



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

**Estudo Exploratório de Modelagem e Simulação
Baseada em Agentes para Compreensão de Aspectos
de Governança do Sistema de Saúde Pública no DF**

Guilherme de Oliveira Silva

Monografia apresentada como requisito parcial
para conclusão do Curso de Engenharia da Computação

Orientador
Prof. Dr. Jorge Henrique Cabral Fernandes

Brasília
2020

Dedicatória

Eu dedico este trabalho a todos os estudantes da UnB, que assim como eu tiveram depressão e passaram por momentos difíceis durante o curso de graduação. Agradeço aos meus familiares e amigos por sempre me apoiar e me incentivar a terminar mais essa etapa na minha vida.

Agradecimentos

Agradeço aos meus pais por sempre me incentivar e apoiar nas horas difíceis. Aos meus amigos e colegas, os quais passamos tantas dificuldades e alegrias juntos. Ao meu orientador e a Fiocruz pela oportunidade de trabalhar neste projeto. E a todos os meus professores, os quais fizeram o seu melhor para me ensinar e transmitir todo o seu conhecimento.

Resumo

Este trabalho explorou a simulação multiagentes para melhor compreender a implementação de políticas públicas de saúde, tomando por base o DF. Dois estudos com simulação foram realizados. Um deles usou a ferramenta SocLab e abordou a modelagem multiagente da governança de intervenções na Cidade Estrutural. Outro utilizou a ferramenta GAMA para a modelagem de agentes móveis no Sistema de Saúde do Distrito Federal. O modelo de agentes com mobilidade levou em consideração os dados disponíveis da Sala de Situação do DF e no IBGE, na modelagem das interações presentes na atenção primária à saúde no cuidado das condições crônicas no Distrito Federal. Os dois modelos buscam prover aos interessados um melhor entendimento sobre o impacto que cada estratégia de melhoria da saúde pode gerar na população de forma geral e em locais específicos. Após a construção, simulação e convergência dos modelos, foi constatado que uma maior cooperação entre os agentes no caso da Governança de Territórios Saudáveis e Sustentáveis na Cidade Estrutural levaria a resultados globais melhores, do ponto de vista de satisfação das partes interessadas. No modelo da atenção primária à saúde no DF, foi possível observar uma diminuição no número de pessoas doentes em relação a situação inicial e um padrão de comportamento da população em relação à sua movimentação e distribuição no mapa. Assim, pode-se averiguar que é possível a construção de modelos e simulações multiagente para o melhor entendimento de estruturas organizacionais envolvendo a população, instituições e serviços de saúde. Também mostra-se ser possível utilizar os resultados para sugerir as melhores estratégias a serem adotadas em cada caso.

Palavras-chave: Simulação multiagente, Territórios Saudáveis e Sustentáveis, Políticas públicas de saúde, Atenção primária à saúde, Agentes móveis, Dilema do prisioneiro, Ferramentas de simulação e modelagem

Abstract

This work explored multi-agent simulation to better understand the implementation of policies for public health system on the DF. Two simulation studies were carried out. One of them used the software SocLab for modeling multi-agent system of the governance in the Cidade Estrutural. Another used the software GAMA for modeling mobile agents in the Distrito Federal Health System. The model created in GAMA took into account the data available from the Sala de Situação do DF and the IBGE, in modeling the interactions present in primary health care in the care of chronic conditions in the Federal District. Both models seek to provide a better understanding of the impact that each health improvement strategy can have on the population in general and in specific locations. After the construction, simulation and convergence of the models, it was found that greater cooperation between the agents in the case of Governance of Healthy and Sustainable Territories in the Cidade Estrutural would lead to better global results, for those involved. In the model of primary health care in DF, it was possible to observe a decrease in the number of sick people in relation to the initial situation and a pattern of behavior of the population in relation to their movement and distribution on the map. Thus, it can be ascertained that it is possible to build multi-agent models and simulations for a better understanding of organizational structures involving the population, institutions and health services. It also shows that it is possible to use the results to suggest the best strategies to be adopted in each case.

Keywords: Multi-agent simulation, Sustainable and Healthy Territories, Public health policies, Primary health care, Modeling Softwares

Sumário

1	Introdução	1
1.1	Questão de Pesquisa	1
1.2	Relevância e Justificativa	1
1.2.1	Governança de Território Saudável e Sustentável da Cidade Estrutural	2
1.2.2	Dinâmica de doença e cura no Entorno das Unidades Básicas de Saúde do DF	3
1.3	Metodologia	4
1.4	Conteúdo da Monografia	5
2	Fundamentos ou Aplicação de Modelagem Baseada em Agentes	7
2.1	Modelagem da Realidade	7
2.2	Modelos Computacionais Baseados em Agentes	8
2.3	Modelos de Sistemas MultiAgentes	8
2.3.1	Aplicação de Modelos Baseados em Agentes a Questões Sociais . . .	9
2.4	Síntese	12
2.5	Governança de Territórios Saudáveis e Sustentáveis	13
2.6	Transição Demográfica e as Condições Crônicas	13
2.7	Crise de Financiamento do SUS	14
2.8	Redes de Atenção à Saúde	14
2.9	Atenção Primária à Saúde	15
2.10	Aplicabilidade de ABMs no campo das Políticas de Saúde	16
3	Plataformas de Modelagem e Simulação Baseada em Agentes	17
3.1	SocLab	17
3.1.1	Bases Teóricas	17
3.1.2	Conceitos Subjacentes aos modelos no SocLab	18
3.1.3	Dilema do Prisioneiro	23
3.1.4	Um exemplo de caso de modelagem e simulação de uma agência de viagem	24

3.1.5	Simulando um modelo no SocLab	28
3.2	GAMA	30
3.2.1	Conceitos Subjacentes aos modelos no GAMA	31
3.2.2	Criação de modelos no GAMA	33
4	Modelos Baseados em Agentes	44
4.1	Modelando a Governança de Território Saudável e Sustentável para a Cidade Estrutural	44
4.1.1	Propósito e padrões do Modelo	44
4.1.2	Entidades e variáveis de estado	44
4.1.3	Visão do processo e escalonamento	47
4.1.4	Design do Modelo	47
4.1.5	Implementação do Modelo: codificação, inicialização/dados de entrada	47
4.1.6	Stakes	48
4.1.7	Effects	48
4.1.8	Constraints	50
4.1.9	Solidarities	50
4.1.10	Fuzzy Stakes e Fuzzy Solidarities	50
4.1.11	Opções de Simulação do SocLab	50
4.1.12	Networks	50
4.1.13	Simulation Explorer	51
4.1.14	Dificuldades encontradas com o uso do SocLab	53
4.2	Modelando a Dinâmica de Doenças e Curas no entorno das Unidades de Saúde do DF	54
4.2.1	Propósito e padrões do Modelo	54
4.2.2	Entidades e variáveis de estado	54
4.2.3	Visão do processo e escalonamento	55
4.2.4	Design do Modelo	55
4.2.5	Implementação do Modelo: codificação, inicialização/dados de entrada	56
4.2.6	Definição dos Atores	57
5	Resultados e Discussões	60
5.1	Governança de Território Saudável e Sustentável para a Cidade Estrutural usando SocLab	60
5.2	Dinâmica de Doenças e Curas no entorno das Unidades de Saúde do DF usando GAMA	62

6 Conclusão	70
6.1 Propostas de Trabalhos Futuros	71
Referências	76

Lista de Figuras

3.1	Meta-modelo básico de uma organização modelada no SocLab.	18
3.2	Meta-modelo subjacente para interpretação dos resultados de uma simulação de ação organizacional modelada no SocLab.	20
3.3	Meta-modelo completo de uma organização modelada no SocLab.	21
3.4	Rede que modela a relação entre a Secretária Agnes e seu Director.	25
3.5	Documentação de um modelo desenvolvido no SocLab	26
3.6	Stakes e effects definidos para as relações e atores no modelo Travel-tour.	27
3.7	Painel de simulação do SocLab, para o modelo Travel-tour.	28
3.8	Síntese de simulação do modelo Travel-tour.	29
3.9	Meta-modelo GAML.	31
3.10	Meta-modelo Completo do GAML.	32
3.11	Gráfico de pizza gerado no código exemplo presente em 3.20	41
3.12	Gráfico de séries gerado no código exemplo 3.20	42
3.13	Histograma gerado no código exemplo 3.20.	42
3.14	Gráfico de radar gerado no código exemplo 3.20.	43
4.1	Painel para edição do modelo GTSSE	48
4.2	Painel para edição do controle das relações do modelo GTSSE	49
4.3	Painel para edição dos Stakes do modelo GTSSE	49
4.4	Painel para edição dos Effects do modelo GTSSE	50
4.5	Painel para edição de Solidarities do modelo GTSSE	51
4.6	Organização estrutural do modelo GTSSE	51
4.7	Painel para definição dos parâmetros iniciais do modelo GTSSE	52
4.8	Painel para definição dos parâmetros dos atores do modelo GTSSE	53
5.1	Resultados da simulação do modelo GTSSE	60
5.2	Resultado das relações do modelo GTSSE	61
5.3	Quantidade de relações concluídas/ satisfação dos atores do modelo GTSSE	61
5.4	Influência/Satisfação dos atores do modelo GTSSE	62
5.5	Poder/Capacidade de Ação dos atores do modelo GTSSE	62

5.6	Mapa no início da simulação do modelo DDCDF	63
5.7	Distribuição da população por idade no modelo DDCDF	64
5.8	Quantidade de pessoas doentes no início da simulação do modelo DDCDF .	65
5.9	Mapa após 600 ciclos de simulação do modelo DDCDF	65
5.10	Doentes após 600 ciclos de simulação do modelo DDCDF	66

Capítulo 1

Introdução

Recentemente, uma comissão da Organização Mundial da Saúde (OMS) apresentou um relatório que tem como foco alertar os governos de todos os países para o crescente número de casos de pessoas com doenças crônicas não transmissíveis (DCNTs), doenças que não são resolvidas num tempo curto, definido usualmente em três meses, de forma que sejam tomadas ações para o tratamento dessas [1, 10]. Para isso, é necessário um compromisso político e o aumento imediato das ações para enfrentar a epidemia de DCNTs, principais causas de morte e de problemas de saúde no mundo [1, 7].

Segundo a Organização Pan-Americana da Saúde, [2] doenças como câncer, diabetes, doenças pulmonares e cardiovasculares matam 41 milhões de pessoas a cada ano, respondendo por 71% de todas as mortes no mundo. Desse número, 15 milhões de mortes ocorrem em indivíduos com idades entre 30 e 70 anos [2].

1.1 Questão de Pesquisa

A Questão de pesquisa investigada nesse trabalho é declarada na forma das seguintes perguntas.

Quais as práticas, desafios e benefícios da utilização de modelagem e simulação baseada em agentes para compreensão da dinâmica de doença e cura da população, e o entendimento das interações presentes em um sistema de governança territorial?

Uma vez realizado as simulações é possível gerar sugestões de estratégias que maximizem os resultados desejados e com isso realizar apoio à tomada de decisões?

1.2 Relevância e Justificativa

Ferramentas de modelagem e simulação multiagente podem ser definidas como sistemas de software para a elaboração de modelos de situações do mundo real, com certo grau de

similaridade definido pelos objetivos desejados, de forma que seja possível testar teorias e avaliar o modelo de acordo com observações feitas sobre a situação real. Tais ferramentas e aplicação em ciências sociais e no estudo de governança de territórios saudáveis e sustentáveis são descritas em mais detalhes na Seção 2.2 deste trabalho.

Por meio dessas formas de modelagem este trabalho busca saber se é possível obter respostas confiáveis para auxiliar no processo de tomada de ações da atenção primária à saúde (APS) que corresponde ao primeiro nível de atenção à saúde que tem como objetivo desenvolver uma atenção integral que impacte positivamente na situação de saúde das coletividades, no cuidado das condições crônicas no DF, que consiste em obter informações obtidas de diversas fontes, em seguida realizar uma análise individual e tentar correlacionar o conhecimento obtido em cada análise com a finalidade de definir estratégias mais eficientes para o combate e monitoramento das doenças crônicas.

Como esse é um problema grave, é necessário que as estratégias adotadas sejam mais eficientes, atingindo uma maior parcela da população e reduzindo o desperdício de recursos.

Este trabalho parte da hipótese de que é possível a utilização de ferramentas de modelagem e simulação de sistemas multiagentes para se investigar o cenário real de um sistema de saúde. Utilizando simulações de nível micro, meso e macro. O nível micro apresenta relações explícitas entre os atores são diretamente modeladas. No nível meso as estruturas organizacionais são determinantes nas relações entre agentes e em sua relação com mudanças na estrutura social superior. No nível macro as estruturas sociais são determinantes da organização dos níveis inferiores [3, 1].

Foi desenvolvido dois diferentes estudos de caso práticos sobre modelagem de sistemas multiagentes: (i) um deles voltado à modelagem do processo de governança de territórios saudáveis e sustentáveis no âmbito da Cidade Estrutural, de nível microsociológico, e (ii) outro voltado à modelagem da dinâmica de surgimento e cura de doenças no âmbito do deslocamento dos usuários e agentes de saúde das Unidades Básicas de Saúde(UBS) do DF, de níveis micro e mesosociológicos [4].

1.2.1 Governança de Território Saudável e Sustentável da Cidade Estrutural

A governança territorial é definida como o processo institucional-organizacional de construção de uma estratégia para compatibilizar os diferentes modos de coordenação entre atores geograficamente próximos em caráter parcial e provisório, atendendo a premissa de resolução de problemas inéditos [5]. Essa forma de organização e gerenciamento está

sendo utilizada em Territórios Saudáveis e Sustentáveis (TSS) que podem ser definidos como [6, p. 4]:

Espaços relacionais e de pertencimento onde a vida saudável é viabilizada, por meio de ações comunitárias e de políticas públicas, que interagem entre si e se materializam, ao longo do tempo, em resultados que visam a atingir o desenvolvimento global, regional e local, em suas dimensões ambientais, culturais, econômicas, políticas e sociais.

Mais detalhes sobre Governança de Territórios Saudáveis e Sustentáveis são descritos na Seção 3.1.

O modelo do processo de Governança de Território Saudável e Sustentável para a Cidade Estrutural (GTSSE) foi desenvolvido para emular as oportunidades e dificuldades na governança da intervenção proposta pela Fiocruz-DF [7], de promoção de um território sustentável e saudável [6] na Cidade Estrutural - DF, com o objetivo de levar a Cidade Estrutural a alcançar os 17 Objetivos de Desenvolvimento Sustentável (ODSs) [8] propostos pela ONU na Agenda 2030.

Esse programa foi viabilizado por financiamento do Governo do Distrito Federal, e conta com a parceria do Instituto Federal de Brasília e da Universidade de Brasília.

Nesse projeto de intervenção, a saúde (ODS 3) foi definida como pilar principal para o desenvolvimento do território, de forma que a vida humana seja o aspecto central dos objetivos das políticas públicas. A saúde foi definida como base porque apenas um indivíduo saudável tem condições de exercer o máximo de sua capacidade física e intelectual para atuar nas atividades de produção e geração de recursos individuais e coletivos. Dessa forma é importante identificar e “repensar as relações entre Estado e a Sociedade, seja do setor Público ou Privado, a partir da perspectiva de uma gestão democrática.” [9, 6].

A construção do modelo de governança em saúde [10, 11] da Cidade Estrutural se justifica pela necessidade de promover o melhor alinhamento de interesses entre as partes da ação coletiva, visando dar maior efetividade e celeridade a essa política pública [12].

1.2.2 Dinâmica de doença e cura no Entorno das Unidades Básicas de Saúde do DF

O modelo de simulação da Dinâmica de Doença e Cura no Entorno das Unidades Básicas de Saúde do DF (DDCDDF) foi desenvolvido para emular os processos decorrentes do deslocamento espacial dos usuários do SUS em suas relações com os agentes de saúde vinculados às unidades de saúde da Secretaria de Saúde do DF, inclusive as unidades básicas.

A construção desses modelos se justifica pela necessidade de melhor criar modelos que permitam investigar a influência dos fatores da espacialização geográfica na dinâmica das doenças crônicas no DF, visando dar maior apoio à tomada de decisões sobre políticas de APS [4].

1.3 Metodologia

Foram feitos dois modelos usando ferramentas de modelagem Soclab[3] e GAMA [13], após o desenvolvimento e simulações iniciais foram realizados ajustes nos parâmetros que regem o comportamento e interação dos agentes presentes na simulação para que fosse melhor refletido os aspectos da realidade.

Para a construção de um agent-based model(ABM) deve-se começar pela modelagem dos agentes, especificando quantos são, quais papéis vão desempenhar no modelo, seus atributos e comportamentos, os quais podem ser divididos em fixos ou mutáveis [14].

No o desenvolvimento do modelo de Governança de Território Saudável e Sustentável da Cidade Estrutural (GTSSE) foi feito um estudo com a finalidade de identificar quais são os principais atores individuais e organizacionais promotores da ação que efetivamente ocorre na Cidade Estrutural e como esses interagem entre si, a partir da análise dos atores e relações presentes no âmbito do programa de atuação da Fiocruz Brasília em Territórios Saudáveis e Sustentáveis [7, 15, 6]. Entretanto para uma melhor representatividade do modelo deveria ter sido feito um sistema bola de neve para a identificação de atores relevantes que promovem ações sociais e de política pública na Cidade Estrutural e construção da rede de interações.

Após a coleta de dados baseada na participação dos atores em suporte às atividades durante dois encontros ocorridos em meados do ano de 2019 [16], no Campus Cidade Estrutural do Instituto Federal de Brasília, foi desenvolvido um modelo multiagente baseado na teoria da Ação Organizada de Crozier e Friedberg [17], simulável através do software SocLab [3]. Esse modelo partiu do pressuposto de que os atores que promovem a governança desse território, de forma geral, possuem comportamento determinado por ajustes simultâneos aos comportamentos exibidos pelos demais ao longo do tempo. Isso é computacionalmente viabilizado através do uso do dilema do prisioneiro [18], usado na Teoria dos Jogos [19]. Todos os atores do modelo foram criados a partir da observação de sua atuação no âmbito dos encontros presenciais frequentados. As regras de interação (padrões de relacionamentos entre agentes) foram definidas de forma que os atores pudessem agir individualmente ou colaborar entre si. Em seguida foi feita a calibragem dos valores dos atributos pertencentes a cada ator e tipo de relação entre eles, definindo assim

os parâmetros da simulação. Os resultados do modelo, após convergência da simulação, foram então interpretados.

No desenvolvimento do modelo da Dinâmica de Doenças e Curas no Entorno das Unidades de Saúde do DF foram utilizados os dados quantitativos populacionais do DF disponíveis no IBGE, além dos dados sobre as UBS disponíveis na Sala de Situação do DF, site que contém informações estratégicas da Secretaria de Saúde em diversos formatos, para entender melhor o funcionamento da estrutura da saúde pública no DF, a divisão da população por faixa etária e as condições referentes aos doentes crônicos.

Em seguida foi modelado o espaço geográfico fixo onde as UBSs estão localizadas, e onde os usuários e agentes se deslocam. Esse espaço geográfico foi modelado em arquivos do tipo *shapefile*, com a finalidade de construir o mapa do DF e suas regionais de saúde, bem como as rodovias e a localização das unidades de saúde. Com o espaço geográfico delimitado, foram modeladas as regras de comportamento dos agentes, algumas das quais definem como esses agentes se deslocam no território, e o seu comportamento dada a condição de saúde, e como interagem entre si.

A construção desse tipo de modelo se justifica pela necessidade de conhecer as potencialidades e limitações da modelagem de agentes móveis, que se relacionam em redes em um território.

Os modelos desenvolvidos apresetam algumas restrições, no modelo GTSSE foram utilizados poucos atores e relações devido a uma restrição da ferramenta e computadores utilizados, no modelo DDCDF não houve georreferenciação das UBS, nem dos atores. E em ambos os modelos não houve participação de usuários.

Os artefatos desenvolvidos nesse trabalho podem ser encontrados no seguinte repositório <https://github.com/guisilva7/Estudo-Exploratorio-de-Modelagem-e-Simulacao-Baseada-em-Agentes>.

1.4 Conteúdo da Monografia

Este trabalho apresenta, no Capítulo 2, conceitos básicos sobre modelos baseados em agentes e de sistemas multiagente, contexto de utilização, descrição dos tipos de atores que podem existir e quais os tipos de regras os rege.

Com a finalidade de apresentar a situação atual do Brasil é feito, no Capítulo 3 uma breve contextualização sobre Sistema Único de Saúde (SUS) e como a utilização de ABM(Agente Based Model) e sistemas multiagentes podem ser empregados com a finalidade de melhorar as perspectivas atuais.

Para um melhor entendimento dos modelos desenvolvidos, é feito no Capítulo 4, o detalhamento dos softwares e linguagens utilizados. Nesse capítulo são explicadas as

funcionalidades, conceitos, exemplos, forma de execução e alguns códigos desenvolvidos no SocLab [3] e GAMA [13].

O Capítulo 5 contém o detalhamento e o passo a passo de como os modelos foram contruídos e simulados.

Os resultados e análises das simulações do modelo de Governança de Território Saudável e Sustentável para a Cidade Estrutural (GTSSE) e do modelo da Dinâmica de Doenças e Curas no Entorno das Unidades de Saúde do DF (DDCDF) são apresentaddos no Capítulo 6.

No Capítulo 7 são apresentados as discussões teóricas dos modelos desenvolvidos levando em conta as teorias apresentadas nos Capítulos 3 e 4.

No Capítulo 8 é apresentado a conclusão deste trabalho além de algumas ideias e pseudocódigos para um futuro refinamento do modelo da Dinâmica de Doenças e Curas no Entorno das Unidades de Saúde do DF, proporcionando uma maior semelhança com a realidade.

Capítulo 2

Fundamentos ou Aplicação de Modelagem Baseada em Agentes

Este capítulo apresenta uma introdução da teoria sobre modelagem de agentes e sistemas baseados em agentes utilizados no desenvolvimento deste trabalho, de forma que a Seção 2.1 contém detalhamentos sobre Modelos Baseados em Agentes e na Seção 2.2 são explicados os Modelos MultiAgentes.

2.1 Modelagem da Realidade

Desde que o ser humano começou a tentar entender os fenômenos que ocorrem ao seu redor, a utilização de modelos vem sendo empregada como uma ferramenta essencial para simplificar sistemas de maior complexidade, sem que esses percam o seu comportamento característico [18]. De fato, a criação de modelos da realidade é base para a construção de qualquer sistema complexo adaptativo [20].

Esses modelos (da realidade) na maioria das vezes são implícitos, com suposições ocultas, consistência interna não testada, relação com os dados e conseqüências lógicas incertas. Nos modelos explícitos, as suposições são descritas de forma clara e com detalhes, facilitando o entendimento e replicação dos resultados obtidos [21].

Além de possibilitar uma visão mais focada e realista da realidade e de seus problemas, os modelos também podem ser utilizados como guias para coleta de dados, para demonstrar *tradeoffs*, treinar profissionais, definir ações para situações críticas, e sugerir formas mais eficientes de se realizar alguma atividade, entre vários outros benefícios [21].

2.2 Modelos Computacionais Baseados em Agentes

Com o advento da computação moderna, foram desenvolvidos diversos *softwares* para ajudar na construção e simulação de modelos computáveis, que passaram a ser desenvolvidos baseados em agentes (*Agent-Based Models* - ABMs). A criação dos ABMs engloba, por exemplo, elementos de teoria dos jogos [19], sistemas multiagentes [22], ciências sociais computacionais [23], e inteligência artificial [24].

Agentes de software inteligentes são entidades computacionais autônomas que interagem entre si e estão situados em um determinado ambiente que provê os estímulos necessários para a tomada de decisões de acordo com as regras comportamentais previamente definidas [25].

As primitivas que regem o comportamento de cada agente podem ser definidas de forma simples ou complexa, onde os agentes podem sofrer de limitações cognitivas e o seu comportamento pode ser modelado a partir de padrões de tomada de decisão humanos [26].

Embora as premissas de racionalidade pareçam ser mais confiáveis que as heurísticas, deve-se levar em consideração o fato de que os modelos de agentes que assumem comportamento racional devem ser solucionáveis, portanto, só devem ser utilizados quando se há um comportamento ideal definido [27].

Por outro lado, modelos de agentes computacionais que assumem regras comportamentais baseadas em psicologia podem se encaixar melhor em dados experimentais e empíricos, quando comparados a modelos de ator racional [28].

2.3 Modelos de Sistemas MultiAgentes

Um Sistema MultiAgente (MAS) é um sistema de Inteligência Artificial Distribuída que têm como objetivo resolver problemas de maneira flexível e robusta, reunindo conjuntos de agentes interagindo de maneira cooperativa ou competitiva, em prol de objetivos possivelmente comuns interagindo com o ambiente [29]. Os agentes nesse caso são autônomos, referente ao fato de que os agentes têm existência própria independente da existência de outros agentes e podem deliberar de forma autônoma, sem interação com o ser humano [30].

A definição de agentes mais bem aceita, no contexto de MAS, denomina um agente como uma entidade real ou abstrata que é capaz de agir sobre ela mesma e sobre seu ambiente, que dispõe de uma representação parcial desse ambiente que, em um universo multiagente, pode comunicar-se com outros agentes, e cujo comportamento é consequência de suas observações, de seu conhecimento e das interações com outros agentes [31].

