

Trabalhando com LuccME

Guia do usuário

Versão 3.1 | Setembro 2017

The logo for LuccME is displayed in a large, bold, sans-serif font. The letters are color-coded: 'L' is blue, 'U' is light blue, 'C' is orange, 'C' is yellow-orange, 'M' is green, and 'E' is light green. The letters are slightly overlapping and have a soft shadow effect.

Trabalhando com LuccME

Guia do usuário

Versão 3.1 | Setembro 2017



Autores:

Equipe LuccME

LuccME

Sumário

1	Introdução ao LuccME	1
1.1	O que é LuccME	1
1.2	A filosofia do LuccME	1
1.3	Para quem se destina este guia?	3
2	Iniciando com LuccME	4
2.1	Instalando o LuccME	4
3	Construindo um modelo com LuccME	5
3.1	Introdução	5
3.2	Configurando os parâmetros do Modelo	17
3.2.1	Definições do Modelo	17
3.2.2	Definições Espaciais	17
3.2.3	Tipos de Uso da Terra	17
3.2.4	Componentes	18
3.2.5	Salvar Parâmetros	18
3.3	Configurando os Componentes	18
3.3.1	Componente de Potencial	19
3.3.1.1	Componentes de Potencial Discretos	19
3.3.1.2	Componentes de Potencial Contínuos	22
3.3.2	Componente de Alocação	25
3.3.2.1	Componentes de Alocação Discretos	25
3.3.2.2	Componentes de Alocação Contínuos	27
3.3.3	Componente de Demanda	30
3.4	Calibração e Validação	32
4	Recursos avançados	34

4.1	Variáveis dinâmicas.....	34
4.2	Regiões	34
4.3	Cenários	35
5	Executando um modelo com LuccME.....	36
5.1	Exercícios.....	36
	Referências	39

1 Introdução ao LuccME

1.1 O que é LuccME

LuccME (*Land Use and Cover Change Modelling Environment*) é um arcabouço de modelagem espacialmente explícita de código aberto para mudanças de uso e cobertura da terra (*LUCC* em inglês) desenvolvido pelo Centro de Ciência do Sistema Terrestre (CCST) e seus colaboradores, como uma extensão do TerraME (Carneiro et al., 2013). Usando o LuccME o usuário pode criar facilmente modelos espacialmente explícitos de desmatamento, expansão agrícola, degradação florestal, expansão urbana e outras mudanças de uso da terra, em diferentes escalas e áreas de estudo, combinando componentes existentes e/ou criando os seus próprios. Estes modelos de uso da terra basicamente quantificam no tempo e no espaço as relações entre fatores determinantes e os padrões espaciais e temporais de mudanças (Verburg et al. 2004). Os modelos também são ferramentas que visam apoiar a análise das causas e consequências destas mudanças.

1.2 A filosofia do LuccME

Existem diferentes tipos de modelos LuccME, os quais podem ser classificados de acordo com a finalidade, escala, abordagem ou teoria subjacente. Apesar da diversidade de modelos, uma estrutura funcional comum pode ser identificada em diversos modelos espacialmente explícitos (Verburg et al., 2006; Eastman et al., 2005), conforme ilustrado na Figura 1. Em geral, três componentes principais podem ser definidos: i) *Demanda* – responsável pelo cálculo da magnitude ou quantidade de mudança; ii) *Potencial* – responsável pelo cálculo da adequação ou suscetibilidade de mudança de cada célula; e iii) *Alocação* – responsável pela distribuição espacial das mudanças com base na demanda por terra e potencial de mudança de cada célula. Estes componentes são organizados, em geral, de maneira *top-down* onde a demanda por terra é alocada espacialmente de acordo com a adequação de cada célula. Diversos modelos LuccME bem conhecidos seguem esta estrutura, incluindo a família de modelos CLUE (Veldkamp e Fresco, 1996; Verburg et al., 1999; Verburg et al., 2001), Dinamica EGO (Soares-Filho et al., 2002) e GEOMOD (Pontius et al., 2001), empregando diferentes técnicas e abordagens para os

três componentes (Eastman et al., 2009; Lesschen et al., 2007). Entretanto, estes modelos são implementados em diferentes plataformas computacionais, geralmente não possuem código aberto, portanto, não podem ser facilmente modificados ou combinados. Neste sentido, o LuccME permite a construção de novos modelos, combinando componentes de *Demanda*, *Potencial* e *Alocação*, desenvolvidos de acordo com as ideias originais dos principais modelos disponíveis na literatura.

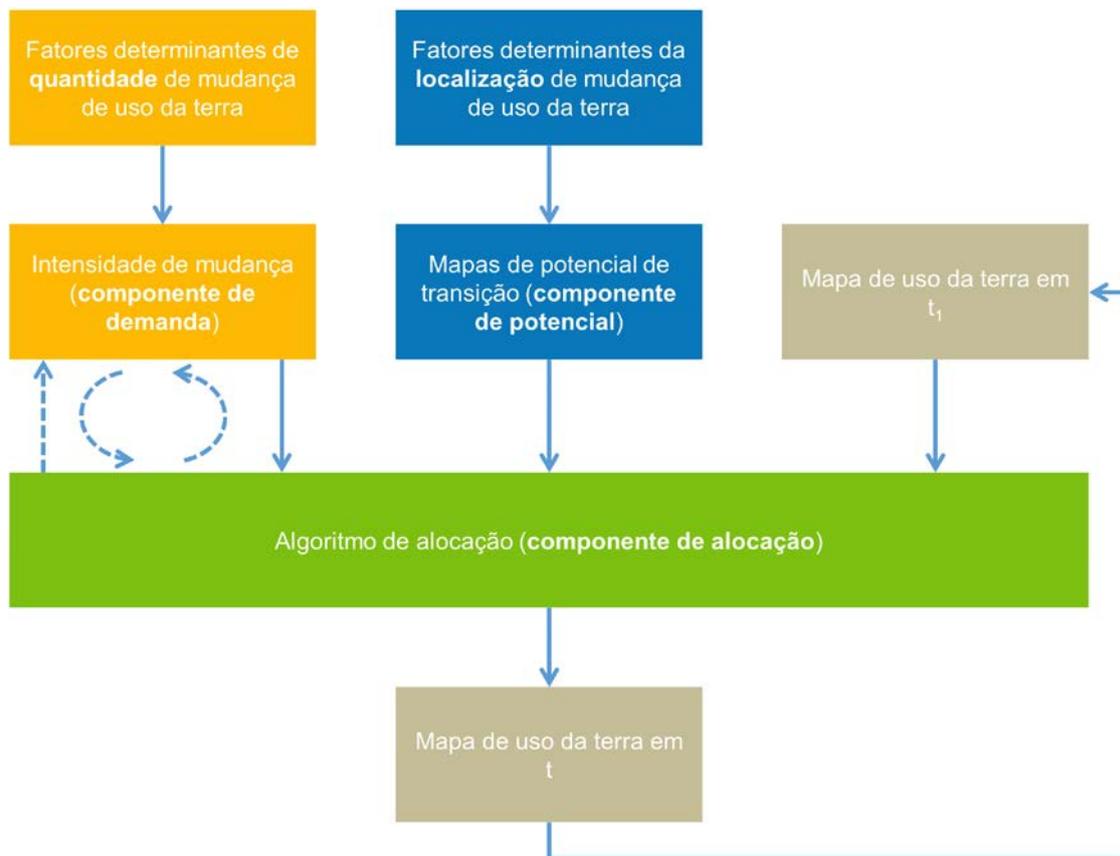


Figura 1 - Estrutura geral dos principais modelos LuccME espacialmente explícitos disponíveis na literatura (adaptados de Verburg et al., 2006).

Uma característica importante do LuccME é sua organização modular. Isso permite ajustar a estrutura do modelo de acordo com as necessidades do usuário. No LuccME, os componentes podem ser selecionados e modificados com base nas necessidades de uma determinada aplicação ou escala de análise e facilmente parametrizados a partir de uma interface simples. O LuccME é construído sobre o TerraME¹, um ambiente de programação genérico para modelagem espacial dinâmica, desenvolvido para suportar modelos de diversos domínios, incluindo, hidrologia, biodiversidade, uso da terra e muitos outros. Sua linguagem de programação

¹ Mais informação: <http://www.terrame.org/doku.php>

tem funções embutidas que tornam mais fácil o desenvolvimento de modelos multiescala e multiparadigma para aplicações ambientais. O TerraME oferece interface com a biblioteca geográfica TerraLib (Camara et al., 2008), o que permite aos modelos acesso direto ao banco de dados geográfico. O LuccME, TerraME e TerraLib são produtos tecnológicos desenvolvidos pelo Instituto Nacional de Pesquisas Espaciais (INPE) e seus parceiros.

Além de proporcionar uma maneira fácil de usar e criar novos modelos, o LuccME foi desenvolvido para suportar o desenvolvimento de modelos que podem ser integrados a modelos ambientais multiescala mais complexos. Neste sentido, os modelos criados utilizando o LuccME são encapsulados em “ambientes”, um conceito central no TerraME, como mostra Carneiro (2006), para que eles possam ser facilmente acoplados a outros modelos, por exemplo, modelos do sistema terrestre ou modelos de uso da terra multiescala.

1.3 Para quem se destina este guia?

Este guia fornece informações sobre o funcionamento dos componentes implementados, bem como, orientação ao usuário para construir e executar um modelo a partir da plataforma LuccME. Entretanto, o uso desta ferramenta requer, no mínimo, o entendimento de como os modelos de uso da terra funcionam de modo geral (Demanda, Potencial, Alocação) e habilidade para manipular banco de dados geográfico. Qualquer modelo construído no LuccME requer um banco de dados espacialmente explícito contendo dados observados de uso e cobertura da terra e diferentes fatores determinantes para cada classe de uso em formato compatível com versão 3.1, conforme detalhado no Capítulo 3.

2 Iniciando com LuccME

Este capítulo descreve como instalar e/ou configurar o LuccME em seu computador.

2.1 Instalando o LuccME

Existem duas formas de se trabalhar com o LuccME, utilizando a interface gráfica ou um editor de texto com *prompt* de comando.

1. LuccME com interface gráfica (recomendado para iniciantes):
Necessário apenas executar o arquivo de instalação (64 bits).

2. LuccME sem interface gráfica:

Para utilizar o LuccME sem interface gráfica existem duas opções:

Utilizando um editor pré-configurado:

1. Baixe a IDE ZeroBrane Studio configurada para executar o LuccME.
2. Descompacte o arquivo em C:.
3. Abra a IDE ZeroBrane e no menu *Project > Lua Interpreter* selecione a opção LuccME.
4. Para executar o modelo clique no ícone com uma seta verde (F5).

Utilizando um editor de sua preferência:

1. Baixe o LuccME (código fonte).
2. Instale o TerraME 2.0.
3. Descompacte o arquivo luccme.zip e coloque o diretório "luccme", no diretório "bin/packages" do TerraME.

Todos os arquivos estão disponíveis em: <http://luccme.ccst.inpe.br>.

3 Construindo um modelo com LuccME

3.1 Introdução

A construção de um modelo no LuccME pode ser dividida em 5 passos, conforme ilustrado na Figura 2. O primeiro passo consiste na definição do objetivo da aplicação como, por exemplo, expansão agrícola, expansão urbana, desertificação, desmatamento. Nesta etapa também costuma ser definida a escala espacial e temporal de trabalho, os tipos de uso da terra, assim como, os fatores determinantes de mudança para o modelo pretendido (distância a centros urbanos, presença de áreas protegidas, relevo, etc).

