0)
561         Tam_Alt = MINFLOAT;
562 }
563 //-----
564
565 void TAutoHidro::Calc_TempoEscoamento()
566 {
567     TImagem *pimgaux;
568     AnsiString pnombandas[3];
569     float pvalor;
570     int plin, pcol, pdir, plinD, pcolD;
571     float pdistancia, pteta, ptangente, pg, pseno, phipotenusa;
572     float  pG = 9.80665, paltura, palturaD, pdeltaalt;
573
574     if(Direcao.Img == NULL)
575     {
576         ShowMessage("Carregar a direção de fluxo antes!!!");
577         return;
578     }
579
580     //Alocar Propriedades
581     TempoEscoa =criar_mat(Direcao.Img->qtlin, Direcao.Img->qtcol);
582     DifAlt     =criar_mat(Direcao.Img->qtlin, Direcao.Img->qtcol);
583     Angulos    =criar_mat(Direcao.Img->qtlin, Direcao.Img->qtcol);
584
585     pdistancia = Get_PixelLarg();
586     TempoPasso = MAXFLOAT ; //Step da linha do tempo
587     for(plin=0;plin<Direcao.Img->qtlin;plin++)
588     {
589         for(pcol=0;pcol<Direcao.Img->qtcol;pcol++)
590         {

```

```

591 //Direção de destino apontada pelo autômato
592     pdir = (int)Get_Direcao(plin,pcol);
593 //Calcular a diferença de altitude com o destino do escoamento:
594     paltura = (float) Get_Altura(plin,pcol); // Altura do pixel
595     plinD = plin + D1[pdir];
596     pcolD = pcol + D2[pdir];
597     palturaD = (float) Get_Altura(plinD,pcolD);
598     pdeltaalt = paltura - palturaD;
599
600     TempoEscoa[plin][pcol] = -1;
601     Angulos[plin][pcol] = -1;
602     DifAlt[plin][pcol] = -1;
603
604 //Só calcular tempo se não for borda da
605 // imagem(D8 não considera bordas) e indicar
606 // o escoamento (pdir!=0 e paltura >0)
607 //Calcular o tempo do escoamento para o autômato plin, pcol
608     if((pdir > 0)&(pdeltaalt >0))
609     {
610 //Calculando a Tangente
611         ptangente = (pdeltaalt/pdistancia);
612 //Calculando o Teta
613         pteta = atan(ptangente);
614 //Calculando o Seno do ângulo Teta
615         pseno = sin(pteta);
616 //Calculando o G' (Proporção da Gravidade)
617         pg = (pG*pseno);
618 //Calculando a hipotenusa (Rampa de descida)
619         phipotenusa = sqrt((pdeltaalt*pdeltaalt)
620                             +(pdistancia*pdistancia));
621 //Calculando o Tempo do Escoamento considerando uma rampa
622         TempoEscoa[plin][pcol] = sqrt((2*phipotenusa)/pg);
623 //Atualizar propriedades Diferença de altitude e ângulos
624         DifAlt[plin][pcol] = pdeltaalt;
625         Angulos[plin][pcol] = pteta;
626
627 //linha do tempo será discretizada com base
628 // no menor tempo de escoamento:
629         if(TempoEscoa[plin][pcol] < TempoPasso)
630             TempoPasso = TempoEscoa[plin][pcol];

```

```

631     }
632     else
633     {
634         TempoEscoa[plin][pcol] = -1;
635     }
636 }
637 }
638
639 //Atualizar matriz auxiliar usada para
640 //                controlar a atualização dos autômatos
641 //Começa com o mesmo conteúdo de TempoEscoa
642 TempoAux = criar_mat(Direcao.Img->qtlin, Direcao.Img->qtcol);
643 for(plin=0;plin<Direcao.Img->qtlin;plin++)
644     for(pcol=0;pcol<Direcao.Img->qtcol;pcol++)
645         TempoAux[plin][pcol] = TempoEscoa[plin][pcol];
646
647 //Gerar uma imagem com os tempos
648 //                de escoamento de cada autômato:
649 FluxoF->Label5->Caption = "Gravando imagem
650                        com tempos de escoamentos...";
651 FluxoF->Refresh();
652 pimgaux = new TImagem();
653 pnombandas[0] = "Tempo de Escoamento";
654 pnombandas[1] = "Diferença de Altitude";
655 pnombandas[2] = "Ângulo";
656 pimgaux->novo_img("AnaliseEscoa.img", "", 3, Direcao.Img->qtcol,
657                Direcao.Img->qtlin, 0, 4, pnombandas, NULL, NULL,
658                Direcao.Img->georef);
659 for(plin=0;plin<Direcao.Img->qtlin;plin++)
660 {
661     for(pcol=0;pcol<Direcao.Img->qtcol;pcol++)
662     {
663         pvalor = TempoAux[plin][pcol];
664         pimgaux->setvalor(&pvalor);
665     }
666 }
667 for(plin=0;plin<Direcao.Img->qtlin;plin++)
668 {
669     for(pcol=0;pcol<Direcao.Img->qtcol;pcol++)
670     {

```

