

Universidade Federal de Santa Catarina
Departamento de Informática e Estatística

NATHAN FROIS PEREIRA PAIVA

**Pesquisa em simulação de evacuação de
multidões utilizando Incremental Neuro-Fuzzy
Gaussian Mixture Network**

Florianópolis - Santa Catarina

2018 / 1

NATHAN FROIS PEREIRA PAIVA

**Pesquisa em simulação de evacuação de multidões
utilizando Incremental Neuro-Fuzzy Gaussian Mixture
Network**

Trabalho de Conclusão de Curso apresentado
à Universidade Federal de Santa Catarina
como parte do requisito de obtenção de grau
de Bacharel em Ciências da Computação.

Universidade Federal de Santa Catarina
Departamento de Informática e Estatística

Orientador: Prof. Dr. Mauro Roisenberg
Coorientador: Prof. Ms. Tiago Mazzutti

Florianópolis - Santa Catarina
2018 / 1

NATHAN FROIS PEREIRA PAIVA

**Pesquisa em simulação de evacuação de multidões
utilizando Incremental Neuro-Fuzzy Gaussian Mixture
Network**

Trabalho de Conclusão de Curso apresentado
à Universidade Federal de Santa Catarina
como parte do requisito de obtenção de grau
de Bacharel em Ciências da Computação.

Prof. Dr. Mauro Roisenberg
Orientador

Prof. Ms. Tiago Mazzutti
Coorientador

Prof. Dr. Elder Rizzon Santos
Membro da Banca

Profa. Dra. Jerusa Marchi
Membro da Banca

Florianópolis - Santa Catarina
2018 / 1

Resumo

Modelar e simular o comportamento humano quando em multidões é essencial para compreender o que ocorre com os indivíduos em situações de pânico. Estas informações são fundamentais para que um arquiteto possa, por exemplo, projetar um espaço fechado da maneira mais segura possível, proporcionando uma circulação mais confortável e o rápido deslocamento dos pedestres até a saída do local em caso de emergência. Helbing elaborou um modelo chamado *Social Force Model*, baseado nos princípios de propagação de partículas, que representa esses comportamentos através de equações diferenciais. Neste modelo, cada indivíduo é uma partícula com uma massa inserida em um ambiente, no qual existem: uma força de atração que atrai a partícula para o ponto de saída, outra que repele ela de obstáculos próximos e também outra ainda que a repele de partículas próximas. Essa pesquisa visa utilizar comportamentos de evacuação simulados através do *Social Force Model*, os quais servem de base para o treinamento de redes neuro-fuzzy. Depois de treinados, estes comportamentos podem ser interpretados como conjuntos fuzzy e regras de um sistema de inferência fuzzy sobre estes conjuntos.

Palavras-chave: *Social Force Model*, simulação de evacuação de multidões, rede neural, lógica fuzzy, redes neuro-fuzzy, *Incremental Neuro-Fuzzy Gaussian Mixture Network*.

Abstract

Modeling and simulating human behavior in crowds is essential to understanding what happens to individuals in panic situations. This information is fundamental so that an architect can, for example, design an enclosed space in the safest way possible, providing a more comfortable circulation and the rapid displacement of the pedestrians until the exit of the place. Helbing has developed a model called Social Force Model based on the principles of particle propagation, which represents these behaviors through differential equations. In this model, each individual is a particle with a mass inserted in an environment where there is a force of attraction that attracts the particle to the point of exit, another that repels it from nearby obstacles and also another that repels it from nearby particles. This research aims to use simulated evacuation behaviors through the Social Force Model, which serve as the basis for the training of neuro-fuzzy systems. Once trained, these behaviors can be interpreted as fuzzy sets and rules of a fuzzy inference system on these sets.

Keywords: Social Force Model, simulation of crowd evacuation, artificial neural networks, fuzzy logic, neuro-fuzzy systems, Incremental Neuro-Fuzzy Gaussian Mixture Network.

Sumário

	Lista de ilustrações	8
	Lista de tabelas	10
	Lista de abreviaturas e siglas	11
	Lista de símbolos	12
1	INTRODUÇÃO	15
1.1	Objetivos	17
1.1.1	Objetivo geral	17
1.1.2	Objetivos específicos	17
1.2	Metodologia de pesquisa	17
1.3	Estrutura do Trabalho	18
2	SOCIAL FORCE MODEL	19
2.1	Definição	19
2.1.1	Força de Desejo	20
2.1.2	Forças de Repulsão	21
2.1.2.1	Força Social	21
2.1.2.2	Força de Obstáculo	22
2.2	Fenômenos em Multidões	24
2.3	Fenômenos em Evacuações de Multidões	25
3	REDES NEURO-FUZZY	28
3.1	Revisão sobre Sistemas de Inferência Fuzzy	29
3.2	Revisão sobre Redes Neurais Artificiais	30
3.3	INFGMN (Incremental Neuro-Fuzzy Gaussian Mixture Network)	32
3.3.1	GMM (<i>Gaussian Mixture Model</i>)	33
3.3.1.1	IGMM (<i>Incremental Gaussian Mixture Model</i>)	34
3.3.2	Arquitetura da INFGMN	35
3.3.3	Exemplo e considerações Finais	39
4	REVISÃO DE LITERATURA	41
4.1	Modelo para Simulação de Multidões Utilizando Lógica Nebulosa	41
4.2	Proposta de Modelo Fuzzy de Forças Sociais para Simulação de Pedestres em Situação de Evacuação	43

4.3	Multi-agent system for simulating human behavior in egress simulations	44
4.4	Fuzzy Logic-Based Model That Incorporates Personality Traits for Heterogeneous Pedestrians	46
5	IMPLEMENTAÇÃO	48
5.1	Software PANIC SIMULATOR	49
5.1.1	Eixos de força no PANIC SIMULATOR	51
5.2	Modelo proposto	51
5.2.1	Captura de dados	52
5.2.2	Treinamento na MLP	53
5.2.3	Treinamento na INFGMN	54
5.2.4	Base de regras obtida	55
6	EXPERIMENTOS E RESULTADOS	57
6.1	Análise qualitativa	58
6.2	Análise quantitativa	60
6.3	Considerações finais	65
7	CONCLUSÃO	67
7.1	Trabalhos Futuros	68
	REFERÊNCIAS	69
	APÊNDICE A – ARTIGO	72

Lista de ilustrações

Figura 1 – Exemplo de Força de Desejo para o pedestre α e o seu objetivo \vec{r}_α^0	21
Figura 2 – Exemplo de Força em situações de pânico, envolvendo um agente α , um outro agente próximo identificado por β e um obstáculo i	23
Figura 3 – Formação de 4 faixas, no círculo branco pedestres em uma direção, no círculo preto são pedestres na direção oposta (Helbing and Molnár, 1995).	24
Figura 4 – Cruzamento do fluxo pedestres em azul com outro fluxo em laranja (Helbing and Johansson, 2010).	25
Figura 5 – Intersecção do fluxo pedestres em azul com outro fluxo em laranja (Helbing and Johansson, 2010).	25
Figura 6 – Comportamento pastoreiro de pedestres em baixa visibilidade, denotadas pela sequência a,b,c,d. (Helbing and Johansson, 2010).	26
Figura 7 – “Congelamento por Aquecimento” em um fluxo oposto de pedestres (Helbing and Johansson, 2010).	27
Figura 8 – Exemplo de conjuntos fuzzy para um universo de discurso idade.	30
Figura 9 – Exemplo de uma topologia de rede, adaptado de Lin e Lee (1996).	32
Figura 10 – Arquitetura da INFGMN (Mazutti et al., 2017).	36
Figura 11 – Conjuntos fuzzy gaussianos para a base de dados da íris.	40
Figura 12 – Conjuntos fuzzy e na imagem abaixo, a base de regras (Nasser, 2011).	42
Figura 13 – Diagrama para implementação do Modelo Fuzzy de Forças Sociais (Sent, 2016).	44
Figura 14 – Módulos do AvatarSim (Sharma and Singh, 2006).	45
Figura 15 – Modelagem dos agentes (Xiu et al., 2017).	46
Figura 16 – Entradas e saídas dos FIS proposto (Xiu et al., 2017).	47
Figura 17 – Diagrama de atividades para a implementação de simulações com a INFGMN.	48
Figura 18 – Tela principal do PANIC SIMULATOR.	49
Figura 19 – Opções de configuração de parâmetros das equações e de selecionar o número de agentes.	50
Figura 20 – Componentes físicos de uma simulação, onde sua numeração identifica o tipo do componente.	50
Figura 21 – Direção e sentido das forças do SFM no PANIC SIMULATOR.	51
Figura 22 – Regressão da MLP para os conjuntos de treinamento da força no eixo- x e para o eixo- y	54
Figura 23 – Regressão da INFGMN para os conjuntos de treinamento da força no eixo- x e para o eixo- y	55
Figura 24 – Cenários de simulação do PANIC SIMULATOR para INFGMN e MLP.	57

Figura 25 – Sequência de simulação para a MLP, no primeiro cenário e ordenada da gravura <i>a</i> até <i>c</i>	59
Figura 26 – Sequência de simulação para a MLP, no segundo cenário e ordenada da gravura <i>a</i> até <i>f</i>	60
Figura 27 – Sequência de simulação para a INFGMN, no segundo cenário e ordenada da gravura <i>a</i> até <i>d</i>	61
Figura 28 – Sequência de simulação para a INFGMN, no segundo cenário, ordenada de <i>a</i> até <i>b</i>	61
Figura 29 – Posicionamento dos agentes para a coleta de dados estatísticos, para o cenário 1 e 2.	62
Figura 30 – Média das forças no cenário 1 para o eixo- <i>x</i> e eixo- <i>y</i>	63
Figura 31 – Média das forças no cenário 2 para o eixo- <i>x</i> e eixo- <i>y</i>	64
Figura 32 – Tempos de saída para os agentes no cenário 1.	65
Figura 33 – Tempos de saída para os agentes no cenário 2.	65

Lista de tabelas

Tabela 1 – Parâmetros do PANIC SIMULATOR	53
Tabela 2 – Parâmetros para experimentos	58
Tabela 3 – Resultados das forças para o cenário 1	62
Tabela 4 – Resultados das forças para o cenário 2	62
Tabela 5 – Tempos para o cenário 1	64
Tabela 6 – Tempos para o cenário 2	64

Lista de abreviaturas e siglas

SFM	<i>Social Force Model</i>
MLP	<i>multi-layer perceptron</i>
INFGMN	<i>Incremental Neuro-Fuzzy Gaussian Mixture Network</i>
FIS	<i>Fuzzy Inference Systems</i>
ANN	<i>Artificial neural networks</i>
ANFIS	<i>Adaptive-Network-based Fuzzy Inference System</i>
EFuNN	<i>Evolving Fuzzy Neural Network</i>
DENFIS	<i>Dynamic Evolving Neuro-Fuzzy Inference System</i>
UCI	<i>University of California, Irvine</i>
IGMN	<i>Incremental Gaussian Mixture Model</i>
GMM	<i>Gaussian Mixture Model</i>
pdf	<i>probability density function</i>
OCEAN	<i>Openness, Conscientiousness, Extraversion, Agreeableness e Neuroticism</i>