Em um MAS os agentes podem ser divididos em tipos que variam de simples a complexos, tais como: agentes passivos ou sem objetivos, agentes ativos ou com objetivos simples e agentes cognitivos que possuem objetivos complexos [32].

Os ambientes dos agentes em um MAS também podem ser organizados de acordo com propriedades, como acessibilidade (se é possível coletar informações completas sobre o ambiente), determinismo (se uma ação causa um efeito definido), dinâmica (quantas entidades influenciam o ambiente no momento), discretude (se o número de ações possíveis no ambiente é finito), episodicidade (se as ações do agente em determinados períodos influenciam outros períodos) [24] e dimensionalidade (se as características espaciais são fatores importantes do ambiente e o agente considera o espaço no ambiente em sua tomada de decisão).

Cada agente em um MAS possui um conjunto de características e comportamentos próprios, que tem como finalidade definir o seu trabalho, propósito e capacidades. Os agentes em um MAS têm várias características importantes [22], tais como:

Autonomia , onde agentes são pelo menos parcialmente independentes, autoconscientes e autônomos;

Visualizações Locais , onde nenhum agente tem uma visão global completa ou o sistema é muito complexo para um agente explorar esse conhecimento.

Descentralização , onde nenhum agente é designado como controlador ou o sistema é efetivamente reduzido a um sistema monolítico [29].

2.3.1 Aplicação de Modelos Baseados em Agentes a Questões Sociais

Os modelos baseados em agentes (ABMs) possuem um grande leque de oportunidades de aplicação para auxiliar no estudo de cenários e eventos pertencentes a área de ciências sociais. Porém, há uma grande variedade de modelos que possuem objetivos, funcionalidades e regras diferentes [33].

No nível mais básico, os ABMs permitem que os pesquisadores vinculem regras comportamentais para agregar padrões, sejam esses de equilíbrio ou padrões complexos. Além disso, também é possível incluir contexto social e geográfico aos ABMs [14].

Na construção políticas públicas, o trabalho em modelos estatísticos aplicados mostra-se importante [34]. As pessoas segregam por raça, renda, religião e outras características, e as evidências sugerem que o comportamento dos indivíduos depende, em parte, das pessoas ao seu redor [34]. De maneira semelhante, os ABMs também podem incluir redes [35], que podem ser redes de políticas, redes de interação ou redes de amizade [36].

Outro fator que levou ao aumento da utilização de ABMs é a possibilidade de se trabalhar simultaneamente com vários processos ou domínios. Esse aspecto é interessante, pois já que os agentes (atores) podem aprender uns com os outros, a junção de domínios pode levar a repercussões e comportamentos que não ocorreriam se os modelos fossem considerados separadamente [37]. Porém, deve-se ter cuidado para não incluir muitos domínios com no modelo com a finalidade de aumentar seu realismo, já que isso pode gerar dificuldades na interpretação dos dados.

No entanto, esse reconhecimento dos limites do realismo não implica que modelos que vinculem muitos domínios não possam produzir resultados significativos[38].

Os ABMs podem incluir heterogeneidade, não apenas em localização, informações, preferências e habilidades, mas também em regras de aprendizado, perspectivas, modelos mentais, repertórios comportamentais e enquadramento cognitivo [38].

Para a construção de um ABM deve-se começar pela modelagem dos agentes, especificando quantos são, quais papéis vão desempenhar no modelo, seus atributos e comportamentos, os quais podem ser divididos em fixos ou mutáveis [14].

Os atributos de um agente são geralmente os mesmos que seriam encontrados em um modelo de teoria dos jogos. Isso inclui afinidade, ações, reputação, recompensas, e assim por diante[19]. Para a criação da situação a ser analisada, deve-se atribuir valores aos atributos, sendo colocado valor 0 para atributos que não fazem parte do escopo de algum agente [38]. O conjunto de todos os atributos pode ser visto como uma configuração do modelo, e que podem ser alterados de acordo com as interações realizadas pelos agentes ao longo da simulação.

Os ABMs geralmente possuem uma quantidade de agentes que pode variar de 2 até alguns milhares para modelos extremamente simples, mas nada impede que este número seja da ordem de milhões, para modelos mais complexos e de alta fidelidade. Porém, vale ressaltar que quanto maior o número de agentes, maior será o esforço computacional e tempo gastos para realizar a simulação [39], bem como para interpretar seus resultados.

Os modelos de ABMs simples são frequentemente usados para gerar elucidar princípios fundamentais, enquanto os modelos de ABMs de alta fidelidade são mais frequentemente usados para elaborar políticas, e funcionar para calibração [38].

Para diminuir o gasto computacional, alguns ABMs desconsideram que os agentes podem interagir aleatoriamente entre si, e consideram apenas interações de agentes de acordo com o estado deles, que pode ser baseado em localização, posição social, disponibilidade, entre outros [40].

Definido quais os atores e seus atributos, deve-se definir as regras que regem o comportamento de cada um. Regras comportamentais simples podem depender apenas da configuração atual do agente, enquanto regras mais sofisticadas também podem depender

de configura ções anteriores (uso de memória) e podem levar em consideração as ações potenciais de outros agentes [41].

Na maioria das vezes, ABMs utilizam regras provenientes de outras disciplinas, tais como teoria dos jogos, levando em conta que as regras comportamentais também podem ser classificadas quanto ao seu estado, que podem ser regras fixas, como no problema do dilema do prisioneiro iterado, e regras adaptativas que permitem o agente aprender a jogar um jogo [18].

Existe uma distinção relevante entre as regras de aprendizado individual e as regras de aprendizado populacional em ABMs. Regras de aprendizado baseadas em indivíduos, como aprendizado de melhor resposta, têm maior probabilidade de localizar os Equilíbrios de Nash [42]. O Equilíbrio de Nash representa uma situação em que, em um jogo envolvendo dois ou mais jogadores, nenhum jogador tem a ganhar mudando sua estratégia unilateralmente. Segundo esse conceito, apesar de os participantes não cooperarem, é possível que a busca individual da melhor solução conduza o jogo a um resultado em que se verifique estabilidade, não havendo incentivo para que nenhum deles altere o seu comportamento [43].

Se os agentes adaptam suas ações como indivíduos, isso é, sem nenhuma coordenação, nenhum equilíbrio que não seja um Equilíbrio de Nash seria sustentável: algum agente encontraria uma estratégia superior com probabilidade positiva. Nas regras de aprendizado populacional, a recompensa depende do desempenho relativo e os agentes tendem a ser influenciados por vizinhos mais bem-sucedidos. Portanto, os agentes podem bloquear comportamentos considerados ruins ou com baixo rendimento [19].

Uma vez definidos os atributos e comportamentos dos agentes, o modelo deve descrever detalhadamente as interações, levando em consideração o tempo, recompensas e efeitos que estas causarão [44].

Nos primeiros modelos ABMs desenvolvidos, a maioria das regras era aplicadas simultaneamente, mas isso se torna problemático quando os agentes podem mudar de local, pois não há tempo suficiente para a reação dos atores a pequenas alterações (adição de um novo vizinho, mudança na taxa de retorno), fazendo com que a simulação avance muito rapidamente para uma convergência ideal, mas que pode não ser próxima do real.

Um problema específico é como impedir que vários agentes se mudem para o mesmo local [44]. Nessa situação todos os agentes se tornam vizinhos próximos e assim algumas regras deixam de ser aplicadas e todos vão se comportar da mesma maneira.

Quando o tempo seqüencial não faz sentido de acordo com o contexto, os ABMs executam as regras do agente sequencialmente. Essa ordem pode ser: aleatória, com base em tags ou geografia; ou endógena, com os agentes que mais se beneficiam chamando suas regras de ação primeiro [44].

Geralmente o objetivo de criar um ABM é procurar informações explicativas sobre o comportamento coletivo dos agentes, sendo utilizado principalmente em sistemas naturais, em vez de resolver problemas práticos ou de engenharia específicos nos quais os modelos de Sistemas Multiagentes (MAS) são mais utilizados [45].

2.4 Síntese

O modelos baseados em agentes tem como principais características:

- Explícito ou Implícito.
- Regras simples ou compostas.
- Agentes racionais ou baseados em psicologia.
- Regras de aprendizado de agentes.

Os modelos baseados em Sistemas MultiAgentes tem como principais características:

- Agentes simples ou complexos.
- Acessibilidade.
- Determinismo.
- Dinâmica.
- Discretude.
- Episodicidade.
- Dimensionabilidade.
- Autonomia, Descentralização e visão local dos agentes.

As próximas seções apresentam um breve histórico da saúde pública no Brasil, a teoria sobre a Governança Territorial e conceitos sobre Territórios Saudáveis e Sustentáveis posteriormente modelados por MAS neste trabalho. A escolha destes temas para a realização do trabalho se deu pelo fato de ter surgido uma oportunidade de participar e acompanhar um trabalho da Fiocruz, além da vontade de explorar as oportunidades que a simulação e modelagem de eventos que traz.

2.5 Governança de Territórios Saudáveis e Sustentáveis

Atualmente há uma crescente demanda por parte da população em participar dos processos que definem as políticas públicas para estes temas. Porém isto levanta questões de como melhorar tais integrações tendo em vista as diferentes necessidades, exigências e problemas presentes nas diversas regiões do Brasil e o fato de que uma grande parte dessas interações são feitas de forma digital [46].

Com a finalidade de atender essas expectativas foi proposto o projeto de desenvolvimento de territórios sustentáveis e saudáveis. De forma que o desenvolvimento se refere ao crescimento socio-econômico das pessoas que vivem no território. Sustentáveis se refere ao fato de que os recursos naturais vão estar disponíveis para gerações futuras. Saudáveis vai além das questões de saúde e engloba os fatos de que se deve ter igualdade entre os gêneros, redução da violência e igualdade social.

Para a implantação deste modelo é necessário a realização de um estudo sobre a governança do território. Onde governança pode ser definido como nova geração de reformas administrativas e de Estado, que têm como objeto a ação conjunta, levada a efeito de forma eficaz, transparente e compartilhada pelo Estado, pelas empresas e pela sociedade civil, visando uma solução inovadora dos problemas sociais e criando possibilidades e chances de um desenvolvimento futuro sustentável para todos os participantes [47].

Desta forma é importante identificar e repensar as relações entre o Estado e Sociedade e Público e Privado a partir da perspectiva de uma gestão democrática [9]. Pois no atual modelo de gestão pública de territórios a governança local que é uma forma autônoma (*self-organizing*) de coordenação e cooperação, por meio de redes interorganizacionais, que podem ser formadas por representantes de organizações políticas e administrativas, associações, empresas e sociedades civis, com ou sem a participação estatal [47].

2.6 Transição Demográfica e as Condições Crônicas

Com o aumento da expectativa de vida e a queda de fecundidade, a população brasileira está passando por um rápido processo de envelhecimento. Nota-se que o percentual de jovens de 0 a 14 anos em 1960 era de 42% e passou para 20,87% em 2020, sendo esperado queda para 14,72% até 2060. Além disso, o percentual de pessoas com mais de 65 anos que era de 2,7% em 1960, passou para 9,83% em 2020, e deve alcançar 25,49% em 2050 [48].

Ainda sobre o percentual de pessoas com mais de 65 anos, é possível observar que o seu aumento está diretamente relacionado ao aumento da incidência de condições crônicas, uma vez que essas enfermidades tendem a afetar mais frequentemente pessoas idosas [49].

Na área da saúde, as condições crônicas abarcam aquelas circunstâncias que afetam a saúde de uma pessoa e que são persistentes no tempo, necessitando de ações contínuas e integradas por parte do sistema de saúde. Dentre essas condições, estão as doenças crônicas que são caracterizadas por sua evolução lenta e gradual, e podendo apresentar diversas causas, e que requerem uma linha de tratamento complexa, não facilmente tratadas [4], podendo durar de 3 meses ou durante toda a vida da pessoa.

Ademais, as DCNTs correspondem a 72% das causas de morte no Brasil. Além disso, dados da Pesquisa Nacional de Saúde — PNS (2013) mostram que mais de 45% da população adulta relata pelo ter menos uma DCNT [50].

As mortes por DCNT afetam predominantemente os países em desenvolvimento, nos quais cerca de um terço dos óbitos ocorrem em pessoas com menos de 60 anos de idade, enquanto nos países desenvolvidos a mortalidade prematura (faixa etária de 30 a 69 anos) corresponde a menos de 13% dos casos [50]. Dos 38 milhões de óbitos em 2012 por DCNT no mundo, mais de 40% foram prematuros e 82% dos óbitos prematuros ocorreram em países de renda baixa e média [51].

2.7 Crise de Financiamento do SUS

A crise contemporânea do SUS se apresenta, também, no que diz respeito na abordagem dada ao tratamento das doenças crônicas. O sistema de saúde brasileiro concentra-se principalmente nas doenças agudas e episódicas, e no controle de crises das doenças crônicas e seus sintomas agudos, sendo sua abordagem reativa e episódica [4].

Tal estratégia não se demonstra eficiente, pois, ao ignorar a causa por detrás do aparecimento dos sintomas, aumenta-se a possibilidade de reincidência desse sintoma, e, conseqüentemente, a uma nova procura de atendimento de saúde. Do mesmo modo, o tratamento apenas sintomático retarda o possível diagnóstico e tratamento das doenças crônicas, tendo como consequência o seu reconhecimento tardio e de tratamento mais difícil e custoso ao sistema de saúde.

2.8 Redes de Atenção à Saúde

Para contornar o problema da crise do SUS [4, 52], é necessário a implantação de Redes de Atenção à Saúde (RAS) [53]. As RASs são organizações de conjuntos de serviços à saúde que ofertam atenção contínua e integral a uma determinada população de uma

região, com o objetivo melhorar sua qualidade de vida, tornar mais eficiente a utilização dos recursos, e garantir a equidade no acesso ao sistema de saúde [4].

A estrutura operacional das RASs têm sua base na APS. A estruturação da APS é de suma importância para o estabelecimento Redes de Atenção à Saúde efetivas, eficientes e de qualidade [4].

2.9 Atenção Primária à Saúde

A APS foi definida pela Organização Mundial da Saúde em 12/09/1978 [54], com o objetivo de atingir em todos os países um nível de bem-estar físico, mental e social para os indivíduos e as comunidades.

Atenção essencial à saúde baseada em tecnologia e métodos práticos, cientificamente comprovados e socialmente aceitáveis, tornados universalmente acessíveis a indivíduos e famílias na comunidade por meios aceitáveis para eles e a um custo que tanto a comunidade como o país possa arcar em cada estágio de seu desenvolvimento, um espírito de autoconfiança e autodeterminação. É parte integral do sistema de saúde do país, do qual é função central, sendo o enfoque principal do desenvolvimento social e econômico global da comunidade. É o primeiro nível de contato dos indivíduos, da família e da comunidade com o sistema nacional de saúde, levando a atenção à saúde o mais próximo possível do local onde as pessoas vivem e trabalham, constituindo o primeiro elemento de um processo de atenção continuada à saúde (Declaração de Alma-Ata) [54].

Portanto a APS é uma estratégia para organizar a atenção a saúde de forma que esta passe a responder de forma regionalizada e contínua às necessidades de uma população, integrando ações preventivas e curativas. Através dela, espera-se atingir uma diminuição da taxa de doenças crônicas e também dos óbitos a causa dessas doenças.

A integração dos conceitos previamente mencionados, tem definido uma estratégia geral no combate às doenças crônicas, porém existem diversos outros detalhes que devem ser considerados para que sejam implementadas políticas públicas que de fato proporcionem uma melhoria na qualidade de vida da população em geral.

Com o intuito de verificar os efeitos de tais políticas, este trabalho visa a elaboração e simulação de um modelo multiagente preliminar que se assemelhe às condições encontradas na interação entre a população e o sistema de saúde do Distrito Federal.

2.10 Aplicabilidade de ABMs no campo das Políticas de Saúde

O uso de sistemas multiagente provê uma alternativa robusta e confiável para modelar e solucionar problemas reais que os sistemas de saúde dos governos vem enfrentando, sendo os mais frequentes a alta demanda por parte da população e os altos custos de manutenção das instalações, procedimentos e salários dos profissionais [55].

Atualmente, as pesquisas em aplicação de MAS à saúde estão concentradas em modelos que prevem cuidados à saúde em casa, assim diminuindo os atendimentos nas unidades de saúde e poupando recursos que são designados para casos mais graves. Tendo essa proposta em mente os modelos desenvolvidos devem possuir alguns elementos básicos como: os pacientes, sensores de diversos tipos, redes de comunicação, profissionais da saúde e as unidades de saúde.

Outra vertente que está surgindo é a utilização de um sistema multiagente com atores humanos e mecanismos adaptáveis trabalhando de forma conjunta, onde os mecanismos provêm serviços de forma especializada [56].

Capítulo 3

Plataformas de Modelagem e Simulação Baseada em Agentes

Existem diversas ferramentas para a realização de modelagem e simulação ABM, como por exemplo: netlogo [57] e Repast Symphony [58].

Para o desenvolvimento dos modelos neste trabalho foram usadas as plataformas de simulação baseadas nos softwares SocLab [3] que é uma plataforma ABMs e GAMA [13] que é uma plataforma de MAS. O soclab foi utilizado pois é um software com boa capacidade para modelar e explorar ambientes compostos de organização e hierarquização de empresas, sociedades e governos. O GAMA foi escolhido devido a possibilidade de modelar espacialmente vários aspectos do ambiente e devido a sua alta capacidade de lidar com uma grande quantidade de atores.

3.1 SocLab

O software SocLab foi desenvolvido [3] com a finalidade de prover um “laboratório virtual para pesquisadores de ciências sociais interessados em modelar e analisar interações e disputas por poder (*power relations*) dentro de organizações e comunidades, criando e analisado Sistemas de Ação Concreta (*Concrete Action Systems*)”. Foi desenvolvido na linguagem Java, e funciona no JDK 1.8. Ele pode ser obtido em [59]. A versão do SocLab utilizada neste trabalho foi a de 14 de outubro de 2016 [60].

3.1.1 Bases Teóricas

A base teórica do SocLab está fundamentada no trabalho de Crozier e Freidberg, sobre Ação Coletiva nas Organizações [17], [61].

Em [61, 17] encontram-se apresentações da teoria da Ação Coletiva nas Organizações, bem como dos conceitos subjacentes ao desenvolvimento de modelos em SocLab.

Na teoria da Ação Coletiva nas Organizações uma organização é [61, 3] uma construção social (meso estrutura de relações sociais formalizadas e não formalizadas) na qual existem relacionamentos entre os atores (agentes, indivíduos) que a constituem. Adicionalmente, os atores agem em relação aos outros usando modelos de racionalidade limitada [61, 3]. Nessa situação, a Teoria possibilita o estudo de modelos neo-estruturais de tradição sociológica nos quais a estrutura social influencia os comportamentos individuais, e os comportamentos individuais influenciam a evolução da estrutura social.

3.1.2 Conceitos Subjacentes aos modelos no SocLab

Modelos desenvolvidos usando o SocLab são exemplificados nos trabalhos de [62] e [63], entre outros.

O objetivo de uma modelagem MAS em SocLab é a obtenção de uma condição estrutural nas relações entre atores de uma organização na qual o grau de satisfação e influência dos atores seja maximizado, enquanto preservando as relações entre os mesmos dentro de parâmetros estabelecidos dentro de limites quantitativamente definidos, possibilitando assim a melhor coordenação da ação coletiva.

Meta-modelo básico do SocLab

O meta-modelo básico de qualquer modelo de organização no SocLab é definido na Figura 3.1.

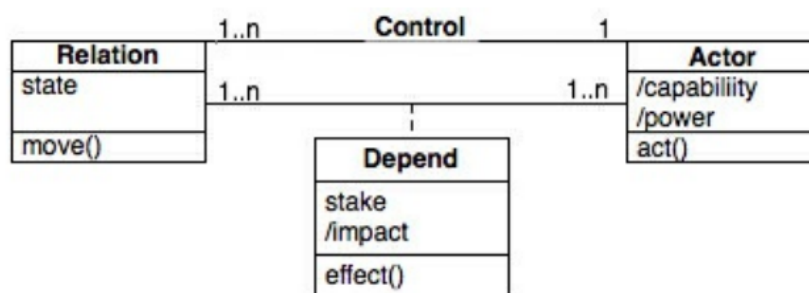


Figura 3.1: Meta-modelo básico de uma organização modelada no SocLab.

No SocLab, uma organização é modelada através da definição de um conjunto de atores, individualmente definidos por sua capacidade e poder. A capacidade de um ator reflete a sua liberdade de ação no modelo. O poder de um ator reflete o sucesso desse ator na distribuição de capacidades para os outros atores com os quais se relaciona, no conjunto das redes de relações que estabelece.

Cada ator numa organização se relaciona com os outros por meio de relações sobre as quais um dos atores exerce controle. Cada um dos dois atores na relação exibir certa dependência (*stake*) perante essa relação, a qual pode ser expressa por meio de stakes (interesses em jogo na relação) ou impactos (decorrentes do exercício da relação). Quando atores agem eles podem exercitar as relações (*move*) sobre as quais detém algum controle, distribuindo capacidades que alteram o estado das relações. Cada relação, então, possui um estado, e pode ser exercitado (*move*) de forma mais ou menos cooperativa dentro de um limite numérico que vai de -10 (menos cooperativo) a +10 (mais cooperativo), enquanto que o valor zero representa relações que são neutras [3, 6].

Quando uma relação é exercitada ela provoca efeitos sobre os atores que dela dependem e que apresentam interesses em jogo (*stakes*). Esses efeitos, gerados a partir do exercício das múltiplas relações que um ator estabelece com os demais, alteram a capacidade de ação dos atores, bem como o poder que eles exercem sobre os outros atores.

A estrutura social é definida por meio de relações que os atores estabelecem uns com os outros.

Construção de modelos no SocLab

Na construção de um modelo no SocLab, as seguintes regras se aplicam:

- Atores podem ser indivíduos ou agentes coletivos (grupos, organizações, populações);
- Cada ator precisa ter controle sobre pelo menos uma relação, posto que é um ator, e não apenas um elemento passivo;
- Cada ator precisa ter alguma dependência (*stake*) perante as relações que estabelece, e sobre essa dependência o ator sofre um impacto, modulado por uma função ou curva de efeito (*effect*);
- O exercício iterado, isso é, ao longo do tempo, do controle e dependência de um ator sobre as suas redes de relações, busca trazer para o ator os recursos dos quais ele necessita para alcançar sua meta;
- Cada ator em um modelo SocLab possui 10 pontos de dependências (*stakes*) para distribuir no conjunto de suas relações.

Metamodelo refinado do SocLab

De modo subjacente à construção de um modelo no SocLab, o metamodelo refinado apresentado na Figura 3.2 apresenta não só os atributos adicionais que podem ser acrescentados aos atores, relações e restrições, mas também os elementos da interpretação analítica

que devem ser investigados após a obtenção dos resultados de simulações de sistemas multiagentes no SocLab.

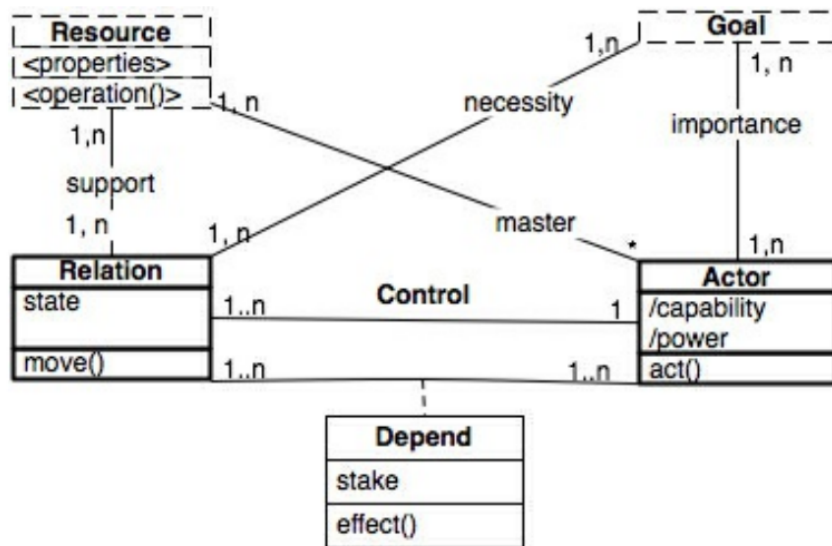


Figura 3.2: Meta-modelo subjacente para interpretação dos resultados de uma simulação de ação organizacional modelada no SocLab.

As metas e recursos apresentadas no modelo da Figura 3.2 não são explicitamente modeladas no SocLab, mas representam, do ponto de vista fenomenológico, que as relações estabelecidas pelos atores visam o acesso desses atores a recursos que são controlados por quem controla a relação, não modelados no SocLab. Esses recursos são necessários para que o ator alcance seus objetivos, também não modelados de forma explícita no SocLab.

De forma prática, embora não modelados no SocLab, recursos poderiam ser [3, 6] “objetos físicos, entidades cognitivas provendo informação factual ou procedural, expectativas, conhecimento, serviços, atitudes etc”. Os recursos aos quais um ator tem acesso, seja os que já possui, ou os que tem porque os obtém em decorrência de relações estabelecidas com outros atores, são a fonte de poder dos atores [3, 6]. Conforme [3, 6], toda relação é um relacionamento de poder desbalanceado. Os atores usam seu poder para alcançar suas metas, ou satisfazer seus interesses.