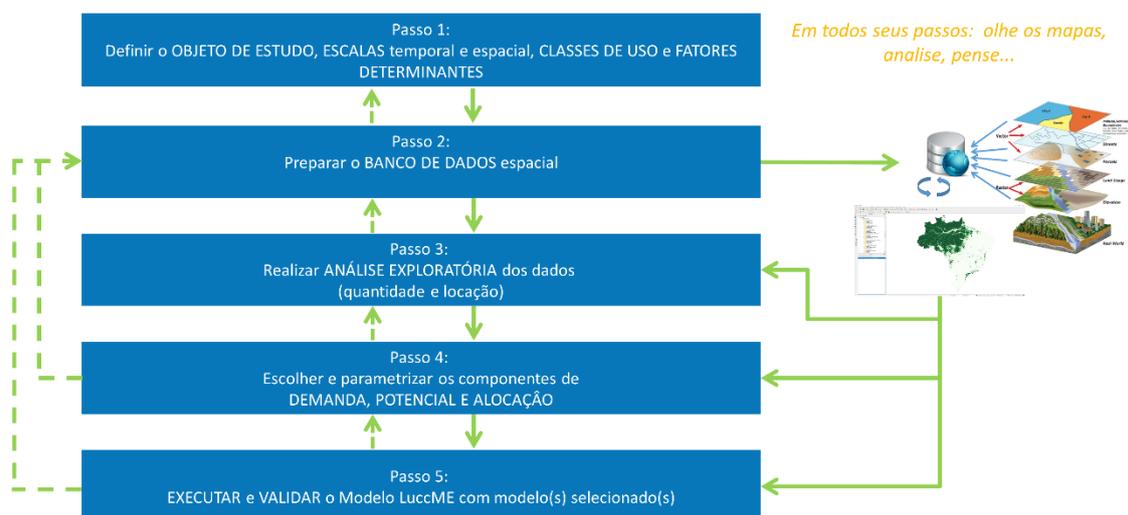


Figura 2 - Etapas para construção de um modelo de uso da terra no LuccME 3.1.

O Segundo passo compreende a preparação do banco de dados, expresso pelo preenchimento de um espaço celular. O espaço celular (Figura 3) é uma estrutura matricial generalizada onde cada célula está associada a vários tipos de atributos, sejam eles classes de uso/cobertura da terra (ex.: floresta, agricultura, pastagem, etc) ou fatores determinantes (ex.: distância a rodovias, presença de assentamentos rurais, declividade média, etc). O objetivo consiste em homogeneizar informações provenientes de diferentes fontes, em formatos distintos (dados vetoriais, matriciais e também outros planos celulares), agregando-os em uma mesma base espaço-temporal. O espaço celular pode ser construído através da interface FillCell, disponível

em (<http://luccme.ccst.inpe.br/category/download-pt/>) ou diretamente no TerraME 2.0, conforme descrito abaixo. O espaço celular pode ainda ser construído a partir de Sistemas de Informação Geográfica (SIG) de preferência do usuário como TerraView, ArcGIS ou QGIS. Para qualquer aplicação o espaço celular deve ser construído e preenchido utilizando um sistema de coordenadas planas (ex: UTM, Policonico, Albers).

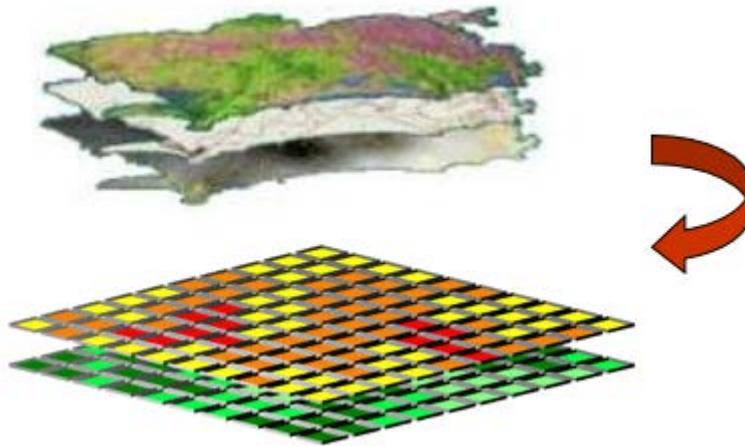


Figura 3 - Espaço celular – processo de homogeneização de diferentes bases de dados em espaço celular.

O gerenciador de scripts FillCell é uma ferramenta oferecida ao usuário de forma gratuita e tem por objetivo auxiliar na construção e preenchimento de um espaço celular compatível com o LuccME 3.1. Através de uma interface intuitiva, o usuário fornece as informações inerentes a sua aplicação sem precisar se importar com a sintaxe de programação. Contudo, esta ferramenta não limita o usuário que desejar construir seus scripts diretamente no TerraME, através do editor de sua preferência. Abaixo encontra-se a janela principal do FillCell, onde é possível acessar as funcionalidades deste gerenciador, tais como:

- Criar um novo script;
- Abrir um script gerado por esta ferramenta;
- Selecionar o idioma de exibição (Português ou Inglês);

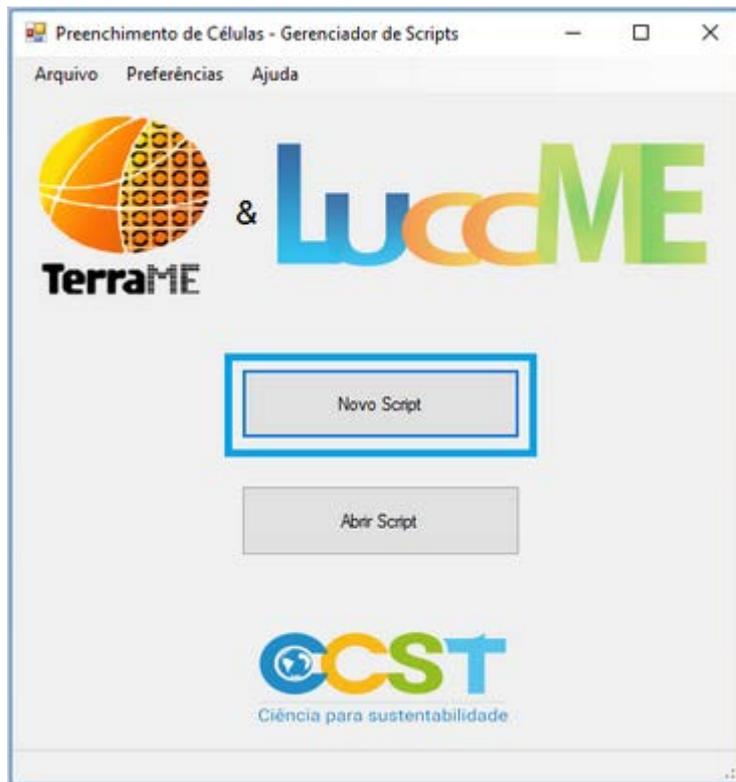


Figura 4 - Janela principal do FillCell.

Outra forma de acessar o formulário para criação de um novo script é através do menu principal, conforme ilustrado na Figura 5.



Figura 5 - Menu Principal - Arquivo > Novo.

Uma vez selecionada, a janela de criação de script é aberta, conforme ilustrado na Figura 6. Nesta etapa o usuário deve, em primeiro lugar, indicar o diretório onde o script será salvo, utilizando a opção “pasta para salvar os scripts”. Após a seleção da pasta aonde o arquivo será salvo, o endereço selecionado será exibido para o usuário. Em segundo lugar, o usuário deve indicar um nome de saída para o script que será criado, utilizando a opção “nome do script”.

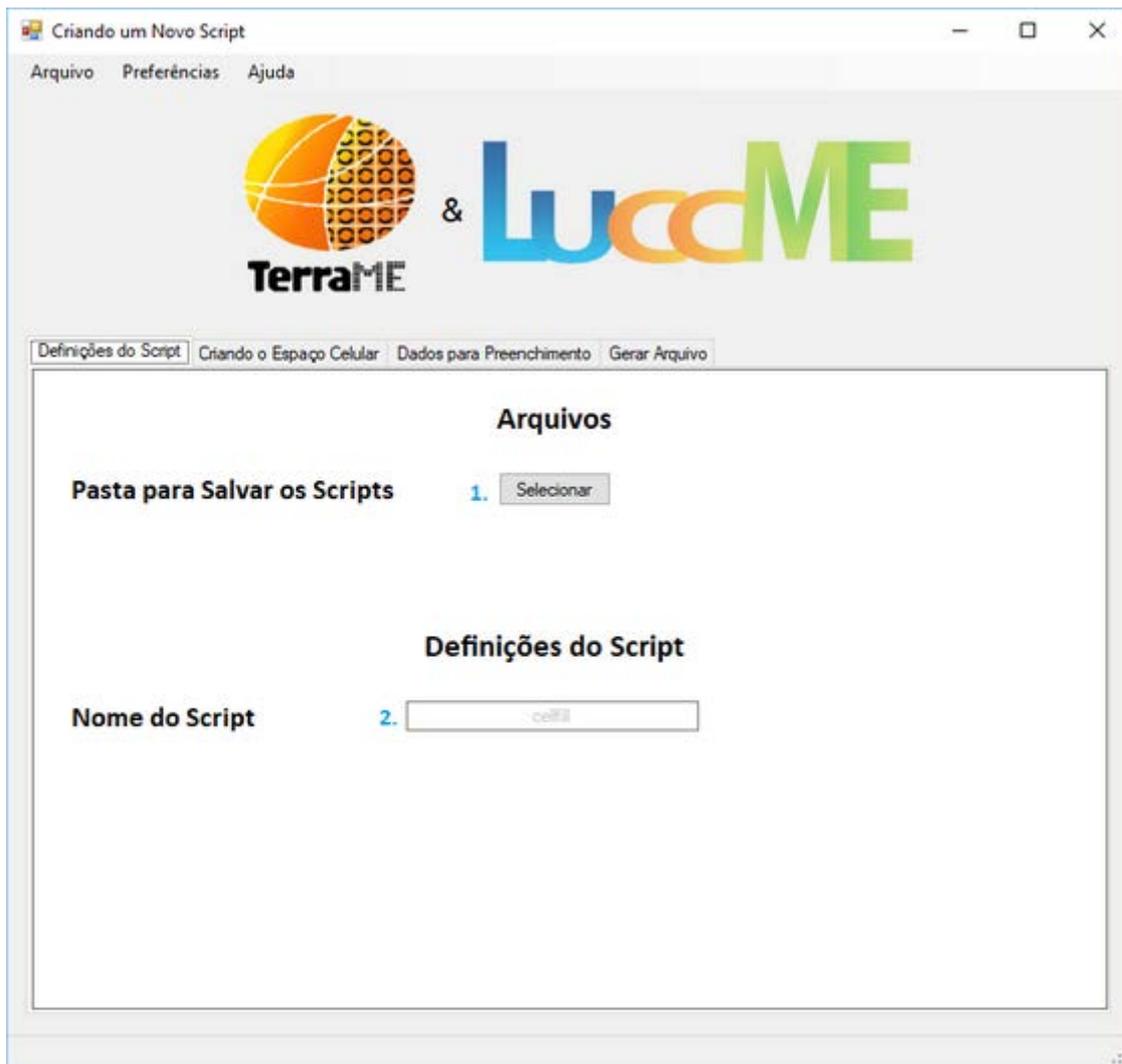


Figura 6 - Definições do script.