Lista de símbolos

α	Agente referenciado na simulação
β	Agente vizinho de α na simulação
i	Obstáculo inserido na simulação
e	Saída inserida na simulação
v	Velocidade
\vec{r}	Coordenada
r	Raio
t	Instante de tempo atual
eixo- x	Abscissa do plano cartesiano
eixo- y	Ordenada do plano cartesiano
Δt	Variação de tempo
$\vec{f}_\alpha(t)$	Soma das forças sociais sob o agente
$\xi_\alpha(t)$	Flutuações sob o agente
\vec{r}_α ou \vec{r}_β	Coordenada ou o raio do agente
\vec{v}_α ou \vec{v}_β	Velocidade atual
m_α	Massa do agente
v_α^0	Velocidade desejada
\vec{e}_α^0	Direção desejada
τ	Tempo de aceleração
\vec{r}_α^0	Local do objetivo do agente
$\vec{f}_{\alpha\beta}(t)$	Força Social
$\vec{f}_{\alpha i}(t)$	Força de Obstáculo
$d_{\alpha\beta}$ ou $d_{\alpha i}$	Distância
A	Força de interação em Newtons

B	Alcance da força de interação
$\vec{f}_{\alpha\beta}^{ph}$ ou $\vec{f}_{\alpha i}^{ph}$	Força de repulsão para multidões em pânico
$\vec{n}_{\alpha\beta}$ ou $\vec{n}_{\alpha i}$	Vetor de direção
$\vec{t}_{\alpha\beta}$ ou $\vec{t}_{\alpha i}$	Vetor de direção tangencial
$\vec{v}_{\beta\alpha}^t$	Diferença de velocidade tangencial
k	Constante de efeitos de obstrução em interações físicas na direção normal
κ	Constante de efeitos de obstrução em interações físicas na direção tangencial
s	Segundos
N	Newtons
$p(j)$	Probabilidade a priori para o j -ésimo componente de mistura
$p(\mathcal{X} j)$	Probabilidade do vetor de entrada pertencer ao j -ésimo componente de mistura
C	Matriz de covariância
μ	Vetor de médias
sp_j	Probabilidade à posteriori acumulada do j -ésimo componente de mistura
j^*	Novo componente de misturas gaussianas
C^*	Matriz de covariância atualizada
u^*	Vetor de médias atualizado
$p(j)^*$	Probabilidade a priori para o j -ésimo componente de mistura atualizada
sp_j^*	Probabilidade à posteriori acumulada do j -ésimo componente de mistura atualizada
τ_{nov}	Limiar de verossimilhança
t_j	Tempo de vida acumulado do j -ésimo componente gaussiano
t_{max}	Tempo de vida máximo de um componente gaussiano
J	Número de Componentes de mistura gaussianas

D	Dimensões do vetor de entrada
M	Dimensões do vetor de saída
I	Número de variáveis fuzzy de entrada
O	Número de variáveis fuzzy de saída
w_k	k -ésimo peso do neurônio
\mathcal{W}_k	k -ésimo peso da regra fuzzy na INFGMN
\mathcal{R}_k	k -ésima regra fuzzy na INFGMN
\mathcal{X}_d	Valor da d -ésima dimensão na entrada
\mathcal{Y}_m	Valor da m -ésima dimensão na saída
\mathcal{G}_j	j -ésima componente de mistura gaussiana
\mathcal{IV}_i	i -ésima variável fuzzy de entrada
\mathcal{OV}_o	o -ésima variável fuzzy de saída
$\mathcal{IV}_{i(j)}$	j -ésimo conjunto fuzzy para i -ésima variável de entrada
$\mathcal{OV}_{o(j)}$	j -ésimo conjunto fuzzy para o -ésima variável de saída
$f_{learning}$	Modo de operação de aprendizagem (<i>learning</i>)
$f_{recalling}$	Modo de operação de recordação (<i>recalling</i>)
$\phi^{\mathcal{OV}_o}$	Função de pertinência para a o -ésima variável fuzzy de saída no modo <i>recalling</i>
$\phi^{\mathcal{IV}_i}$	Função de pertinência para a i -ésima variável fuzzy de entrada no modo <i>recalling</i>
a_k	Força de disparo da k -ésima regra fuzzy da INFGMN
y_m	Universo de discurso da m -ésima variável de saída
$crisp_{y_m}$	Valor crisp para m -ésima variável de saída

1 Introdução

Modelar e simular o comportamento humano quando em multidões é essencial para compreender o que ocorre com os indivíduos em situações de pânico, como um ataque terrorista ou um incêndio. Isto é fundamental para que um arquiteto possa, por exemplo, projetar um espaço fechado da maneira mais segura possível, evitando congestionamento em evacuações, uma circulação mais confortável e o rápido deslocamento dos pedestres até a saída do local.

O estudo do comportamento de multidões começou há quarenta anos atrás e tem ganhando notoriedade por diversas razões como a ampla quantidade de materiais multimídia contendo dados destes comportamentos e de dados coletados empiricamente, que são comparáveis à parâmetros de modelos simulados, os quais por sua totalidade são mensuráveis, tais como posicionamento e velocidade dos pedestres. Além disso, são constatadas algumas similaridades entre ações em aglomerações de pessoas com o comportamento de gases e fluídos (Helbing and Molnár, 1995). Assim, no início das pesquisas, se coletavam os parâmetros das simulações através de fotos, vídeos e observações diretas. Por muito tempo, o objetivo deste tipo de pesquisa era o fornecimento de serviços com intenção de projetar áreas para pedestres, tendo estas a forma de relações de regressão, que não eram adequadas para demonstrar o comportamento de fluxos de pedestres em situações de evacuações mais intensas ou quando a arquitetura do ambiente fosse mais complexa (Helbing and Johansson, 2010).

Com o passar do tempo surgiram outros modelos que simulavam atuações humanas em multidões, ao serem analisados os modelos estocásticos e o modelo de “comportamento de escolha da rota”, constatou-se que nenhum deles levava em conta efeitos de organização espontânea dos pedestres, citando-se como exemplo a formação de um padrão, como uma faixa de fluxo entre eles em uma direção uniforme, ou a oscilação do fluxo de pedestre em gargalos. (Helbing and Johansson, 2010).

Estes fenômenos descritos acima ocorrem devido as interações entre os pedestres e o ambiente onde estão inseridos. Com isso, podemos assumir a existência de uma “coordenação inteligente” entre eles, principalmente em casos de evacuações simples. No entanto, existem situações complexas, onde esse mesmo tipo de coordenação dá lugar ao completo caos. As consequências disso são frequentes ultrapassagens entre pedestres, desaceleração do fluxo ou mudanças abruptas na direção do movimento de várias pessoas, devido à agitação da multidão.

Atualmente existem duas abordagens aceitas para modelar e simular o comportamento humano: a macroscópica e a microscópica (Helbing and Molnár, 1995). A primeira

é mais generalista, leva em conta a multidão inteira, enquanto a segunda, estuda o movimento e as interações de cada pedestre. Embora computacionalmente mais complexa, é uma modelagem que oferece maior precisão.

Dentro do modelo microscópico existem três alternativas de se representá-lo: *Cellular Automaton*, que apesar de seu baixo custo computacional, é pouco precisa na interação entre pedestres. A segunda alternativa é o *Agent Based Methods*, onde são definidas regras específicas para cada pedestre, o que se torna computacionalmente ineficiente. Por fim, temos o modelo no qual esse trabalho se baseia, que é o *Social Force Model*, sendo este o que melhor equilibra confiabilidade e desempenho. (Sent and Roisenberg et al., 2015).

Esse modelo foi elaborado durante a década de 1990 por Helbing e se baseia em pesquisas anteriores como a de Henderson, na década de 1970, a qual obteve sucesso na comparação do fluxo de pedestres com as equações de NAVIER-STOKES (Helbing and Molnár, 1995). O princípio de sua proposta é representar o comportamento dessas pessoas numa evacuação de forma análoga à propagação de partículas (Helbing and Molnár, 1995), ou seja, cada indivíduo é uma partícula com uma massa, inserido em um ambiente, onde existe um tipo de força que atrai a partícula para o ponto de saída, outra que repele o pedestre de obstáculos próximos e também outra que o repele de pessoas próximas, sendo todo este processo explicado por equações matemáticas de difícil compreensão.

Revisando as literaturas existentes sobre o assunto, é possível constatar a existências de pesquisas que utilizaram lógica fuzzy para substituir esse modelo complicado de equações, por uma descrição linguística do relacionamento entre elementos ambientais, sensações do agente e as ações tomadas por ele, acentuando assim a interpretabilidade do problema e permitindo reproduzir situações específicas, como pânico e visibilidade restrita. Por outro lado, nem sempre é fácil para o projetista elaborar regras que simulem esses comportamentos. Assim sendo, essa pesquisa utiliza comportamentos de evacuação simulados através do *Social Force Model*, os quais servem de base para o treinamento de redes neuro-fuzzy. Depois de treinados, estes comportamentos podem ser interpretados como conjuntos fuzzy e regras de um sistema de inferência fuzzy sobre estes conjuntos.

1.1 Objetivos

1.1.1 Objetivo geral

Deseja-se analisar a substituição das equações de forças do *Social Force Model*, considerando as constantes fixas para o contexto de evacuações de multidões, por redes neuro-fuzzy, conhecidas na literatura como INFGMN (*Incremental Neuro-Fuzzy Gaussian Mixture Network*). Isso deverá oferecer uma maneira de extrair conjuntos fuzzy e regras linguísticas do tipo Mamdani-Larsen, oferecendo a melhor relação de acurácia e interpretabilidade, para um sistema de inferência fuzzy equivalente a geração de forças do SFM.

1.1.2 Objetivos específicos

- Estudar o comportamento de multidões através do *Social Force Model*, apresentando também suas equações.
- Introduzir fundamentos teóricos relativos as redes neuro-fuzzy, apresentando a INFGMN e suas vantagens.
- Gerar conjuntos de treinamento para redes neurais *multi-layer perceptron*, a partir de um simulador denominado PANIC SIMULATOR para MATLAB.
- Utilizar a INFGMN, através de uma biblioteca desenvolvida para MATLAB, para treinar os mesmos conjuntos de dados obtidos para as redes neurais *multi-layer perceptron*.
- Substituir as equações de força do SFM, pela INFGMN, mantendo as variáveis que capturem as sensações dos agentes.
- Avaliar se as regras geradas pela INFGMN promovem uma experiência mais simples de *software* para o especialista do domínio, ou seja, uma melhor compreensão sobre o comportamento dos agentes na simulação.
- Analisar a relação interpretabilidade *versus* precisão em relação a substituição das equações do *Social Force Model* pela INFGMN.