Meta-modelo completo do SocLab

O meta-modelo completo de qualquer modelo de organização no SocLab é definido na Figura 3.3.

No SocLab, uma organização é modelada de forma plena através da definição de um conjunto de atores, individualmente definidos por sua capacidade, poder, satisfação e influência.

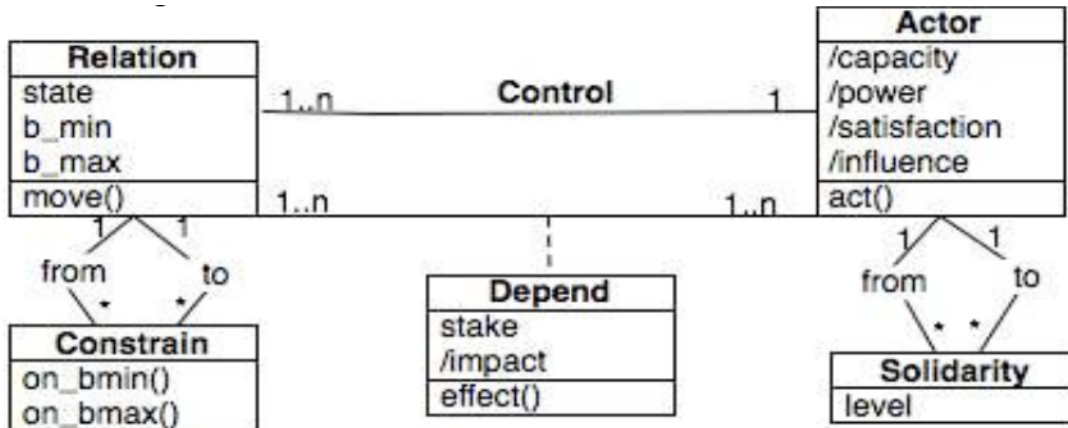


Figura 3.3: Meta-modelo completo de uma organização modelada no SocLab.

A estrutura social já definida pelas relações que os atores estabelecem uns com os outros é complementada por relações de solidariedade, cujos níveis são imutáveis do ponto de vista do modelo.

Como já dito, a capacidade de um ator reflete a sua liberdade de ação no modelo. O poder de um ator reflete o sucesso desse ator na distribuição de capacidades para os outros atores com os quais se relaciona. Adicionalmente, a satisfação de um ator em relação a outro decorre do produto da solidariedade desse ator em relação ao outro, multiplicada pelo interesse (*stake*) em jogo na relação, e pela curva de efeito a ser aplicada nessa situação de dependência do outro ator perante a relação em pauta. A influência de um ator perante o outro corresponde à satisfação desse ator perante o outro nas situações em que o primeiro ator controla a relação.

Ao agir, um ator opera sobre suas relações estabelecidas com os demais atores. O objetivo da ação é maximizar sua meta. Cada relação pode ser ajustada dentro de uma faixa de limites máximos e mínimos de valoração na estrutura, sendo que existem restrições entre as relações, disparadas quando o estado de uma relação atinge certo limite inferior (relação não cooperativa) ou superior (relação cooperativa).

Criação de modelos no SocLab

As relações entre atores são consideradas colaborações. Ou seja, os atores no modelo de uma organização agem movidos por interesses ou metas, que podem ser explícitas ou implícitas, e buscam alcançar essas metas obtendo acesso a recursos.

Os passos recomendados por [3] para a construção de modelos no SocLab são os seguintes:

1. Identificação de atores relevantes;

2. Levantamento das relações presentes na organização, e dos pares de atores que dela participam;
3. Levantamento dos interesses (stakes) que cada ator tem na relação;
4. Interpretação dos estados de cada relação, em relação ao comportamento do ator que a controla;
5. Escolha da função de efeitos que melhor se aplica à relação, seja de forma linear, sigmóide, gaussiana etc;
6. Definição de restrições entre relações;
7. Definição das solidariedades entre pares de atores.

Simulação de modelos no SocLab

A simulação de modelos no SocLab consiste na aplicação iterativa de regras presentes numa base de regras de cada ator individual, cuja execução altera o estado da relação entre atores, usando as capacidades disponíveis no ator, visando obter-se uma situação estacionária (convergência do modelo) no qual a satisfação global dos atores é maximizada.

Essa aplicação ocorre por meio da utilização de uma base de regras mantidas no estado interno de cada agente, constantemente atualizada por um algoritmo de aprendizagem por reforço. Nessa base, cada regra é formada por uma tripla da forma apresentada na equação 3.1 [3, 14].

$$regra ::= \langle \textit{situação}, \textit{ação}, \textit{força} \rangle \quad (3.1)$$

Uma *situação* é um vetor com todos os impactos recebidos dos outros atores com os quais o ator interagiu, em uma interação anterior. Uma *ação* é um vetor de mudanças no estado das relações controladas por um agente, que será aplicado quando a atual situação da simulação for próxima da *situação* estabelecida na regra; Uma *força* é uma avaliação da efetividade dessa regra.

Ainda antes de execução da simulação, o SocLab permite a definição de parâmetros psico-cognitivos para os atores, que são específicos para a simulação a ser executada, e que são:

Scope , que determina com que autonomia um ator vai buscar na base de regras aquela mais aproximada à situação da simulação, sendo que se tiver o valor 1 (mínimo), então qualquer regra é aplicável em qualquer situação. Se o escopo (autonomia) tiver valor 5 (máximo) então a base de regras do ator será expandida para representar um grande número de situações;

Tenacity , que determina com que grau de esforço ou tenacidade o ator vai explorar o espaço de estados da simulação, visando gerar a *ação* que melhor se aplica à *situação* encontrada, visando alcance de sua meta individual;

Group Identification , que determina com que grau o ator se identifica com a organização simulada, e assim atribui maior peso à sua influência sobre os demais atores;

Repartition of Reward , que determina a porcentagem de pontos que o ator distribui quando aumenta sua satisfação em relação à interação anterior.

Type of Rules , que indica que o ator vai aplicar um algoritmo de aprendizagem por reforço, para recompensar as regras que resultaram em maior ganho de pontos, ou se será usada uma ponderação fixa para cada regra criada.

A simulação se inicia com todos os atores definindo internamente o seu grau esperado de alcance da meta com um valor máximo, sendo que esse valor se inicia com 100, e vai sendo reduzido visando a convergência do modelo. O grau atual de alcance da meta de um ator é definido pela soma do grau de satisfação [3, 7] e o grau de influência desse ator sobre os demais, ponderado pelo grau de identificação do ator com a organização simulada.

3.1.3 Dilema do Prisioneiro

Para verificar a adequação do SocLab aos requisitos desejados foi desenvolvido um modelo inicial baseado no Dilema do Prisioneiro, que foi originalmente formulado em 1950 por Merrill Flood e Melvin Dresher que tem o seguinte funcionamento:

Dois suspeitos, A e B, são presos pela polícia. A polícia tem provas insuficientes para os condenar, mas, separando os prisioneiros, é oferecida a ambos o mesmo acordo: se um dos prisioneiros, confessando, testemunhar contra o outro e esse outro permanecer em silêncio, o que confessou sai livre enquanto o cúmplice silencioso cumpre 10 anos de sentença. Se ambos ficarem em silêncio, a polícia só pode condená-los a 6 meses de cadeia cada um. Se ambos traírem o comparsa, cada um leva 5 anos de cadeia. Cada prisioneiro faz a sua decisão sem saber que decisão o outro vai tomar, e nenhum tem certeza da decisão do outro. A questão que o dilema propõe é: o que vai acontecer? Como os prisioneiros vão reagir?

A simulação de modelos no SocLab nos permite obter respostas para estas perguntas, de forma que o resultado esperado é a convergência do modelo para um cenário de equilíbrio onde todos os atores alcancem os seus objetivos sem prejudicar os demais.

3.1.4 Um exemplo de caso de modelagem e simulação de uma agência de viagem

A distribuição padrão do SocLab vem acompanhada de alguns modelos de simulação já prontos, dentre os quais se destaca um modelo simples composto por dois atores, situado no âmbito de uma situação de conflito que ocorre numa agência de viagens com dois escritórios. O modelo se mostra útil para familiarizar o leitor com as metodologias de criação de modelos, simulação e interpretação de dados, bem como com a interface gráfica do software.

O modelo exemplo encontra-se no arquivo `TravelTour.org`, no diretório `Models` da distribuição do SocLab usada.

O caso da contratação de Agnes

A apresentação a seguir é adaptada de [64].

Travel-tour é uma empresa que opera com duas agências, TRO1 e TRO2, numa mesma cidade.

Agnes é uma pessoa que atua como secretária com contrato temporário em tempo parcial em TRO1, e que passa uma parte do seu horário em TRO2, movimentando-se entre as duas agências. Devido à melhoria dos resultados de TRO1, o alto executivo da empresa propõe formalizar o contrato de Agnes e deixá-la trabalhando fixa apenas em TRO1. Isso daria ao Diretor de TRO1 uma secretária com dedicação total, evitando também que ela tenha que se deslocar entre as agências, o que lhe pareceu um ajuste bem atrativo para ambas as partes.

Surpreendentemente, tanto Agnes quanto o Diretor recusaram a proposta do alto executivo da empresa, e uma análise da situação revela que ambos tem bons motivos para recusar a proposta, e que são:

- A agência TRO2 é mais criativa em seu trabalho, embora TRO1 tenha uma equipe mais produtiva. Sabendo disso, Agnes traz para o Diretor em TRO1 as ideias que vê sendo desenvolvidas em TRO2, as quais são implementadas de modo mais efetivo por TRO1, alavancando seu desempenho;
- Agnes não tem interesse em ficar com um emprego formalizado e em tempo integral, ela prefere a renovação periódica do contrato em tempo parcial;
- Agnes aprecia que nem o Diretor de TRO1, nem o de TRO2 possam controlar o seu trabalho, tendo em vista sua dupla atividade e deslocamento entre as agências.

- Se Agnes ficar trabalhando em tempo integral em TRO1, o Diretor dessa agência perderá acesso às informações de TRO2, e Agnes passará a ser mais controlada no emprego.

De que forma um modelo no SocLab conseguiria simular essa situação, gerando resultados que representariam com boa fidelidade o melhor arranjo de ação entre as partes envolvidas?

Um modelo SocLab para representar a situação de Agnes e do Diretor de TRO1

A Figura 3.4, gerada pela interface do usuário do SocLab, apresenta o grafo das relações entre os atores do Caso na Travel-tour.

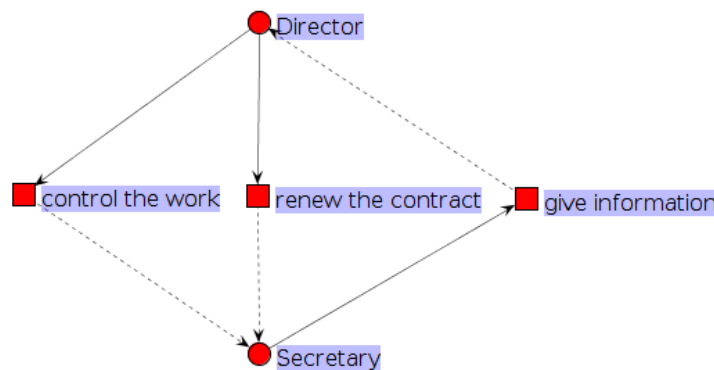


Figura 3.4: Rede que modela a relação entre a Secretária Agnes e seu Diretor.

Dois atores, Secretaria (Secretary) e Diretor (Director) compartilham três relações entre si, sendo que (i) o controle do trabalho (control the work) e a (ii) renovação do contrato (renew the contract) são controladas pelo Diretor, enquanto que a relação (iii) prover informação sobre TRO2 (give information) é controlada pela Secretária.

A fim de permitir uma interpretação dos resultados gerados pelo modelo, foram pré-definidos os seguintes parâmetros de interpretação para cada uma das relações:

give information , representa uma situação na qual a Secretária passa para o Diretor informação sobre os novos produtos em desenvolvimento em TRO2, que é a outra agencia da empresa Travel-tour. Se após a convergência da simulação give information tiver o seu state numa faixa entre -10 e -1 (faixa de valores que indica comportamento não cooperativo), representa que estará havendo desinformação (informação equivocada ou errada) da Secretária para o Diretor; Se o state de give information estiver na faixa de -1 a 2, significa que nenhuma informação estaria sendo repassada;

Se o state da relação estiver na faixa de 2 a 10 (faixa de valores que indica comportamento cooperativo), isso significa que está sendo passada informação satisfatória para o Diretor;

control the work , se após a convergência da simulação, tiver seu state numa faixa entre -10 e -2 isso significa que não está havendo cooperação entre o Diretor e a Secretária, sugerindo que o Diretor está controlando o trabalho da Secretária;

renew the contract , se após a convergência da simulação, estiver com seu state numa faixa entre -10 e -3 (comportamento não cooperativo), indica que a renovação do contrato poderá ser adiada, gerando preocupação na Secretária; se estiver na faixa de -3 a 3 sugere neutralidade do ponto de vista do Diretor; se estiver na faixa de +3 a +10 significa que a renovação do contrato será feita com facilidade.

A Figura 3.5 apresenta o local, na interface gráfica do SocLab, onde devem ser inseridas informações de subsídio à interpretação e análise dos resultados das simulações do modelo.

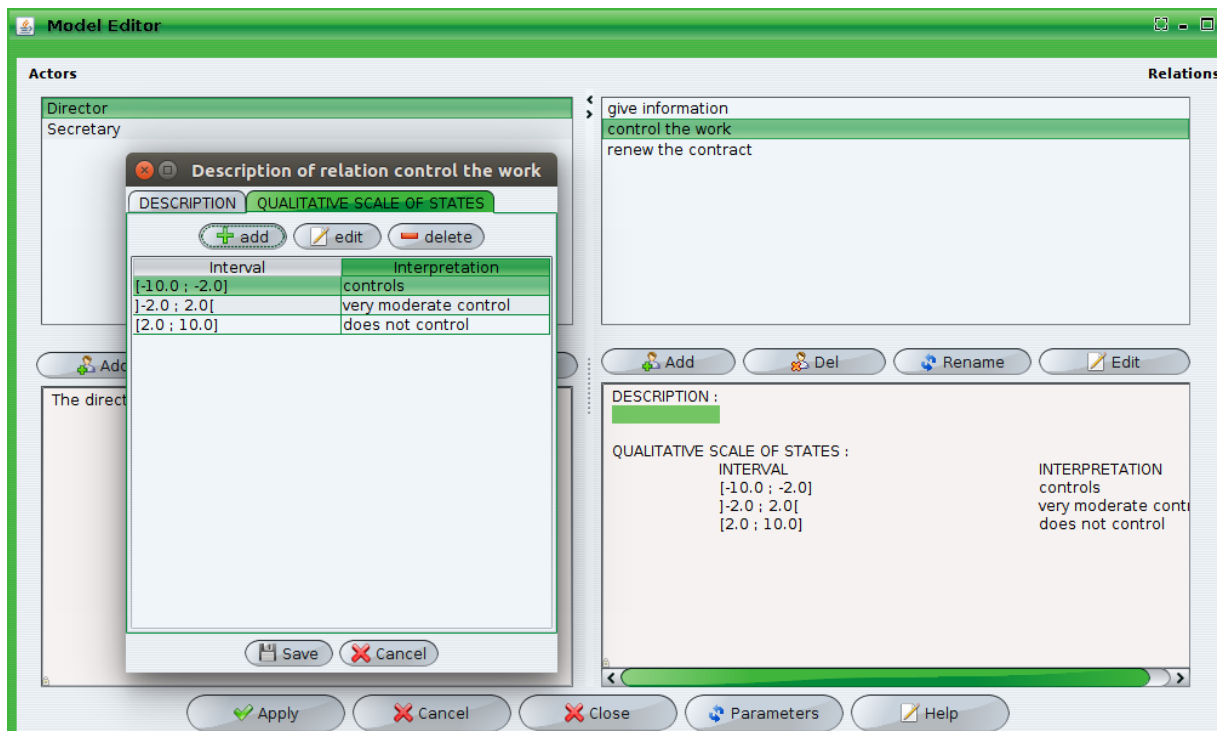


Figura 3.5: Documentação de um modelo desenvolvido no SocLab .

A descrição textual da relação e as escalas quantitativas precisas que permitem a interpretação dos intervalos são formalizadas na própria documentação do modelo, e auxiliam na interpretação dos resultados pós-simulação.

Definido interesses e efeitos sobre relações

A Figura 3.6 resume os valores para os interesses (stakes) e a função de efeitos definidos para cada uma das relações entre os dois atores.


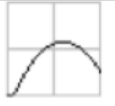
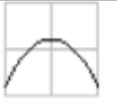

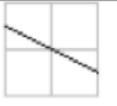

	Stakes		Effect functions	
	Director	Secretary	Director	Secretary
Give information	7.0	2.0		
Control the work	2.0	6.0		
Renew the contract	1.0	2.0		

Figura 3.6: Stakes e effects definidos para as relações e atores no modelo Travel-tour.

Os valores dos stakes indicam o interesse dos correspondentes atores na relação, estando grafado em negrito o destaque para o ator que controla a relação. Ou seja:

- A Secretária controla a relação give information, mas seu interesse na relação é casual.
- O Diretor controla as duas outras relações, e também tem interesse casual nas mesmas;
- O Diretor está mais interessado na relação give information, por meio da qual obtém os recursos dos quais necessita para alcançar seus objetivos (saber quais pacotes inovadores estão sendo discutidos na agência TRO2 da Travel-tour);
- A Secretária está mais interessada em não ter o seu trabalho controlado, e não tem muita preocupação com a renovação do contrato.

As formas do gráfico das funções de efeito indicam de que forma um exercício mais cooperativo da relação (metade direita de cada gráfico), ou menos cooperativo da relação (metade esquerda de cada gráfico) afeta as capacidades do correspondente ator. A título de exemplo:

- Na relação give information, quanto maior foi a cooperação na relação, melhor isso atende aos interesses do Diretor. Já para a Secretária, um excessivo exercício da cooperação nessa relação pode fazer com que a sua posição em TRO2 seja prejudicada;

- Na relação control the work, o diretor quer manter-se neutro em relação a quão cooperativo é o controle, mas para a Secretária ela tem interesse em uma maior cooperação;
- Na relação renew the contract, o Diretor deseja uma postura menos cooperativa na relação, enquanto que a Secretária busca maior cooperação, embora ambos tenham poucos interesses (stakes) em jogo.

3.1.5 Simulando um modelo no SocLab

A Figura 3.7 apresenta a janela de exploração de simulações do SocLab, com foco na definição do estado inicial do modelo Travel-tour.

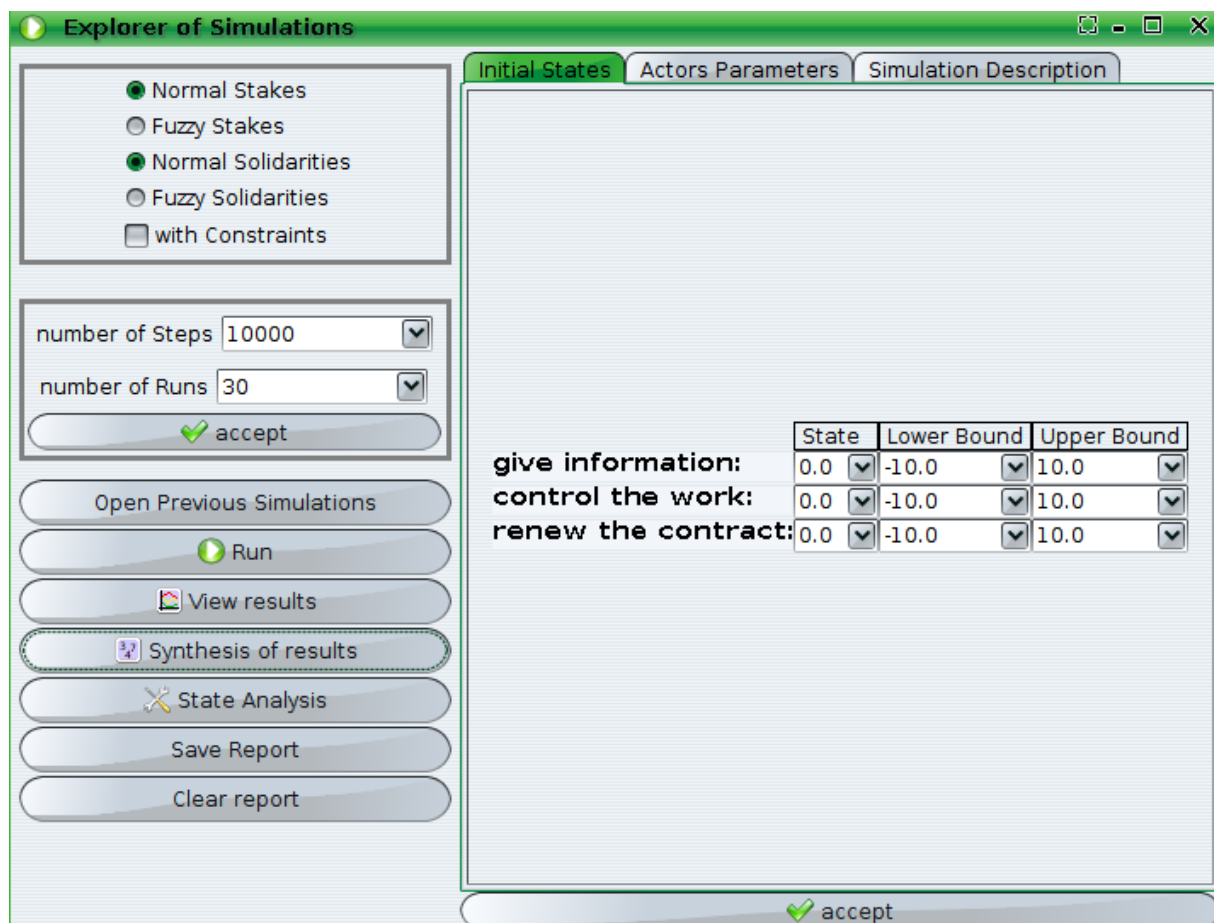


Figura 3.7: Painel de simulação do SocLab, para o modelo Travel-tour.

Conforme apresenta a Figura 3.7, o valor 0 (zero) é atribuído ao estado inicial das três relações, com os limites inferior (-10, relação não cooperativa) e superior (+10, relação cooperativa) para as relações definidos conforme o padrão.

À esquerda da janela, a quantidade de interações (runs) está definida em 30, e para cada uma das interações, são executados até 10.000 passos de exploração do espaço de estados pelos agentes, na busca da aplicação da melhor regra.

A simulação é executada acionando-se o botão run, e, após a conclusão da simulação, seus resultados são armazenados em uma pasta no sistema de arquivos, e uma janela síntese dos resultados da simulação pode ser apresentada.

A Figura 3.8 apresenta uma síntese de uma simulação com 100 runs, e 10.000 passos por run, como pode ser observado na parte superior da Figura (Simulation Parameters).

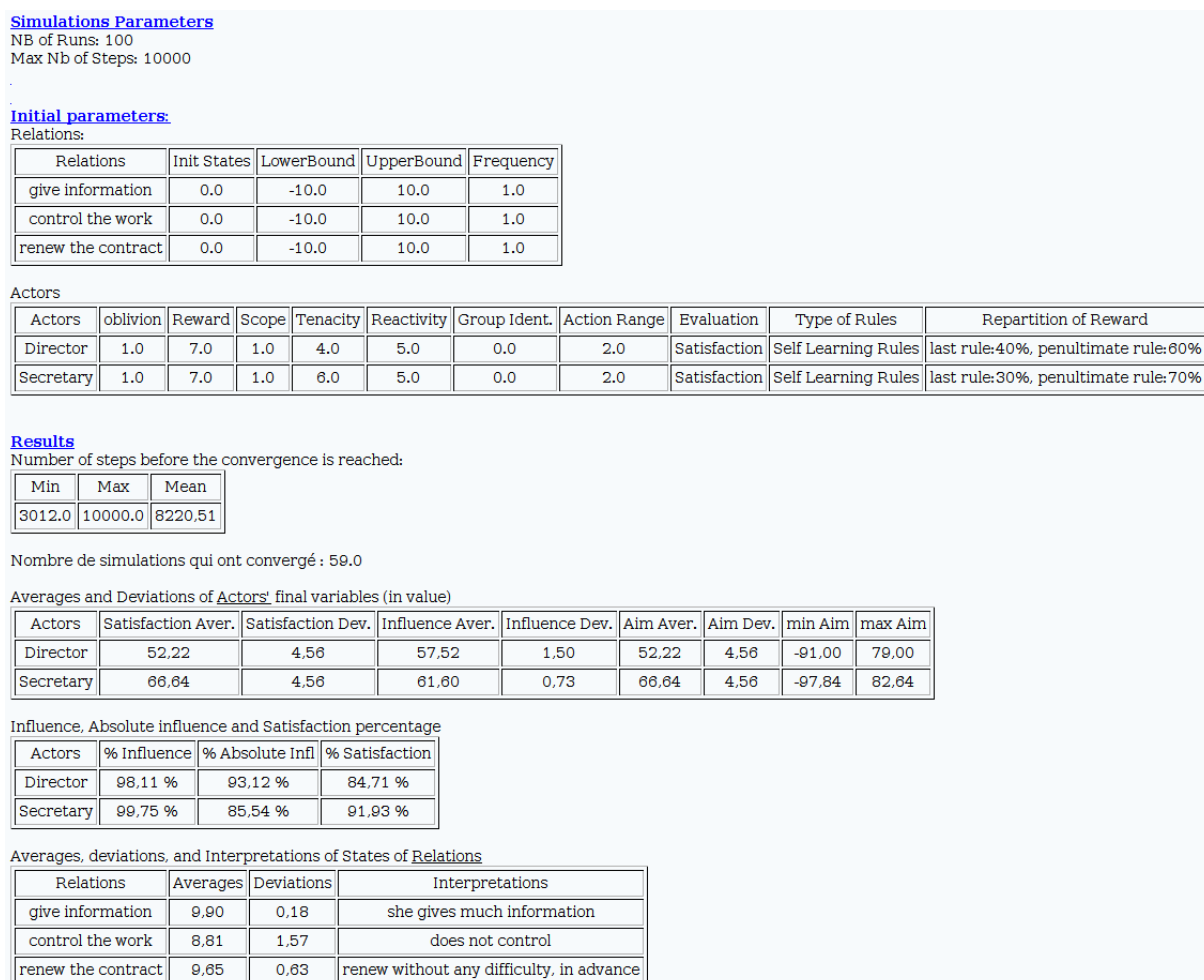


Figura 3.8: Síntese de simulação do modelo Travel-tour.