O próximo passo consiste na definição dos parâmetros para criação do espaço celular, conforme ilustrado na Figura 7. Neste momento cabe ao usuário indicar:

1. Arquivo que contém o limite do espaço celular (ex. limite da área de estudo). Este arquivo deve ser um vetor, do tipo polígono, no formato *shapefile*;
2. Nome do espaço celular a ser criado;
3. Resolução espacial das células (lado x lado);



Figura 7 - Criando um espaço celular.

Uma vez que os parâmetros do espaço celular tenham sido definidos, o usuário pode iniciar o seu preenchimento através no menu "atributos de preenchimento". Nesta seção o usuário deve informar quais dados serão usados para preencher as células (ex: usos da terra), conforme ilustrado na figura 8. Para adicionar os arquivos que contém os dados, o usuário deve clicar no botão "+" localizado no canto inferior esquerdo da janela. Após clicar neste botão a janela de seleção de arquivos aparecerá. Os formatos de arquivos suportados atualmente na interface são *shapefile* para dados vetoriais (linhas, pontos e polígonos) e GeoTIFF para dados matriciais.

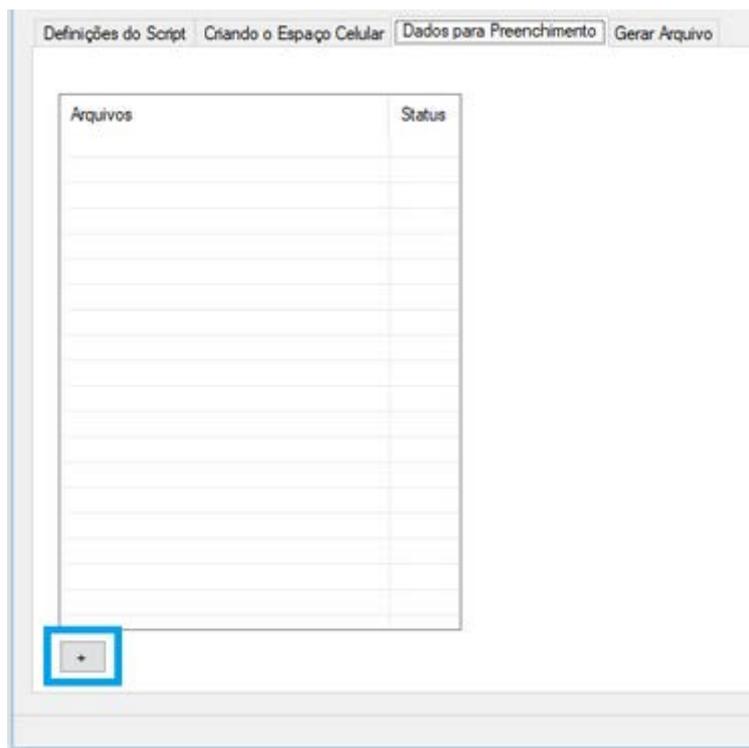


Figura 8 - Adicionando atributos de preenchimento.

Quando a seleção dos dados a serem preenchidos for concluída, os nomes dos arquivos (.tif e .shp) vão aparecer na coluna atributos, conforme ilustrado na Figura 9.

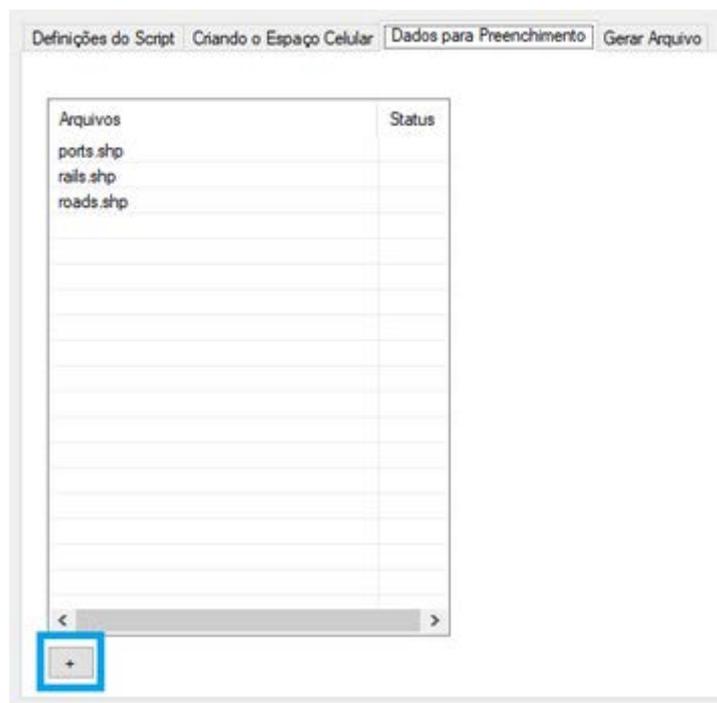


Figura 9 - Dados incluídos na coluna atributos.

Uma vez que um dado é adicionado à lista de atributos, o usuário pode selecionar a operação de preenchimento desejada, para que esta variável fique disponível no espaço celular. Para configurar uma operação de preenchimento, basta clicar no nome do arquivo presente na lista, então a janela entrará no modo de configuração, conforme ilustrado na Figura 10.

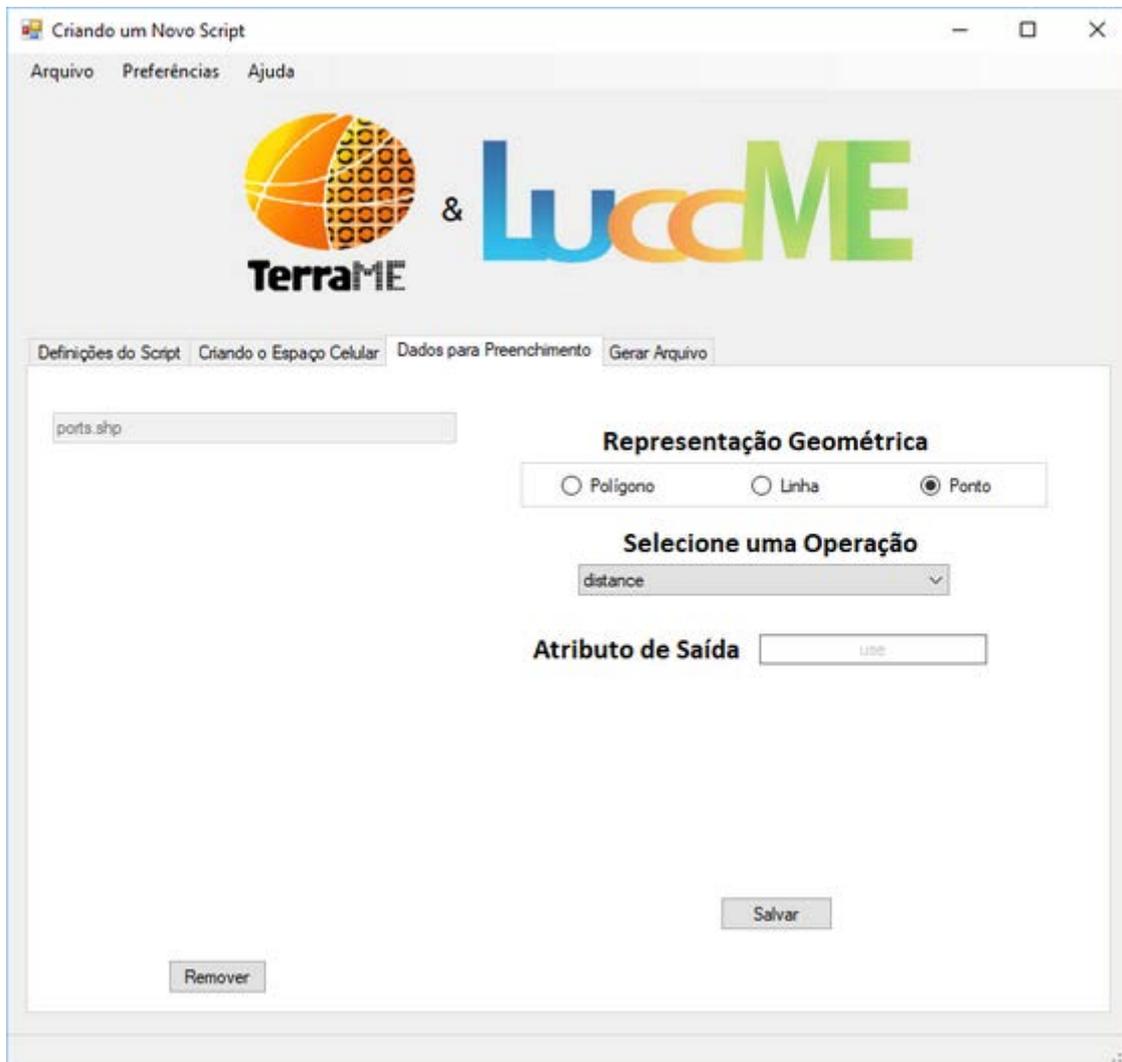


Figura 10 - Selecionando uma operação de preenchimento.

A lista de operadores disponíveis para preenchimento é gerada de acordo com o formato dos arquivos utilizados (.shp ou .tif). Quando a operação de preenchimento desejada for selecionada, os parâmetros de configuração da operação irão aparecer.

A Tabela 1 apresenta uma síntese dos principais operadores disponíveis para preenchimento de células via TerraME.

O exemplo de *script* apresentado a seguir pode ser utilizado para criação e preenchimento de um espaço celular diretamente no TerraME (gerado com a interface FillCell). Cada tipo de uso da terra, bem como, os diferentes fatores determinantes serão representados por uma coluna no espaço celular.

```

-----
--      This file contains a Fill Cell Script      --
--      Compatible with LuccME 3.1                --
--      Generated with Fill Cell Script Configurator --
--      19/06/2017 at 09:24:50                   --
-----

local x = os.clock()
import("terralib")

local projFile = File("t3mp.tview")
if(projFile:exists()) then
    projFile:delete()
end

-- CREATING PROJECT --
print("-- Creating Project --\n")

proj = Project {
    file = "t3mp.tview",
    clean = true
}

-- ADDING LAYERS --
print("-- Adding Layers to the Project --")

l1 = Layer{
    project = proj,
    name = "limit",
    file = "C:\\Treinamento\\preenchimentoDeCelulas\\limitePA_src29101_pol.shp"
}
print("Added Limit: limitePA_src29101_pol.shp")

l2 = Layer{
    project = proj,
    name = "layer2",
    file = "C:\\Treinamento\\preenchimentoDeCelulas\\rodoviasPA_src29101_lin.shp"
}
print("Added Layer2: rodoviasPA_src29101_lin.shp")

-- Checking EPSGs --
print("\n-- Checking EPSGs--")
local epsgVector = {l1.epsg, l2.epsg}
local fileVector = {"limitePA_src29101_pol.shp", "rodoviasPA_src29101_lin.shp"}
local checkEPSG = true

for i = 1, #epsgVector, 1 do
    if (epsgVector[i] ~= l1.epsg) then
        print("Error: EPSG does not math - limit : "..l1.epsg.." "..fileVector[i].." : "..epsgVector[i])
        checkEPSG = false
    end

    if checkEPSG then print("EPSG - limit : "..l1.epsg.."\\t"..fileVector[i].." : "..epsgVector[i]) end
end

if not checkEPSG then os.exit() end

-- CREATING CELLULAR SPACE --
print("\n-- Creating Cellular Space --\n")

local cs = Layer{
    project = proj,
    source = "shp",
    name = "cs",

```

```

input = 11.name,
resolution = 50000,
file = "csExemplo.shp",
clean = true
}

-- FILLING CELLULAR SPACE --
print("Filling rodoviasPA_src29101_lin.shp into Cellular Space using distance operation")
cs:fill{
    layer = "layer2",
    operation = "distance",
    attribute = "rod",
    dataType = "line"
}

projFile = File("t3mp.tview")
if(projFile.exists()) then
    projFile.delete()
end
print(string.format("\nElapsed time : %.2fs\n", os.clock() - x))
print("\nEnd of Script")

```

Tabela 1 - Síntese dos principais operadores e funções disponíveis para preenchimento de células via TerraME.