1.2 Metodologia de pesquisa

Essa pesquisa envolve duas frentes, sendo que a primeira tem por objetivo realizar uma apresentação teórica relativa ao *Social Force Model* e redes neuro-fuzzy através de artigos científicos e livros técnicos. A segunda parte dessa pesquisa engloba uma implementação prática da geração de forças pela INFGMN. Para isso, existe um simulador

de código aberto para MATLAB R2017 chamado PANIC SIMULATOR (Schmidt and Späh, 2014) que implementa o *Social Force Model*, e que possui diversos recursos e possibilidades. Esse processo de substituição das equações de força contempla uma etapa de aprendizagem supervisionada, que coleta dados dos agentes para treinamento, através de simulações que ainda usam as equações originais e posicionam estes agentes de forma aleatória no cenário.

Ao fim desta etapa é concebida uma rede neural do tipo *multi-layer perceptron* (MLP) para cada força de um eixo cartesiano. A meta neste estágio é avaliar empiricamente, o quão bom é esse conjunto de treinamento, bem como, se é capaz de reproduzir adequadamente as forças do *Social Force Model*. Por último, essas redes neurais são substituídas, cada uma, por uma INFGMN, que treina o mesmo conjunto de treinamento obtido anteriormente. Nesta última etapa são avaliados os resultados de saída, isto é as regras geradas (interpretabilidade) e o passo de defuzzificação (precisão) ao reproduzir o *Social Force Model*. A avaliação da precisão é dividida em duas análises: uma qualitativa, relativa a reprodução de fenômenos ocorridos em multidões e outra quantitativa, analisando a intensidade da força gerada em cada eixo cartesiano e os tempos das simulações.

1.3 Estrutura do Trabalho

Este trabalho é dividido em sete capítulos. Além da introdução já abordada no primeiro capítulo, no segundo, o *Social Force Model* é explicado com mais detalhes, apresentando suas equações e fenômenos decorrentes em multidões. No capítulo três são apresentadas as redes neuro-fuzzy, revisando conceitos de sistemas de inferência fuzzy, redes neurais e por fim uma explanação completa sobre a INFGMN, a rede neuro-fuzzy a ser adotada nessa pesquisa. O quarto capítulo apresenta um resumo do estado da arte. Já nos capítulos cinco e seis, são respectivamente descritos a implementação, ou seja, a descrição passo a passo do processo que substitui as equações de força pela INFGMN no MATLAB e uma análise dos resultados obtidos com a aplicação da INFGMN, comparando-a com as redes neurais e o *Social Force Model*. Por fim, o sétimo capítulo conclui essa pesquisa e sugere trabalhos futuros.

2 *Social Force Model*

Conforme já discutido, o *Social Force Model* é um modelo microscópico que simula realisticamente o fluxo de pedestres em diversas situações, como evacuações e tráfego intenso, através de equações que representam os pedestres individualmente como partículas de gases e fluídos. Também neste modelo, leva-se em conta a formação dos padrões de fluxo devido às interações não lineares entre os pedestres e os efeitos que eles podem causar em situações normais ou de pânico na multidão. Tudo isto é levado em consideração na formulação matemática descrita neste modelo.

Neste capítulo, a primeira seção define matematicamente o *Social Force Model*. Inicialmente, a fórmula geral é apresentada e, em seguida, cada componente de força deste modelo é detalhado. Uma vez que o PANIC SIMULATOR contempla situações de pânico, apresentaremos especificamente os componentes de força de repulsão que realizam isso. Já as duas próximas seções abordam os fenômenos decorrentes em multidões e evacuações, os quais são replicados por este modelo.

2.1 Definição

Considere um pedestre α inserido em um ambiente qualquer. De acordo com Helbing e Johansson 2010, $\vec{r}_\alpha(t)$, é a mudança de um local \vec{r} em um instante de tempo t do agente α , a partir de sua velocidade, sendo descrito pela seguinte equação:

$$\frac{d\vec{r}_\alpha(t)}{dt} = v_\alpha(t) \quad (2.1)$$

Ainda segundo, Helbing e Johansson (2010) e Helbing et al. (2000), podemos descrever a mudança de velocidade deste pedestre α , com uma massa m_α , no instante t a partir de uma equação de aceleração:

$$m_\alpha \frac{dv_\alpha}{dt} = \vec{f}_a(t) + \xi_\alpha(t) \quad (2.2)$$

Nesta equação, $\vec{f}_a(t)$ é a força de aceleração para o pedestre α , correspondendo a soma das forças sociais que influenciam no movimento dele. Já, $\xi_\alpha(t)$ são “flutuações” individuais que refletem a variação do comportamento de α , mas que não são implementadas no PANIC SIMULATOR.

A fórmula para força de aceleração é definida abaixo:

$$\vec{f}_\alpha(t) = \frac{m_\alpha}{\tau_\alpha}(v_\alpha^0 \vec{e}_\alpha^0 - \vec{v}_\alpha) + \sum_{\alpha \neq \beta} \vec{f}_{\alpha\beta}(t) + \sum_i \vec{f}_{\alpha i}(t) \quad (2.3)$$

Em que $\frac{m_\alpha}{\tau_\alpha}(v_\alpha^0 \vec{e}_\alpha^0 - \vec{v}_\alpha)$ denota o cálculo força de desejo para α . O segundo termo desta equação denota soma de forças sociais no instante de tempo t , entre α e cada outro indivíduo β presente na simulação. O terceiro termo da equação, determina, também, uma soma forças de obstáculos para o mesmo instante t , entre α e cada obstáculo i .

Uma grande vantagem desse modelo é a sua flexibilidade, sendo possível generalizar quaisquer possíveis cenários específicos de evacuação e fluxo de multidões, apenas pela equação (2.3), sendo isto essencial para reproduzir simulações complexas de maneira fiel às observações empíricas (Helbing and Johansson, 2010). As próximas subseções detalham cada componente de força.

2.1.1 Força de Desejo

O primeiro termo da equação (2.3) é representado pela Figura 1, temos que cada indivíduo α se move com uma velocidade desejada v_α^0 , para uma direção desejada \vec{e}_α^0 , onde temos um certo tempo de aceleração τ_α , com $v_\alpha^0 \vec{e}_\alpha^0$ tendendo a v_α^0 e sob esse valor subtrai-se a velocidade atual \vec{v}_α , na qual, com baixos valores de τ_α , os pedestres caminham mais agressivamente (Helbing and Molnár, 1995). Essa direção desejada toma como base o local em que o pedestre α tem como objetivo final denominado por \vec{r}_α^0 , que se deseja alcançar da maneira mais confortável e rápida possível. Esse objetivo, geralmente, é a saída mais próxima de onde se encontra o agente. Ainda de acordo com Helbing e Molnár (1995) denotamos a seguinte equação para a direção desejada:

$$\vec{e}_\alpha^0(t) = \frac{\vec{r}_\alpha^0 - \vec{r}_\alpha}{\|\vec{r}_\alpha^0 - \vec{r}_\alpha\|} \quad (2.4)$$

Nesta equação \vec{r}_α é a posição atual em que se encontra a entidade α . Já o denominador dessa fração corresponde a distância vetorial entre a posição de α e o local do objetivo.

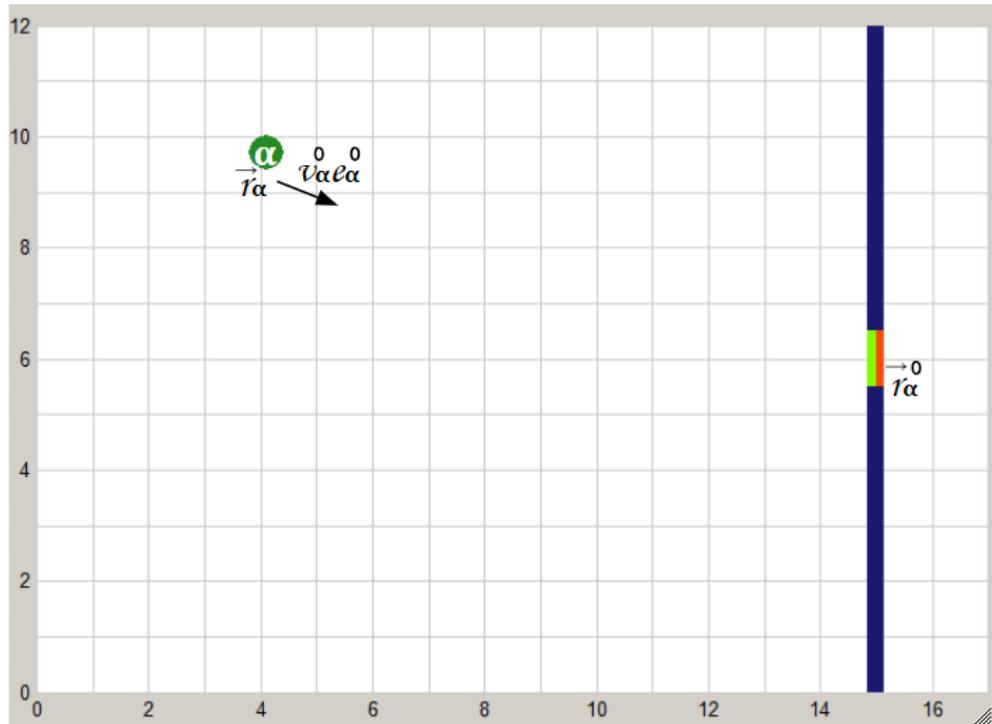


Figura 1 – Exemplo de Força de Desejo para o pedestre α e o seu objetivo r_{α}^0 .

2.1.2 Forças de Repulsão

Na medida em que um pedestre α se aproxima de outro pedestre β , α gradualmente se sente mais desconfortável e a interação entre ambos pode acabar sendo agressiva. Esse é o modo como se representa conceitualmente a força de repulsão entre ambos os pedestres. Também no fluxo de pedestres existem obstáculos inseridos no universo onde se encontra a pessoa, como muros, paredes, balcões e outros objetos no caminho.

Conforme um pedestre α se aproxima de obstáculo i ele tende a reduzir a sua velocidade para não colidir com este, evitando se machucar. Análogo com a força de repulsão entre pedestres já descrita acima, isso seria a força de repulsão entre pedestres e obstáculos.