Os parâmetros iniciais da simulação (Initial Parameters) prescrevem que a Secretária deveria adotar mais tenacidade na busca por regras que maximizem suas metas (valor 6). Ambos atores tiveram como nula a sua identificação com o grupo, e trabalharam com regras de auto-aprendizagem por reforço.

Os resultados possibilitam a interpretação dos valores obtidos, que devem ser confrontados com a realidade observada no caso, que é de recusa de ambos os atores (Diretor e Secretária) em seguir a ação sugerida pelo executivo chefe.

As médias e desvios padrão das variáveis satisfação e influência, determinantes do alcance dos objetivos dos atores, tiveram valores aproximadamente iguais, sendo que os da secretária são ligeiramente maiores que os do diretor.

No que se refere aos estados finais das relações, a média e desvio padrão dos estados das relações *give information*, *control the work* e *renew the contract* indicam que os dois atores adotariam comportamento cooperativo, no qual:

- A Secretária se ajusta a passar muita informação para o Diretor;
- O Diretor não controla o trabalho da Secretária;
- O contrato temporário da secretária será renovado de forma antecipada.

O exemplo apresentado apresenta de forma bastante simples a potencial utilidade do SocLab na modelagem de um sistema de atores, onde interações sociais de nível micro-sociológico explicitamente definidas por relacionamentos diversos entre um conjunto de atores possibilitam explicação de fenômenos sociais cooperativos, como os observados no caso indicado.

3.2 GAMA

o GAMA é um ambiente de desenvolvimento de código aberto para modelagem e simulação espacialmente explícitas baseadas em agentes. GAMA foi projetado com o objetivo de apoiar a definição de modelos complexos com um forte suporte da dimensão espacial. A plataforma GAMA é um ambiente de desenvolvimento integrado (IDE) para a construção de modelos baseados em agentes espacialmente explícitos, especialmente focado na modelagem de sistemas socioambientais e utiliza a linguagem de programação GAML (GAMA Modeling Language) que é altamente orientada a agentes, o que significa que tudo "ativo"(entidades de um modelo, sistemas, processos e atividades) pode ser representado no GAML como um agente.

Este software é mais recomendado para a realização de análises nos níveis meso e macro devido a sua característica de modelagem espacial, sendo que não é ideal para modelar relações de poder e iteração entre atores em pequenas organizações(nível micro).

3.2.1 Conceitos Subjacentes aos modelos no GAMA

Como no paradigma orientado a objetos, onde a noção de classe é usada para fornecer uma especificação para objetos, os agentes em GAML são especificados por suas espécies, que lhes fornecem um conjunto de atributos (o que eles sabem), ações (o que eles podem fazer), comportamentos (o que eles realmente fazem) e também especifica propriedades de sua população, por exemplo, sua topologia (como estão conectados).

Qualquer espécie pode herdar habilidades, atributos e ações de outra espécie (espécie-mãe), criando uma relação semelhante à especialização do paradigma orientado a objetos.

Meta-Modelo básico do GAMA

As relações entre espécies, modelos e planos de experimentos são codificados no meta-modelo do GAML na forma de um quadro composto por três espécies abstratas, respectivamente chamadas de *agent* (pai direto ou indireto de todas as espécies), *model* (pai de todas as espécies que definem um modelo) e *experiment* (pai de todas as espécies que definem um plano de experimento).

Neste meta-modelo, as instâncias dos filhos de *agent* conhecem a instância do filho de *model* em que estão inseridos como seu mundo, enquanto a instância de *experiment* identifica o mesmo *agent* como uma das simulações sob sua responsabilidade. O diagrama a seguir resume essa estrutura:

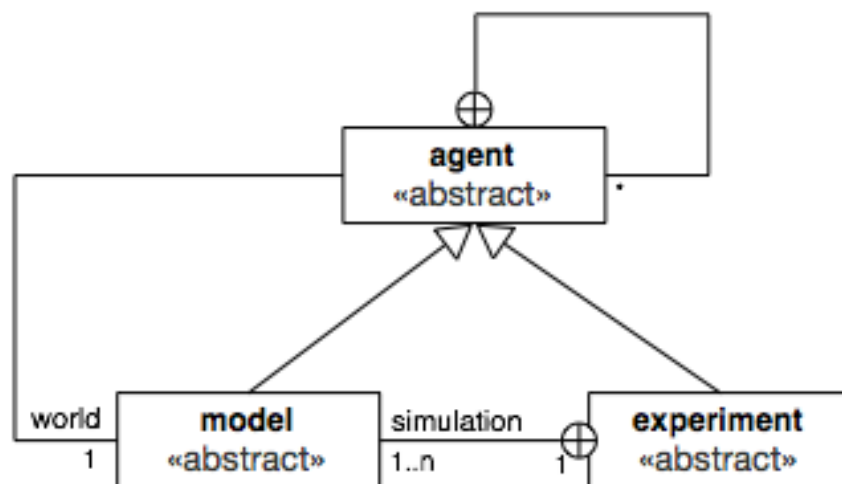


Figura 3.9: Meta-modelo GAML.

Portanto para escrever um modelo em GAML é necessário definir uma espécie que herda de *model*, espécies que herdaram de *agent* (representam as entidades que povoam o modelo) e os planos de *experiment* que serão executados.

Ao executar um experimento no GAMA, um agente de *experiment* é criado, de forma que este irá criar agentes de simulação (instância de *model*) e executá-los de acordo com

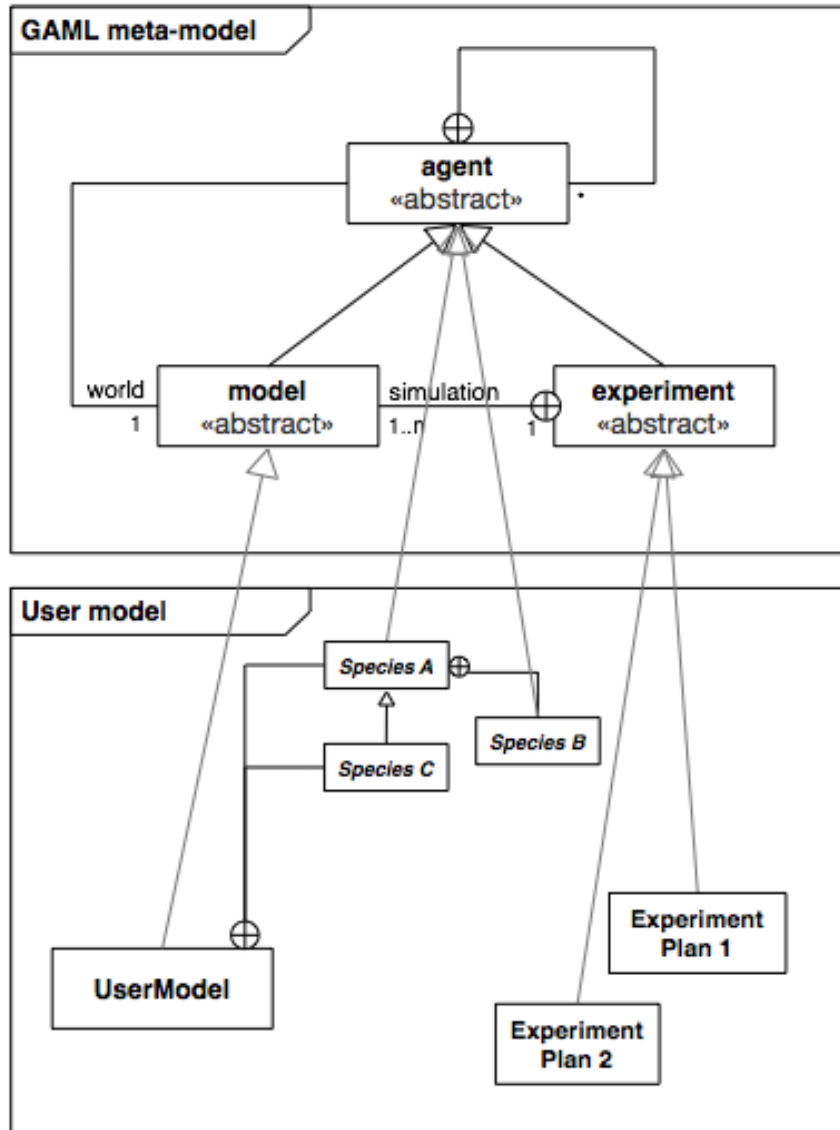


Figura 3.10: Meta-modelo Completo do GAML.

o comportamento definido em *experiment*. Recursivamente, a inicialização de um agente de simulação criará a população de *agent* das espécies definidas no modelo, onde cada um desses agentes pode criar a população de suas micro-espécies.

Em resumo, a criação de um modelo em GAML equivale a definir uma espécie do modelo, que mais tarde permite instanciar um agente do modelo (também conhecido como simulação), que pode ou não conter micro-espécies e contém um ou mais planos de experimentos para serem simulados.

3.2.2 Criação de modelos no GAMA

A seguir é explicado o formato padrão que um modelo desenvolvido em GAML segue, também é mostrado trechos de códigos exemplificando como é feito a definição do modelo.

Definição da espécie global para representar a espécie modelo

Todo modelo começa com a declaração do *header*, onde 3.1 contém o nome do modelo e 3.2 é a definição da espécie global, que é a definição da espécie modelo, a espécie global não tem nome e é única em seu modelo, nela são declarados todos os atributos / ações / comportamentos globais.

```
1 model name_of_the_model
```

Listing 3.1: Declaração do nome da espécie global(*model*)

```
1 global {  
2 // definition of global attributes, actions, behaviors  
3 }
```

Listing 3.2: Declaração da espécie global(*model*)

Definição das diferentes micro-espécies(espécies regulares)

Após a definição do *header* é declarado e definido todas as espécies de *agent* que iram popular o modelo.

As espécies regulares de *agent* podem ser declaradas com a palavra reservada *species*. Pode-se declarar várias espécies regulares, mas todas elas devem ser nomeadas e cada espécie define seus atributos, ações e comportamentos e aspectos.

```
1 species nameOfSpecies {  
2 // definition of your species attributes, actions and behaviors and  
3 aspects  
4 }
```

Listing 3.3: Declaração de espécie regular(*agent*)

Note porém, que a declaração de espécies regulares pode ser feita após ou dentro da declaração da espécie global, de tal forma que 3.4 é equivalente a 3.5.

```
1 global {  
2 // definition of global attributes, actions, behaviors  
3 }  
4  
5 species A {...}  
6
```

```
7 species B {...}
```

Listing 3.4: Primeiro local de declaração das espécies regular

```
1 global {  
2     // definition of global attributes, actions, behaviors  
3  
4     species A {...}  
5  
6     species B {...}  
7 }
```

Listing 3.5: Segundo local de declaração das espécies regular

Definição dos diferentes planos de experimentos para o modelo

Os experimentos geralmente são declarados no final do arquivo e começam a utilização da palavra reservada *experiment*. Pode-se declarar quantos experimentos forem necessários onde cada um contém *inputs*(os parâmetros de uma simulação) e os *outputs* (saídas tais como telas, monitores ou inspetores).

```
1 experiment first_experiment {  
2     // definition of parameters (inputs)  
3  
4     // definition of output  
5     output {...}  
6 }  
7  
8 experiment second_experiment {  
9     // definition of parameters (inputs)  
10  
11    // definition of output  
12 }
```

Listing 3.6: Declaração de espécie *experiment*

Existe 4 tipos de espécie de *experiment*, sendo elas:

- *Bach experiment* que permite a execução de sucessivas simulações (geralmente usadas para exploração de modelo) e é declarado conforme 3.7

```
1 experiment batch_experiment type:batch {  
2     [parameters]  
3     [exploration method]  
4     [...]  
5 }
```

Listing 3.7: Declaração de *Bach experiment*

- *Test experiment* permite que a realização testes de unidade em um modelo (usado para garantir sua qualidade). É declarado conforme 3.8

```

1 experiment test_experiment type:test autorun: true {
2     [setup]
3     [tests]
4     [...]
5 }

```

Listing 3.8: Declaração de *Test experiment*

- *Memorize experiment* permite o armazenamento de cada etapa da simulação na memória e o retorno às etapas anteriores. É declarado com a seguinte estrutura conforme 3.9

```

1 experiment test_experiment type:memorize {
2     [parameters]
3     [output]
4     [...]
5 }

```

Listing 3.9: Declaração de *Memorize experiment*

- *Gui experiment* permite a exibição de uma interface gráfica com parâmetros de entrada e saídas. Esse foi o tipo de experimento utilizado neste trabalho. É declarado conforme 3.10

```

1 experiment gui_experiment type:gui {
2     [parameters]
3     [output]
4     [...]
5 }

```

Listing 3.10: Declaração de *Gui experiment*

Código base de um modelo GAMA

Após a junção dos elementos definidos acima é já possível se ter a definição básica do modelo a ser construído, conforme 3.11.

```

1 model name_of_the_model
2
3 global {
4     // definition of [global attributes](GlobalSpecies#declaration),
5     actions, behaviours
6 }

```

```

7 species my_specie {
8     // definition of attributes, actions, behaviors
9 }
10
11 experiment my_experiment /* + specify the type : "type:gui", "type:batch
12     ", "type:test", or "test:memorize" */
13 {
14     // here the definition of your experiment, with...
15     // ... your inputs
16     output {
17         // ... and your outputs
18     }
19 }

```

Listing 3.11: Declaração básica de modelo GAMA

Definindo características da espécie global

A espécie global é uma espécie como as outras e pode ser manipulada como elas, tendo como diferença que há somente um agente mundial por simulação, sendo criado quando um usuário executa um experimento e é responsável por inicializar a simulação por meio de seu escopo *init*.

Após a declaração de todos os atributos globais e definir o tamanho do ambiente do modelo, pode-se definir um estado inicial (antes de lançar a simulação), o qual inicializa as variáveis globais e instancia das espécies. A seguir é apresentado alguns atributos globais:

- *Environment size*: É possível definir o tamanho e a forma do ambiente de simulação através do comando *shape* que é do tipo *geometry*. Por padrão o ambiente é um quadrado com 100m de lado.

```

1     geometry shape <- circle(50#mm);
2     geometry shape <- rectangle(10#m,20#m);
3     geometry shape <- polygon([1#m,2#m},{3#m,50#cm},{3.4#m
4     ,60#dm}]);

```

Listing 3.12: Exemplos de alteração do formato do ambiente de simulação

- *cycle*: Designa o número de execuções dos ciclos da simulação, aceitando apenas números inteiros e com o primeiro ciclo denominado de 0.
- *step*: É a duração de um intervalo entre dois ciclos, em segundos. Seu valor padrão é 1 (segundo). A cada ciclo o valor do tempo é incrementado pelo valor de *step*,

gerando assim o tempo total de simulação. A definição de *step* deve ser coerente com as variáveis dos agentes, como velocidade.

```
1     global { ...
2         float step <- 10 #h;
3     }
4
```

Listing 3.13: Declaração de *step*

- *time*: Representa o tempo simulado atual em segundos (a unidade padrão). É a hora na hora do modelo, onde o tempo é calculado pela quantidade de *cycle* multiplicado pelo valor de *step*.

```
1     global { ...
2         int nb_minutes function: { int(time / 60)};
3         ...
4     }
5
```

Listing 3.14: Utilização de *time*

- *starting_date* e *current_date*: Representam a data de início e a data atual da simulação. A *current_date* é atualizada a partir da *starting_date* pelo valor de *step* em cada ciclo da simulação.
- *agents*: Retorna uma lista de todos os agentes do modelo que são considerados "ativos" (ou seja, todos os agentes com comportamentos, excluindo os locais). Observe que obter essa lista pode ser bastante demorado, pois o mundo tem que passar por todas as espécies e obter seus agentes antes de montar o resultado.
- *pause*: Pausa a simulação, que pode então ser continuada pelo usuário.

```
1     global { ...
2         reflex toto when: time = 100 {
3             do pause;
4         }
5     }
6
```

Listing 3.15: Utilização de *pause*

- *die*: Encerra a simulação.

```
1     global { ...
2         reflex halting when: empty (agents) {
3             do die;
4         }
5     }
6
```



```

4         }
5     }
6

```

Definição *gui experiment*

Na execução da simulação muitas vezes é necessário mostrar algumas informações, tais como entradas, saídas e comportamentos.

- As entradas serão compostas por parâmetros manipulados pelo usuário para cada simulação.
- Os comportamentos serão usados para definir o comportamento executado em cada etapa do experimento.
- As saídas serão compostas por displays, monitores. Eles serão definidos dentro da saída do escopo. A definição de seu layout também pode ser definida com a declaração de layout.

```

1  experiment exp_name type: gui {
2      [input]
3      [behaviors]
4      output {
5          layout [layout_option]
6          [display statements]
7          [monitor statements]
8      }
9 }

```

Listing 3.16: Organização padrão *gui experiment*

A definição de parâmetros(*input*) permite tornar o valor de uma variável global definível pelo usuário por meio da interface gráfica. Ao iniciar a simulação, o parâmetro aparecerá no painel "*Parameters*", com o nome que escolhido para identifica-lo.

```

1  global {
2      int my_integer_global_value;
3  }
4
5  experiment MyExperiment type: gui {
6      parameter "My integer global value" var: my_integer_global_value
7      init: 5;
8  }

```

Listing 3.17: Definição de um parâmetro

A variável *my_integer_global_value* também pode ser inicializada na declaração global, deste modo deve-se remover o comando *init* da declaração do experimento.

Também é possível declarar especificações como *category*(utilizada para criação de labels), *min*, *max* e *among*

```
1 global {
2     string fruit <- "none" ;
3     string vegetable <- "none";
4     int integer_variable <- 5;
5 }
6
7 experiment MyExperiment type: gui {
8     parameter "fruit" category:"food" var: fruit <- "none" among:["none
9     ", "apple", "banana"] ;
10    parameter "vegetable" category:"food" var: vegetable <- "none" among
11    :["none", "cabbage", "carrot"];
12    parameter "integer variable" category:"other" var: integer_variable
13    <- 5 min:0 max:100 ;
14 }
```

Listing 3.18: Exemplos de definição de parâmetros

Um *display* é uma das saídas gráficas da simulação, pode-se definir vários *display* relacionados ao que deseja representar na execução do modelo. Para definir um *display* é utilizado a palavra reservada *display* dentro do escopo de *output* e especificando um nome(*label* para identifica-lo).

É possível mudar a cor de fundo do *display* através do comando *background*, para definir de quanto em quanto tempo o *display* será atualizado para exibição é necessário a utilização do comando *refresh:every(s)* dado o tempo em segundos. Para a exibição do *display* existe dois métodos, *java2D* o qual apresenta a visualização em 2D e o *opengl* que permite a visualização em 3D.

```
1     experiment my_experiment type: gui {
2         output {
3             display "display1" background: #red refresh:every(10)
4         type: opengl {
5             }
6             display name:"display2" background: #green refresh:every
7             (20) type: java2D {
8             }
9         }
10    }
```

Listing 3.19: Definição básica de *display*

Para uma melhor exibição dos resultados da simulação é possível adicionar gráficos dentro de um *display*, para isso deve-se utilizar a palavra reservada *chart*. OS gráficos devem ser nomeados e o tipo deve ser especificado entre histograma, pizza, série, dispersão, xy, radar, mapa de calor ou *box_whisker*.

```
1 global {
2   int numberA <- 2 update: numberA*2;
3   int numberB <- 10000 update: numberB-1000;
4   init {
5     create people number: 10000;
6   }
7 }
8
9 species people {
10   float age <- gauss(40.0, 15.0);
11 }
12
13 experiment my_experiment type: gui {
14   output {
15     display "my_display" {
16       chart "my_chart1" type: pie {
17         data "numberA" value: numberA color: #red;
18         data "numberB" value: numberB color: #blue;
19       }
20       chart "my_chart2" type: series {
21         data "numberA" value: numberA color: #red;
22         data "numberB" value: numberB color: #blue;
23       }
24       chart "my_chart3" type: histogram {
25         datalist (distribution_of(people collect each.age
26 ,20,0,100) at "legend")
27         value:(distribution_of(people collect each.age,20,0,100)
28 at "values");
29       }
30       chart "my_chart3" type: radar background: #white axes:#black
31 {
32         data "numberA" value: numberA color: #red
33 accumulate_values: true;
34         data "numberB" value: numberB color: #blue
35 accumulate_values: true;
36       }
37     }
38   }
39 }
```

Listing 3.20: Definindo vários tipos de gráfico

O gráfico de pizza mostra em um único diagrama como em 3.11 a proporção de cada série de dados sobre a soma de todas as séries.

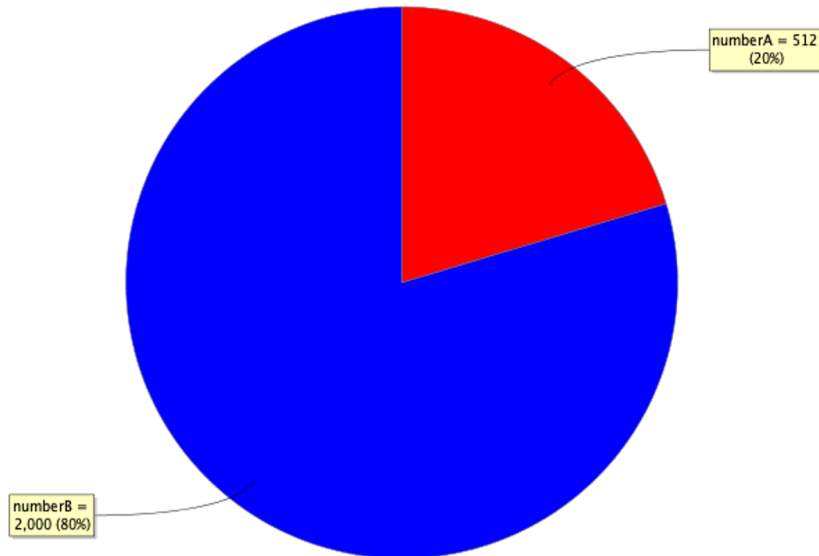


Figura 3.11: Gráfico de pizza gerado no código exemplo presente em 3.20 .

O gráfico de série exibe em um espaço de coordenadas xy o valor de cada série de dados ao longo do tempo(ciclo da simulação): o eixo x exibe o tempo, o eixo y representa o valor da série de dados. O gráfico gerado no exemplo está em 3.12.

Os gráficos de histograma representam com barras o valor de várias séries de dados. O exemplo anterior pode é exibido em 3.13.

Os gráficos xy são usados para mostrar um valor em função de outro (ao invés de plotar um valor em função do tempo): neste caso, o eixo x não representa o tempo em geral.

Um gráfico de radar exibe a evolução da expressão ao longo do tempo em uma espécie de representação circular, o exemplo descrito anteriormente obtemos o seguinte gráfico 3.14:

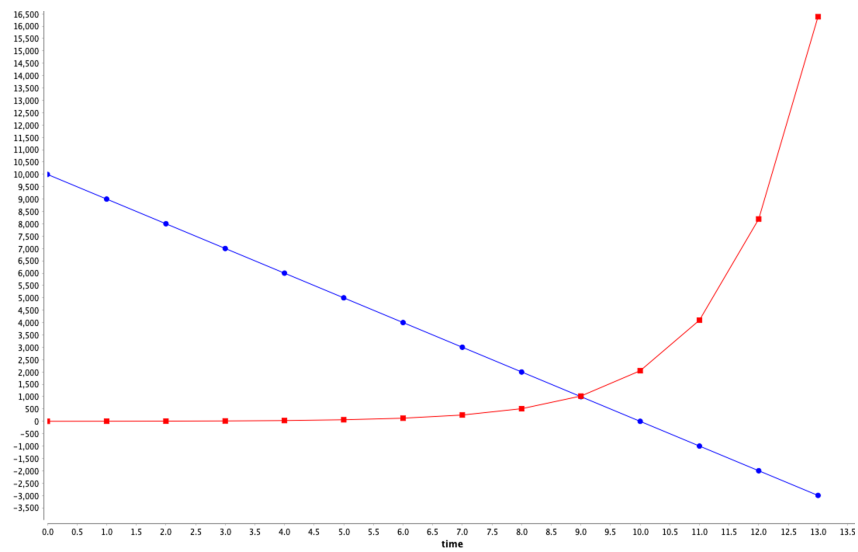


Figura 3.12: Gráfico de séries gerado no código exemplo 3.20 .