"area"	Área total de sobreposição entre a célula e um <i>layer</i> de polígonos. Os valores gerados podem variar de 0-1, indicando sua área de cobertura.
"average"	Média de valores quantitativos dos objetos que tem alguma intersecção com a célula, sem levar em consideração suas propriedades geométricas. Quando o argumento área é utilizado, ele calcula a média ponderada pela proporção das respectivas áreas de intersecção. Este operador costuma ser útil para distribuir atributos que representam médias como renda <i>percapita</i> .
"count"	Número de objetos que tem alguma sobreposição com a célula.
"distance"	Distância ao objeto mais próximo. A distância é calculada a partir do centroide da célula até o ponto, linha ou borda mais próxima de um polígono.
"mode"	Valor qualitativo mais comum dos objetos que tem alguma intersecção com a célula sem levar em consideração suas propriedades geométricas. Esta operação converte o resultado para <i>string</i> . Quando há dois ou mais valores com a mesma contagem, o valor resultante vai conter todos eles separados por vírgulas.
"maximum"	Valor quantitativo máximo entre os objetos que tem alguma intersecção com a célula sem levar em consideração suas propriedades geométricas.
"minimum"	Valor quantitativo mínimo entre os objetos que tem alguma intersecção com a célula sem levar em consideração suas propriedades geométricas.
"coverage"	Porcentagem de cada valor qualitativo cobrindo a célula, utilizando dados de polígonos ou matriz. Este operador cria um novo atributo para cada valor disponível, onde atributo é o valor passado como argumento para preenchimento e valor representa os diferentes valores nos dados de entrada. A soma dos valores do atributo criado para uma dada célula irá variar de 0-100 de acordo com a cobertura da célula. Quando utilizar arquivos <i>shapefile</i> , lembre-se do limite de 10 caracteres para nomes. Esta operação vai ser interrompida caso 2 nomes de saída tenham nomes iguais.

"presence"	Valores booleanos indicando se algum objeto tem sobreposição com a célula.
"stdev"	Desvio padrão dos objetos quantitativos que tem alguma sobreposição com a célula, sem levar em consideração suas propriedades geométricas.
"sum"	Soma dos valores quantitativos dos objetos que tem alguma sobreposição com a célula, sem levar em consideração suas propriedades geométricas. Quando o argumento área é utilizado, o operador calcula a soma com base na proporção da área de intersecção. Este operador costuma ser útil para preservar a soma total em ambos os <i>layers</i> , como, por exemplo, o tamanho da população.

Fonte: Adaptado de Andrade e Avancini, 2016 (<https://github.com/TerraME/terrame/wiki/Fill>).

Para modelos Contínuos, cada célula (linha) deve ser preenchida com o percentual ocupado por cada tipo de uso da terra. O somatório de todas as classes de uso/cobertura da terra deve totalizar 1 em todas as células. A área total ocupada por cada classe de uso da terra deve ser igual ao valor informado no primeiro passo de tempo no componente de demanda. Para modelos Discretos, cada célula deve ser preenchida com o tipo de uso da terra majoritário onde 1 deve indicar presença e 0 ausência de cada tipo de uso na célula, conforme ilustrado na Figura 4.

Os nomes utilizados para identificar os atributos (colunas com classes de uso/cobertura da terra e fatores determinantes) no preenchimento do espaço celular também devem ser utilizados para parametrizar o modelo desenvolvido a partir da plataforma LuccME. Caso o usuário tenha optado por utilizar alguma transformação das variáveis explicativas (ex.: logarítmica, radiciação, etc) durante as análises estatísticas, deve-se fazer o mesmo com as respectivas colunas do espaço celular, caso ainda não estejam transformadas. As variáveis dependentes (tipos de uso/cobertura da terra) só poderão ser transformadas com o Log10, o que pode ser indicado no LuccME. Para as demais variáveis (variáveis independentes) podem ser utilizadas qualquer transformação.

O terceiro passo visa, fundamentalmente, apoiar a parametrização do componente de Potencial, mas também dos componentes de Demanda e Alocação com base na análise do ritmo e aglutinação das mudanças no tempo e espaço. Esta etapa é destinada a análise visual dos dados de uso da terra (extensão e diferença), análise da quantidade de mudança e análise visual dos fatores determinantes (relação qualitativa com os usos). O terceiro passo também inclui a análise exploratória dos dados: correlações, histogramas e transformações necessárias visando estabelecer relações potenciais a partir de diferentes métodos (regressão linear, logística,

espacial). Estas análises são feitas fora do LuccME, normalmente em softwares como R, GeoDA, SPSS entre outros.

Com base na aplicação e análise das diferentes classes de uso/cobertura da terra e fatores determinantes, o usuário pode iniciar o quarto passo, que consiste na parametrização do modelo LuccME. Primeiro, o usuário deve selecionar os componentes de *Potencial*, *Demanda* e *Alocação* mais adequados para a aplicação. Caso necessário, o usuário pode ainda implementar novos componentes. Os componentes do LuccME (descritos nas próximas seções) seguem uma abordagem estrutural (Verburg et al., 2004), onde os fatores determinantes espaciais são relacionados a diferentes padrões de uso da terra, com base num único mapa. Ou seja, os modelos de uso da terra construídos no LuccME não trabalharam com as transições observadas entre classes de uso (ex.: floresta para pastagem). Em outras palavras, a variável de uso da terra não é a mudança que ocorreu entre duas classes de usos/cobertura num determinado período, mas o padrão histórico acumulado. Além disso, os componentes do LuccME são organizados em dois grupos (diretórios) que não podem ser combinados: modelos i) *Contínuos* e ii) *Discretos*. Na prática, algoritmos *Contínuos* devem ser utilizados quando as variáveis de uso da terra são fornecidas como percentual de cada célula no espaço celular (Figura 4). Por outro lado, algoritmos *Discretos* devem ser utilizados quando os tipos de uso da terra são fornecidos como variáveis categóricas, representando a presença de cada classe numa célula. Em cada um destes grupos (*Contínuo* e *Discreto*), todos os componentes podem ser indistintamente utilizados. Novos componentes implementados por usuários LuccME devem respeitar esta regra. Para maiores informações sobre a criação de novos componentes, por favor, verificar o *Guia LuccME Criando Componentes*, disponível na página do LuccME.

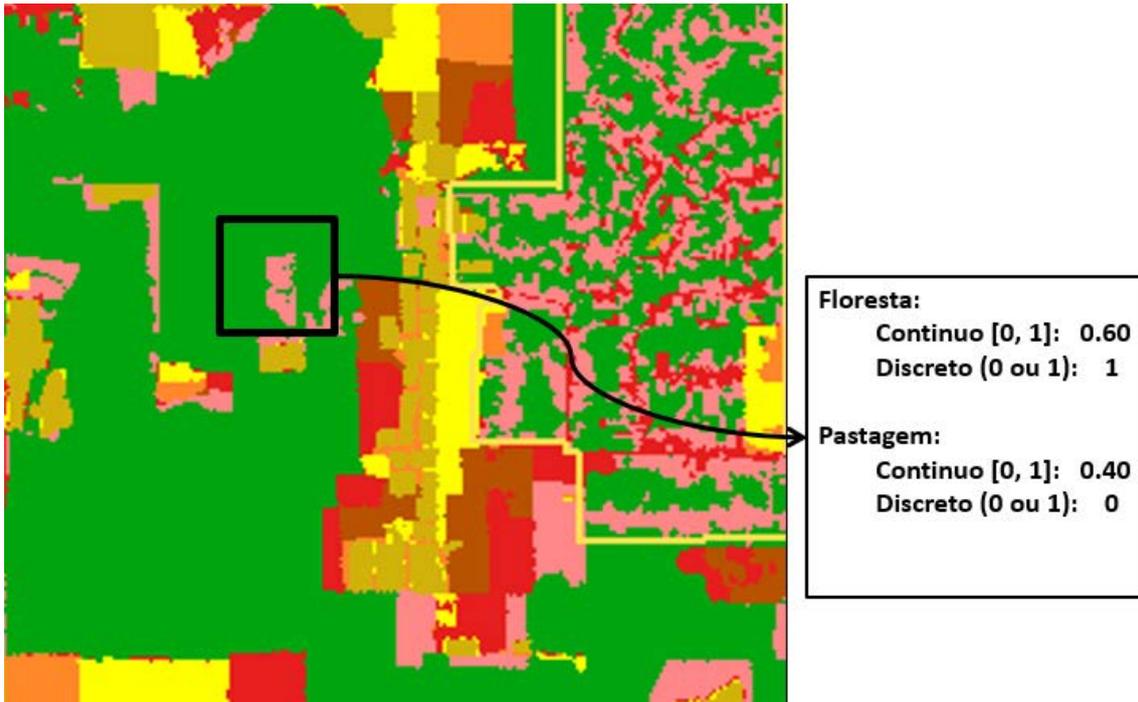


Figura 11 - Diferença entre modelos com dados de uso da terra “contínuos” e “discretos”.

Novos usuários são convidados a explorar o LuccME combinando diferentes componentes de *Potencial*, *Alocação* e *Demanda* já implementados (Figura 5); desta forma, o usuário tem apenas que parametrizar um modelo inicial a partir da interface gráfica levando em consideração as orientações apresentadas seção 3.2.

	Demanda	Potencial	Alocação
Discretos Variável uso da terra: Categórica	Pre Computed Values Compute Two Dates Compute Three Dates External Model	Neigh Simple Rule Inverse Distance Rule Neigh Inverse Distance Rule Logistic Regression Logistic Regression Neigh Attraction Logistic Regression Attract Repulsion Sample Based	Simple Ordering ClueS Like ClueS Neigh Ordering
Contínuos Variável de uso da terra: % de cada classe		Linear Regression Spatial Lag Regression Spatial Lag Linear Regression Mix Sample Based	Clue Like Clue Like Saturation

Figura 12 - Componentes do LuccME.

O quinto e último passo para construção de um modelo no LuccME consiste em rodar o modelo após sua parametrização e, sobretudo, na calibração e validação no modelo proposto. Esta etapa será descrita em detalhes na última seção deste capítulo.

3.2 Configurando os parâmetros do Modelo

Com a criação da interface gráfica do LuccME, o processo de criação de um novo modelo tornou-se simplificado, bastando ao usuário apenas configurar os parâmetros do modelo. Nas próximas sessões serão explicados todos os parâmetros de um modelo LuccME 3.1, usando a estrutura: **nome na interface** (*nome no código Lua*): *explicação*. Aos usuários avançados é recomendado que crie uma primeira versão na interface gráfica e posteriormente modifique os arquivos gerados.