2.1.2.1 Força Social

Seja o componente da soma $\vec{f}_{\alpha\beta}(t)$ da equação (2.3), ele é um somatório da força de repulsão entre α e cada outro indivíduo β entre uma lista de indivíduos (Helbing and Johansson, 2010). Assim, em uma multidão muito densa, teremos que essa força será maior sobre α . Deste modo, as forças sociais sobre eles aumentam drasticamente a intensidade quando passa-se pela seguinte condição de contato físico: $d_{\alpha\beta} < r_{\alpha\beta}$ e $r_{\alpha\beta} = r_{\alpha} + r_{\beta}$. Ou seja, a distância entre o pedestre α e β deve ser menor que o raio de ambos, sendo que este é obtido pela soma dos raios individuais destes pedestres. Abaixo,

a equação de Força Social é demonstrada:

$$\vec{f}_{\alpha\beta}(t) = \{A \exp^{(r_{\alpha\beta} - d_{\alpha\beta})/B} + k\Theta(r_{\alpha\beta} - d_{\alpha\beta})\} \times \vec{n}_{\alpha\beta} + \kappa\Theta(r_{\alpha\beta} - d_{\alpha\beta})\Delta v_{\beta\alpha}^t \vec{t}_{\alpha\beta} \quad (2.5)$$

Observe-se que temos $d_{\alpha\beta} = \|\vec{r}_a - \vec{r}_b\|$, ou seja a distância vetorial entre a posição do pedestre α e o pedestre β . Assim sendo $A \exp^{(r_{\alpha\beta} - d_{\alpha\beta})/B} \times \vec{n}_{\alpha\beta}$ o alcance da força repulsiva, $k\Theta(r_{\alpha\beta} - d_{\alpha\beta}) \times \vec{n}_{\alpha\beta}$ a força do corpo que contrai a compressão do corpo, e $\kappa\Theta(r_{\alpha\beta} - d_{\alpha\beta})\Delta v_{\beta\alpha}^t \vec{t}_{\alpha\beta}$ uma “força de atrito deslizante”, impedindo o movimento tangencial relativo. A função Θ retorna o próprio parâmetro, $\Theta(r_{\alpha\beta} - d_{\alpha\beta}) = r_{\alpha\beta} - d_{\alpha\beta}$, se o resultado for ≥ 0 , caso contrário retorna zero.

Ainda individualmente, a variável A é a força da interação em newtons e B o alcance da mesma em metros. Helbing e Johanson (2010), mencionam que podemos generalizar esses dados individuais de A e B para uma população de características homogêneas, o que é o caso para o simulador. Já k e κ representam grandes constantes. Também $\vec{n}_{\alpha\beta} = \frac{(\vec{r}_\alpha - \vec{r}_\beta)}{d_{\alpha\beta}}$ é o vetor de direção normalizado do agente β para o agente α . Por fim, tem-se também $\vec{t}_{\alpha\beta}$ e $\Delta v_{\beta\alpha}^t$ que significam respectivamente o vetor normalizado na direção tangencial e a diferença de velocidade tangencial entre os agentes, sendo esta dada por:

$$\Delta v_{\beta\alpha}^t = (\vec{v}_\beta - \vec{v}_\alpha) \times \vec{t}_{\alpha\beta} \quad (2.6)$$

2.1.2.2 Força de Obstáculo

Seja o componente da soma $\vec{f}_{\alpha i}(t)$ da equação (2.3) ele é um somatório da força de repulsão entre α e cada outro obstáculo i entre uma lista deles (Helbing and Johansson, 2010). Da mesma forma que na interação entre indivíduos, em uma multidão cheia de obstáculos, teremos que esse tipo de força será maior sobre α . Ainda de acordo com Helbing e Johansson (2010), podemos tratar todas as interações de um pedestre α com obstáculos i , da mesma forma que tratamos os outros pedestres β . Ou seja, se trocarmos β por i obteríamos a mesma equação da Força Social, para a Força de Obstáculo. Abaixo ela é apresentada:

$$\vec{f}_{\alpha i}(t) = \{A \exp^{(r_{\alpha i} - d_{\alpha i})/B} + k\Theta(r_{\alpha i} - d_{\alpha i})\} \times \vec{n}_{\alpha i} + \kappa\Theta(r_{\alpha i} - d_{\alpha i})\Delta v_{i\alpha}^t \vec{t}_{\alpha i} \quad (2.7)$$

Como os obstáculos estão em repouso, ou seja, $\vec{v}_i = 0$, então apenas a velocidade do agente α é considerada.

A Figura 2 ilustra os principais componentes dessa força, sobretudo a condição de contato entre os pedestres e o obstáculo. Ainda nesta Figura é possível observar

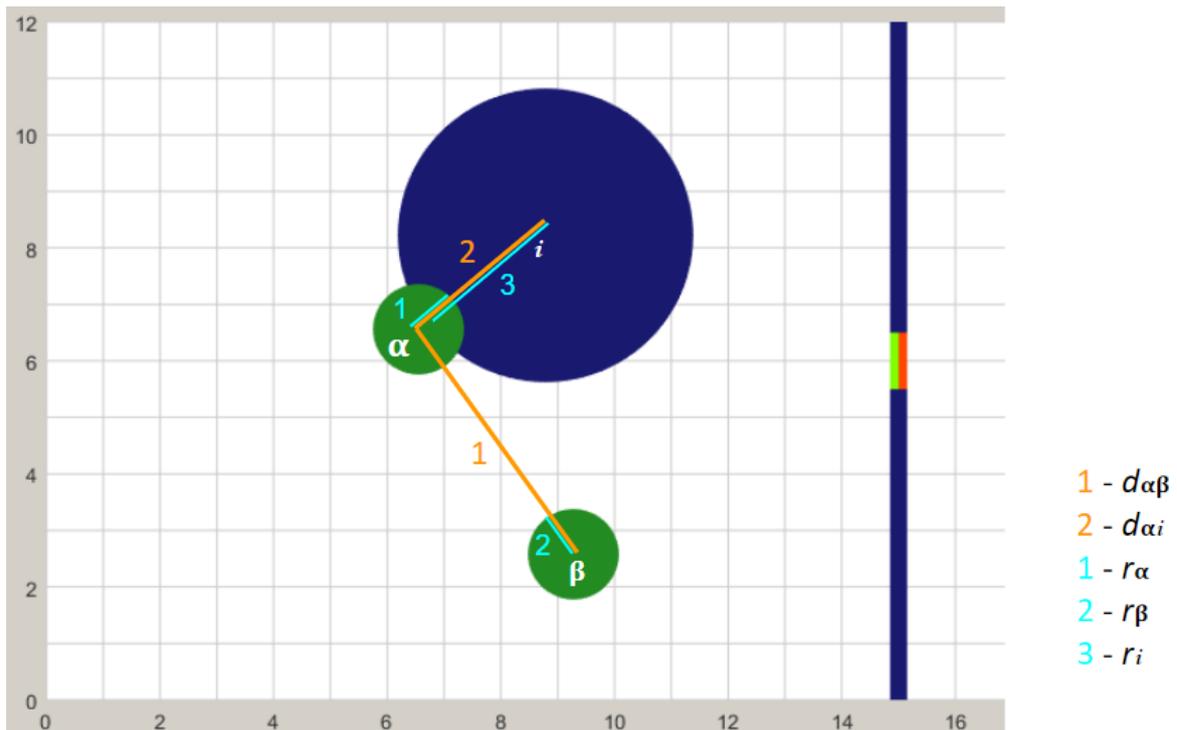


Figura 2 – Exemplo de Força em situações de pânico, envolvendo um agente α , um outro agente próximo identificado por β e um obstáculo i .

visualmente que $d_{\alpha\beta} > r_{\alpha} + r_{\beta}$ e $d_{\alpha i} < r_{\alpha} + r_i$, ou seja apenas no segundo caso é contemplado a condição de contato físico entre o agente α e obstáculo i .

Esse modelo de força descrito nessa seção é o que o simulador utiliza. Entretanto, as forças de atrito deslizante e a força de compressão para pedestres e obstáculos são desconsideradas nesta pesquisa, tanto para as coletas dos conjuntos de treinamento, quanto para obtenção de estatísticas com simulações envolvendo o *Social Force Model*. A razão é que a ocorrência dessas forças acontece apenas quando há toque entre as entidades da simulação, como especifica a função Θ . Também, a matemática é extremamente complexa, ao realizar muitos cálculos apenas para estas situações.

Desta maneira, necessitaria-se de muitas variáveis para que uma rede neural ou neuro-fuzzy reproduzisse estes eventos com uma precisão minimamente aceitável. A medida que mais variáveis fossem consideradas, como direção tangencial e diferença de velocidade para cada entidade vizinha, o número de regras ficaria exponencialmente maior e o problema muito pouco interpretável. Ainda assim é possível reproduzir muitos dos fenômenos que são descritos nas próximas duas seções, inclusive as forças de repulsão, pois as de interação e o alcance da mesma são simulados para a captura da força.

2.2 Fenômenos em Multidões

O *Social Force Model* é um modelo robusto, sendo capaz de reproduzir fielmente fenômenos que ocorrem em multidões na vida real e que já foram mencionados brevemente no capítulo introdutório, como a auto-organização dos pedestres, na qual existem interações não-lineares entre eles, ou seja, não há nenhuma comunicação explícita entre eles que determine isto. Essa auto-organização é responsável pela formação de padrões de movimento de pedestres, que ocorrem sem que eles mesmos possam perceber essa coordenação inteligente (Helbing and Johansson, 2010). A durabilidade desses padrões é inversamente proporcional a quantidade de transformações que eles sofrem durante o fluxo.

A “Formação de Faixas” (*Lane Formation*), apresentada na Figura 3, é um exemplo de padrão de movimento contendo filas de indivíduos em uma direção uniforme por onde o fluxo passa. Assim, essas faixas contêm poucas manobras entre os pedestres. Este fenômeno surge quando pedestres estão se movendo opostamente e sem intersecção entre eles, a uma velocidade relativa alta e à uma distância muito curta entre um e outro. Este padrão de movimento dos pedestres ocorre, por exemplo, em lugares muito fechados, como um centro comercial. Também entram aqui as forças de repulsão já descritas e que são muito atuantes quando pedestres estão se movendo em direções opostas. Todavia, essa força é minimizada quando a “Formação de Faixas” acontece e esses agentes que se repulsavam são separados em faixas diferentes.

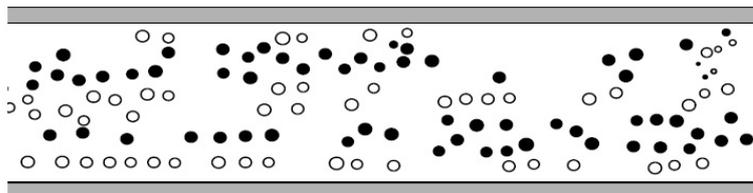


Figura 3 – Formação de 4 faixas, no círculo branco pedestres em uma direção, no círculo preto são pedestres na direção oposta (Helbing and Molnár, 1995).

Outro tipo de fenômeno em multidões é a “Oscilação de Fluxos em Gargalos” (*Oscillatory Flows at Bottlenecks*), demonstrada por meio da Figura 4. Isso ocorre quando há um cruzamento de fluxos de pedestres, e segundo Helbing e Johanson (2010), em uma analogia com a propagação de partículas, podemos pensar, assim que um indivíduo (partícula) consegue uma passagem para seguir o seu caminho atravessando o gargalo, os outros atrás com a mesma direção fazem o mesmo, ocorrendo uma diferença de pressão entre um fluxo e outro, sendo suficientemente grande para que um fluxo pare dando passagem ao outro e eventualmente, com as devidas alterações, faça com que a diferença de pressão se inverta e outro fluxo consegue a passagem. Esse fenômeno pode ocorrer em lugares nos quais se tem uma passagem muito estreita.