```

671     pvalor = DifAlt[plin][pcol];
672     pimgaux->setvalor(&pvalor);
673 }
674 }
675 for(plin=0;plin<Direcao.Img->qtlin;plin++)
676 {
677     for(pcol=0;pcol<Direcao.Img->qtcol;pcol++)
678     {
679         pvalor = Angulos[plin][pcol];
680         pimgaux->setvalor(&pvalor);
681     }
682 }
683 pimgaux->Fecha();
684
685 }
686
687 //-----
688 //Decrementa TempoPasso da matriz TempoAux
689 void TAutoHidro::Atualiza_Tempo()
690 {
691     int plin, pcol;
692     for(plin=0;plin<Direcao.Img->qtlin;plin++)
693     {
694         for(pcol=0;pcol<Direcao.Img->qtcol;pcol++)
695         {
696             TempoAux[plin][pcol] -= TempoPasso;
697             if(TempoAux[plin][pcol] < 0.0)
698                 TempoAux[plin][pcol] = 0.0;
699         }
700     }
701 }
702
703 void TAutoHidro::Atualiza_LinhasTempo()
704 {
705     int ppos, pqtde;
706     float pintervalo, ptempo;
707     AnsiString pmensa;
708     //Menor passo do tempo:
709     pintervalo = TempoPasso;
710     if(pintervalo > 1)

```

```

711     pintervalo = 1;
712 //Quantidade máxima de passos:
713     pqtde = Duracao/pintervalo;
714 //Atualizar linhas do tempo:
715     LTEIP          = criar_vet_bool(pqtde);
716     LTEScoamento  = criar_vet_bool(pqtde);
717     LTFrames       = criar_vet_bool(pqtde);
718     LinhaTempo     = criar_vet(pqtde);
719     pmensa = "Tempos: ";
720     for(ppos=0;ppos<pqtde;ppos++)
721     {
722         LTEIP[ppos]          = false;
723         LTEScoamento[ppos]  = false;
724         LTFrames[ppos]       = false;
725         LinhaTempo[ppos]     = ppos*ppos;
726         pmensa += FloatToStr(LinhaTempo[ppos]) + "#";
727     }
728
729 //Marcar posições de atualização da infiltração,
730 //                               evaporação, precipitação
731 //Intervalo: 1s em 1s
732     pmensa += "\n";
733     pmensa += "IEP: ";
734     ptempo = 0;
735     for(ppos=0;ppos<pqtde;ppos++)
736     {
737         if((LinhaTempo[ppos]) >= ptempo)
738         {
739             LTEIP[ppos] = true;
740             ptempo += 1;
741             pmensa += IntToStr(ppos);
742         }
743
744     }
745
746 //Marcar posições de atualização do escoamento
747 //Intervalo: propriedade TempoPasso
748     pmensa += "\nEsc: "+FloatToStr(TempoPasso)+"\n";
749     ptempo = 0;
750     for(ppos=0;ppos<pqtde;ppos++)

```

```
751     {
752         if((LinhaTempo[ppos]) >= ptempo)
753         {
754             LTEscoamento[ppos] = true;
755             ptempo += TempoPasso;
756             pmensa += IntToStr(ppos);
757         }
758
759     }
760     ShowMessage(pmensa);
761     //Marcar posições de gravação dos frames
762     //Intervalo: propriedade Intervalo
763     LT_QTframes = 0;
764     ptempo = 0;
765     pmensa += "\nFrm: ";
766     for(ppos=0;ppos<pqtde;ppos++)
767     {
768         if((LinhaTempo[ppos]) >= ptempo)
769         {
770             LTFrames[ppos] = true;
771             ptempo += Intervalo;
772             LT_QTframes++;
773             pmensa += IntToStr(ppos);
774         }
775
776     }
777     ShowMessage(pmensa);
778
779     //Atualizar o tamanho dos vetores de linhas do tempo:
780     LT_Tam = pqtde;
781 }
```

ÍNDICE

Anexo A, 81	Resultados, 42
Apendice A	
Apresentação do código fonte, 58	
Conclusão, 54	
Descrição dos Métodos, 37	
Desenvolvimento	
Análise e interpretação dos Dados, 42	
Autômato Celular, 32	
Autômato Finito, 31	
Autômato Híbrido, 35	
Autômatos, 31	
Ciclo Hidrológico, 20	
Definição do Sistema, 38	
Descrição dos Métodos, 37	
Literatura, 21	
Metodologias para o Cálculo da Direção de Fluxo da Água, 25	
Modelo Digital de Terreno - (MDT), 23	
Primeiro Teste, 43	
Segundo Teste, 45	
Terceiro Teste, 48	
Teste do Acúmulo de Água Superficial, 49	
Fundamentação Teórica, 20	
Introdução, 15	
Delimitação do Tema, 15	
Justificativas, 17	
Metodologia, 18	
Objetivos, 16	
Tema, 15	