3.2.1 Definições do Modelo

Os parâmetros relacionados as definições do modelo são:

Pasta para Salvar o Modelo (*dofile*): Local em que os arquivos gerados serão salvos.

Nome do Modelo (*name*): Nome para o modelo LuccME.

Ano de Início (*startTime*): Ano que o modelo inicia.

Ano de Término (*endTime*): Último ano de simulação.

3.2.2 Definições Espaciais

Os parâmetros relacionados a definições espaciais são:

Arquivo (*file*): Informa o nome do arquivo que contém o espaço celular, seja ele um arquivo de projeto (*Terraview*) ou um arquivo vetorial (*Shape*).

Nome da Camada (*layer*): Nome da *Camada* onde o espaço celular está localizado no projeto TerraView (*não utilizado usando Shape*).

Tamanho da Célula (*cellArea*): A resolução espacial da célula (tamanho da célula).

3.2.3 Tipos de Uso da Terra

Os parâmetros relacionados aos tipos de uso da terra são:

Tipos de Uso da Terra (*landUseTypes*): Nome de cada tipo de uso da terra conforme definido no banco de dados.

Tipo de Uso com "No Data" (*landUseNoData*): Tipo de uso da terra sobre o qual será acrescentado eventuais *no data* gerados pelo sistema, como efeito de borda, por exemplo (normalmente tipo de uso estático).

3.2.4 Componentes

O usuário deverá escolher os componentes que se adequam a sua aplicação/modelo.

A explicação de cada componente e seus parâmetros se encontra na sessão 3.3.

3.2.5 Salvar Parâmetros

Os parâmetros relacionados as saídas do modelo, o que será salvo, são:

Nome de Saída (*outputTheme*): Caso seja utilizado um projeto Terraview, o nome utilizado será o nome do *layer* de saída no projeto, se for um shape, será o nome do arquivo de saída.

Salvar Anualmente (*yearly*): Se selecionado na interface (*true* no código) será salvo todos os anos da simulação.

Anos a serem Salvos (*saveYears*): Informa os anos de saída a serem salvos.

Atributos a serem Salvos (*saveAttrs*): Informa os tipos de uso da terra a serem salvos. Para cada tipo de uso 3 saídas serão gravadas:

_out: Padrão de alocação para cada tipo de uso da terra.

_chtot: Mudança entre o tempo inicial e tempo final de simulação.

_chpast: Mudança entre ano salvo e o ano anterior de simulação.

_pot: Superfície do potencial de mudança.

3.3 Configurando os Componentes

Outro passo para construir um modelo espacial no LuccME 3.1 é configurar os Componentes utilizados na aplicação/modelo (ver a pasta Labs no pacote LuccME para um exemplo). Os componentes de *Potencial*, *Demanda* e *Alocação* são definidos e parametrizados pelo usuário na sessão Componentes da interface e serão gravados no arquivo de submodelo. Os algoritmos e parâmetros disponíveis para cada componente no LuccME 3.1 são descritos da seção 3.3.1 até 3.3.3.

3.3.1 Componente de Potencial

Este componente indica a adequação ou potencial de mudança de cada célula para um dado tipo de uso da terra. Tal potencial é baseado em fatores determinantes espaciais (e.g. distância a estradas, distância a centros urbanos, presença de áreas protegidas), e/ou características de vizinhança indicados pelo usuário como responsáveis pela localização das mudanças no uso da terra. A seleção de fatores determinantes, bem como, a definição de seus pesos relativos é normalmente baseada em análises empíricas, as quais devem ser consistentes com a abordagem de modelagem selecionada no LuccME. Por exemplo, se o usuário seleciona um algoritmo baseado em regressão linear para calcular o potencial de mudança de cada célula, então, a seleção de fatores determinantes e definição dos pesos relativos deve ser feita com base no mesmo tipo de abordagem. Conforme descrito anteriormente, o processo de seleção de fatores determinantes e seus coeficientes são entradas para o componente de Potencial, portanto, definidos fora do LuccME, no terceiro passo do processo de modelagem. Os algoritmos de Potencial disponíveis no LuccME 3.1 para modelos *Contínuos* e *Discretos* são sintetizados abaixo.

3.3.1.1 Componentes de Potencial Discretos

Os algoritmos de Potencial disponíveis no LuccME 3.1 para modelos *Discretos* são:

PotentialDNeighSimpleRule: Modelo simples desenvolvido como material para treinamento e iniciação no LuccME. Não deve ser utilizado em aplicações reais. Estima o potencial de mudança de cada célula para um dado tipo de uso da terra de acordo com o percentual de células do mesmo tipo numa vizinhança de Moore;

Parâmetros específicos de alocação não estão disponíveis para este algoritmo.

PotentialDInverseDistanceRule: Modelo simples desenvolvido como material para treinamento e iniciação no LuccME. Não deve ser utilizado em aplicações reais. Estima o potencial de mudança de cada célula para um dado tipo de uso da terra de acordo com a soma do inverso do valor de um determinado atributo, ponderado por um peso (normalmente um atributo de distância).

Para cada tipo de usos da terra:

Peso (*const*): Informa a importância dos múltiplos (de 0 a 1).

Atributo (*betas*): Informa o nome do atributo e seu peso.

PotentialIDNeighInverseDistanceRule: Modelo simples desenvolvido como material para treinamento e iniciação no LuccME. Não deve ser utilizado em aplicações reais. Estima o potencial de mudança de cada célula combinando os dois métodos anteriores.

Para cada tipo de usos da terra:

Peso (*const*): Informa a importância dos múltiplos (de 0 a 1).

Atributo (*betas*): Informa o nome do atributo e seu peso.

PotentialIDLogisticRegression: Utiliza técnicas de regressão logística para calcular o potencial de mudança da célula para cada tipo de uso da terra. Baseado na abordagem original para cálculo de potencial do modelo CLUE-S (Verburg et al., 2002). Para cada uso, define-se também uma elasticidade, que indicará quão difícil é retirar aquele uso da célula (varia de 0-1, sendo 1 muito fácil e 0 muito difícil). Esta elasticidade é somada à probabilidade estimada pela regressão logística para o uso atual da célula (área como uma "âncora" – quanto maior, mais difícil de ocorrer uma mudança para outro tipo de uso).

Para cada tipo de usos da terra:

Const (*const*): Informa a constante da regressão logística.

Elasticity (*elasticity*): Informa a parâmetro que deve indicar quão difícil é retirar um uso de determinada célula variando de [0-1], sendo 1, muito fácil, e 0 muito difícil.

Betas (*betas*): Informa o nome e peso (*betas*) de cada variável independente estimado por regressão logística.

PotentialIDLogisticRegressionNeighAttract: Modificação do componente *LogisticRegression* combinando ideias de modelos baseados em autômato celular. O potencial da célula é modificado de acordo com a atratividade da mesma classe numa dada vizinhança (genérica). A vizinhança pode ser construída externamente e carregada através de um arquivo (*gal*). Caso não seja informada, uma vizinhança de Moore é construída. O potencial da célula é alterado (aumentado) a partir de uma determinada quantidade de vizinhos informado pelo parâmetro *percNeighborsUse*.

filename: Informa o arquivo de vizinhança no formato (gal). Este é um parâmetro opcional, caso não seja informado uma vizinhança Moore será construída. *(Não disponível na interface)*

Para cada tipo de usos da terra:

Const (*const*): Informa a constante da regressão logística.

Elasticity (*elasticity*): Informa a parâmetro que deve indicar quão difícil é retirar um uso de determinada célula variando de [0-1], sendo 1, muito fácil, e 0 muito difícil.

percNeighborsUse (*percNeighborsUse*): Informa a partir de qual percentual de vizinhos [0-1] será alterado o potencial.

Betas (*betas*): Informa o nome e peso (betas) de cada variável independente estimado por regressão logística.

PotentialDLogisticRegressionNeighAttractRepulsion: Modificação do componente *LogisticRegression* combinando ideias de modelos baseados em autômato celular. O potencial da célula é modificado de acordo com a matriz de atratividade numa dada vizinhança (genérica). A vizinhança pode ser construída externamente e carregada através de um arquivo (gal). Caso não seja informada, uma vizinhança de Moore é construída. O potencial da célula é alterado (aumentado) a partir de uma determinada quantidade de vizinhos informado pelo parâmetro *percNeighborsUse*.

filename: Informa o arquivo de vizinhança no formato (gal). Este é um parâmetro opcional, caso não seja informado uma vizinhança Moore será construída. *(Não disponível na interface)*

Para cada tipo de usos da terra:

Const (*const*): Informa a constante da regressão logística.

Elasticity (*elasticity*): Informa a parâmetro que deve indicar quão difícil é retirar um uso de determinada célula variando de [0-1], sendo 1, muito fácil, e 0 muito difícil.

percNeighborsUse (*percNeighborsUse*): Informa a partir de qual percentual de vizinhos [0-1] será alterado o potencial.

Betas (*betas*): Informa o nome e peso (betas) de cada variável independente estimado por regressão logística.

Matrix de Afinidade (*affinityMatrix*): Informa se os usos se atraem (valores entre 0.1 a 1, sendo 1 atração máxima), se repelem (valores entre -0.1 a -1, sendo -1 repulsão máxima) ou se é indiferente (valores iguais a 0).

PotentialDSampleBased: Componente baseado nos conceitos de máxima entropia. O algoritmo utiliza as variáveis informadas no modelo para selecionar um conjunto de amostras que representa cada uso. Com os dados das amostras, busca no espaço celular as células com potencial para aquele uso, fazendo uma série de ponderações para determinar o valor do potencial da célula. Para todas as variáveis informadas quanto mais perto for o valor de cada parâmetro da média das amostras, maior será o potencial da célula.

Para cada tipo de usos da terra:

Porcentagem para considerar um uso (*cellUsePercentage*): Este valor para componentes discretos será sempre 100 (valor em porcentagem).

Atributos de Intervalo (*attributesPerc*): Informa os atributos de intervalo (range), tais como temperatura, distância, etc.

Atributos Categóricos (*attributesClass*): Informa os atributos categóricos, tais como presença ou não, classificações, etc.

3.3.1.2 Componentes de Potencial Contínuos

Os componentes de Potencial disponíveis no LuccME 3.1 para modelos *Contínuos* (*LinearRegression*, *SpatialLagRegression*, *SpatialLagLinearRoads*, *SampleBased*) são descritos abaixo:

PotentialCLinearRegression: Utiliza técnicas de regressão linear para estimar o potencial de mudança de cada célula. Baseado na proposta original do modelo CLUE contínuo, o potencial de mudança para cada tipo de uso (aumento ou redução no percentual de cada célula num dado período de tempo) é estimado com base na diferença entre a superfície de regressão e o percentual atual do uso da terra num dado período de tempo (veja Verburg et al., 1999 para uma descrição detalhada). As mudanças serão distribuídas posteriormente pelo componente de alocação proporcionalmente ao potencial de cada célula (de acordo com a demanda). Uma diferença do modelo CLUE original é permitir que as variáveis de uso sejam log10

transformadas. Para cada uso da terra no modelo, em cada região (ver item...), o usuário deve fornecer os dados do modelo de regressão.

Para cada tipo de usos da terra:

isLog (*isLog*): Informa se o tipo de uso da terra considerado é Log10 transformado.