A “formação de faixa em intersecção de fluxo” (*Stripe Formation in Intersecting*

Flows) indicada pela Figura 5, ocorre quando há um cruzamento em direção oposta de somente dois fluxos. Todavia, diferentemente dos padrões acima, um fluxo consegue atravessar outro sem que haja algum tipo de parada em ambos, permitindo também a minimização dos efeitos de interações obstrutivas a maximização da velocidade média dos indivíduos (Helbing and Johanson 2010).

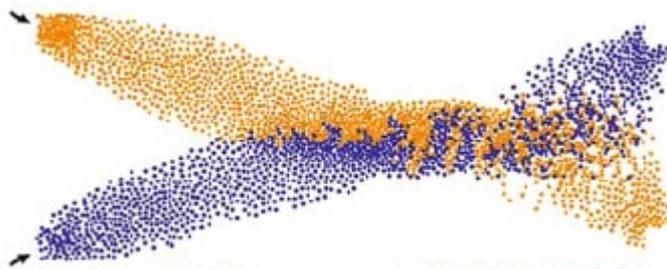


Figura 4 – Cruzamento do fluxo pedestres em azul com outro fluxo em laranja (Helbing and Johansson, 2010).

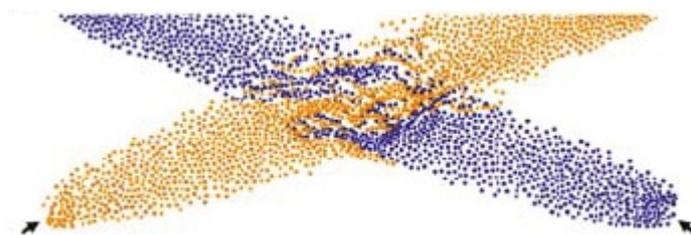


Figura 5 – Intersecção do fluxo pedestres em azul com outro fluxo em laranja (Helbing and Johansson, 2010).

2.3 Fenômenos em Evacuações de Multidões

Pode-se determinar que há uma evacuação de multidões quando tem-se um ambiente com uma extrema densidade de pedestres e eles estão sob estresse psicológico devido a necessidade de sair do local o mais rápido possível, resultando no incremento de suas velocidades desejadas e de sua *flutuação*. Situações como essas são coloquialmente chamadas “pânico”. Muitas evacuações conseguem ser relativamente organizadas, mas pode haver alguns casos que resultam em grandes desastres com perdas humanas (Helbing and Johansson, 2010).

Além disso, temos um “comportamento pastoreiro” por parte dos pedestres, que sem saberem o que fazer, seguem o comportamento daqueles que estão próximos deles, fazendo com que se movam coletivamente em direção a mesma saída. Isso torna a evacuação muito mais lenta, principalmente em situações com baixa visibilidade, (veja Figura 6) e menos eficiente, pois muitas saídas acabam sendo ignoradas e outras excessivamente ocupadas.

Acontece que, com isso, ocorre um atolamento na saída, um aumento drástico na pressão que um pedestre faz ao outro, devido a uma maior quantidade de interação entre eles e, por fim, mudanças abruptas na direção e intensidade da força, que podem fazer com que os pedestres caiam e sejam pisoteados, servindo de obstáculo para os outros na evacuação. Segundo Helbing e Johanson (2010), as melhores chances de sobrevivência nessas situações se encontram na busca por saídas alternativas, seguindo outros pedestres.

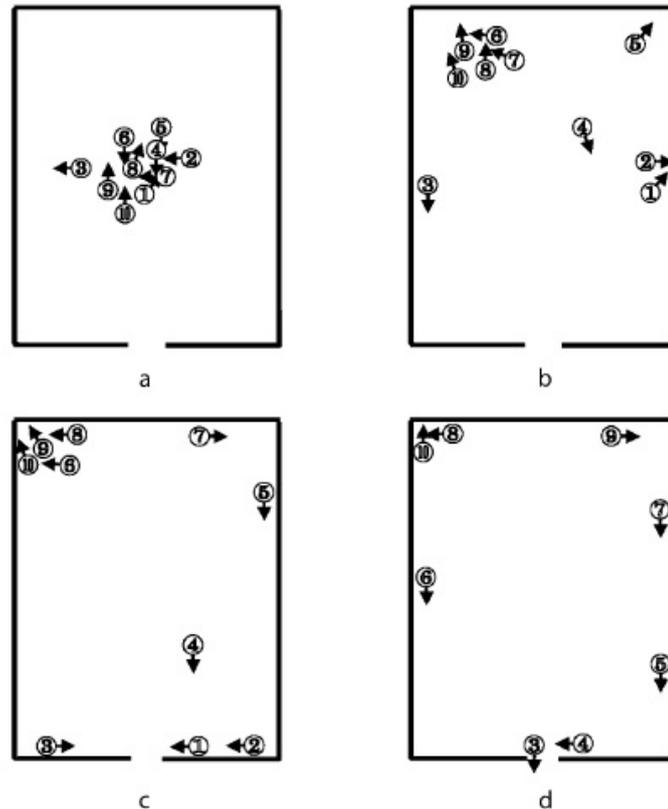


Figura 6 – Comportamento pastoreiro de pedestres em baixa visibilidade, denotadas pela sequência a,b,c,d. (Helbing and Johansson, 2010).

Situações de pânico costumam resultar na destruição de faixas de fluxo de pedestres, sejam elas opostas ou não, (isso é ilustrado pela Figura 7). Quando isso ocorre, em analogia com a propagação de partículas, teríamos naturalmente que os fluídos (faixa contendo fluxo de pedestre) se transformassem em gases e os pedestres se moveriam descontroladamente. No entanto, o que ocorre é exatamente oposto. Com a destruição das faixas, os pedestres tendem a ficar em estado sólido, ou seja, bloqueados pelo menos temporariamente. Por isso, esse fenômeno ganha o nome de “Congelamento por Aquecimento” (*Freezing by Heat*) (Helbing and Johansson, 2010).

Ainda no contexto de evacuações de massas, quando o fluxo de entrada em um gargalo é maior que o fluxo de saída, uma fila de pedestres é formada na entrada dessa passagem estreita e os pedestres ficam agressivos ao competir pelas mesmas passagens no local. A consequência disso são os grandes problemas na coordenação de movimento

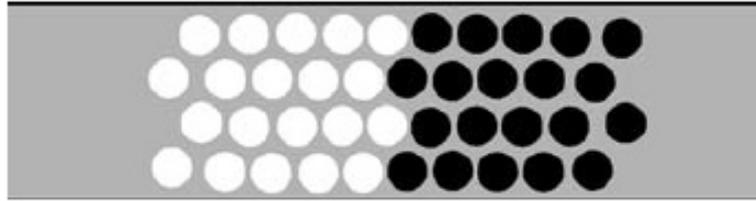


Figura 7 – “Congelamento por Aquecimento” em um fluxo oposto de pedestres (Helbing and Johansson, 2010).

dos pedestres em uma multidão densa, trazendo pausas constantes do deslocamento na saída do gargalo. Todos estes fatores tornam os indivíduos passivos a maiores interações físicas e de efeitos de atrito, causando lentidão na evacuação. Assim, o conceito para isso é chamado de “Quanto Mais Rápido, Mais Devagar” (*Faster-Is-Slower Effect*) (Helbing and Johansson, 2010). Visto isso, é importante que as áreas de saída sejam largas o suficiente, minimizando gargalos e contrafluxos causados por alguns pedestres neste tipo de evento.

3 Redes Neuro-Fuzzy

As redes neuro-fuzzy podem ser compreendidas como uma combinação das características de sistemas fuzzy e de redes neurais. De acordo com Lin e Lee (1996), sistemas fuzzy e redes neurais são estimadores numéricos com a capacidade comum de otimizar a inteligência de sistemas em ambientes caracterizados por imprecisões e incertezas. Apesar das similaridades, elas divergem quanto à estrutura. Em sistemas que utilizam lógica fuzzy, existe a necessidade do especialista do domínio estruturar o conhecimento do mundo real em valores linguísticos que correspondem à conjuntos fuzzy e, por fim, associá-los às regras IF-THEN, que são altamente interpretáveis, porém têm desvantagens, tais como: a exigência de trabalho manual árduo para definir as regras e, segundo Lin e Lee (1996), a dificuldade de refinar o conjunto de regras e as funções de pertinência.

Já em relação as redes neurais, os mesmos autores argumentam que elas são sistemas dinamicamente treináveis, cujas habilidades de aprendizagem, tolerância à ruído e generalização, crescem a partir de suas estruturas conexionistas, suas dinâmicas e suas representações de dados distribuídos. Essas estruturas são compostas por vários neurônios, os quais são capazes de aprender e generalizar, por meio de padrões de treinamento, ou quaisquer dados fornecidos como entrada à rede.

Embora a capacidade de generalização e aprendizagem seja uma grande vantagem, é muito difícil para um especialista em um domínio no qual se quer resolver um problema, compreender essas estruturas conexionistas e suas relações com os valores resultantes. De toda maneira tanto um sistema fuzzy, como as redes neurais, são capazes de modelar um comportamento complexo não-linear a diferentes níveis de acurácia (Lin and Lee, 1996).

Estas diferenças entre ambas as técnicas de inteligência computacional motivou o interesse dos pesquisadores em integrar esses sistemas em um só, no qual se permita que as redes neurais provenham a capacidade de aprendizagem para sistemas fuzzy e estes permitam que as redes neurais possam ser entendidas como regras IF-THEN de alto nível. Assim, Lin e Lee (1996) definem redes neuro-fuzzy como “uso de redes neurais como ferramentas em modelos fuzzy”, enquanto que Kasabov (2006), estabelece que os sistemas de inferência neuro-fuzzy consistem em um conjunto de regras e um método de inferência que são incorporados ou combinados com uma estrutura conexionista para uma melhor adaptação.

Dada a motivação inicial, este capítulo busca revisar conceitos básicos a respeito do sistema de inferência fuzzy, também sobre redes neurais, percorrendo muito brevemente sobre a rede neural *multi-layer perceptron*, pois o uso dela neste trabalho tem apenas um fim secundário, que envolve apenas a coleta dos dados da simulação original para o

treinamento da INFGMN. Já a INFGMN é a rede neuro-fuzzy usada para substituir as equações de força do SFM. Por fim, esse capítulo finaliza argumentando em detalhes o funcionamento da INFGMN, incluso aqui suas vantagens e fundamentos matemáticos.