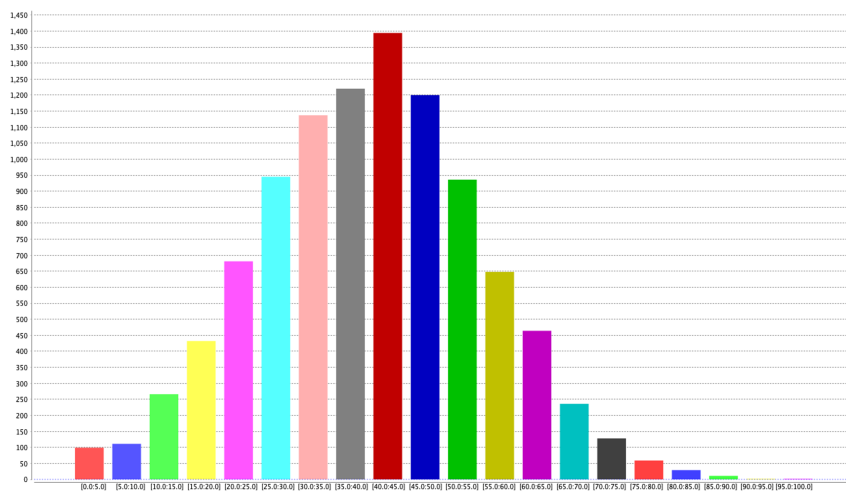


Figura 3.13: Histograma gerado no código exemplo 3.20.

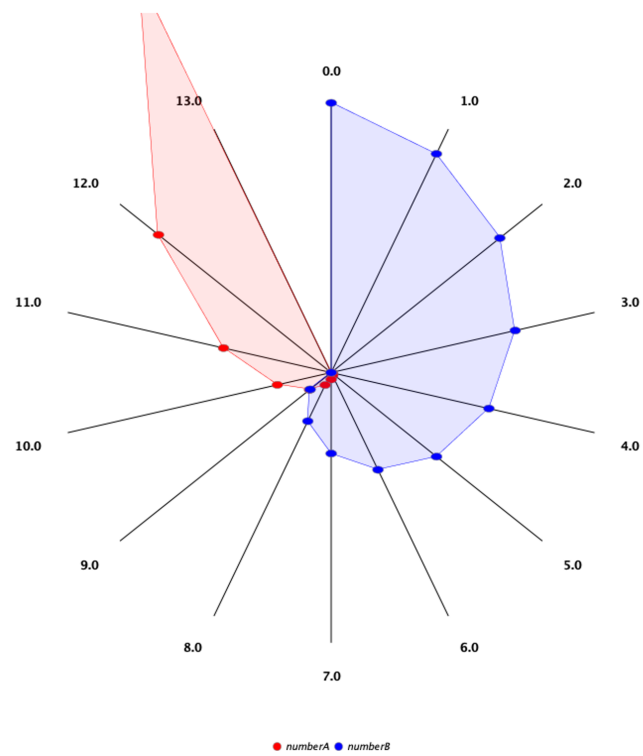


Figura 3.14: Gráfico de radar gerado no código exemplo 3.20.

Capítulo 4

Modelos Baseados em Agentes

Este capítulo apresenta as etapas, configurações e códigos utilizados para a criação e simulação dos modelos idealizados.

4.1 Modelando a Governança de Território Saudável e Sustentável para a Cidade Estrutural

O modelo do processo de Governança de Território Saudável e Sustentável para a Cidade Estrutural (GTSSE) foi desenvolvido para emular as oportunidades e dificuldades na governança da intervenção proposta pela Fiocruz-DF [7], de promoção de um território sustentável e saudável [6] na Cidade Estrutural - DF.

4.1.1 Propósito e padrões do Modelo

Este modelo tem como propósito observar as relações de poder entre entidades dentro da organização do território da estrutural, para tentarmos entender o melhor funcionamento da comunidade como um todo e identificar atores que têm a capacidade de influenciar todo o modelo através de ações. A convergência do modelo foi definida como um dos padrões, de forma que foi feita a calibragem do modelo para obter convergência em todas as simulações realizadas. Outro padrão adotado foi a expectativa de que os atores iam colaborar entre si.

4.1.2 Entidades e variáveis de estado

Nesta subseção são apresentadas as entidades, suas descrições e quais as relações que cada uma pode iniciar.

- Fiocruz é um dos atores principais do modelo por ter contato tanto com a parte acadêmica e governamental quanto o representante da parte social/civil do modelo. A fiocruz controla as relações:
 - Obter dados da saúde
 - Acesso a professores e alunos
 - Obter dados dos projetos na estrutural
- SES corresponde a parte governamental do modelo responsável por repassar dados da saúde para a Fiocruz, sendo que o SES controla a relação: Obter recursos computacionais.
- UnB corresponde a Universidade de Brasília, a qual realiza parcerias com a fiocruz para diversos estudos. A UnB controla a relação: Disseminação de conhecimento.
- Coraci é outro ator principal do modelo, pois é o representante da parte social/civil e está em contato direto com a Fiocruz. Coraci controla as relações:
 - Analise de dados da estrutural
 - Obter dados do coletivo
- Coletivo da Cidade controla é um ator que reúne diversas informações da comunidade da estrutural e as repassa para o Coraci. Controla a relação: Identificação de possíveis projetos

Os valores e descrições das relações implementadas estão descritos abaixo:

- Obter dados da saúde: A fiocruz obtém e utiliza diversos dados ligados a saúde pública do Distrito Federal, dados estes que são disponibilizados pela Secretaria de Saúde do Distrito Federal.
- Obter recursos computacionais: A SES utiliza a estrutura disponível na fiocruz com a intenção de realizar processamentos nos dados da saúde.
- Acesso a professores e alunos: Através dessa relação a Fiocruz cria parceria com a UnB de forma a utilizar o conhecimento e trabalho tanto dos alunos como dos professores.
- Disseminação de conhecimento: A parceria com a Fiocruz permite a UnB realizar novas pesquisas, disponibilizar iniciação científica para os alunos e com isso promover a disseminação de conhecimentos.

- Obter dados de projetos da estrutural: Através dessa relação a Fiocruz obtém dados dos projetos os quais estão sendo implantados na Cidade Estrutural.
- Análise de dados da estrutural: Essa relação descreve a utilização de recursos materiais e intelectuais da Fiocruz para a realização de análises dos dados dos projetos da Cidade Estrutural com a finalidade de gerar conhecimento.
- Obter dados do coletivo: Nessa relação, as informações referentes a diversos projetos e pessoas da cidade Estrutural são repassados para o Coraci que é o representante da comunidade local.
- Identificação de possíveis projetos: Após a análise dos dados da estrutural, os resultados são repassados da Fiocruz para o Coraci, o qual junto com o Coletivo da cidade farão a identificação de novos projetos sociais e avaliação dos projetos já existentes.
- Fiocruz
 - Obter dados da saúde: 2.0
 - Obter recursos computacionais: 1.0
 - Acesso a professores e alunos: 1.5
 - Disseminação de conhecimento: 1.5
 - Obter dados de projetos na estrutural: 2.0
 - Análise de dados da estrutural: 2.0
- SES
 - Obter dados da saúde: 4.0
 - Obter recursos computacionais: 6.0
- UnB
 - Acesso a professores e alunos: 4.0
 - Disseminação de conhecimento: 6.0
- Coraci
 - Obter dados de projetos na estrutural: 2.0
 - Análise de dados da estrutural: 3.0
 - Obter dados do coletivo: 2.0
 - Identificação de possíveis projetos: 3.0

- Coletivo da cidade
 - Obter dados do coletivo: 3.0
 - Identificação de possíveis projetos: 7.0

4.1.3 Visão do processo e escalonamento

O modelo foi construído tendo como base o dilema do prisioneiro, o qual é um problema de teoria dos jogos. Nesse dilema tem-se um caso de soma não nula, onde o ganho de um ator não necessariamente corresponde a perda de outro ator, porém todos os atores tentam maximizar o seu ganho sem se importar com o efeito que suas ações teram sobre os outros atores.

Os atores decidem qual ação vão tomar ao mesmo tempo, e somente no próximo ciclo as variáveis de satisfação serão atualizadas para que os atores possam avaliar se suas ações foram benéficas ou não e assim realizar um novo julgamento sobre qual a melhor estratégia a ser tomada.

4.1.4 Design do Modelo

- Fenômenos emergentes: O fator de satisfação dos agentes pode ser considerado como uma propriedade emergente do modelo, uma vez que este depende do resultado de todas as interações do agente durante os ciclos da simulação.
- Sensibilidade: Os atores não possuem a completa informação sobre as opiniões e incertezas dos os atores com os quais se relacionam.
- Estocasticidade: O modelo desenvolvido não apresenta comportamento estocástico, uma vez que os parâmetros, interações e outros aspectos não definidos de forma aleatória.
- Observação: As variáveis que podem ser observadas durante a simulação são: influência, satisfação, média da quantidade de vezes que cada ação foi executada, convergência do modelo. Também é possível observar a rede de relações.

4.1.5 Implementação do Modelo: codificação, inicialização/dados de entrada

Model Editor

Para começar o desenvolvimento de um modelo no SocLab deve-se iniciar pelo painel Model Editor(exemplificado na Figura 4.1) onde o usuário define quais são os atores e

relações envolvidos no modelo, também é possível editar a descrição das relações e incluir intervalos que variam de $[-10,10]$ e são utilizados para quantificar as relações dos atores.

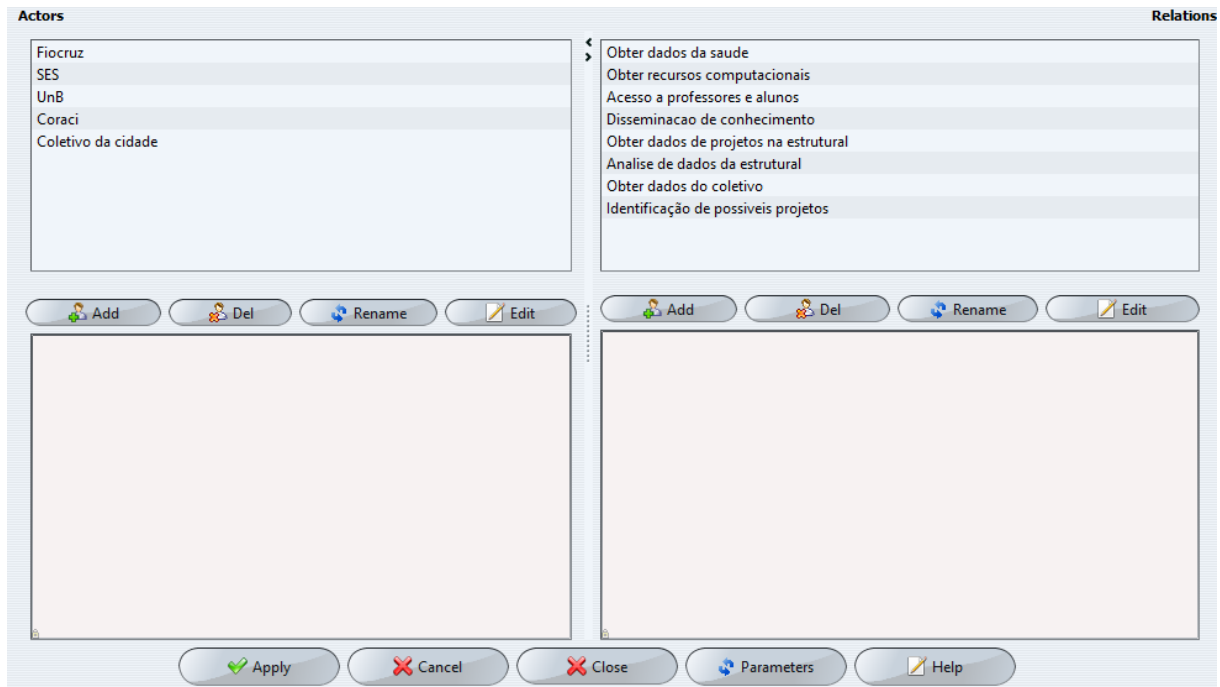


Figura 4.1: Painel para edição do modelo GTSSE

Controls

Neste painel(Figura 4.2) é possível configurar qual ator inicia e controla uma determinada relação, além disto pode-se determinar com qual frequência isto acontece e os valores máximos e mínimos que as relações podem ter.

4.1.6 Stakes

Em stakes(Figura 4.3) é possível definir valores para o interesse que cada ator possui nas relações em que está envolvido, de forma que a soma desses interesses tem que ser obrigatoriamente igual a 10.

4.1.7 Effects

Em effects(Figura 4.4) é possível definir uma função matemática para caracterizar o a satisfação dos atores de acordo com as ações em que participa. Para facilitar a análise dos resultados os efeitos foram definidos como uma equação linear, de forma que quanto mais uma ação é tomada maior será satisfação do ator envolvido.

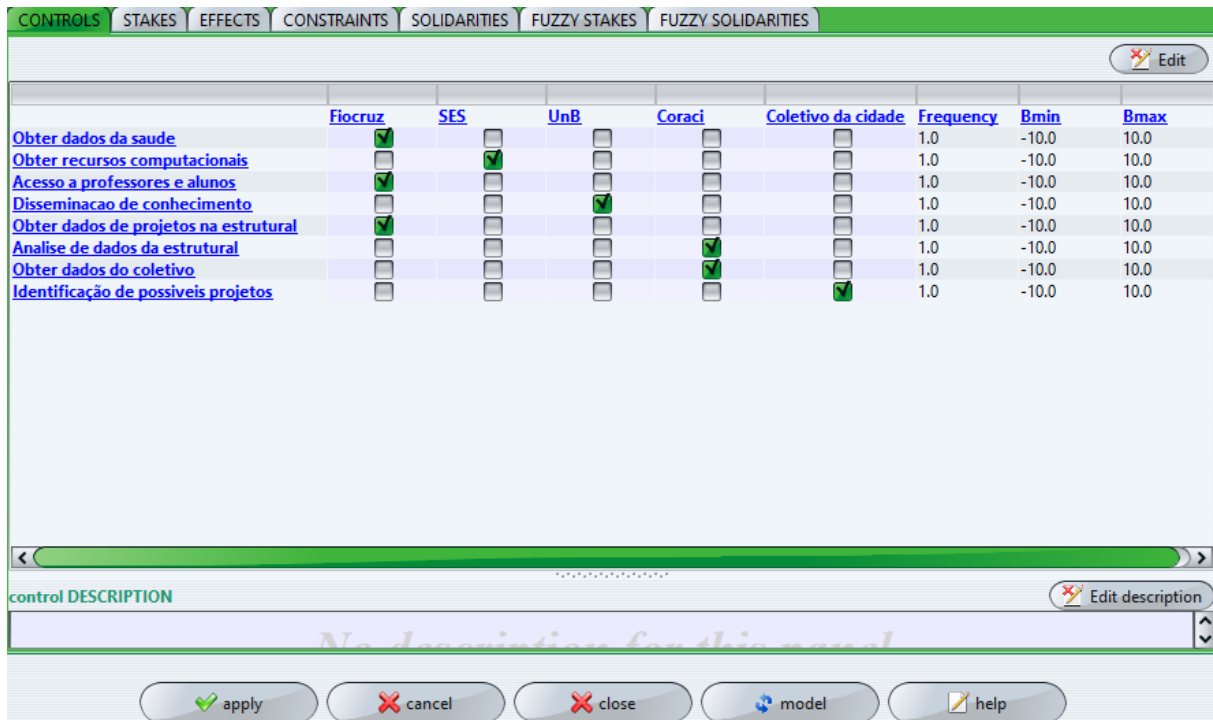


Figura 4.2: Painel para edição do controle das relações do modelo GTSSE

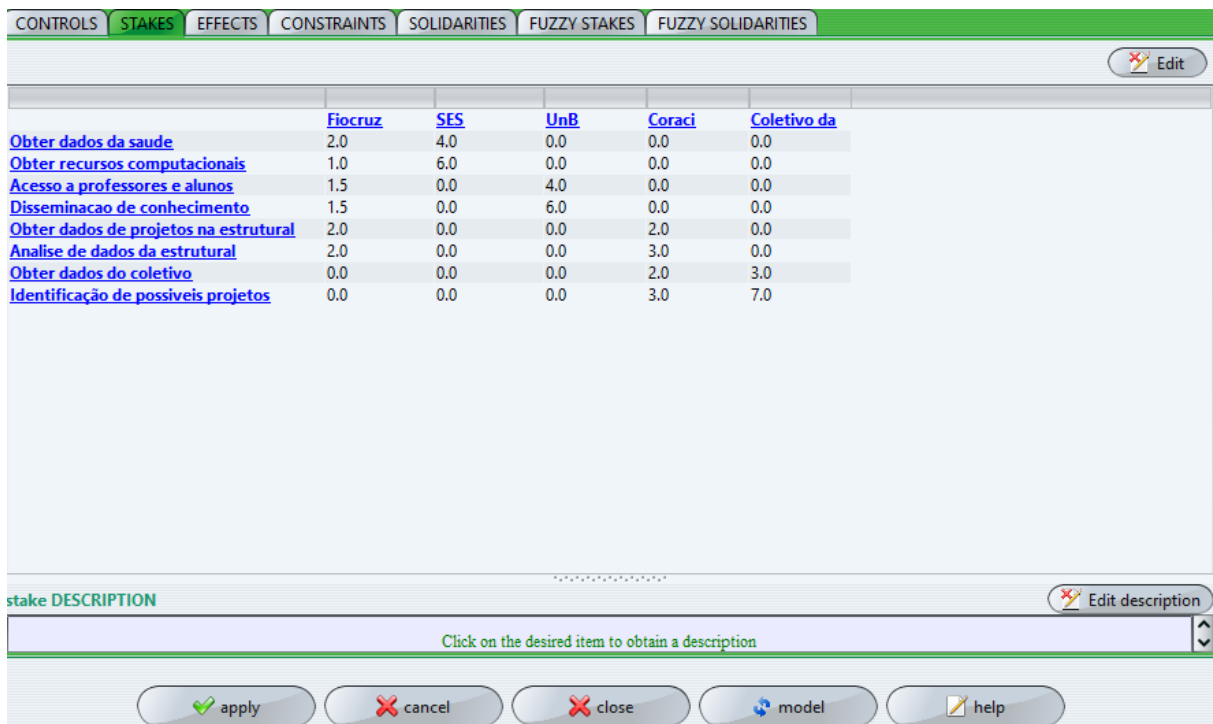


Figura 4.3: Painel para edição dos Stakes do modelo GTSSE

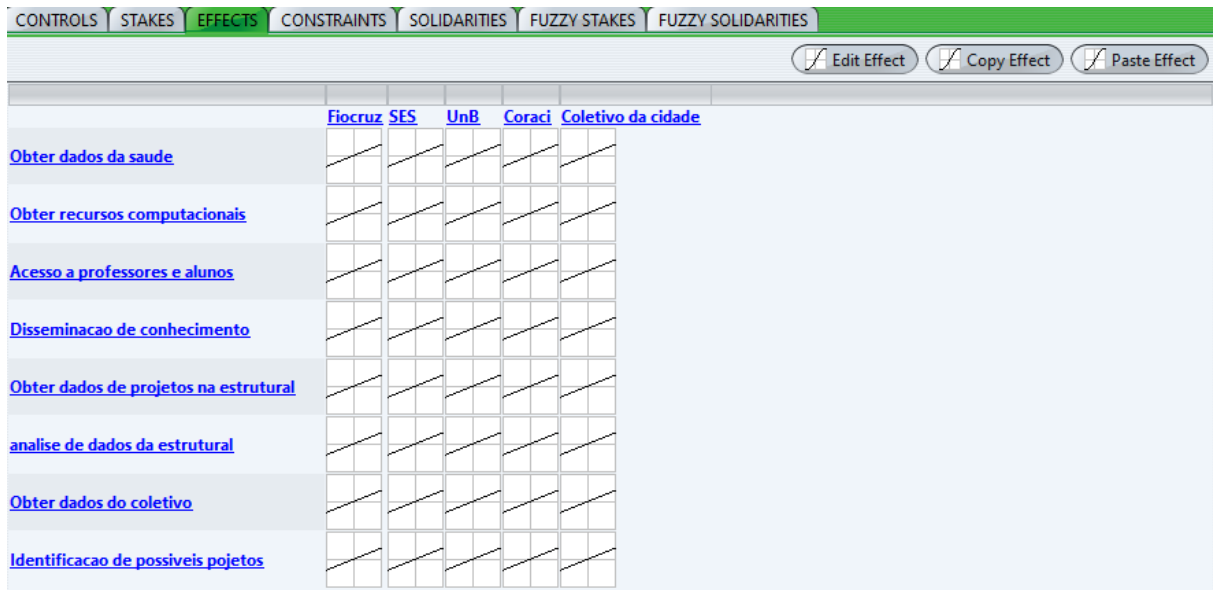


Figura 4.4: Painel para edição dos Effects do modelo GTSSE

4.1.8 Constraints

Constraints se refere a forma como uma ação influencia a tomada de outras ações. Para este modelo preliminar não foi definido nenhuma restrição.

4.1.9 Solidarities

Neste painel(Figura 4.5) é possível definir o quanto um ator se importa com os outros, fato que irá interferir na decisão de qual ação deverá ser executada em um determinado passo da simulação.

4.1.10 Fuzzy Stakes e Fuzzy Solidarities

As solidariedades e os valores de interesses também podem ser definidos nesses painéis de forma que passam a se comportar como variáveis difusas, cujos valores variam dentro de um intervalo de tempo configurado pelo usuário.

4.1.11 Opções de Simulação do SocLab

4.1.12 Networks

Utilizando a função networks é possível visualizar como o modelo está organizado. Sendo que se pode visualizar a importância de cada relação, a autonomia e subordinação de cada autor, a satisfação máxima e mínima, a relevância de cada relação.

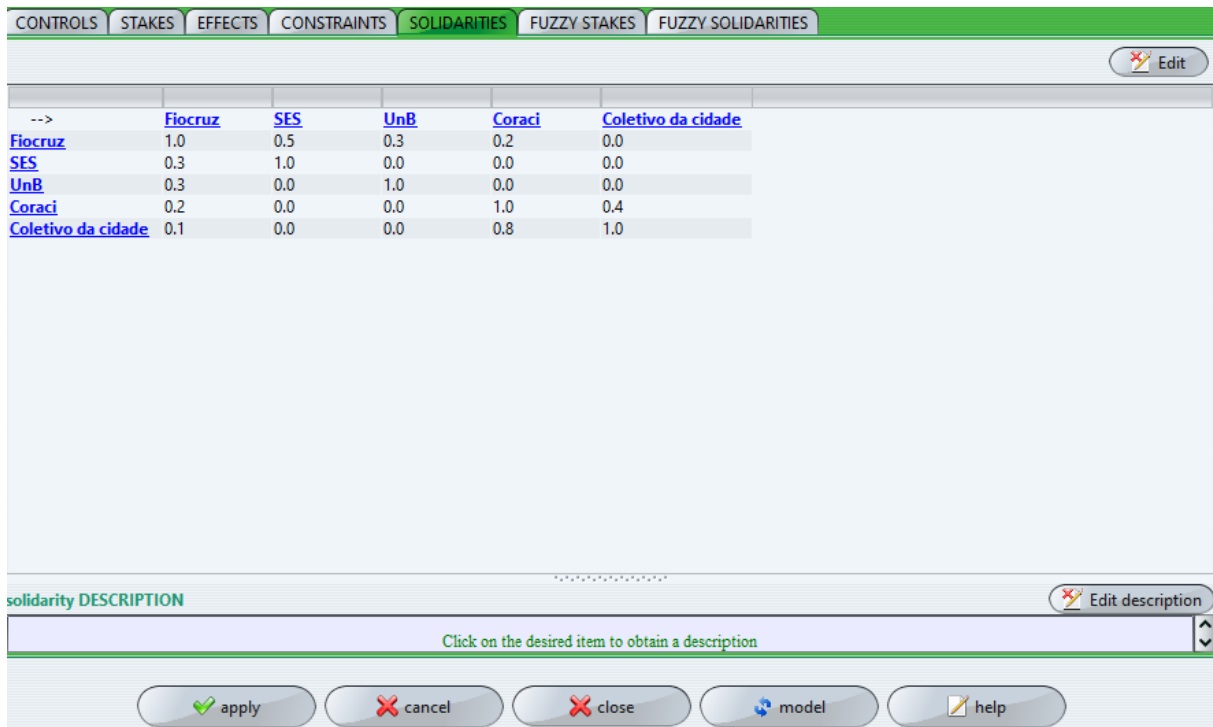


Figura 4.5: Painel para edição de Solidarities do modelo GTSSE

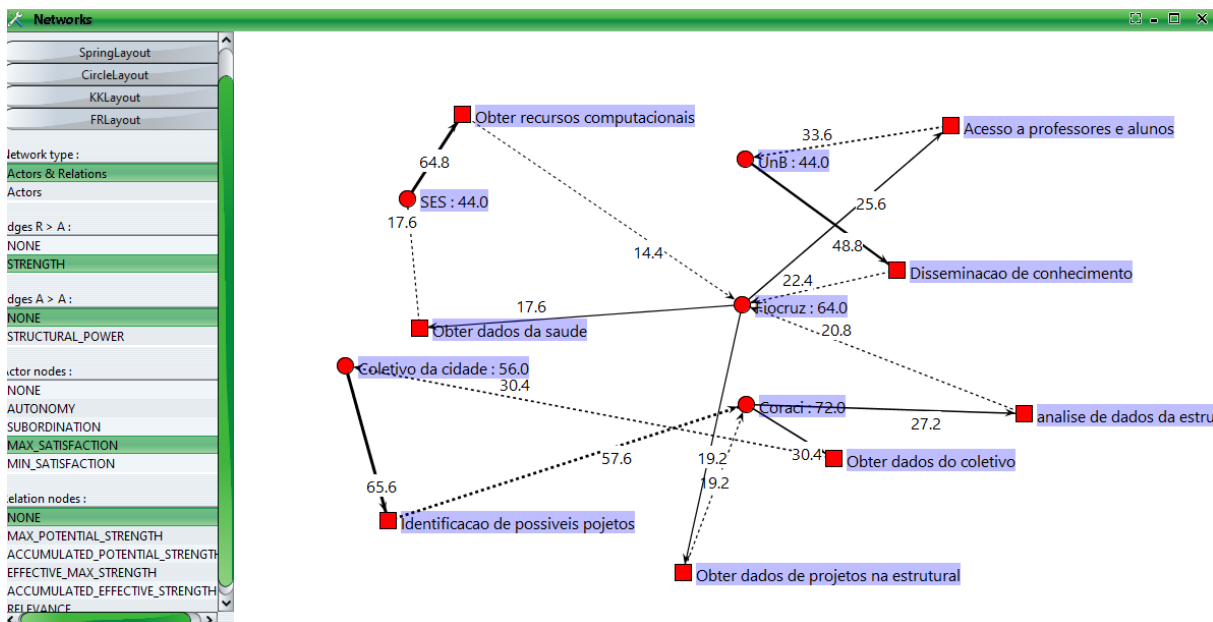


Figura 4.6: Organização estrutural do modelo GTSSE

4.1.13 Simulation Explorer

A Figura 4.6 mostra o painel utilizado para a definição dos parâmetros iniciais que serão utilizados na simulação para cada relação.