Const (*const*): Informa a constante da regressão.

Betas (*betas*): Informa o nome e peso (betas) de cada variável independente. O nome das variáveis independentes deve ser exatamente o mesmo do banco de dados.

PotentialCSpatialLagRegression: Semelhante a abordagem *LinearRegression*, mas se baseia em técnicas de regressão espacial (Anselin, 2002) para calcular a superfície de regressão. Este componente deve ser utilizado (em casos em que os usos da terra apresentem alta dependência espacial). Como no componente *LinearRegression*, as mudanças serão distribuídas posteriormente pelo componente de alocação proporcionalmente ao potencial de cada célula (de acordo com a demanda). Porém, para evitar o espalhamento em células com baixo potencial (que posteriormente são potencializadas pelos efeitos da vizinhança), foram introduzidos os controle *minReg* e *maxReg* descritos abaixo.

Para cada tipo de usos da terra:

isLog (*isLog*): Informa se o tipo de uso da terra considerado é Log transformado.

Const (*const*): Informa a constante da regressão.

minReg (*minReg*): Informa o valor mínimo da regressão (0-1) que deve ser considerado para evitar espalhamento de um determinado uso em células com baixo potencial. Deve ser usado para classes de uso unidirecionais e que tenham sempre demanda por aumento de um ano para outro (por exemplo, desmatamento). Caso não seja adequado, utilizar valor 0.

maxReg (*maxReg*): Informa o valor máximo da regressão (0-1) que deve ser considerado para evitar o decréscimo do uso em células com alto potencial. Deve ser usado para classes de uso unidirecionais e que tenham sempre demanda por diminuição de um ano para outro (por exemplo, floresta primária). Caso não seja adequado, utilizar valor 1.

ro (*ro*): Informa o coeficiente auto regressivo da regressão espacial (0-1).

Betas (*betas*): Informa o nome e peso (betas) de cada variável independente. O nome das variáveis independentes deve ser exatamente o mesmo do banco de dados.

PotentialCSpatialLagLinearRegressionMix: Modificação do componente *SpatialLagRegression* para permitir o uso de regressão linear simples em casos específicos, quando estradas são criadas ou pavimentadas. Este componente é um exemplo de como a plataforma pode ser estendida para aplicações específicas. Foi desenvolvido poder possibilitar o surgimento de novas fronteiras de desmatamento na Amazônia Brasileira, quando novas estradas são criadas, por exemplo (pois o método de regressão espacial tende a favorecer os locais onde já existe concentração de desmatamento).

Para cada tipo de usos da terra:

isLog (*isLog*): Informa se o tipo de uso da terra considerado é Log transformado.

Const (*const*): Informa a constante da regressão espacial.

minReg (*minReg*): Informa o valor mínimo da regressão (0-1) que deve ser considerado para evitar espalhamento de um determinado uso em células com baixo potencial. Deve ser usado para classes de uso unidirecionais e que tenham sempre demanda por aumento de um ano para outro. (por exemplo, desmatamento). Caso não seja adequado, utilizar valor 0.

maxReg (*maxReg*): Informa o valor máximo da regressão (0-1) que deve ser considerado para evitar o decréscimo do uso em células com alto potencial. Deve ser usado para classes de uso unidirecionais e que tenham sempre demanda por diminuição de um ano para outro (por exemplo, floresta primária). Caso não seja adequado, utilizar valor 1.

ro (*ro*): Informa o coeficiente auto regressivo da regressão espacial (0-1).

Betas (*betas*): Informa o nome e peso (betas) de cada variável independente. O nome das variáveis independentes deve ser exatamente o mesmo do banco de dados.

Roads Model (*roadsModel*): Informa os dados de regressão, tendo em conta as novas estradas construídas ou pavimentadas.

Roads Model Attributes (*attrs*): Informa o nome das variáveis relacionadas a novas estradas.

Const (*const*): Informa a constante da regressão linear.

Change (*change*): Informa se o valor do atributo da célula (relacionado a estradas) deve mudar a partir de um determinado nível (e.g. -1.5)

Betas (*betas*): Informa o nome e peso (betas) de cada variável independente. O nome das variáveis independentes deve ser exatamente o mesmo do banco de dados.

PotentialCSampleBased: Componente baseado nos conceitos de máxima entropia. O algoritmo utiliza as variáveis informadas no modelo para selecionar um conjunto de amostras que representa cada uso. Com os dados das amostras, busca no espaço celular as células com potencial para aquele uso, fazendo uma série de ponderações para determinar o valor do potencial da célula. Para todas as variáveis informadas quanto mais perto for o valor de cada parâmetro da média das amostras, maior será o potencial da célula.

Para cada tipo de usos da terra:

Porcentagem para considerar um uso (*cellUsePercentage*): Informa a porcentagem do uso presente na célula para que esta seja considerada uma amostra (0-100).

Atributos de Intervalo (*attributesPerc*): Informa os atributos de intervalo (range), tais como temperatura, distância, etc.

Atributos Categóricos (*attributesClass*): Informa os atributos categóricos, tais como presença ou não, classificações, etc.

3.3.2 Componente de Alocação

Os componentes de alocação fazem a distribuição espacial das mudanças de uso e cobertura da terra com base na demanda (seção 3.3.3) e potencial de mudança de cada célula (seção 3.3.1). Diversas regras de alocação também podem ser definidas no componente de alocação (e.g. quantidade e velocidade de mudança permitida em cada passo de tempo), mas sua parametrização depende das necessidades do usuário. Os algoritmos e parâmetros de alocação disponíveis no LuccME 2.1 para modelos *Contínuos* e *Discretos* são apresentados abaixo.

3.3.2.1 Componentes de Alocação Discretos

Os algoritmos de alocação disponíveis no LuccME 3.1 para modelos *Discretos* são:

AllocationDClueSLike: Baseado no processo de competição entre classes de uso da terra dentro de uma mesma célula, ajustado de modo iterativo para satisfazer a demanda quando

todas as células são consideradas, conforme descrito em Verburg et al., (2002), mas estendido para incorporar novas características, tais como mudanças em bloco (opcional, parametrizado por célula).

maxIteration (*maxIteration*): Informa o número máximo de interações permitidas para cada passo de tempo do modelo (deve ser número inteiro > 0; valor recomendado = 1000). Caso este número de iterações seja alcançado sem que o modelo obtenha sucesso na alocação da demanda informada, então, o modelo interrompe sua execução e retorna uma mensagem de erro informando que não conseguiu alocar a demanda.

factorIteration (*factorIteration*): Valor inicial do parâmetro que controla o fator de iteração da alocação.

maxDifference (*maxDifference*): Informa a diferença máxima entre o valor de demanda informado pelo usuário e a quantidade alocada pelo modelo para cada tipo uso (deve ser informado na mesma unidade de área da célula);

Transition Matrix (*transitionMatrix*): Informa as transições permitidas (1) e não permitidas (0) numa matriz uso da terra vs uso da terra (deve ter pelo menos uma região).

AllocationDSimpleOrdering: Modelo simples desenvolvido como material de aula. Não deve ser utilizado em aplicações reais. Ao invés de utilizar o processo iterativo empregado no modelo CLUE, este componente utiliza uma abordagem de ordenamento simples.

maxDifference (*maxDifference*): Informa a diferença máxima entre o valor de demanda informado pelo usuário e a quantidade alocada pelo modelo para cada tipo uso (deve ser informado na mesma unidade de área da célula);

AllocationDClueSNeighOrdering: Este componente é uma modificação do componente *AllocationClueSLike*, onde, antes da execução da alocação o potencial de cada uso é ordenado de acordo com a vizinhança da cada célula.

maxIteration (*maxIteration*): Informa o número máximo de interações permitidas para cada passo de tempo do modelo (deve ser número inteiro > 0; valor recomendado = 1000). Caso este número de iterações seja alcançado sem que o modelo obtenha sucesso na alocação da demanda informada, então, o modelo interrompe sua execução e retorna uma mensagem de erro informando que não conseguiu alocar a demanda.

factorIteration (*factorIteration*): Valor inicial do parâmetro que controla o fator de iteração da alocação.

maxDifference (*maxDifference*): Informa a diferença máxima entre o valor de demanda informado pelo usuário e a quantidade alocada pelo modelo para cada tipo uso (deve ser informado na mesma unidade de área da célula);

Transition Matrix (*transitionMatrix*): Informa as transições permitidas (1) e não permitidas (0) numa matriz uso da terra vs uso da terra (deve ter pelo menos uma região).

3.3.2.2 Componentes de Alocação Contínuos

Os algoritmos de Alocação disponíveis no LuccME 3.1 para modelos *Contínuos* são:

AllocationCCLueLike: Baseado no processo de competição entre classes de uso da terra dentro de uma mesma célula, ajustado de modo iterativo para satisfazer a demanda quando todas as células são consideradas, conforme descrito em Verburg et al., (1999). As células recebem um percentual da mudança que deve ser alocada em cada passo de tempo para toda a área, proporcionalmente ao seu potencial. A versão do INPE difere do modelo original CLUE, pois é adaptado para o contexto brasileiro, por exemplo, para representar o cumprimento do Código Florestal (Aguiar, 2006). Existem ainda parâmetros para controlar a quantidade e velocidade de mudança que pode ocorrer em cada célula.

maxDifference (*maxDifference*): Informa a diferença máxima entre o valor de demanda informado pelo usuário e a quantidade alocada pelo modelo para cada tipo uso (deve ser informado na mesma unidade de área da célula);

maxIteration (*maxIteration*): Informa o número máximo de interações permitidas para cada passo de tempo do modelo (deve ser número inteiro > 0; valor recomendado = 1000). Caso este número de iterações seja alcançado sem que o modelo obtenha sucesso na alocação da demanda informada, então, o modelo interrompe sua execução e retorna uma mensagem de erro informando que não conseguiu alocar a demanda.

initialElasticity (*initialElasticity*): Informa o valor inicial de elasticidade (*iterationFactor*) (flutuante > 0; valor recomendado = 0.1).

minElasticity (*minElasticity*): Informa o valor mínimo de elasticidade (*iterationFactor*) (flutuante > 0; valor recomendado = 0.001).

maxElasticity (*maxElasticity*): Informa o valor máximo de elasticidade (iterationFactor) (flutuante > 1; valor recomendado = 1.5).

complementarLU (*complementarLU*): Informa o tipo de uso da terra que será recalculado no final caso a soma das percentagens de uso não esteja somando exatamente 100% (null ou um nome de uso da terra válido).