3.1 Revisão sobre Sistemas de Inferência Fuzzy

A ideia de Sistemas de Inferência Fuzzy (FIS - *fuzzy inference systems*) é ser um modelo computacional, que controla um determinado sistema através de regras fuzzy IF-THEN, definidas por um humano especialista. Essas regras relacionam as entradas do sistema com a saída do mesmo, sendo que as primeiras são os antecedentes das regras condicionadas a operações booleanas e por último, o lado conseqüente das regras. Todas essas entradas e saídas correspondem a valores linguísticos, que aumentam o poder de expressão para modelar problemas do mundo real. Estes valores são representados como conjuntos fuzzy e estão associados a um universo de discurso ou variável linguística.

Lin e Lee (1996), explicam que conjuntos fuzzy são como uma generalização da teoria dos conjuntos clássicos e Zadeh (1965), define que para cada elemento de um conjunto fuzzy, existe uma função de pertinência, a qual mapeia esse elemento, associando-o ao respectivo conjunto no intervalo entre $[0,1]$. Isso é compreendido como o grau de pertinência do elemento ao conjunto. Há vários tipos de funções de pertinência como triangular, trapezoidal e gaussiana. A Figura 8, aponta um exemplo hipotético de três conjuntos fuzzy: idoso (em vermelho), jovem (em verde) e adulto (em azul), inseridos em um universo de discurso “idade”, onde os dois primeiros conjuntos são representados por uma função sigmoide e o último por uma diferença de funções sigmoides.

Baseando-se na explicação de Lin e Lee (1996), o funcionamento básico de um FIS pode ser resumido em três etapas principais: a fuzzificação, na qual valores numéricos são convertidos a valores linguísticos através das funções de pertinência; a fase de inferência, onde ocorre a análise de cada regra, relacionando matematicamente os graus de pertinência dos termos antecedentes, formando um subconjunto fuzzy na variável de saída. O objetivo nessa etapa é agregar esses subconjuntos, criando um novo, para que na última etapa, na defuzzificação, esse conjunto fuzzy resultante seja transformado em um valor *crisp*, isto é, a saída do sistema fuzzy e não mais um valor difuso. Entre os variados métodos de defuzzificação, inclui-se a centroide, na qual esse valor é o centro de gravidade do conjunto obtido na saída.

Dentre vários modelos de FIS, os dois mais conhecidos são o Takagi-Sugeno e o modelo Mamdani-Larsen. O primeiro, apresenta maior precisão e considera que na parte conseqüente de cada regra, o valor de saída é calculado por alguma função que tem as entradas dessa regra como parâmetros. No outro modelo, a parte conseqüente é constituída apenas de um valor linguístico, o que torna o modelo muito mais interpretável

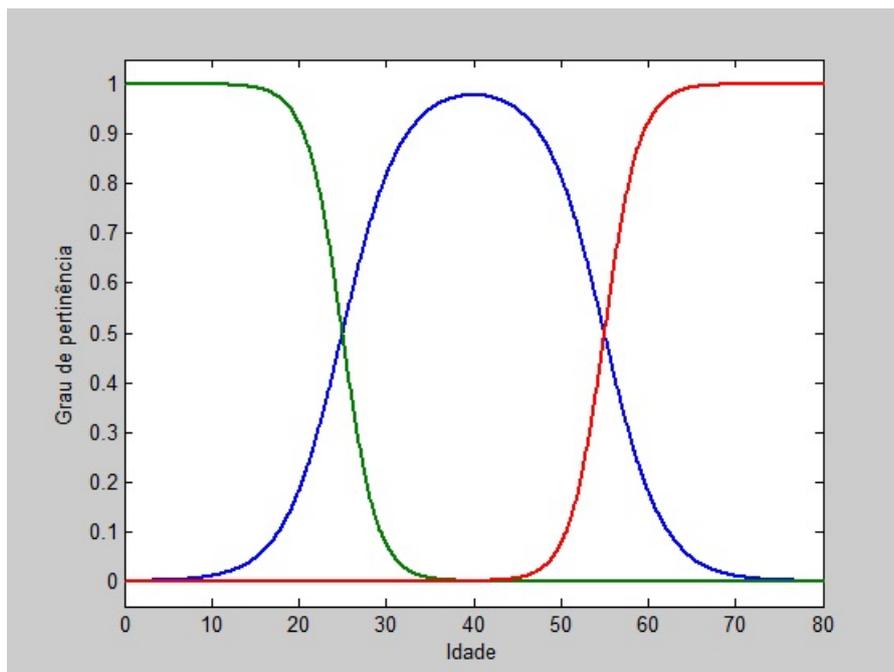


Figura 8 – Exemplo de conjuntos fuzzy para um universo de discurso idade.

para o especialista.

3.2 Revisão sobre Redes Neurais Artificiais

As Redes Neurais Artificiais (ANN - *Artificial neural networks*) são sistemas de processamento de informações, que possuem habilidades de aprender, recordar e generalizar a partir de padrões de treinamento ou dados, analogamente ao cérebro humano (Lin and Lee, 1996). Ainda acordo com estes mesmos autores, as redes neurais são compostas por uma estrutura conexionista formada por nodos de processamento, os quais são comumente conhecidos na literatura como neurônios artificiais ou simplesmente neurônios. Esse modelo computacional é altamente utilizado para resolver problemas de busca por correspondência de padrões ou classificação dos mesmos, ou aproximação de funções e clusterização de dados (Lin and Lee, 1996).

Lin e Lee (1996), explicam que os modelos de redes neurais são especificados por três entidades: quanto ao processamento de elementos na própria rede, a topologia da rede e a aprendizagem da rede. Em relação a primeira delas, cada neurônio recebe como entrada todos os padrões de treinamento ou saídas de todos outros neurônios, sendo este último o caso de um modelo *multi-layer*. Associada à cada conexão de entrada desse neurônio, existe um peso numérico determinado pela regra de aprendizagem, o qual permite determinar um nível de ativação ao neurônio. Um peso maior que zero é considerado excitatório, ou seja, favorece a ativação do neurônio. Já o contrário, tem um efeito inibitório sobre esse. Finalmente sobre toda essa entrada, tem-se uma operação

matemática, e que, segundo, Lin e Lee (1996), geralmente é uma soma que combina toda essa informação vinda em um valor denominado *net input*. Sobre ele é aplicada uma função de ativação, a qual produz valor de saída para esse neurônio.

Em relação à topologia da rede, é analisada a forma geométrica com que os neurônios estão conectados. Eles podem estar agrupados em uma camada, onde cada neurônio dela produz sua própria saída. Na situação em que todas essas saídas se conectam à entradas de neurônios de outra camada e quando não há conexão entre neurônios de uma mesma camada, pode-se ter o que a literatura conceitua como *multi-layer feedforward network*.

A estrutura das redes com múltiplas camadas pode ser compreendida assim: a camada de neurônios que recebe a entrada da rede, já descrita no parágrafo acima, é denominada *input layers* e a camada contendo a saída final da rede, por sua vez, recebe o nome de *output layer*. Já as camadas escondidas ou *hidden layers*, são todas camadas intermediárias entre a *input layer* e a *output layer*. Lin e Lee (1996), argumentam que as camadas intermediárias tem como característica apenas receber como conexões de entrada, as saídas da camada antecessora. Sobre o número de camadas intermediárias, ele pode ser variável, ou até mesmo não existirem na rede, o que nesse caso implicaria na camada de entrada se conectar diretamente à camada de saída.

O terceiro e último componente das redes neurais é a regra de aprendizagem. De acordo com Lin e Lee (1996), existem dois tipos de aprendizagem: de parâmetros (*parameter learning*) e a aprendizagem estrutural (*structure learning*), não sendo necessariamente mutuamente exclusivos. O primeiro tipo basicamente faz com que a rede aprenda através da atualização automática dos pesos das conexões e o segundo permite que a rede altere a sua estrutura, variando no número de neurônios e no tipo das conexões.

Ambos tipos de aprendizagem estão associadas a três categorias: aprendizado supervisionado (*supervised learning*), *reinforcement learning*, e *unsupervised learning*. Aqui é destacado o papel do aprendizado supervisionado, pois é utilizado neste trabalho e nele a rede é treinada com exemplos prontos, contendo entradas e saídas pré-definidas. O objetivo principal do aprendizado supervisionado é ajustar o peso das conexões, para que as saídas geradas da rede se aproximem das saídas originais dos padrões de entrada da rede.

A Figura 9, é um exemplo de uma *multi-layer feedforward network* com 3 camadas e 7 neurônios. O vetor, x , dado por $x_1 \dots x_5$, denota a entrada da rede sobre os neurônios da *input layer*, numerados de 0 a 4. A camada intermediária é composta pelos neurônios 5 e 6, dos quais tem suas saídas conectadas a camada de saída, composta pelo neurônio 7. Por último, nessa figura, w_{ij} , é peso que da conexão entre um neurônio j e um outro i . Assim, w_{50} é o peso da conexão do neurônio 0 para o 5.

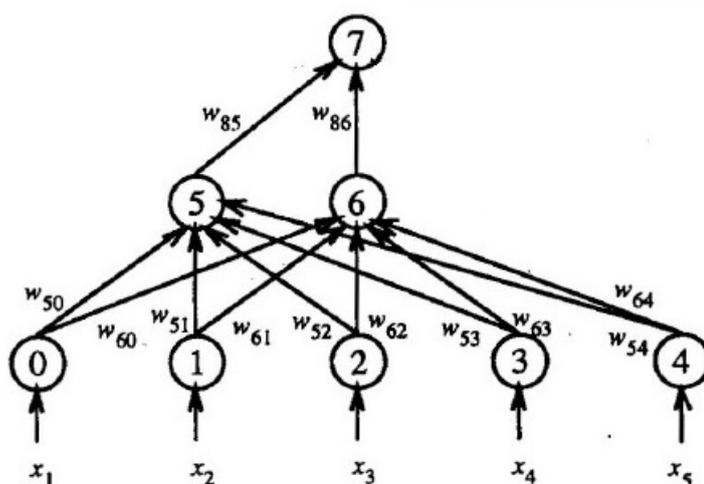


Figura 9 – Exemplo de uma topologia de rede, adaptado de Lin e Lee (1996).

Entre vários modelos de redes neurais, Lin e Lee (1996), destacam a *multi-layer feedforward network* com algoritmo de aprendizado *back-propagation learning rule*, também conhecida na literatura como *multi-layer perceptron* (MLP). Nesse algoritmo, os pesos iniciais das conexões são pequenos valores definidos aleatoriamente, e a saída da rede é propagada para as camadas anteriores, para que os pesos das conexões possam ser ajustados, com o objetivo de minimizar o erro de saída. Lin e Lee (1996), relatam que esse algoritmo é a resposta para um dos principais problemas no campo de redes neurais, que é a aprendizagem em redes complexas de múltiplas camadas e esse modelo de rede neural é quase sempre a primeira abordagem a ser usada em aplicações.