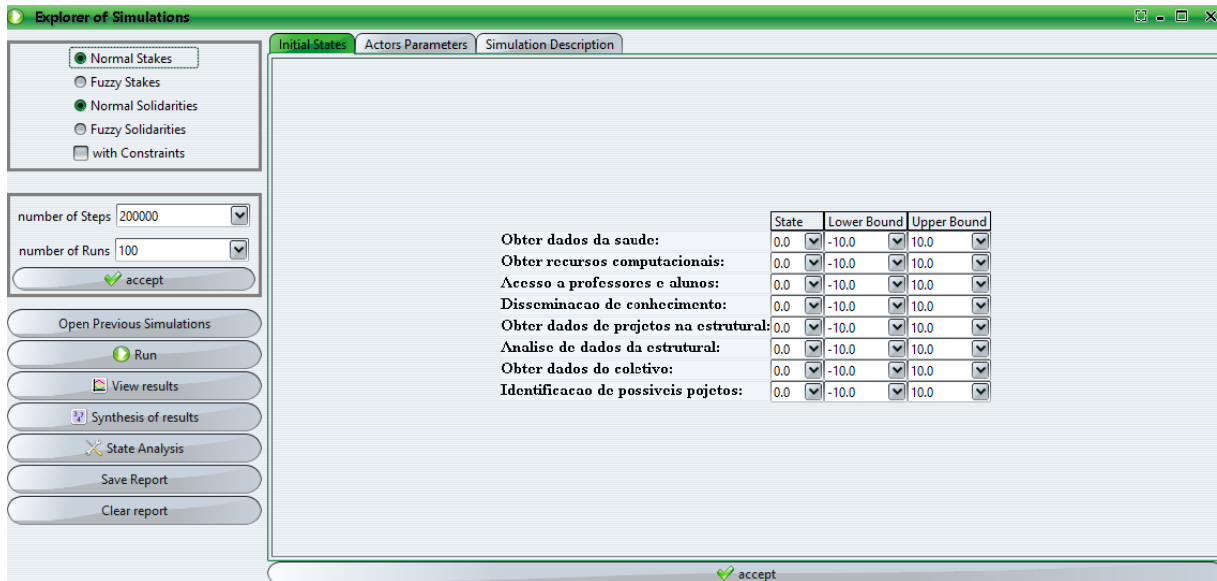


Figura 4.7: Painel para definição dos parâmetros iniciais do modelo GTSSE

A Figura 4.7 mostra o painel utilizados para a definição dos parâmetros de comportamento de cada ator durante a simulação. Onde tem-se:

Distance min/max satisfaction: É a distância euclidiana entre a satisfação mínima e máxima do ator.

Scope: Reflete a capacidade do ator de discriminar situações e, portanto, sua capacidade de selecionar as regras aplicáveis de sua base na situação atual. Com um Escopo de 1, as regras são sempre aplicáveis, com um Escopo de 3, as situações são discriminadas em ruins, médias e boas.

Reactivity : Reflete a rapidez com que um ator atualiza sua ambição e sua taxa de exploração, de acordo com a satisfação que recebe na situação atual. Valores pequenos aumentam a duração da simulação.

Repartition of reward : Repartição da recompensa entre a última regra aplicada e a penúltima regra aplicada.

Types of rules:

Simple: As regras são reforçadas com uma recompensa fixa, independentemente do ganho de satisfação (ou perda) que sua aplicação produz. Essa recompensa é adicionada (ou deduzida) à força da regra até um certo limite. As regras são progressivamente removidas da base de regras com um fator de esquecimento.

Self learning: As regras são reforçadas com uma recompensa proporcional ao ganho de satisfação (ou perda) que seu aplicativo produz. Este reforço segue a fórmula:

$$strength \leftarrow al * strength + (1 - al) * reward$$

,onde al aumenta progressivamente, seguindo a taxa de exploração.

A Figura 4.8 mostra o painel onde é realizado a configuração inicial dos parâmetros dos atores, sendo possível alterar as regras de aprendizado, divisão da recompensa, entre outros.

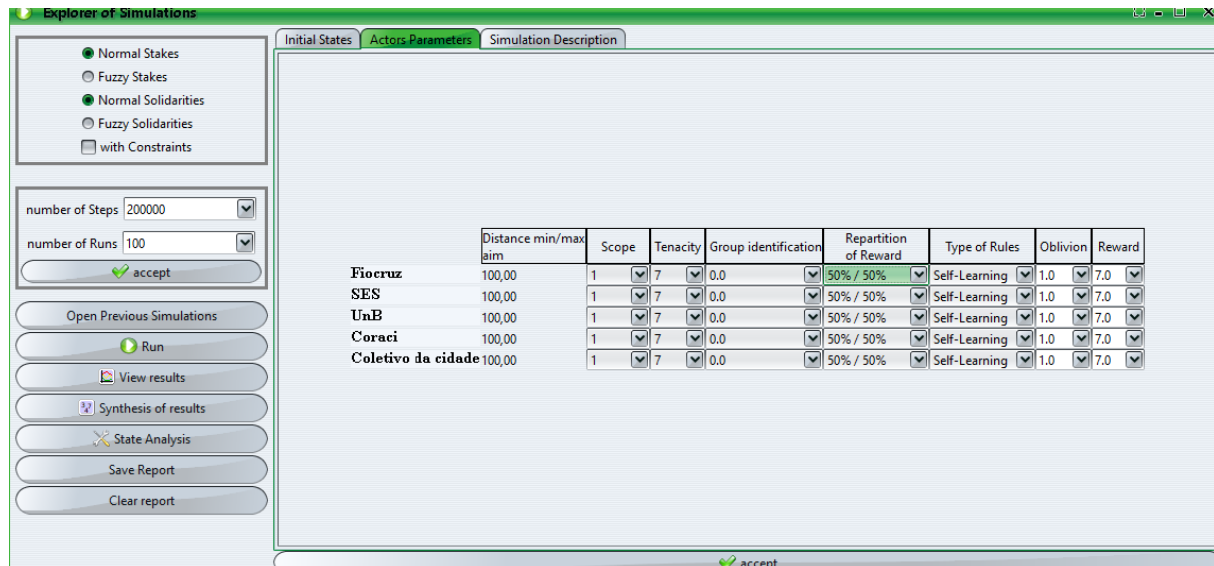


Figura 4.8: Painel para definição dos parâmetros dos atores do modelo GTSSE

4.1.14 Dificuldades encontradas com o uso do SocLab

Para o desenvolvimento de modelos com muitos atores e relações o programa apresentou alguns problemas, como demora para a exibir os resultados e em alguns casos até travou, sendo necessário interromper o seu processo. Também foi identificado que quanto mais efeitos são definidos como constantes o tempo de processamento aumenta. Com isto podemos verificar que este software é mais indicado para modelagem em nível micro, já que nas abordagens meso e macro há uma grande quantidade de atores envolvidos.

Devido a estes problemas encontrados o software não foi considerado ideal para a proposta de estudo deste trabalho. Entretanto o SocLab apresenta grande potencial para ajudar em pesquisas sobre análise de relações de poder e tomada de decisões em governança de territórios, seja estes quais forem. Porém para a geração de um modelo mais próximo do real se faz necessário um estudo mais amplo do território a fim de detectar mais relações entre os atores.

No entanto para valores que refletem pensamentos e emoções humanas como stakes, solidariedades e a interconexão das relações é necessário a aplicação de entrevistas e questionários com fundamentos psicológicos e sociológicos a fim de obter valores mais adequados, pois estes dados são de difícil modelamento matemático.

4.2 Modelando a Dinâmica de Doenças e Curas no entorno das Unidades de Saúde do DF

Esta seção tem como objetivo descrever o modelo DDCDF para uma melhor compreensão do objetivo para o qual foi realizado essa modelagem.

4.2.1 Propósito e padrões do Modelo

O modelo de simulação da Dinâmica de Doença e Cura no Entorno das Unidades Básicas de Saúde do DF (DDCDF) foi desenvolvido para emular os processos decorrentes do deslocamento espacial dos usuários do SUS em suas relações com os agentes de saúde vinculados às unidades de saúde da Secretaria de Saúde do DF, inclusive as unidades básicas. Para padrão do modelo foi definido que a simulação seria encerrada quando tivesse a estabilização do número de pessoas em crise, fato que ocorreu por volta de 600 ciclos de simulação.

4.2.2 Entidades e variáveis de estado

As entidade definidas no modelo são:

- Mapa do DF: Essa entidade foi utilizada para delimitar os limites do ambiente de simulação, onde os agentes se movem.
- Rodovias: Entidade utilizada para simular as vias de locomoção presentes no DF, são utilizadas pelos agentes pra locomoção até as unidades de saúde.
- Ubs: Entidade para onde as pessoas em crise se dirigem com a finalidade de receberem tratamento e conseqüentemente ficarem saudáveis.
- Health agents: Atores os quais se locomovem dentro de uma subárea administrativa da saúde do df, com a finalidade de prestarem serviços a comunidade e assim diminuir o risco das pessoas ficarem doentes.
- People: Corresponde a todas as pessoas do modelo e representam a população do DF.

As variáveis de estado são descritas abaixo:

- shape: Corresponde a forma e a cor designados para cada tipo de agente.
- type: Presente na entidade ubs tem como objetivo definir em qual subárea administrativa da saúde a unidade está inserida.

- **sintoms:** Variável utilizada para definir se uma pessoa está doente ou não, o seu valor é definido aleatoriamente no início da simulação juntamente com a criação dos atores do tipo `people`.
- **faixa etaria:** Variável definida aleatoriamente de acordo com a porcentagem de distribuição jovens, adultos e idosos da população do DF.
- **jovem,adulto e idoso:** Classifica o valor da variável faixa etária.
- **cronologico:** Variável utilizada para definir se uma pessoa é doente crônica ou não, de acordo com a probabilidade de cada classificação da faixa etária.
- **sick:** Variável utilizada para definir que uma pessoa está doente assim que o valor de `sintoms` fica igual ou superior a 18.
- **sick crono:** Variável utilizada para definir que uma pessoa que é doente crônica está doente assim que o valor de `sintoms` fica igual ou superior a 18.
- **ubs place:** Define para qual unidade de saúde a pessoa deverá se dirigir caso esteja doente.
- **last point:** Variável utilizada para verificar a posição atual da pessoa no mapa.
- **target ubs:** Define o ponto no mapa onde se encontra a `ubs` para calcular a rota para a pessoa chegar até lá.

4.2.3 Visão do processo e escalonamento

O deslocamento das pessoas para as UBS ocorre através do reflexo `move`, o qual verifica a sua posição atual, define para qual UBS a pessoa deverá ir e com isso seta um ponto no mapa como destino final do movimento. Quando as pessoas estão em condições normais de saúde, elas se movimentam aleatoriamente no mapa.

O valor das doenças são definidos aleatoriamente na inicialização do modelo, e esses valores são atualizados de acordo com a probabilidade de uma pessoa ter uma piora, não sofrer alterações ou ter uma melhora no seu estado de saúde. Os valores também são atualizados se as pessoas se encontrarem com os agentes de saúde, ocasionando uma diminuição nas chances de ficarem doentes. Outra forma de atualização dessa variável é através das UBS as quais diminui o valor até a pessoa ficar saudável.

4.2.4 Design do Modelo

- **Fenômenos emergentes:** A variável `sintomas` presentes nos agentes do tipo `people` pode ser considerada como uma propriedade emergente do modelo, uma vez que

este depende da interação com os agentes Health Agents e dos valores definidos de forma aleatória no início da simulação.

- Sensibilidade: Os atores não possuem a completa informação sobre as opiniões e incertezas dos os atores com os quais se relacionam.
- Estocasticidade: O modelo desenvolvido apresenta comportamento estocástico, uma vez que a forma de interação entre os agentes e o valor da variável sintomas são definidos de forma aleatória.
- Observação: É possível observar a movimentação de todos os agentes, assim como se uma pessoa está em crise ou não, é possível acompanhar um determinado agente específico da classe people durante a simulação, também é apresentado um gráfico mostrando a evolução do número de pessoas em crise.

4.2.5 Implementação do Modelo: codificação, inicialização/dados de entrada

Definição do Mapa geográfico do DF

Arquivos no formato shapefile foram utilizados para definir o mapa do espaço geográfico de simulação onde os agentes móveis estão inseridos.

os limites geográficos do Distrito Federal estão definidos no arquivo de nome df.shp.

Em ubs.shp foram definidas as localidades das ubs e hospitais, entre outras unidades organizacionais que realizam atendimentos relacionados a saúde no DF.

Os arquivos leste.shp, sul.shp, oeste.shp, central.shp, norte.shp, sudoeste.shp, e centro_sul.shp foram criados para definir as regiões de saúde do DF.

O arquivo roads.shp definiu o mapa das rodovias do DF, com a finalidade de serem utilizadas para a movimentação dos usuários do SUS, inclusive em direção às unidades de saúde.

O arquivo df.shp, criado a partir de dados obtidos no site do IBGE contendo todas as rodovias do Brasil, bem como os demais arquivos, foram criados ou modificados através do programa QGIS3, que permite a criação de arquivos shapefiles com base em outros arquivos e mapas de satélite.

Após a geração dos arquivos shapefile, esses foram sobrepostos, gerando o espaço geográfico de simulação.

Depois, foi feita a criação dos agentes móveis, para definir suas fronteiras e como estes são exibidos no ambiente gráfico. A definição de cada um está exemplificada abaixo:

Código do agente building utilizado em conjunto com o arquivo df.

```
1 species DF {
```

```

2 aspect base {
3   draw shape color: #white border: #black;
4 }
5 }

```

Listing 4.1: Declaração do ator DF

Código do agente ubs utilizado em conjunto com o arquivo ubs.

```

1 species ubs {
2   int type;
3   rgb color <- #gray ;
4   aspect default {
5     draw shape color: color;
6   }
7 }

```

Listing 4.2: Declaração de comportamento de um agente da espécie unidade de saúde

Código do agente rodovia utilizado em conjunto com o arquivo roads.

```

1 species rodovia {
2   aspect default {
3     draw shape color: #gray;
4   }
5 }

```

Listing 4.3: Declaração de comportamento de um agente da espécie rodovia

4.2.6 Definição dos Atores

Com o ambiente de simulação completo, o próximo passo é a definição dos agentes envolvidos na simulação, que no caso são as pessoas e os agentes de saúde. A descrição e código de agente está abaixo.

Pessoas: São classificadas quanto a idade em jovens, adultos e idosos e quanto a saúde em saudáveis, doentes em crise, doentes crônicos e doentes crônicos em crise. Pessoas saudáveis e doentes crônicos se movem aleatoriamente no mapa. A condição de saúde das pessoas pode melhorar, ficar estável ou piorar de acordo com o resultado da variável `condit`. Quando uma pessoa entra em crise, ou seja, o atributo `sintoms` fica acima de 18, ela deve-se dirigir à ubs designada para receber tratamento médico, sendo que o deslocamento se dá através das rodovias e a ubs de destino fica localizada na região de origem da pessoa. Assim que a pessoa em crise se recupera, ou seja, quando o `sintoms` fica abaixo de 18, ela se desloca para fora da ubs.

```

1 species people skills: [moving]{
2

```

```

3  int sintoms;
4  int faixa_etaria;
5  int cronologico;
6  bool jovem;
7  bool adulto;
8  bool idoso;
9  bool sick;
10 bool sick_crono;
11 int condit;
12 ubs ubs_place;
13 point last_point;
14 point target_ubs <- nil;
15 point target_region <- nil;
16 int last_checkup;
17
18 reflex sick when: sick = true{
19   if(target_ubs = nil){
20     target_ubs <- any_location_in (ubs_place);
21   }
22   if(location = target_ubs){
23     sintoms <- sintoms -1;
24     if(sintoms < 18){
25       last_point <- target_ubs;
26       target_ubs <- nil;
27     }
28   }
29 }
30 reflex condition{
31   if (sintoms <= 18){
32     condit <- rnd_choice([0.247,0.5,0.253]);
33     //0 para piorar , 1 para estavel , 2 para melhorar
34     if (condit = 2){
35       sintoms<- sintoms -1;
36     }
37     if (condit = 0){
38       sintoms<- sintoms +1;
39     }
40     if (sintoms > 18){
41       sick <- true;
42     }
43     if (sintoms < 18){
44       sick <- false;
45     }
46   }
47 }

```

```

48 reflex move{
49   do goto target: target_ubs on: the_graph;
50 }
51 reflex health {
52   if(sintoms < 18 ){
53     do wander speed: 0.01 # m / # h;
54   }
55 }
56 aspect default {
57   draw circle(0.001) color: sick ? #red:#green;
58 }
59 }

```

Listing 4.4: Declaração dos comportamentos e atributos de people

Agentes de saúde: São designados para uma das ubs definidas no arquivo shapefile e tem como habilidade se movimentar no mapa com a finalidade de simular visitas a domicílio, que caso tenham efeito diminuem o valor da variável sintoms, tal ação só pode ocorrer caso a pessoas não esteja em crise. A probabilidade de uma visita ser positiva é definida pela variável proba_work.

```

1 species health_agents skills: [moving] {
2   ubs working_place <- nil ;
3   reflex move {
4     do wander;
5     ask people at_distance visit_distance{
6       if(sintoms < 17){
7         if (flip(proba_work)){
8           sintoms <- sintoms -1;
9         }
10      }
11    }
12  }
13  aspect default {
14    draw circle(0.001) color: #blue;
15  }
16 }

```

Listing 4.5: Declaração de comportamentos e atributos de health_agents

Capítulo 5

Resultados e Discussões

5.1 Governança de Território Saudável e Sustentável para a Cidade Estrutural usando SocLab

Apos a simulação do Modelo Governança de Território Saudável e Sustentável para a Cidade Estrutural (GTSSE) ter sido realizada 100 vezes com 200000 passos em cada, foi obtido a convergencia em todas as vezes.

A Rfigura 5.1 mostra os resultados para a satisfação e influência dos atores.

Nombre de simulations qui ont convergé : 100.0

Averages and Deviations of Actors' final variables (in value)

Actors	Satisfaction Aver.	Satisfaction Dev.	Influence Aver.	Influence Dev.	Aim Aver.	Aim Dev.	min Aim	max Aim
Fiocruz	53,67	10,24	60,46	19,49	53,67	10,24	-64,00	64,00
SES	39,83	5,56	40,76	0,09	39,83	5,56	-44,00	44,00
UnB	43,01	0,86	36,77	0,07	43,01	0,86	-44,00	44,00
Coraci	61,72	11,37	47,78	21,69	61,72	11,37	-72,00	72,00
Coletivo da cidade	51,24	5,60	63,92	0,15	51,24	5,60	-56,00	56,00

Influence, Absolute influence and Satisfaction percentage

Actors	% Influence	% Absolute Infl	% Satisfaction
Fiocruz	89,66 %	79,32 %	91,98 %
SES	100,00 %	100,00 %	95,32 %
UnB	100,00 %	100,00 %	98,86 %
Coraci	88,37 %	76,74 %	92,94 %
Coletivo da cidade	100,00 %	100,00 %	95,83 %

Figura 5.1: Resultados da simulação do modelo GTSSE

Como pode-se observar todos os atores apresentaram valores para satisfação consideravelmente altos e com pouco desvio, isto mostra uma certa estabilidade no sistema como um todo.

A Rfigura 5.2 mostra o resultado de quantas vezes cada relação foi feita, juntamente com a derivação de cada uma e uma análise do valor obtido de acordo com parâmetros estabelecidos no model editor.

Averages, deviations, and Interpretations of States of Relations

Relations	Averages	Deviations	Interpretations
Obter dados da saude	5,89	6,36	Dados passados na medida certa
Obter recursos computacionais	9,99	0,02	Acesso a quantidade adequada de recursos
Acesso a professores e alunos	9,94	0,11	Acesso a uma grande quantidade de alunos e professores
Disseminacao de conhecimento	9,99	0,02	Há uma grande disseminação de conhecimento
Obter dados de projetos na estrutural	7,03	4,84	Grande quantidade de dados obtidos
Analise de dados da estrutural	6,69	5,34	Quantidade razoável de dados analisados
Obter dados do coletivo	8,58	2,60	Grande quantidade de dados repassados
Identificacao de possiveis pojetos	9,99	0,02	Grande quantidade de possiveis projetos identificados

Figura 5.2: Resultado das relações do modelo GTSSE

Algumas relações como: "Obter dados da saúde", "Obter dados de projetos na estrutural", "Obter dados do coletivo" e "Análise de dados da estrutural" apresentaram um elevado valor de desvio, podendo apresentar até mesmo valores negativos no caso de "Obter dados da saúde", tal fato provavelmente teve como consequência uma maior variação no nível de satisfação dos atores Fiocruz e Coraci.

A Rfigura 5.3 mostra a esquerda o gráfico para a quantidade de vezes que uma relação foi estabelecida e a direita a satisfação total dos atores.

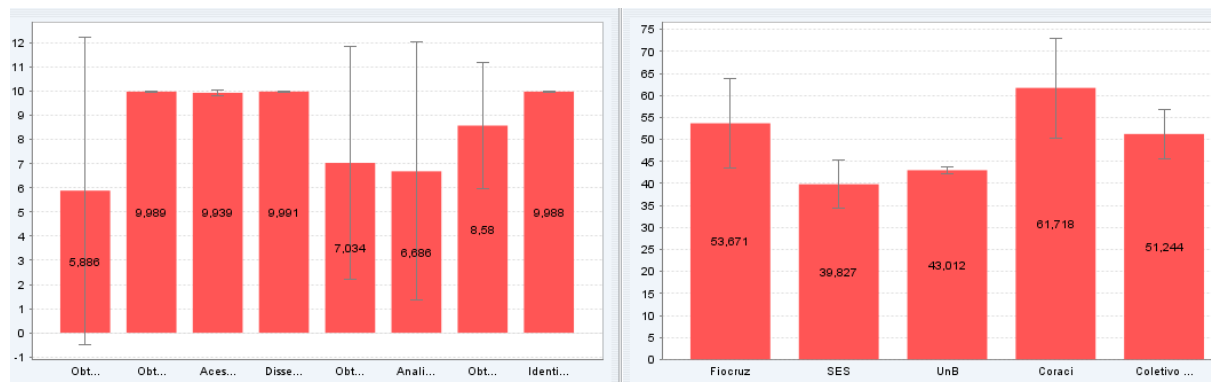


Figura 5.3: Quantidade de relações concluídas/ satisfação dos atores do modelo GTSSE

Também pode-se fazer análise de estado da simulação, onde é possível verificar algumas relações de indicadores como Influência/Satisfação, Poder/Capacidade de Ação. Os valores obtidos em cada relação estão apresentados nas figuras abaixo.