Allocation Data (*allocationData*): Parâmetros para definir regras de alocação específicas para cada tipo de uso da terra.

static (*static*): Informa se a variável pode aumentar ou diminuir em cada célula ou apenas mudar na direção da demanda. Possíveis valores: -1 (mudanças apenas na direção da demanda); 0 (mudanças em qualquer direção); 1 (estático);

minValue (*minValue*): Informa o valor percentual mínimo de um dado tipo de uso da terra permitido na célula (como resultado das novas mudanças – o percentual original pode estar fora do limite) (flutuante [0,1]). Normalmente definido de acordo com regras de uso da terra tais como Código Florestal. Note que as restrições devem ser consistentes com os diferentes tipos de uso da terra, pois o somatório deve ser 100% (por exemplo, o usuário não deve informar que todos os tipos de uso da terra têm um limite mínimo de 50%, caso ele tenha 3 ou mais tipos de uso).

maxValue (*maxValue*): Informa o valor percentual máximo permitido para um dado tipo de uso da terra na célula (como resultado das novas mudanças – o percentual original pode estar fora do limite) (flutuante [0,1]).

minChange (*minChange*): Informa a mudança mínima permitida para um dado tipo de uso da terra na célula (como resultado das novas mudanças – o percentual original pode estar fora do limite) (flutuante [0,1]).

maxChange (*maxChange*): Informa a mudança máxima permitida para um dado tipo de uso da terra na célula em 1 passo de tempo até o limite de saturação (flutuante [0,1]).

changeLimiarValue (*changeLimiarValue*): Informa o limiar de mudança para um dado tipo de uso da terra na célula. A partir deste limiar, a velocidade de mudança de um dado tipo de uso pode ser modificada (flutuante [0,1]). Deve ser calibrado de acordo com a quantidade média de mudança registrada de um ano para outro em dados observados.

maxChangeAboveLimiar (*maxChangeAboveLimiar*): Informa a mudança máxima permitida para um dado tipo de uso da terra na célula após o limiar de saturação (flutuante [0,1]).

AllocationCCLueLikeSaturation: Modificação do componente *AllocationClueLike*. Neste caso, a velocidade de mudança em cada célula utiliza uma variável espaço-temporal, atualizada dinamicamente cada ano indicando se a célula está numa área consolidada ou área de fronteira. O limiar de saturação considera uma vizinhança de 10x10, não considerando área protegidas.

maxDifference (*maxDifference*): Informa a diferença máxima entre o valor de demanda informado pelo usuário e a quantidade alocada pelo modelo para cada tipo uso (deve ser informado na mesma unidade de área da célula;

maxIteration (*maxIteration*): Informa o número máximo de interações permitidas para cada passo de tempo do modelo (deve ser número inteiro > 0; valor recomendado = 1000). Caso este número de iterações seja alcançado sem que o modelo obtenha sucesso na alocação da demanda informada, então, o modelo interrompe sua execução e retorna uma mensagem de erro informando que não conseguiu alocar a demanda.

initialElasticity (*initialElasticity*): Informa o valor inicial de elasticidade (iterationFactor) (flutuante > 0; valor recomendado = 0.1).

minElasticity (*minElasticity*): Informa o valor mínimo de elasticidade (iterationFactor) (flutuante > 0; valor recomendado = 0.001).

maxElasticity (*maxElasticity*): Informa o valor máximo de elasticidade (iterationFactor) (flutuante > 1; valor recomendado = 1.5).

complementarLU (*complementarLU*): Informa o tipo de uso da terra que será recalculado no final caso a soma das percentagens de uso não esteja somando exatamente 100% (null ou um nome de uso da terra válido).

saturationIndicator (*saturationIndicator*): Informa o nome de um atributo que será dinamicamente atualizado (pode ser salvo para fins de calibração).

attrProtection (*attrProtection*): Informa o atributo do banco de dados indicando o percentual de áreas protegidas a ser excluído a partir do nível de saturação).

Allocation Data (*allocationData*): Parâmetros para definir regras de alocação específicas para cada tipo de uso da terra.

static (*static*): Informa se a variável pode aumentar ou diminuir em cada célula ou apenas mudar na direção da demanda. Possíveis valores: -1 (mudanças

apenas na direção da demanda); 0 (mudanças em qualquer direção); 1 (estático);

minValue (*minValue*): Informa o valor percentual mínimo de um dado tipo de uso da terra permitido na célula (como resultado das novas mudanças – o percentual original pode estar fora do limite) (flutuante [0,1]). Normalmente definido de acordo com regras de uso da terra tais como Código Florestal. Note que as restrições devem ser consistentes com os diferentes tipos de uso da terra, pois o somatório deve ser 100% (por exemplo, o usuário não deve informar que todos os tipos de uso da terra têm um limite mínimo de 50%, caso ele tenha 3 ou mais tipos de uso).

maxValue (*maxValue*): Informa o valor percentual máximo permitido para um dado tipo de uso da terra na célula (como resultado das novas mudanças – o percentual original pode estar fora do limite) (flutuante [0,1]).

minChange (*minChange*): Informa a mudança mínima permitida para um dado tipo de uso da terra na célula em 1 passo de tempo até o limite de saturação (flutuante [0,1]).

maxChange (*maxChange*): Informa a mudança máxima permitida para um dado tipo de uso da terra na célula em 1 passo de tempo até o limite de saturação (flutuante [0,1]).

changeLimiarValue (*changeLimiarValue*): Informa o limiar de mudança para um dado tipo de uso da terra na célula. A partir deste limiar, a velocidade de mudança de um dado tipo de uso pode ser modificada (flutuante [0,1]). Deve ser calibrado de acordo com a quantidade média de mudança registrada de um ano para outro em dados observados.

maxChangeAboveLimiar (*maxChangeAboveLimiar*): Informa a mudança máxima permitida para um dado tipo de uso da terra na célula após o limiar de saturação (flutuante [0,1]).

3.3.3 Componente de Demanda

Este componente define a quantidade ou magnitude das mudanças de uso da terra a serem alocadas em cada passo de tempo. Esta informação é frequentemente uma entrada dos modelos de uso da terra construídos no LuccME, especialmente quando aplicados em estudos de cenários. Diversos métodos podem ser utilizados para fazer tais estimativas, tais como projeções a partir de linhas de base, premissas de cenários, ou valores de demanda derivados

de modelos econômicos. Estimativas de demanda também podem ser geradas internamente quando dois ou mais períodos de uso da terra passados são fornecidos no banco de dados (seção 3.2.3.1). Todos os algoritmos e parâmetros de demanda disponíveis no LuccME 3.1 são sintetizados abaixo e podem ser utilizados tanto para modelos *Contínuos* como para modelos *Discretos*.

DemandPreComputedValues: Utiliza valores de demanda predefinidos pelo usuário para cada tipo de uso da terra em cada passo de tempo.

Demanda Anual (*annualDemand*): Informa a demanda anual por terra para cada tipo de uso (deve ser informado na mesma unidade de área da célula), na mesma ordem informada no *Tipos de Uso*.

DemandComputeTwoDates: Os valores de demanda são calculados internamente com base em 2 períodos diferentes de dados de uso da terra do passado fornecidos pelo usuário no banco de dados.

Ano Final para Interpolação (*finalYearForInterpolation*): O ano para ser considerado o final da interpolação para o cálculo da demanda.

Colunas para Interpolação (*finalLandUseTypesForInterpolation*): Informa o nome da coluna com os valores de demanda para o final da interpolação de cada tipo de uso (Ex.: Uso – F, D, O; Colunas Finais – F_final, D_final, O_final).

DemandComputeThreeDates: Os valores de demanda são calculados internamente com base em 3 períodos diferentes de dados de uso da terra do passado fornecidos pelo usuário no banco de dados.

Ano Intermediário (*middleYearForInterpolation*): O ano para ser considerado o final da primeira interpolação para o cálculo da demanda.

Ano Final (*finalYearForInterpolation*): O ano para ser considerado o final da segunda interpolação para o cálculo da demanda.

Colunas para Interpolação (*middleLandUseTypesForInterpolation, finalLandUseTypesForInterpolation*): Informa o nome da coluna com os valores de demanda

para os dois cálculos de interpolação de cada tipo de uso (Ex.: Uso – F_00, D_00, O_00; Colunas – F_05, D_05, O_05 (*middle*); F_10, D_10, O_10 (*final*)).

DemandExternalModel: Faz chamada de um modelo externo *.lua* para realizar o cálculo da demanda. É disponibilizado para o modelo externo todos os dados da simulação, e é esperado em seu retorno o preenchimento da demanda anual (*annualDemand*) para o ano corrente.

Arquivo (*file*): Endereço do arquivo *.lua* do modelo externo.

Demanda Anual (*annualDemand*): Informa a demanda anual por terra para cada tipo de uso (deve ser informado na mesma unidade de área da célula), na mesma ordem informada no *Tipos de Uso*, inicialmente vem zerada por padrão, deve ser preenchida pelo modelo externo.

3.4 Calibração e Validação

Uma vez que o modelo proposto tenha sido efetivamente parametrizado e executado no LuccME é preciso ajustar e testar o despenho do modelo. Estas etapas consistem basicamente nas atividades de Calibração e Validação do modelo. A escolha do melhor componente de Potencial e/ou Alocação, bem como, o método utilizado para parametrização destes componentes é parte da calibração. Trata-se de um processo iterativo onde normalmente diversas rodadas preliminares são necessárias até a identificação dos melhores componentes e variáveis do modelo. Os diferentes componentes oferecem uma série de parâmetros que podem ser ajustados para melhor capturar os processos de mudança que estão sendo simulados. Contudo, cabe ressaltar que o horizonte temporal utilizado para calibração do modelo (ex.: 1995-2005), não pode ser o mesmo período utilizado para validação. Idealmente, o modelo calibrado deve ser validado para um período independente (ex.: 2005-2015). Outros recursos avançados como a utilização de variáveis dinâmicas, assim como, a definição de sub-regiões também podem contribuir significativamente para a calibração e validação dos modelos, como será discutido no próximo capítulo.

Os resultados das simulações podem ser validados pela métrica da validação de ajuste por múltiplas resoluções, adaptada de Costanza (1989), implementada no LuccME. Essa métrica permite estabelecer o nível de similaridade entre os mapas simulado e observado em diferentes

resoluções, por meio de janelas de amostragem que aumentam a cada passo de tempo, conforme ilustrado na Figura 6.

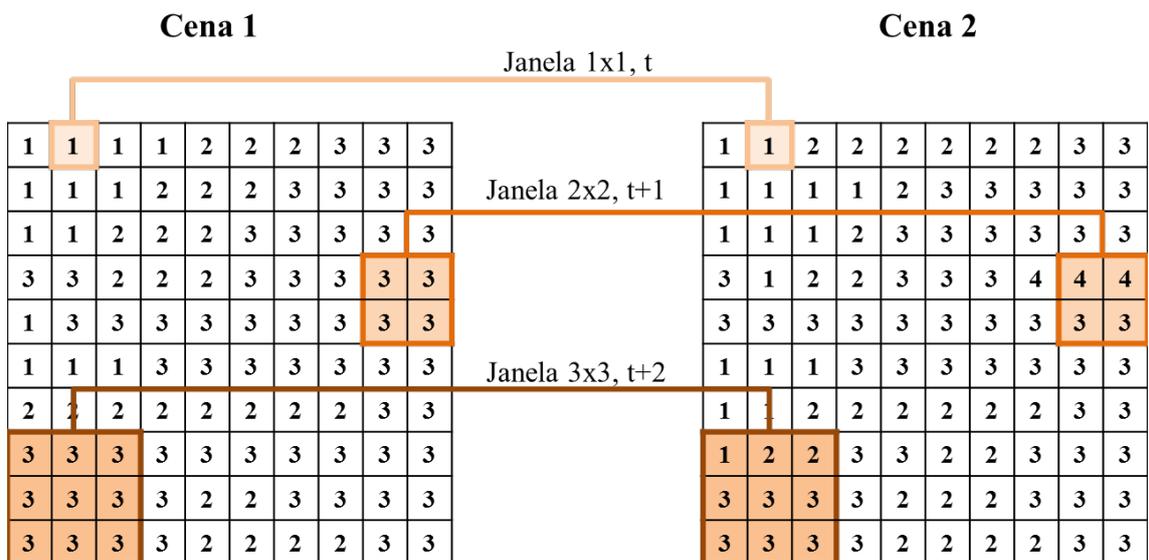


Figura 13 - Exemplo de validação por múltipla resolução (agregação de células por meio de janelas de amostragem).