3.3 INFGMN (*Incremental Neuro-Fuzzy Gaussian Mixture Network*)

A INFGMN (*Incremental Neuro-Fuzzy Gaussian Mixture Network*) é um sistema neuro-fuzzy proposto por Mazzutti et al. (2017), e têm como característica o aprendizado incremental permitindo à rede aprender constantemente, sem a necessidade de retreinar todo o conjunto de dados e de gerar um novo conjunto de regras cada vez que um novo padrão é apresentado à rede.

Em relação à base de regras do sistema, ela também é mantida de forma incremental, sendo que novas regras podem ser adicionadas e outras removidas, sem quaisquer prejuízos de desempenho à rede. Ainda em relação ao conjunto de regras formado na INFGMN, Mazzutti et al. (2017) afirmam que elas são obtidas a partir de um modelo de misturas gaussianas, confirmando o que Gan et al. (2015) já haviam sentenciado quando demonstraram a equivalência entre esse modelo e um FIS com regras do tipo Mamdani-Larsen.

Outra característica da INFGMN, é conseguir lidar com o dilema da acurácia-

interpretabilidade, obtendo um excelente equilíbrio entre ambos, pois em geral, esses requisitos se tornam contraditórios em sistemas fuzzy. Rodrigues et al. (2014), dá um exemplo disso, quando analisa que um sistema com milhares de regras acaba tendo excelente precisão, porém um nível de compreensão bem baixo para um ser humano e o inverso também ocorre, quando um sistema fuzzy com uma regra é altamente interpretável, porém não será preciso em reproduzir o fenômeno desejado.

Além da INFGMN, também existem outros sistemas neuro-fuzzy na literatura, mas a grande maioria deles implementa regras do tipo Takagi-Sugeno, conforme apontam Mazzutti et al. (2017). Estão entre eles, a ANFIS (*Adaptive-Network-based Fuzzy Inference System*), Jang (1993), e a DENFIS (*Dynamic Evolving Neuro-Fuzzy Inference System*), Kasabov e Song (2002), que contam com bons desempenhos. Já a EFuNN (*Evolving Fuzzy Neural Network*), Kasabov (2001), trabalha com regras do tipo Mamdani-Larsen, porém, Rodrigues et al. (2014) afirmam que apesar da performance rápida, tem como desvantagens a difícil configuração dos parâmetros de aprendizagem, a geração automática do tipo e da quantidade de funções de pertinência para cada variável de saída. Ainda em relação a EFuNN, Mazzutti et al. (2017) destacaram que em testes realizados sobre três conjuntos dados da UCI (University of California, Irvine), Dheeru e Taniskidou (2017), tais quais, o Servo, o Vinho e a Íris, resultaram em acurácia inferior, se comparada a INFGMN e em um número de regras gerados muito maior que a INFGMN.

Com isso, as próximas três subseções se destinam a apresentar respectivamente o modelo de misturas gaussianas (*Gaussian Mixture Model*), a arquitetura da INFGMN, contemplando suas camadas em conjunto com o funcionamento da rede pelos seus modos de operação e, por fim, um exemplo prático com o conjunto de dados da íris é discutido em conjunto com algumas observações.

3.3.1 GMM (*Gaussian Mixture Model*)

O GMM (*Gaussian Mixture Model*), detalhado em Yu e Deng (2014), é um modelo probabilístico e que serve de base para criação dos termos linguísticos das variáveis e das regras dessa rede neuro-fuzzy. Assim, Mazzutti et al. (2017), mencionam que a função de densidade de probabilidade de misturas gaussianas (pdf - *probability density function*) para uma variável aleatória ou para o vetor de entrada da INFGMN, \mathcal{X} , de dimensão D , é definida por:

$$G(\mathcal{X}; p, u, C) = \sum_{j=1}^J p(j) \mathcal{N}^D(\mathcal{X}; \mu_j, C_j) \quad (3.1)$$

Nesta equação, $p(j)$, $\forall j \in [1, J]$, é a j -ésima proporção de mistura independente entre J componentes. Existem duas restrições relativas às proporções de mistura: a

primeira é para o valor de proporção de mistura de um componente, o qual sempre pertence à um intervalo fechado entre 0 e 1, e a segunda é referente à soma de todas as proporções de mistura, a qual é sempre igual a 1. Nesta mesma equação acima, C é uma matriz de dimensões: $D \times JD$ e μ é uma matriz com dimensões: $D \times J$. Por último, $\mathcal{N}^D(\mathcal{X}; \mu_j, C_j)$, é a densidade multivariada normal para a j -ésima componente de mistura, cuja fórmula segue abaixo:

$$\mathcal{N}^D(\mathcal{X}; \mu_j, C_j) = (2\pi)^{-D/2} |C_j|^{-1/2} \times \exp \left\{ \frac{-1}{2} (\mathcal{X} - \mu_j) C_j^{-1} (\mathcal{X} - \mu_j) \right\} \quad (3.2)$$

Nessa equação, C_j é a matriz de covariância para o j -ésimo componente de mistura e μ_j o vetor de médias para o j -ésimo componente de mistura.

3.3.1.1 IGMM (*Incremental Gaussian Mixture Model*)

O IGMM (*Incremental Gaussian Mixture Model*) é a versão incremental do GMM, baseada em modelos probabilísticos e que utiliza a distribuição de misturas gaussianas conforme visto anteriormente. A vantagem deste algoritmo ser incremental é permitir que a rede inicie a fase de aprendizagem com um número não fixo de padrões de treino, para que os novos possam ser acomodados na rede sem a necessidade de retreino completo. A IGMM utiliza a mesma equação (3.1) para definir a função densidade de probabilidade dos dados de entrada. Já as proporções de mistura, também entendidas como a probabilidade *a priori*, $p(j)$, são atualizadas e as funções de densidade de componentes são normalizadas para respeitarem as duas restrições de p , anteriormente discutidas (Mazzutti et al., 2017).

Adicionalmente, $p(\mathcal{X}|j)$, é a probabilidade de que o vetor de entrada, \mathcal{X} , pertença ao j -ésimo componente de mistura. Ela é calculada pela equação (3.2) e quando esse número for inferior a uma fração, que envolve um valor configurável pelo usuário, chamado de limiar de verossimilhança, τ_{nov} . Então, diz-se, que \mathcal{X} é rejeitado por esse j -ésimo componente de mistura. Quando \mathcal{X} é rejeitado para todos os J componentes, então cria-se novo componente, j^* , sendo que inicialmente o vetor de médias, $u_{j^*} = \mathcal{X}$, e a matriz de covariância, C_{j^*} , é equivalente a matriz identidade multiplicada por uma constante δ definida pelo usuário. Essa constante influencia no *spread* de uma componente de mistura recém adicionada, sendo sugerido para ela, a adoção de um valor baixo, como $\frac{1}{100}$. Essa constante é também multiplicada junto à variância total de cada d -ésima dimensão do vetor de entrada.

Segundo Mazzutti et al. (2017), para a atualização das distribuições do IGMM, são definidas as seguintes equações:

$$p(j|\mathcal{X}) = \frac{p(\mathcal{X}|j)p(j)}{\sum_{j=1}^J p(\mathcal{X}|j)p(j)} \quad (3.3)$$

$$sp_j^* = sp_j + p(j|\mathcal{X}), \forall j \in J \quad (3.4)$$

$$w_j = \frac{sp_j^*}{p(j|\mathcal{X})}, \forall j \in J \quad (3.5)$$

$$\mu_j^{d*} = \mu_j + w_j(\mathcal{X} - \mu_j^d), \forall j \in J, \forall d \in D \quad (3.6)$$

$$C_j^{d*} = C_j^d - (\mu_j^{d*} - \mu_j^d)(\mu_j^{d*} - \mu_j^d)' + w_j [(\mathcal{X} - \mu_j^{d*})(\mathcal{X} - \mu_j^{d*})'], \forall j \in J, \forall d \in D \quad (3.7)$$

$$p(j)^* = \frac{sp_j^*}{\sum_{q=1}^J sp_q}, \forall j \in J \quad (3.8)$$

Nas equações acima, sp_j e sp_j^* , são respectivamente a probabilidade *a posteriori* acumulada do j -ésimo componente e o seu novo valor atualizado. Já μ_j^* , é o novo vetor de médias do j -ésimo componente de mistura, C_j^* , a nova matriz de covariância do j -ésimo componente de mistura e $p(j)^*$, a nova probabilidade *a priori* atualizada.

Por último, um dado j -ésimo componente é removido do modelo sempre que essas duas condições sejam verdades: $t_j > t_{max}$ e $sp_j < sp_{min}$. A primeira condição significa que o tempo de vida acumulado do j -ésimo componente não deve ser maior do que um tempo máximo estipulado pelo usuário, para mostrar que esse componente é útil ao modelo, cuja utilidade é verificada pela segunda condição, ou seja, a probabilidade *a posteriori* acumulada do j -ésimo componente não pode estar abaixo de uma constante definida pelo usuário, sp_{min} .

3.3.2 Arquitetura da INFGMN

A arquitetura da INFGMN é dividida em cinco camadas, conforme ilustra a Figura 10 e sobre ela observam-se as seguintes notações: \mathcal{X} (vetor de entrada), \mathcal{Y} (vetor de saída), \mathcal{IV} (variáveis fuzzy de entrada ou antecedentes das regras fuzzy), \mathcal{OV} (variáveis fuzzy de saída ou consequentes das regras fuzzy), \mathcal{G} (conjunto de J componentes da mistura gaussiana), \mathcal{W} (conjunto dos pesos das regras), \mathcal{R} (conjunto de regras), K (número de regras) e M (dimensões do vetor de saída). Seguindo a ordem numérica dessa figura, as camadas são organizadas da seguinte forma: na primeira camada têm-se a apresentação do vetor de entrada; a segunda camada é relativa a IGMM; já na terceira camada, cada neurônio é associado a uma variável fuzzy, seja ela um antecedente ou consequente das regras fuzzy, enquanto na quarta camada, cada nó corresponde a uma regra fuzzy, e na quinta e última camada, os nós representam os elementos do vetor de saída.

A INFGMN têm dois modos de operação. O primeiro deles, *learning*, é ativado nas primeiras quatro camadas, seguindo essa ordem, e tem como objetivo gerar e atualizar um FIS do tipo Mamdani-Larsen, a partir dos padrões apresentados na primeira camada. Já o modo *recalling* é pré-condicionado a uma rede já estar parcialmente treinada e é ativado para as três últimas camadas. Em sua ocorrência, nenhum parâmetro ou estrutura da rede são alterados, apenas tem-se a apresentação de um ou mais padrões sem saída(s) conhecida(s) à rede, para que ela seja responsável por inferir as regras e defuzzificar a saída.