States/Interpretation	Influence/Satisfaction	Power/ActionCapacity	Influence/Satisfaction %	Power/ActionCapacity %		
	Fiocruz	SES	UnB	Coraci	Coletivo da cidade	SATISFACTION
Fiocruz	24.6	7.2	11.2	8.3	2.4	53.7
SES	6.5	32.4	0.4	0.5	0.0	39.9
UnB	17.7	0.4	24.4	0.5	0.0	43.0
Coraci	9.2	0.8	0.8	22.2	28.8	61.8
Coletivo da cidade	2.2	0.0	0.0	16.3	32.8	51.3
INFLUENCE	60.3	40.8	36.8	47.9	64.0	50.0
ABSOLUTE INFLU...	60.3	40.8	36.8	47.9	64.0	
POSITIVE INFLUE...	60.3	40.8	36.8	47.9	64.0	

Figura 5.4: Influência/Satisfação dos atores do modelo GTSSE

States/Interpretation	Influence/Satisfaction	Power/ActionCapacity	Influence/Satisfaction %	Power/ActionCapacity %		
	Fiocruz	SES	UnB	Coraci	Coletivo da cidade	Action Capacity
Fiocruz	18.2	4.0	4.0	5.4	0.0	31.6
SES	4.7	32.0	0.0	0.0	0.0	36.7
UnB	15.8	0.0	24.0	0.0	0.0	39.8
Coraci	5.6	0.0	0.0	14.9	12.0	32.5
Coletivo da cidade	0.0	0.0	0.0	10.3	28.0	38.3
POWER	44.4	36.0	28.0	30.6	40.0	35.8
ABSOLUTE POWER	44.4	36.0	28.0	30.6	40.0	
COOPERATIVE PO...	44.4	36.0	28.0	30.6	40.0	

Figura 5.5: Poder/Capacidade de Ação dos atores do modelo GTSSE

Através da simulação podemos concluir que a cooperação entre os atores é muito benéfica para a situação geral da organização, sendo que as ações realizadas nas extremidades do modelo, ou seja, ações que tem impacto direto sobre a população foram as que obtiveram os maiores valores.

5.2 Dinâmica de Doenças e Curas no entorno das Unidades de Saúde do DF usando GAMA

Foram realizadas 50 simulações com o intuito de verificar possíveis divergências nos resultados, o que não ocorreu, tendo em vista que foi obtido convergência para um cenário onde não há mais pessoas doentes.

A Rfigura 5.6 mostra a disposição inicial onde as pessoas foram alocadas em cada uma das regiões e a definição de em qual das ub's os agentes de saúde foram designados de acordo com os parâmetros estabelecidos no modelo. Os pontos em verde representam as pessoas saudáveis, os pontos em vermelho representam as pessoas que necessitam de atendimento médico, os triângulos em amarelo mostram a localização das ub's, as linhas em cinza mais claro são as vias que possibilitam a movimentação das pessoas.

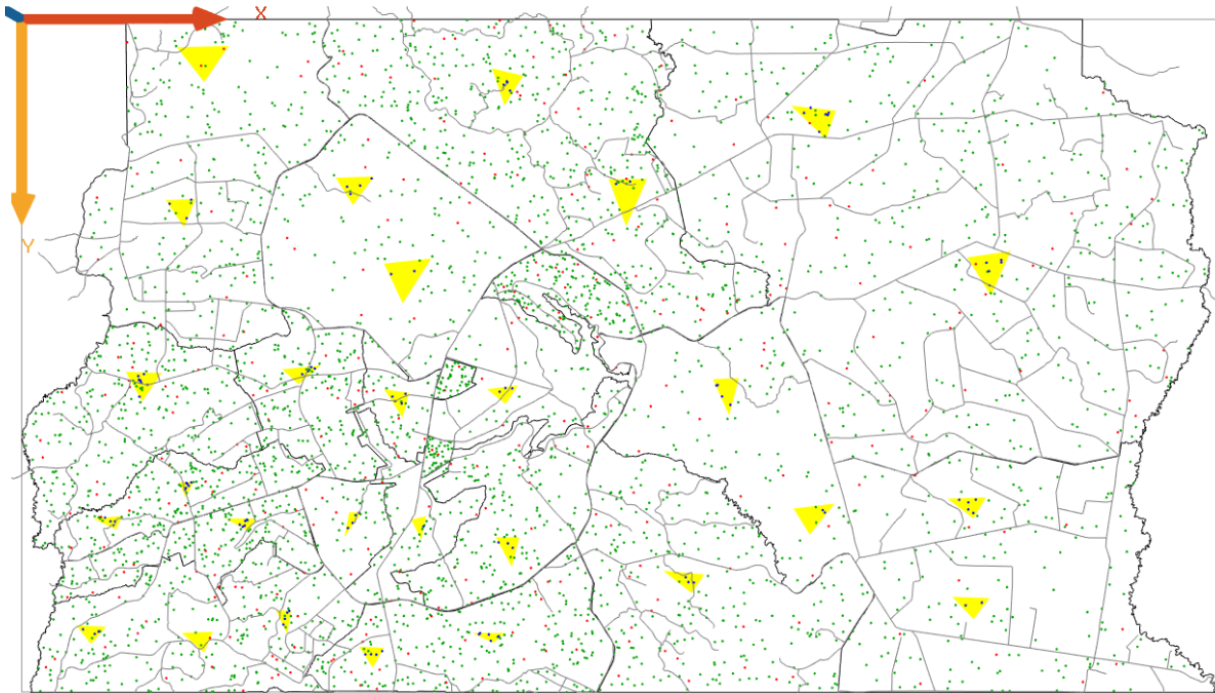


Figura 5.6: Mapa no início da simulação do modelo DDCDF

A Rfigura 5.7 mostra a distribuição da população por faixa etária de acordo com as probabilidades disponibilizadas pelo IBGE. Para a obtenção melhores efeitos visuais do comportamento dos agentes, as simulações foram realizadas com um número reduzido de pessoas, mas sempre respeitando a porcentagem de cada faixa etária.

a Rfigura 5.8 é referente ao estado inicial da simulação e detalha a quantidade de pessoas consideradas doentes crônicas, a quantidade de pessoas que estão em crise e quantas delas são doentes crônicos. O gráfico apresenta uma reta crescente com alta inclinação pois no instante inicial não há nenhuma pessoa.

A Rfigura 5.9 mostra a posição no mapa onde as pessoas e os agentes de saúde se encontram após a ter decorrido o tempo de 600 ciclos de simulação. Como pode-se observar houve um padrão de comportamento onde as pessoas tendem a ficar perto das ubs e dos agentes de saúde, de forma que essa é uma situação extremamente benéfica individualmente e que no cenário global da simulação leva a diminuição do desenvolvimento de condições crônicas e a estabilidade dos sintomas em pessoas que já possuem alguma doença crônica.

Porém tal padrão não corresponde a realidade, pois há outros fatores que regem o comportamento da população em geral.

A Rfigura 5.10 mostra a evolução do número de pessoas que se encontram em crise e dos doentes crônicos em crise ao decorrer dos 600 ciclos de simulação.

No início da simulação ocorreu um aumento no número de pessoas doentes, porém com

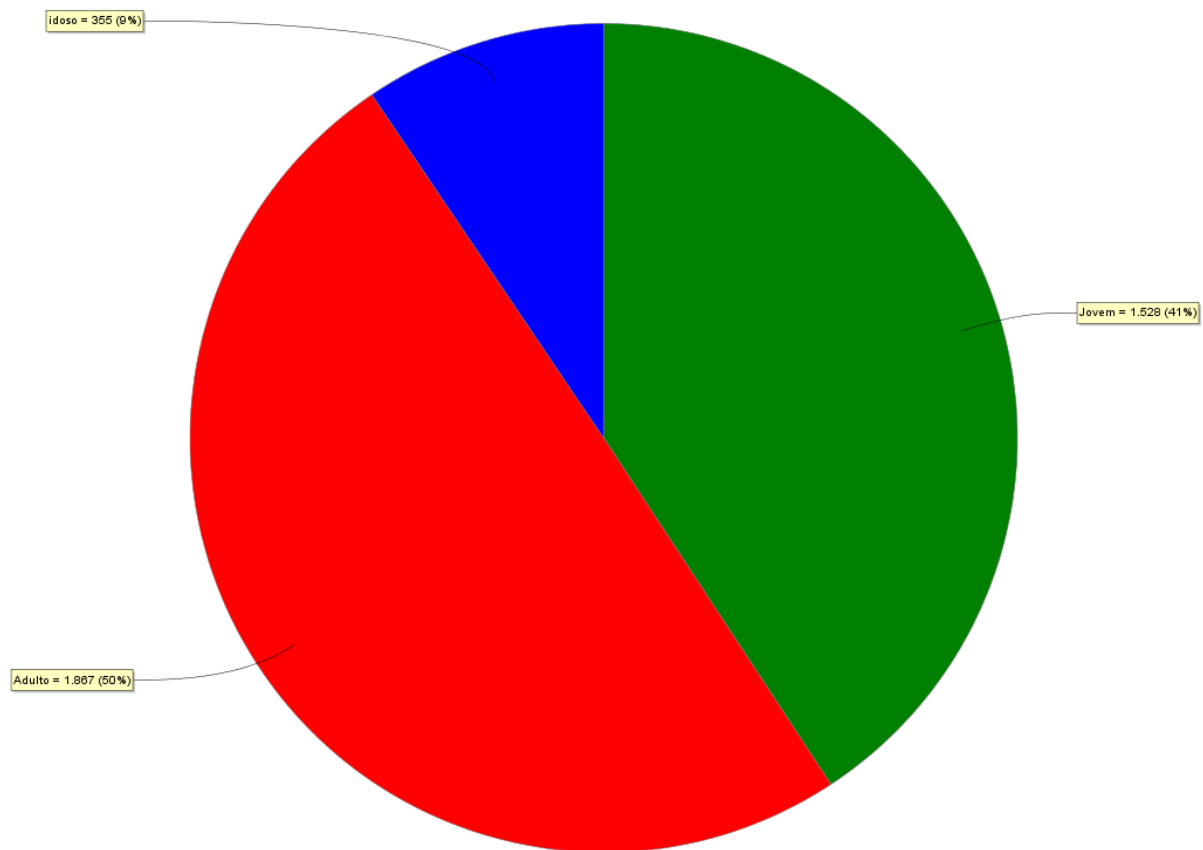


Figura 5.7: Distribuição da população por idade no modelo DDCDF

os passar dos ciclos este número apresentou uma redução até apresentar uma estabilidade em 600 ciclos.

Este capítulo apresenta uma análise das características dos modelos desenvolvidos, levando em consideração as teorias descritas nos capítulos 2 e 3.

Os modelos desenvolvidos podem ser considerados como explícitos, pois possuem regras de comportamento bem definidas e é possível observar relações entre diversos atributos e os impactos que podem gerar no resultado final da simulação.

Foi utilizado conceitos de ciências sociais computacionais para a definição e melhor refinamento dos modelos, uma vez que as ciências sociais procuram compreender não apenas como os indivíduos se comportam, mas também como a interação de muitos indivíduos leva a resultados em grande escala. A compreensão de um sistema político ou econômico requer mais do que a compreensão dos indivíduos que o compõem. Também requer a compreensão de como os indivíduos interagem uns com os outros e como os resultados podem ser mais do que a soma das partes[23].

Para a construção do modelo Governança de Territórios Saudáveis e Sustentáveis da Cidade Estrutural(GTSSE) foi utilizado o Dilema do Prisioneiro, o qual apresenta um problema derivado da teoria de jogos chamado de soma não nula, onde o ganho de um

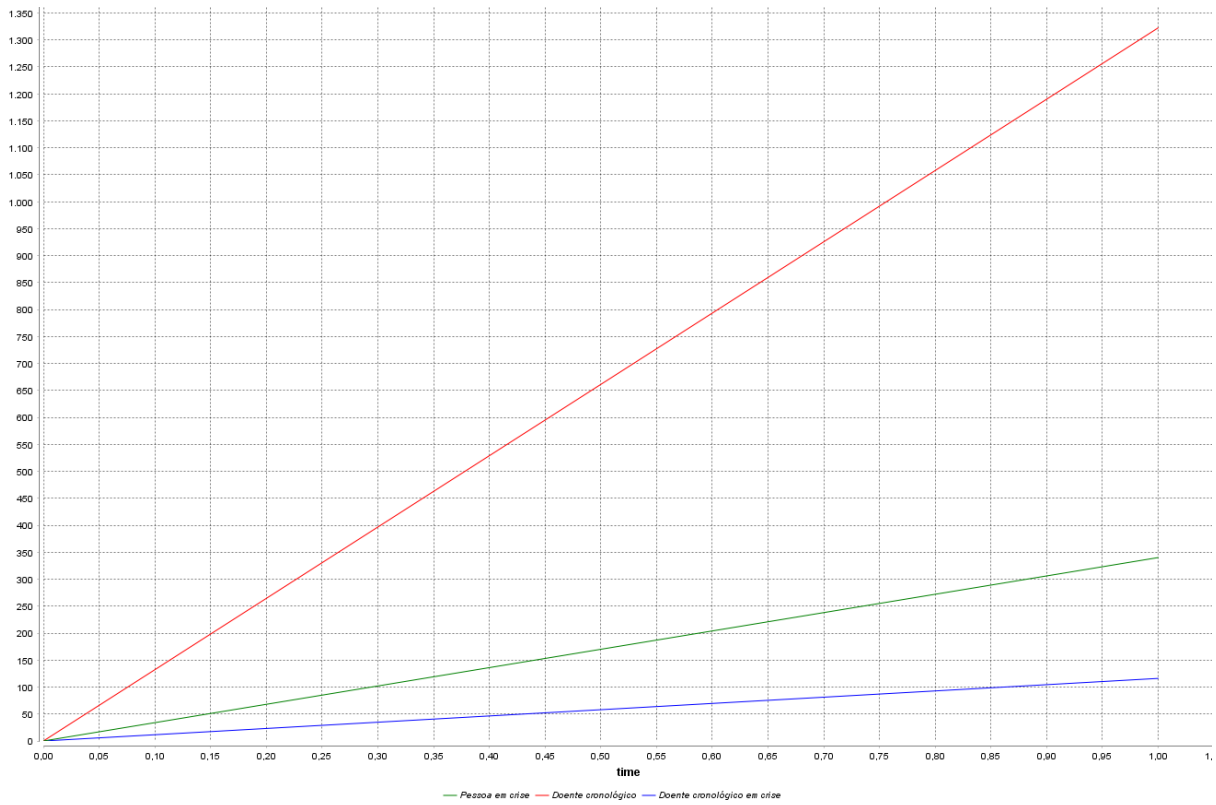


Figura 5.8: Quantidade de pessoas doentes no início da simulação do modelo DDCDF

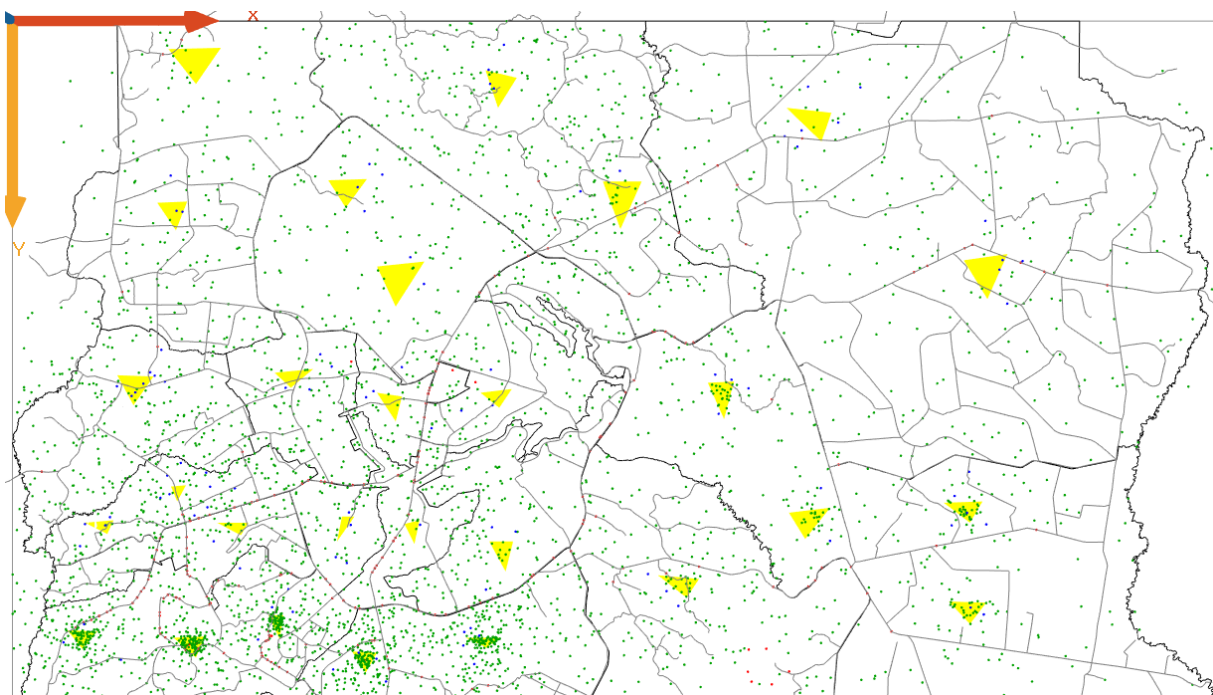


Figura 5.9: Mapa após 600 ciclos de simulação do modelo DDCDF

jogador(ator) não necessariamente corresponde à perda dos outros atores. No modelo de Dinâmica de doença e cura no entorno das Unidades Básicas de Saúde do DF (DDCDF)

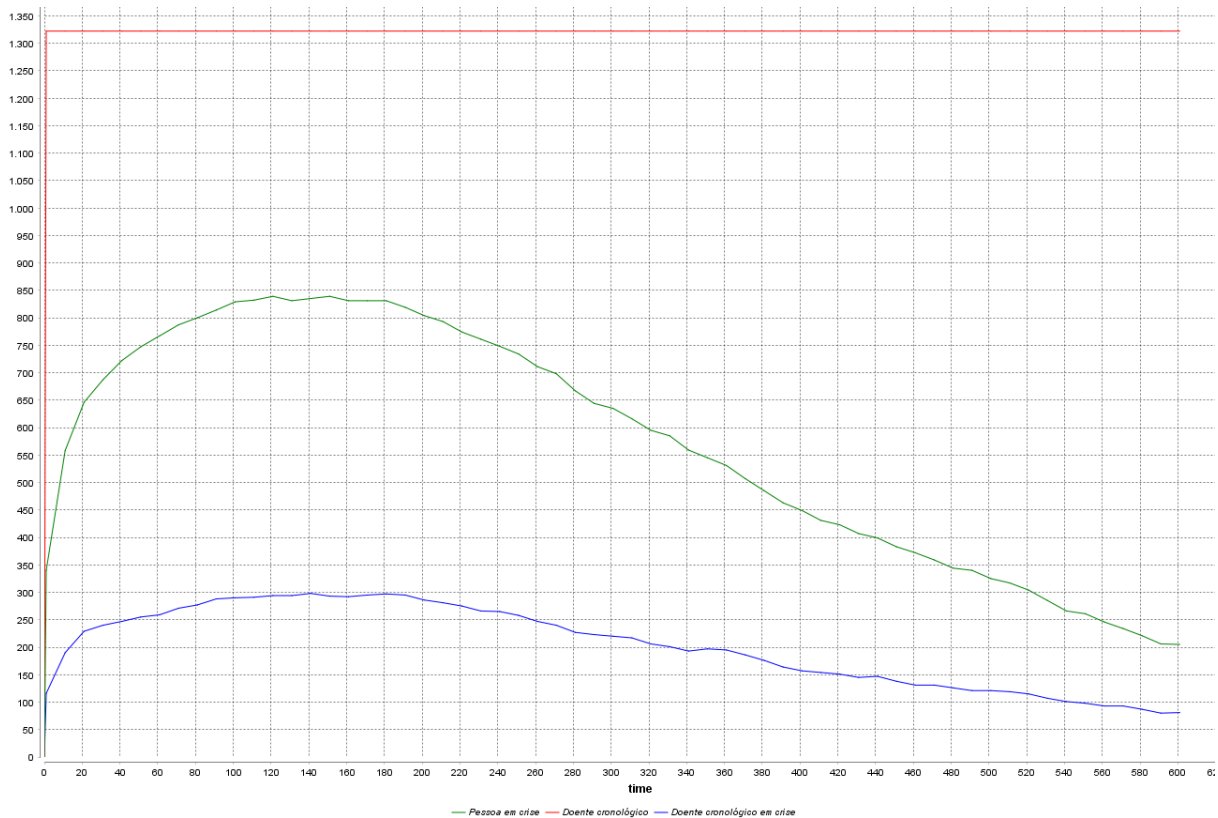


Figura 5.10: Doentes após 600 ciclos de simulação do modelo DDCDF

apresenta a utilização de uma inteligência artificial básica, onde cada pessoa ao perceber que está doente se dirige para uma unidade de saúde.

Não foram adotadas premissas de racionalidades como descrito em [25], pois não há comportamento ideal definido para os atores; logo foi adotado premissas heurísticas obtidas a partir de observações e de estudo prévio acerca dos atores e relacionamentos presentes nos modelos.

Em ambos os modelos os atores computacionais foram definidos de acordo com algumas características como:

- São autônomos e interagirem entre-si, apresentando diferenças apenas em quais ações são possíveis de se realizar.
- Possuem apenas visualização local, nenhum agente possui acesso ou tem conhecimento de todo o sistema .
- São descentralizados, nenhum agente é designado como controlador do sistema.

No modelo GTSSE, os atores decidem se é melhor colaborarem ou não de acordo com os estados iniciais definidos para um, permitindo que observemos como ocorre o *tradeoff* entre atores envolvidos em alguma relação. E no modelo DDCDF os atores podem decidir

se é necessário ir para uma unidade de saúde ou se podem continuar com a sua rotina normal.

Os agentes foram modelados de acordo com primitivas simples e com a limitação cognitiva. No modelo GTSSE, os atores só podem escolher se colaboram ou não atores os quais já possuem relacionamento, não sendo possível criarem novas relações. No modelo DDCDF os atores só procuram as unidades de saúde dentro da região onde estão inseridos, quando na realidade as pessoas tendem a procurar as unidades mais próximas de onde se encontram.

Em ambos os modelos houve a definição de agentes ativos, os quais possuem poder de decisão sobre suas ações como por exemplo: Agentes de saúde, pessoas, Fiocruz, UnB. Os únicos agentes passivos foram definidos no modelo DDCDF (ubs, rodovias, DF) e não tem poder de decisão sobre nenhuma ação e apenas são alvos de ações dos outros atores.

Como apresentado em [24] os ambientes de simulação possuem algumas propriedades que permitem classifica-los.

Por possuir como foco de experimento a relação entre atores, o ambiente do modelo GTSSE possui pouco detalhamento e importância, por isso, o ambiente não apresenta acessibilidade, dinâmica e dimensionalidade. Porém apresenta características como:

- Determinismo: Onde pode-se observar como a decisão de um ator influencia a satisfação de outro.
- Discretude: Pois há um limite de 2 ações para cada ator.
- Episodicidade: Os atores tem o poder de determinar se uma relação foi benéfica ou não, e com isso ter uma base para a próxima tomada de decisão.

O ambiente modelado no SocLab não apresenta acessibilidade (pois não é possível obter algumas informações sobre o ambiente como por exemplo: número de pessoas atendidas em uma determinada UBS) e dinamismo (nenhum agente tem a capacidade de influenciar o ambiente em si). Mas possui as características de :

- Determinismo: As ações dos agentes de saúde causam um efeito definido de reduzir os fatores de risco das pessoas, a decisão de ir para uma UBS quando doente provoca o efeito de cura.
- Discretude: Os agentes de saúde só possuem a ação de se mover pelo mapa, enquanto as pessoas podem se mover aleatoriamente pelo mapa ou se locomoverem em direção a uma UBS.
- Episodicidade: Quando uma pessoa decide ir para a UBS, ela ficará parada na UBS até que esteja curada, e só então poderá tomar outras decisões.

- Dimensionalidade: Os fatores do ambiente como rodovias e regiões administrativas influenciam a forma como os atores tomam suas decisões tanto na hora de escolher para qual UBS deverá se dirigir, qual rodovia será utilizada, e os agentes de saúde só podem se locomover dentro da sua região administrativa.

Neste trabalho as regras comportamentais foram definidas para que fosse possível a identificar a presença ou não de padrões comportamentais nos atores durante as simulações até que fosse atingindo um critério de parada, que no caso do modelo GTSSE foi a convergência dos resultados e no modelo DDCDF foi a execução de 100 ciclos de simulação.

Os modelos foram desenvolvidos de forma a incluir contextos sociais, os quais tem como função emular a segregação dos atores, que no caso do modelo DDCDF as pessoas são divididas por faixa etária e se estão saudáveis ou não, e no modelo GTSSE as pessoas são segregadas através de redes de interação como descrito em [36]. Porém não foram adicionados muitos domínios, pois um grande número destes poderia provocar dificuldades em interpretar os dados resultantes da simulação.

Além da inclusão de contexto social, os agentes apresentam heterogeneidade em relação a localização onde estão inseridos (apenas no Modelo DDCDF), habilidades (o que reflete quais ações podem ser realizadas), preferências (definidas no modelo GTSSE em forma de *solidarities*) e de recompensas (definidos como função matemática em *effects* no modelo GTSSE).