O cálculo do nível de similaridade pode ser descrito segundo a equação abaixo:

$$NS_i = 1 - \left[\frac{\sum_{j=1}^n (|\sum_{c=1}^k \mathbf{dif}_{sim,c} - \sum_{c=1}^k \mathbf{dif}_{real,c}|)}{2 * \sum_{j=1}^n \sum_{c=1}^k \mathbf{dif}_{real,c}} \right] * 100$$

em que **NS** corresponde ao nível de similaridade entre os mapas real e simulado em uma determinada resolução *i*; *j* é a janela considerada; *n* estabelece o número de janelas a ser considerado; *c* é a quantidade de células em uma resolução *k* (*i**); e $\mathbf{dif}_{real} = \% \text{real}_{if} - \% \text{real}_{ij}$ e $\mathbf{dif}_{sim} = \% \text{sim}_{tfinal} - \% \text{real}_{tinicial}$ sendo *ti* e *tf* os anos inicial e real, respectivamente, considerados na validação.

Nesta versão do LuccME duas opções de validação estão disponíveis:

1. Considerando todo o espaço celular (Ext): Esta opção leva em consideração todo o padrão histórico acumulado para os diferentes tipos de uso da terra;
2. Considerando apenas as áreas modificadas (Dif): Esta opção leva em consideração apenas as áreas que passaram por algum processo de mudança, portanto, trata-se de um sistema mais específico de validação para cada tipo de uso da terra.

4 Recursos avançados

4.1 Variáveis dinâmicas

Um recurso avançado para construção de modelos no LuccME 3.1 é a utilização de variáveis dinâmicas. Este tipo de recurso é extremamente útil ao usuário quando os fatores determinantes que explicam o padrão de distribuição espacial dos tipos de uso da terra mudam significativamente ao longo do período considerado. Neste caso, as variáveis dinâmicas podem ser atualizadas em diferentes passos de tempo para melhorar o desempenho do modelo.

Quando o usuário desejar utilizar variáveis dinâmicas, novos espaços celulares devem ser criados no TerraME 2.0 ou outro SIG de preferência do usuário com a seguinte organização:

<NomeEspaçoCelular>_<AnoAtualização> (e.g. CS_2015).

Isso significa que no ano de atualização indicado no modelo, o LuccME vai ler o espaço celular atualizado. Para um exemplo da utilização deste recurso, por favor, veja o Lab 6 para modelos Contínuos disponibilizado junto com o pacote LuccME. Outro exemplo de aplicação completa utilizando este tipo de recurso pode ser observado em Aguiar et al. (2015).

4.2 Regiões

Outro recurso avançado para construção de modelos no LuccME 3.1 é a utilização de duas ou mais sub-regiões para definição de *Potencial* (tanto em modelos contínuos quanto discretos). Este tipo de recurso é extremamente útil ao usuário quando os fatores determinantes que explicam o padrão de distribuição espacial dos tipos de uso da terra mudam significativamente no espaço. Neste caso, a definição de sub-regiões permite ao usuário utilizar variáveis e coeficientes mais adequados as peculiaridades locais e assim melhorar o desempenho do modelo. A ocorrência de sub-regiões deve ser informada pelo usuário no componente de potencial.

Para um exemplo da utilização deste recurso, por favor, veja o Lab Region para modelos Contínuos disponibilizado junto com o pacote LuccME. Um exemplo de aplicação construída

utilizando este tipo de recurso (com versões anteriores do LuccME) é Coelho (2009) e Pimenta et al. (2010).

4.3 Cenários

Uma vez que o modelo desenvolvido seja capaz de representar de modo consistente os padrões de uso da terra observados no passado, o LuccME 3.1 oferece a possibilidade de explorar cenários futuros de mudança. Neste caso, o usuário deve informar o tempo inicial e nome do cenário de interesse, assim como, os períodos para atualização de variáveis dinâmicas (se aplicável).

Anos com Variáveis a serem Atualizadas (*updateYears*): Anos de atualização das variáveis dinâmicas.

Ano de Início do Cenário (*scenarioStartTime*) = <value>: Informa o ano de início do cenário.

Nome do Cenário (*scenarioName*): Informa o nome do cenário.

A *Demanda* para os cenários passa a ser uma entrada para o modelo e pode ser calculada por diferentes métodos, conforme descrito no capítulo anterior. Para um exemplo da utilização deste recurso, por favor, veja o Lab 7 para modelos Contínuos disponibilizado junto com o pacote LuccME. Um exemplo de aplicação completa utilizando este tipo de recurso é Aguiar et al. (2015).

5 Executando um modelo com LuccME

5.1 Exercícios

Neste capítulo nós convidamos novos usuários a explorar a Plataforma LuccME 3.1 através de exercícios práticos fornecidos no site do LuccME.

Estes exercícios, chamados LuccME Labs, incluem tanto exemplos de modelos *Discretos* como modelos *Contínuos* organizados em diferentes níveis de complexidade. Os usuários são convidados a rodar cada um dos Labs para comparar a configuração dos modelos e resultados. Uma síntese das principais diferenças entre Labs para modelos *Contínuos* e *Discretos* é apresentada nas Tabela 2 e 3.

Uma vez que o usuário esteja familiarizado com LuccME e suas ferramentas, os exemplos de modelos fornecidos neste Guia podem ser utilizados como base para construção de novos modelos. Em outras palavras, o usuário pode economizar tempo configurando novos modelos com base em exemplos anteriores tais como os LuccME Labs.

Tabela 2 - Resumo dos exercícios do tipo *Discreto* (Labs).

Exercícios	Componentes		
	Potencial	Alocação	Demanda
Lab1	<i>PotentialDNeighSimpleRule</i>	<i>AllocationDSimpleOrdering</i>	<i>DemandPreComputedValues</i>
Lab2	<i>PotentialDInverseDistanceRule</i>	<i>AllocationDSimpleOrdering</i>	<i>DemandPreComputedValues</i>
Lab3	<i>PotentialDNeighInverseDistanceRule</i>	<i>AllocationDSimpleOrdering</i>	<i>DemandPreComputedValues</i>
Lab4	<i>PotentialDLogisticRegression</i>	<i>AllocationDSimpleOrderingg</i>	<i>DemandPreComputedValues</i>
Lab5	<i>PotentialDLogisticRegressionNeighAttract</i>	<i>AllocationDClueSLike</i>	<i>DemandPreComputedValues</i>
Lab6	<i>PotentialDLogisticRegression</i>	<i>AllocationDClueSLike</i>	<i>DemandPreComputedValues</i>
Lab7	<i>PotentialDLogisticRegression</i>	<i>AllocationDClueSLike</i>	<i>DemandComputeTwoDates</i>
Lab8	<i>PotentialDLogisticRegression</i>	<i>AllocationDClueSLike</i>	<i>DemandComputeThreeDates</i>
Lab9	<i>PotentialDSampleBased</i>	<i>AllocationDClueSLike</i>	<i>DemandPreComputedValues</i>
Lab10	<i>PotentialDLogisticRegressionNeighAttractRepulsion</i>	<i>AllocationDClueSNeighOrdering</i>	<i>DemandPreComputedValues</i>
Lab11	<i>PotentialDLogisticRegression</i>	<i>AllocationDClueSNeighOrdering</i>	<i>DemandComputeTwoDates</i>
Lab12	<i>PotentialDLogisticRegressionNeighAttractRepulsion</i>	<i>AllocationDClueSLike</i>	<i>DemandPreComputedValues</i>

Tabela 3 - Resumo dos exercícios do tipo *Contínuo* (Labs).

Exercícios	Componentes		
	Potencial	Alocação	Demanda
Lab1	<i>PotentialCLinearRegression</i>	<i>AllocationCCLueLike</i>	<i>DemandPreComputedValues</i>
Lab2	<i>PotentialCSpatialLagRegression</i>	<i>AllocationCCLueLike</i>	<i>DemandPreComputedValues</i>
Lab3	<i>PotentialCSpatialLagRegression</i>	<i>AllocationCCLueLikeSaturation</i>	<i>DemandPreComputedValues</i>
Lab4	<i>PotentialCSpatialLagRegression</i>	<i>AllocationCCLueLike</i>	<i>DemandComputeTwoDates</i>
Lab5	<i>PotentialCSpatialLagRegression</i>	<i>AllocationCCLueLike</i>	<i>DemandComputeThreeDates</i>
Lab6	<i>PotentialCSpatialLagRegression</i>	<i>AllocationCCLueLike</i>	<i>DemandPreComputedValues</i>
Lab7	<i>PotentialCSpatialLagRegression</i>	<i>AllocationCCLueLike</i>	<i>DemandPreComputedValues</i>
Lab8	<i>PotentialCSpatialLagLinearRegressionMix</i>	<i>AllocationCCLueLike</i>	<i>DemandPreComputedValues</i>
Lab9	<i>PotentialCSampleBased</i>	<i>AllocationCCLueLike</i>	<i>DemandPreComputedValues</i>
Lab10	<i>PotentialCLinearRegression</i>	<i>AllocationCCLueLike</i>	<i>DemandExternalModel</i>

Referências

AGUIAR, A.P.D. Modeling land use change in the Amazon: Exploring intra-regional heterogeneity. 2006. 153 p. (INPE- 08.10.18.21-TDI). Thesis (Doctoral in Remote Sensing) - Brazilian National Institute for Space Research (INPE), São José dos Campos, 2006. Retrieved from: <http://urlib.net/6qtX3pFwXQZGivnJSY/M7t7e>.

CARNEIRO, T. G. S., ANDRADE, P. R., CÂMARA, G., MONTEIRO, A. M. V., & PEREIRA, R. R. (2013). An extensible toolbox for modeling nature–society interactions. *Environmental Modelling & Software*, 46, 104-117.

VERBURG, P. H.; KOK, K.; PONTIUS JR, R. G.; VELDKAMP, A. Modeling land-Use and land-cover change. In: LAMBIN, E.F.; GEIST, H. (Eds.). *Land-use and land-cover change: local processes and global impacts*. Berlin: Springer, 2006, p. 117-135.

VELDKAMP, A.; FRESCO, L. CLUE-CR: an integrated multi-scale model to simulate land use change scenarios in Costa Rica. *Ecological Modeling*, 91: 231-248, 1996.

VERBURG, P.; DE KONING, G.; KOK, K.; VELDKAMP, A.; BOUMA, J. A spatial explicit allocation procedure for modeling the pattern of land use change based upon actual land use. *Ecological Modeling*, 116: 45-61, 1999.

VERBURG, P.; VELDKAMP, A. The role of spatially explicit models in land-use change research: a case study for cropping patterns in China. *Agriculture, Ecosystems and Environment*, 85: 177–190, 2001.

SOARES-FILHO, B.; CERQUEIRA, G.; PENNACHIN, C. DINAMICA – a stochastic cellular automata model designed to simulate the landscape dynamics in an Amazonian colonization frontier. *Ecological Modeling*, 154 (3): 217 – 235, 2002.

PONTIUS, R. G.; CORNELL, J. D.; HALL, C. A. Modeling the spatial pattern of land-use change with GEOMOD2: application and validation for Costa Rica. *Agriculture, Ecosystems & Environment*, v. 85, n. 1, p. 191-203, 2001.