Outro fator importante no desenvolvimento de modelos computacionais é a questão da quantidade de agentes presentes na simulação, dado que um grande número pode gerar problemas de execução devido ao alto esforço computacional. Fato o qual foi observado neste trabalho, gerando assim a necessidade de um refinamento dos modelos para definir uma quantidade adequada de agentes a qual não afetaria o resultado, mas que também pudesse ter uma execução viável aos equipamentos disponíveis. Seguindo o mesmo intuito de diminuir o esforço computacional, os agentes do modelo DDCDF foram configurados para interagirem aleatoriamente entre si, considerando apenas a tipo do agente e seus estados (doente, saudável). Com o refinamento do modelo e conseqüentemente melhoria na performance da simulação foi possível a definição de regras compostas, as quais utilizam memória dos estados anteriores para definir o estado atual, proporcionando assim uma maior fidelidade dos modelos desenvolvidos.

Apesar do modelo GTSSE ter utilizado o Dilema do Prisioneiro como base, ao contrário deste também foram empregadas regras adaptativas nas quais os agentes podem aprender com as suas experiências passadas e julgar se uma relação trouxe ou não benefícios e com isso definir qual a ação será tomada nos próximos ciclos. As regras fixas foram definidas para o cálculo do nível de satisfação, importância dada a uma relação e no comportamento de todos os agentes do modelo DDCDF.

As regras adaptativas empregadas também podem ser classificadas como regras de aprendizado individuais, as quais aumentam as chances de se encontrar o Equilíbrio de Nash, o qual pode indicar uma convergência dos resultados da simulação do modelo. O Equilíbrio de Nash consiste no fenômeno em que os atores buscam realizar ações as quais trazem para si mesmos os melhores benefícios e por isso nenhum tem a ganhar mudando a sua estratégia. Para o Equilíbrio de Nash não é necessário que os atores cooperem, porém na simulação do modelo do GTSSE o equilíbrio e convergência foram alcançados através da cooperação entre todos atores, fato que significa que dentre todas as opções, a de colaborar gera um melhor benefício individual.

Assim como nos primeiros modelos ABM, o modelo DDCDF também aplica regras de forma simultânea e foi utilizado para procurar informações explicativas sobre o comportamento dos atores no sistema natural que é o sistema de saúde público.

A aplicação das regras simultâneas resultou em um padrão onde todos os atores doentes se dirigiam instantaneamente (após um ciclo) para as unidades de saúde e lá permanecessem sem se locomover, com isso o valor dos sintomas não sofria alterações e a pessoa nunca mais ficava doente. Para corrigir este problema, foi limitado o raio e a velocidade de locomoção dos atores.

Capítulo 6

Conclusão

Este trabalho teve como pergunta de pesquisa a seguinte questão:

É possível aplicar a teoria e ferramentas de modelagem e simulação multiagente para investigar as interações presentes no sistema de saúde oferecido à uma população de um território, e na tomada de decisões na atenção primária à saúde no cuidado das condições crônicas do DF, e com isto obter as estratégias que maximizem os resultados desejados?

Foram obtidos resultados positivos quanto a utilização e aplicação da teoria de Modelagem e Simulação de Agentes para investigação das interações presentes na Governança Territorial, equilíbrio de poder entre os agentes e deslocamento de agentes em um território específico.

Apesar de ser possível a obtenção de sugestões de políticas públicas a partir da análise dos cenários obtidos nas simulações, este trabalho não gerou tomada de decisões. Pois para gerar tal efeito seria necessário um maior aprofundamento com aplicações práticas e participação de usuários, os quais de fato têm o poder para tomar tais decisões, no entanto esses requerimentos estão fora do escopo do projeto.

Para uma melhor compreensão do problema enfrentado pela saúde pública no Distrito Federal, é recomendado a utilização de mais de um software de simulação de modelos multiagentes uma vez que cada um possui aspectos positivos e negativos diferentes, sendo que determinados softwares se comportam melhor em cenários diferentes.

Por isso se faz necessário um estudo mais aprofundado sobre quais são as esferas que influenciam a tomada de decisões pertinentes ao sistema público de saúde do Distrito Federal, de forma que uma vez identificado estes espaços também seja feito uma análise estrutural e organizacional de cada um com a finalidade de se construir modelos menores.

Com a junção dos resultados obtidos tanto em simulações mais genéricas e globais como em simulações específicas e locais é possível obter uma visão mais completa do sistema como um todo e com isto definir políticas públicas realmente efetivas.

6.1 Propostas de Trabalhos Futuros

De acordo com a matriz de prevenção das condições crônicas os grupos populacionais são divididos em crianças, adolescentes, adultos, idosos, grupos vulneráveis e outros [65].

Portanto o modelo desenvolvido deveria contemplar mais faixas etárias, sendo que para implementar esta alteração é necessário apenas incluir a porcentagem da população que se encaixa em cada um dos grupos e a probabilidade de serem doentes crônicos. O pseudocódigo para tal mudança se encontra abaixo.

```
1 //incluir nas variaveis
2
3 float proba_crianca <- 0.212;
4 float proba_adolescente <- 0.190;
5 float proba_vulneravel <- 0.065;
6 float crianca_crono <- 0.113;
7 float adolescente_crono <- 0.163;
8 float vulneravel_crono <- 0.551;
```

Listing 6.1: Probabilidades para classificação dos agentes people

```
1 //incluir em todos create people number:
2
3 faixa_etaria <- rnd_choice([proba_crianca,proba_adolescente,proba_adulto
4 ,proba_idoso,proba_vulneravel]);
5
6 if (faixa_etaria = 1){
7     adulto <- true;
8     cronologico <- rnd_choice([1-crianca_crono,crianca_crono]);
9     if(cronologico = 1){
10        sick_crono <- true;
11    }
12 }
13 if (faixa_etaria = 2){
14     adulto <- true;
15     cronologico <- rnd_choice([1-adolescente_crono,adolescente_crono
16 ]);
17     if(cronologico = 1){
18        sick_crono <- true;
19    }
20 }
21 if (faixa_etaria = 5){
22     adulto <- true;
23     cronologico <- rnd_choice([1-vulneravel_crono,vulneravel_crono])
24 ;
25     if(cronologico = 1){
```

```

24     sick_crono <- true;
25   }
26 }

```

Listing 6.2: Definindo randomicamente quais atores são doentes crônicos de acordo com as probabilidades para cada classificação de people

No modelo não foi considerado os fatores de risco que quando atuam durante muito tempo podem levar uma pessoa a desenvolver alguma doença crônica. Tal cenário pode ser implementado com a definição de variáveis para os fatores de risco, sendo que deve-se definir pesos para cada um de forma a calcular o valor da taxa morbidade que juntamente com o tempo de exposição aos fatores de risco nos dão a probabilidade da pessoa desenvolver alguma doença crônica.

```

1 //incluir nas variaveis
2
3 float proba_classe_fumar <- x;
4 float proba_classe_alimen_inade <- y;
5 float proba_classe_sobrepeso <- z;
6 float proba_classe_sedentarismo <- w;
7 float proba_classe_alcool <- t;
8
9 //Onde classe pode ser crianca, adolescente, adulto, idoso ou vulneravel
.
10 // x, y, z, w, t sao as probabilidades dos fatores de risco acometerem
    uma pessoa em uma das classes.

1 //adicionar em species people
2
3 float morbidade;

```

Listing 6.3: Variável utilizada para calcular a morbidade de cada agente

```

1 //adicionar em species people
2
3 int test_tabagismo;
4 int test_alimentacao_inadequada;
5 int test_sobrepeso;
6 int test_sedentarismo;
7 int test_excesso_alcool;
8
9 bool tabagismo;
10 bool alimentacao_inadequada;
11 bool sobrepeso;
12 bool sedentarismo;

```

```
13 bool excesso_alcool;
```

Listing 6.4: código para verificação se o agente possui algum dos fatores de risco

```
1 //adicionar em species people
2
3 int tempo_tabagismo;
4 int tempo_alimentacao_inadequada;
5 int tempo_sobrepeso;
6 int tempo_sedentarismo;
7 int tempo_excesso_alcool;
```

Listing 6.5: Variáveis para definir a quanto tempo o agente possui um determinado fator de risco

```
1 //adicionar em species people
2
3 int quant_tabagismo;
4 int quant_alimen_inadequada;
5 int quant_sobrepeso;
6 int quant_sedentarismo;
7 int quant_excesso_alcool;
```

Listing 6.6: Variáveis para definir em qual o nível ou quantidade de consumo dos fatores de risco

```
]
```

```
1 //adicionar em species people
2
3 reflex risco {
4
5     if(tabagismo = true){
6         tempo_tabagismo <- tempo_tabagismo + 1;
7     }
8     if(alimentacao_inadequada = true){
9         tempo_alimen_inadequada <- tempo_alimen_inadequada + 1;
10    }
11    if(sobrepeso = true){
12        tempo_sobrepeso <- tempo_sobrepeso + 1;
13    }
14    if(sedentarismo= true){
15        tempno_sedentarismo <- tempo_sedentarismo + 1;
16    }
17    if(excesso_alcool = true){
18        tempo_excesso_alcool <- tempo_excesso_alcool + 1;
19    }
```

Listing 6.7: Atualização do tempo que os agentes possuem os fatores de risco

```

]
1 //adicionar em species people
2
3 morbidade <- tempo_tabagismo * quant_tabagismo * fator +
4 tempo_alimen_inadequada * quant_alimen_inadequada * fator +
5 tempo_sobrepeso * quant_sobrepeso * fator +
6 tempo_sedentarismo * quant_sedentarismo * fator +
7 tempo_excesso_alcool * quant_excesso_alcool * fator;
8
9 if(morbidade > valor){
10     cronico <- true;
11 }
12 }

```

Listing 6.8: Código para calculo do fator de morbidade de cada agente

```

1 //Alterar em todos create people number:
2
3
4 if (faixa_etaria = x){ // Com x podendo variar entre 0 e 4
5
6     classe <- true; // Classe pode ser crianca, adolescente, adulto,
7     idoso ou vulneravel de acordo com o valor de x.
8     cronologico <- rnd_choice([1-classe_crono, classe_crono]);
9     if(cronologico = 1){
10         sick_crono <- true;
11     }
12     teste_tabagismo <- rnd_choice (1-proba_classe_fumar,
13     proba_classe_fumar)
14     if (teste_tabagismo = 1){
15         tabagismo <- true;
16         quant_tabagismo <- rnd(1,5);
17     }
18     test_alimentacao_inadequada <- rnd_choice (1-
19     proba_classe_alimen_inade,
20     proba_classe_alimen_inade)
21     if (test_alimentacao_inadequada = 1){
22         alimentacao_inadequada <- true;
23         quant_alimen_inadequada; <- rnd(1,5);
24     }
25     test_sobrepeso <- rnd_choice (1-proba_classe_sobrepeso,
26     proba_classe_sobrepeso)
27     if (test_sobrepeso = 1){
28         sobrepeso <- true;
29         quant_sobrepeso; <- rnd(1,5);
30     }
31 }

```

```

28     test_sedentarismo <- rnd_choice (1-proba_classe_sedentarismo ,
29     proba_classe_sedentarismo)
30     if (test_sedentarismo = 1){
31         sedentarismo <- true;
32         quant_sedentarismo; <- rnd(1,5);
33     }
34     test_excesso_alcool <- rnd_choice (1-proba_classe_alcool ,
35     proba_classe_alcool)
36     if (test_excesso_alcool = 1){
37         excesso_alcool <- true;
38         quant_excesso_alcool; <- rnd(1,5);
39     }
40 }
41 }

```

Listing 6.9: Definição se os atores possuem fatores de riscos e se são doentes crônicos ou não

Outra ação necessária é a definição de como o governo pode adotar estratégias para diminuir o índice de morbidade dos diferentes setores da população e em diferentes escalas. Primeiramente deve-se definir faixa etária da população e qual fator de risco será alvo da campanha, de forma que apenas as pessoas que se encaixem nos requisitos sejama afetadas, também deve-se definir a propabilidade de que a ação governamental irá funcionar e em seguida reduzir os valores do risco de morbidade das pessoas.

Para melhorar tornar a simulação mais realista também pode-se considerar a adição de alguns tipos de doenças crônicas, com a finalidade de definir diferentes tipos de tratamentos e as necessidades das pessoas.

Referências

- [1] World Health Organization. *Time to deliver: report of the WHO Independent high-level commission on noncommunicable diseases*. World Health Organization, Geneva, 2018. 1
- [2] OPAS/OMS. OPAS/OMS Brasil - Comissão da OMS pede ação urgente contra doenças crônicas não transmissíveis, June 2018. 1
- [3] Christophe Sibertin-Blanc, Pascal Roggero, Françoise Adreit, Bertrand Baldet, Paul Chapron, Joseph El-Gemayel, Matthias Mailliard, and Sandra Sandri. SocLab: A Framework for the Modeling, Simulation and Analysis of Power in Social Organizations. *Journal of Artificial Societies and Social Simulation*, 16(4):8, 2013. 2, 4, 6, 17, 19, 20, 21, 22, 23
- [4] Eugênio Vilaça Mendes. *O cuidado das condições crônicas na atenção primária à saúde: o imperativo da consolidação da estratégia da saúde da família*. Organização Pan-Americana da Saúde, Brasília, 2012. 2, 4, 14, 15
- [5] Lucas Labigalini Fuini. Compreendendo a governança territorial e suas possibilidades: Arranjos Produtivos Locais (APL) e circuitos turísticos. *Interações (Campo Grande)*, 13(1):93–104, June 2012. Publisher: Editora da Universidade Católica Dom Bosco. 2
- [6] Jorge Mesquita Huet Machado, Wagner de Jesus Martins, Maria do Socorro de Souza, André Luiz Dutra Fenner, Missifany Silveira, and Aletheia de Almeida Machado. Territórios saudáveis e sustentáveis: contribuição para saúde coletiva, desenvolvimento sustentável e governança territorial. *Comunicação em ciências da saúde*, 28(2):243-249, 2017. 3, 4, 44
- [7] Fiocruz. Atuação da Fiocruz Brasília em Territórios Saudáveis e Sustentáveis é destaque no 2º Ciclo de Debates de Saúde, Ambiente e Sustentabilidade, April 2019. 3, 4, 44
- [8] ONU. Transformando Nosso Mundo: A Agenda 2030 para o Desenvolvimento Sustentável, October 2015. 3
- [9] Edmundo Gallo and Andréia Faraoni Freitas Setti. Abordagens ecossistêmica e comunicativa na implantação de agendas territorializadas de desenvolvimento sustentável e promoção da saúde. *Ciência & Saúde Coletiva*, 17(6):1433–1446, June 2012. 3, 13

- [10] Ilona Kickbusch and David Gleicher. *Governance for Health in the 21st Century*. World Health Organization, Denmark, 2012. 3
- [11] David V. McQueen, Matthias Wismar, Vivian Lin, Catherine M. Jones, and Maggie Davies. *Intersectoral Governance for Health in All Policies: Structures, actions and experiences*. World Health Organization, Switzerland, 2012. 3
- [12] Gemma Carey and Sharon Friel. Understanding the Role of Public Administration in Implementing Action on the Social Determinants of Health and Health Inequities. *International Journal of Health Policy and Management*, 4(12):795–798, 2015. 3
- [13] GAMA Platform. GAMA-Platform, 2020. 4, 6, 17
- [14] Nigel Gilbert and Pietro Terna. How to build and use agent-based models in social science. *Mind & Society*, 1(1):57–72, March 2000. 4, 9, 10
- [15] Paulo Eduardo Guedes Sellera, Maria Francisca Santos Abritta Moro, Reginaldo de Holanda e Albuquerque, Lua Isis Braga, Maria do Socorro de Souza, Antônia Sheila Gomes Lima, and Wagner de Jesus Martins. A ativação de redes sociotécnicas na cidade estrutural/DF, Brasil:construindo um território saudável e sustentável. *Ciência & Saúde Coletiva*, 24:2185–2191, 2019. 4
- [16] Fiocruz. Cidade do Futuro - Estrutural Saudável e Educadora, July 2019. 4
- [17] Camilo Olaya and Michael Ruess. The Sociological Theory of Crozier and Friedberg on Organized Action Seen Through a Simulation Model. *InterJournal of Complex Systems*, (1469, CX.4 Human Systems):1–14, 2004. 4, 17, 18
- [18] Isaac Epstein. O dilema do prisioneiro e a ética. *Estudos Avançados*, 9(23):149–163, April 1995. 4, 7, 11
- [19] Ronaldo Fiani. *Teoria Dos Jogos: com aplicações em economia, administração e ciências sociais*. Campus, Rio de Janeiro, 2009. 4, 8, 10, 11
- [20] John Holland. *Hidden order: how adaptation builds complexity*. Helix Books, Reading, Mass, 1995. 7
- [21] Joshua M. Epstein. Why Model? *Journal of Artificial Societies and Social Simulation*, 11(4):12, October 2008. 7
- [22] Michael Wooldridge. *An Introduction to MultiAgent Systems*. John Wiley & Sons, June 2009. 8, 9
- [23] Robert Axelrod and Leigh Tesfatsion. On-Line Guide for Newcomers to Agent-Based Modeling in the Social Sciences, January 2020. 8, 64
- [24] Stuart J. Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Prentice Hall series in artificial intelligence. Prentice Hall, Englewood Cliffs, N.J, 1995. 8, 9, 67

- [25] Wulfrano Luna-Ramirez and Maria Fasli. Bridging the Gap between ABM and MAS: A Disaster-Rescue Simulation Using Jason and NetLogo. *Computers*, 7(2):24, April 2018. 8, 66
- [26] Shu-Heng Chen. Varieties of agents in agent-based computational economics: A historical and an interdisciplinary perspective. *Journal of Economic Dynamics and Control*, 36(1):1–25, January 2012. 8
- [27] Tatiana Massaroli Melo and José Ricardo Fucidji. Racionalidade limitada e a tomada de decisão em sistemas complexos. *Revista de Economia Política*, 36(3):622–645, September 2016. 8
- [28] Colin F. Camerer. Behavioural studies of strategic thinking in games. *Trends in Cognitive Sciences*, 7(5):225–231, May 2003. 8
- [29] Liviu Panait and Sean Luke. Cooperative Multi-Agent Learning: The State of the Art. *Autonomous Agents and Multi-Agent Systems*, 11(3):387–434, November 2005. 8, 9
- [30] Andrio dos Santos Pinto. *Simulação e avaliação de comportamentos em sistemas multi-agentes baseados em modelos de reputação e interação*. Dissertação de Mestrado, UNISINOS, RS, February 2008. 8
- [31] J. Ferber and O. Gutknecht. A meta-model for the analysis and design of organizations in multi-agent systems. In *Proceedings International Conference on Multi Agent Systems (Cat. No.98EX160)*, pages 128–135, July 1998. 8
- [32] Yoann Kubera, Philippe Mathieu, and Sébastien Picault. Everything Can Be Agent! In *Proceedings of the 9th International Conference on Autonomous Agents and Multi-agent Systems: Volume 1 - Volume 1*, AAMAS '10, pages 1547–1548, Richland, SC, 2010. International Foundation for Autonomous Agents and Multiagent Systems. event-place: Toronto, Canada. 9
- [33] Jacinto Davila and Kay Tucci. Towards a Logic-Based, Multi-Agent Simulation Theory. In *In International Conference on Modeling, Simulation and Neural Networks MSNN'2000, IEEE*, pages 37–51, 2000. 9
- [34] Peter D. Hoff and Michael D. Ward. Modeling Dependencies in International Relations Networks. *Political Analysis*, 12(2):160–175, 2004. 9
- [35] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, January 2003. arXiv: cond-mat/0303516. 9
- [36] David Lazer. The co-evolution of individual and network. *The Journal of Mathematical Sociology*, 25(1):69–108, 2001. 9, 68
- [37] Alexis Drogoul, Diane Vanbergue, and Thomas Meurisse. Multi-agent Based Simulation: Where Are the Agents? In Jaime Simão Sichman, François Bousquet, and Paul Davidsson, editors, *Multi-Agent-Based Simulation II*, volume 2581, pages 1–15. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003. 10

- [38] Scott de Marchi and Scott E. Page. Agent-Based Models. *Annual Review of Political Science*, 17(1):1–20, May 2014. 10
- [39] Muaz Niazi and Amir Hussain. Agent-based computing from multi-agent systems to agent-based models: a visual survey. *Scientometrics*, 89(2):479–499, November 2011. 10
- [40] Nigel Gilbert. Agent-based social simulation: dealing with complexity. Technical report, Complex Systems Network of Excellence, EU, 2004. 10
- [41] John H. Miller and Scott E. Page. *Complex Adaptive Systems: An Introduction to Computational Models of Social Life*. Princeton University Press, student edition edition, 2007. 11
- [42] Nicolaas J Vriend. An illustration of the essential difference between individual and social learning, and its consequences for computational analyses. *Journal of Economic Dynamics and Control*, 24(1):1–19, January 2000. 11
- [43] Paulo José de A. L. de Almeida, Maja Kajin, and Marcus V. Vieira. Equilíbrio de Nash e Estratégias Evolutivamente Estáveis: A teoria dos jogos na ecologia de populações. *Oecologia Australis*, 16(1):127–140, 2012. 11
- [44] Richard J. Gaylord and Louis J. D’Andria. Modeling a Society of Mobile Heterogeneous Individuals. In Richard J. Gaylord and Louis J. D’Andria, editors, *Simulating Society: A Mathematica® Toolkit for Modeling Socioeconomic Behavior*, pages 3–17. Springer, New York, NY, 1998. 11
- [45] Paul Davidsson. Multi Agent Based Simulation: Beyond Social Simulation. In Scott Moss and Paul Davidsson, editors, *Multi-Agent-Based Simulation*, volume 1979, pages 97–107. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000. 12
- [46] Wagner Martins. Saúde como sustentabilidade do desenvolvimento. page 17, 2016. 13
- [47] Leo Kissler and Francisco G. Heidemann. Governança pública: novo modelo regulatório para as relações entre Estado, mercado e sociedade? *Rev. Adm. Pública*, 40(3):479–499, June 2006. 13
- [48] IBGE. IBGE:Projeção da população. site: <https://www.ibge.gov.br/apps/populacao/projecao/>, 2020. 13
- [49] Fausto Brito. A transição demográfica no Brasil: as possibilidades e os desafios para a economia e a sociedade. Technical report, Cedeplar, Universidade Federal de Minas Gerais, Minas Gerais, 2007. 14
- [50] Deborah Carvalho Malta, Silvânia Suely Caribé de Araújo Andrade, Taís Porto Oliveira, Lenildo de Moura, Rogério Ruscitto do Prado, and Maria de Fátima Marinho de Souza. Probabilidade de morte prematura por doenças crônicas não transmissíveis, Brasil e regiões, projeções para 2025. *Rev. bras. epidemiol.*, 22:e190030, 2019. 14

- [51] Organisation mondiale de la santé. *Global status report on noncommunicable diseases 2014: attaining the nine global noncommunicable diseases targets; a shared responsibility*. World Health Organization, Geneva, 2014. OCLC: 907517003. 14
- [52] Barbara Starfield. Efetividade e Equidade em Serviços de Saúde através da Atenção Primária em Saúde. In *Anais*, Recife - PE, December 2007. Ministério da Saúde. 14
- [53] Eugênio Vilaça Mendes. As Redes de Atenção à Saúde no SUS, September 2009. 14
- [54] Andréia Patrícia Gomes, José Roberto Bittencourt Costa, Túlio da Silva Junqueira, Mariana Beatriz Arcuri, and Rodrigo Siqueira-Batista. Atenção primária à saúde e formação médica: entre episteme e práxis. *Revista Brasileira de Educação Médica*, 36(4):541–549, December 2012. Publisher: Associação Brasileira de Educação Médica. 15
- [55] Elhadi Shakshuki and Malcolm Reid. Multi-Agent System Applications in Healthcare: Current Technology and Future Roadmap. *Procedia Computer Science*, 52:252–261, 2015. 16
- [56] Dante I. Tapia and Juan M. Corchado. An Ambient Intelligence Based Multi-Agent System for Alzheimer Health Care. *International Journal of Ambient Computing and Intelligence (IJACI)*, 1(1):15–26, 2009. 16
- [57] W. WILENSKY, U.; RAND. An introduction to agent-based modeling: Modeling natural, social and engineered complex systems with netlogo. 2015. 17
- [58] P NORTH NICHOLSON T. COLLIER, J. O. E. R. T. C. M. M. M. B. M. J.; SYDELKO. Complex adaptive systems modeling with repast simphony. complex adaptive systems modeling, v. 1, n. 3, 2013. 2013. 17
- [59] SocLab : the Sociology Laboratory. Library Catalog: sourceforge.net. 17
- [60] SocLab Project. SocLab : the Sociology Laboratory, 2016. 17
- [61] Christophe Sibertin-Blanc, Frédéric Amblard, and Matthias Mailliard. A Coordination Framework Based on the Sociology of Organized Action. pages 3–17, 2005. 17, 18
- [62] Françoise Adreit, Pascal Roggero, Christophe Sibertin-Blanc, and Claude Vautier. Using Soclab for a Rigorous Assessment of the Social Feasibility of Agricultural Policies, 2012. 18
- [63] Marcos Aurélio Santos da Silva. O Território Como um Sistema Social Complexo. In Bernardo Alves Furtado, Patrícia A. M. Sakowski, and Marina H. Tóvolli, editors, *Modelagem de Sistemas Complexos Para Políticas Públicas*, pages 403–436. IPEA, Brasília - DF, 2015. 18
- [64] Examples, April 2012. Library Catalog: soclabproject.wordpress.com. 24
- [65] Eugênio Vilaça Mendes. O cuidado das condições crônicas na atenção primária à saúde. *RBPS*, 31(2):1–3, June 2018. 71