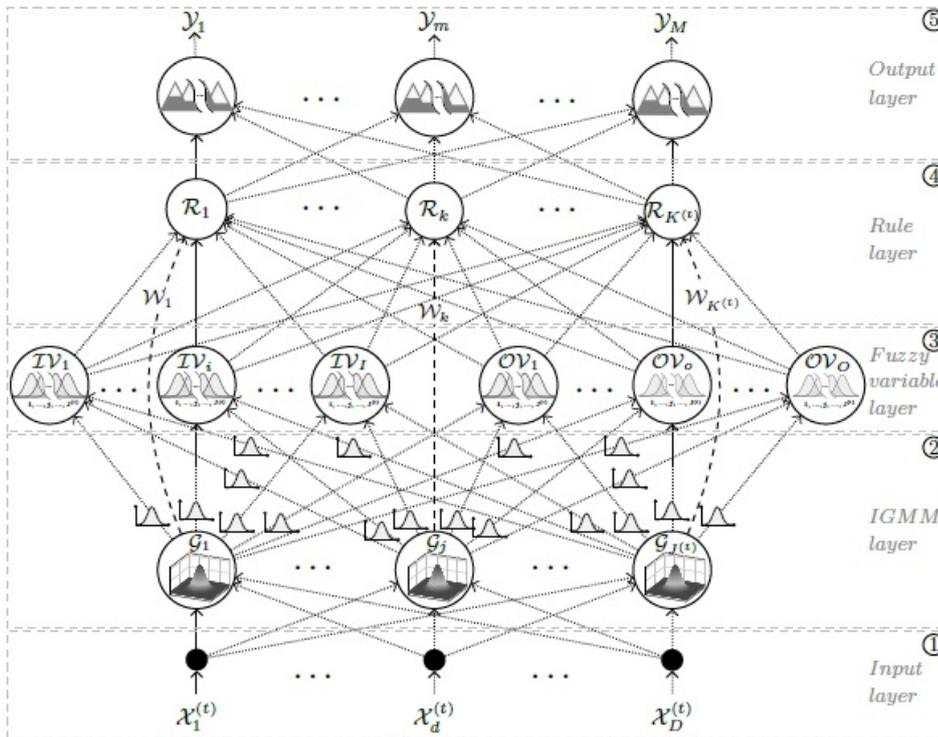


Figura 10 – Arquitetura da INFGMN (Mazutti et al., 2017).

A primeira camada (*Input Layer*) é composta essencialmente pelo vetor de dados da entrada, \mathcal{X} , de dimensão D , onde \mathcal{X}_d é o valor da d -ésima dimensão desse vetor, associado ao d -ésimo nodo, entre os D nodos distribuídos nessa camada. Note-se que esse vetor contém, também, os elementos do vetor de saída. Finalmente, todo o vetor \mathcal{X} correspondendo aos D nodos dessa camada é transmitido para todos os nós da segunda camada, sem quaisquer alterações sobre \mathcal{X} .

A segunda camada (*IGMM layer*) trabalha sobre o IGMM, após o vetor \mathcal{X} ser apresentado a essa camada. Suas funções são: criar e ajustar os parâmetros de cada distribuição gaussiana, de acordo com as equações (3.3), (3.4), (3.5), (3.6), (3.7) e (3.8). Após a ocorrência deste processo, esses valores são computados pela função, $\mathcal{G}_j(\mathcal{X}; p^*, \mu^*, C^*), \forall j \in [1, J]$, a qual produz como saída o j -ésimo componente de mistura gaussiana atualizado. Ela é representada pelo gráfico 3D da Figura 10 e também matematicamente operada pela equação (3.1), que é, por fim, o modo de operação de *learning* dessa camada. Quanto aos

valores passados para as outras camadas, as médias atualizadas, definidas por meio da variável μ^* e as covariâncias atualizadas, representadas por, C^* , são passadas à terceira camada, enquanto as probabilidades *a priori* atualizadas, identificadas por p^* , e o vetor de pesos, \mathcal{W} , são repassados diretamente a quarta camada.

A terceira camada (*Fuzzy variable layer*), compreende a criação e/ou atualização dos antecedentes e consequentes de cada regra fuzzy do tipo Mamdani-Larsen, após o treinamento de todos os padrões terem ocorrido na segunda camada. Esta camada é composta por nodos que correspondem às variáveis linguísticas, sendo que cada uma é constituída de termos linguísticos, que são conjuntos fuzzy com formato gaussiano. Assim, \mathcal{IV}_i e \mathcal{OV}_o , significam, respectivamente, i -ésima variável fuzzy de entrada, $\forall i \in [1, I]$, e a o -ésima variável fuzzy de saída, $\forall o \in [1, O]$. Cada variável linguística está associada à alguma d -ésima dimensão de \mathcal{X}_d , $\forall d \in [1, D]$. Assim, conclui-se que, $D = I + O$. Logo, a camada tem D nodos, e ainda, $O = M$, sendo a última significa que o número de variáveis fuzzy de saída é equivalente ao número de saídas da rede, ou, para um vetor de entrada quaisquer, no modo *learning*, com O dimensões de saída, teremos M saídas na rede.

Também $\mathcal{IV}_{i(j)}$ e $\mathcal{OV}_{o(j)}$ são respectivamente o j -ésimo conjunto fuzzy de uma variável linguística de entrada e o outro de saída. Isso implica que cada variável linguística, seja ela de entrada ou saída, possui fixadamente J conjuntos fuzzy. Ainda, cada j -ésimo conjunto fuzzy é composto pelos parâmetros de resultantes da segunda camada, como, $C_{i(j)}$ ou $C_{o(j)}$ e $\mu_{i(j)}$ ou $\mu_{o(j)}$. Em relação à geração e/ou atualização das funções de pertinências para cada termo linguístico dessas variáveis, de acordo com Mazzutti et al. (2017), ocorrem no modo operação *learning*, em que $\mathcal{IV}_{i(j)}(\mu_{i(j)}, C_{i(j)})$, é a $i(j)$ -ésima função de pertinência de entrada, e $\mathcal{OV}_{o(j)}(\mu_{o(j)}, C_{o(j)})$, é a $o(j)$ -ésima função de pertinência de saída.

Por último, Mazzutti et al. (2017), alegam que, no modo de operação *recalling*, essa camada tem a função de fuzzificar o vetor de entrada \mathcal{X} . Ou seja, para \mathcal{X}_i , é executada i -ésima função de pertinência da variável linguística de entrada \mathcal{IV}_i , definida por: $\phi^{(\mathcal{IV}_i)} = N^1(\mathcal{X}_i, u_{i(j)}, C_{i(j)})$. Já para o cálculo da o -ésima variável linguística de saída, \mathcal{OV}_o , a função de pertinência é fornecida por: $\phi^{(\mathcal{OV}_o)} = \int_{y_o} N^1(y_o, u_{o(j)}, C_{o(j)}) dy_o$, onde, N^1 é a função gaussiana definida anteriormente pela equação (3.2) para $D = 1$ e y_o , a saída para essa variável.

Na quarta camada (*Rule layer*), para o modo de operação *learning*, as regras fuzzy do tipo Mamdani-Larsen são criadas e/ou atualizadas por meio das variáveis linguísticas de entrada e saída geradas na terceira camada. Isso se dá a partir da equação (3.9), pela qual o número de regras geradas é igual ao número de componentes de mistura, e, portanto, $K = J$. Além disso, cada regra é composta por I termos antecedentes, os quais são sempre ligados pelo operador *AND*, e O termos consequentes.

Nessa equação $\forall m \in [1, M]$, \mathcal{X}_{I+m} é equivalente à \mathcal{Y}_m , ou seja, a m -ésima saída

da rede vista na Figura 10. Também associada a cada regra, está \mathcal{W}_k , o peso da k -ésima regra, sendo ele equivalente a probabilidade *a priori*, $p(j)$, respeitando as suas restrições já mencionadas nesse capítulo.

No modo de operação *recalling*, Mazzutti et al. (2017) definem que as regras são inferidas por meio da equação (3.10). Nessa equação, a força de disparo de cada k -ésima regra é obtida pelo resultado de um produtório entre todos termos antecedentes dessa regra, multiplicado pelo peso desta k -ésima regra, \mathcal{W}_k . Obtida as forças de disparo, então uma operação de implicação multiplicativa é incidida sobre os conjuntos fuzzy de saída de todas as regras. O próximo passo é obter um conjunto fuzzy agregado, OV^{*m} , para cada m -ésima variável de saída, através da operação de agregação que soma cada conjunto fuzzy obtido por essa m -ésima variável em todas as regras ativadas. Ao final do *recalling*, cada m -ésima variável de saída, tem um conjunto fuzzy agregado, OV^{*m} , e sua função de pertinência agregada, ϕ^{*m} .

$$f_{learning}^4 = \left\{ \begin{array}{l} \mathcal{R}^k : IF \bigwedge_{i=1}^I \mathcal{X}_i \text{ is } \mathcal{IV}_{i(k)} \\ \text{THEN} \left[\mathcal{OV}^{*m} : \bigvee_{m=o=1}^{O=M} \mathcal{X}_{I+m} \text{ is } \mathcal{OV}_{o(k)} \right] \cdot (\mathcal{W}_k) \mid k \in K, K = J \text{ and } D = I + O \end{array} \right\} \quad (3.9)$$

$$f_{recalling}^4 = \left\{ a_k : \mathcal{W}_k \cdot \prod_{i=1}^I \phi^{(\mathcal{IV}_{i(k)})} \mid k \in K \right\} \Rightarrow \left\{ \phi^{*m=o} : \sum_{k=1}^K a_k \cdot \phi^{(\mathcal{OV}_{o(k)})} \mid o \in O \text{ and } M = O \right\} \quad (3.10)$$

A quinta camada (*Output layer*) é ativada apenas no modo *recalling* e tem como função obter um valor para m -ésima variável de saída fuzzy. A equação (3.11) defuzzifica cada m -ésima saída da INFGMN, através do método da centroide, o qual é computado para cada m -ésimo conjunto fuzzy resultante do processo de agregação e aplicando sobre ele, a sua função de pertinência, ϕ^{*m} .

$$crisp_{y_m} = \left\{ \frac{\int_{y_m} y_m \phi^{*m}(y_m) dy_m}{\int_{y_m} \phi^{*m}(y_m) dy_m} \mid m \in M \right\} \quad (3.11)$$

Em que, y_m , é o universo de discurso da m -ésima variável de saída da INFGMN e $crisp_{y_m}$, o valor *crisp* obtido para m -ésima variável de saída.

3.3.3 Exemplo e considerações Finais

Para demonstrar um uso prático da INFGMN, suponha o clássico problema de classificação de dados para íris (R.A Fisher, 1936). Cada elemento deste conjunto é composto de quatro atributos de entrada, tamanho e largura de pétalas, tamanho e largura de sépalas, e um atributo de saída, a classe. Para essa última são três as possibilidades: *versicolor*, *virginica* e *setosa*. Isso faz cada vetor de entrada ter sete dimensões. O conjunto de dados total é de 150 elementos, sendo que aleatoriamente 80% deste foi treinado e o restante para fins de teste. Para o treinamento da INFGMN, a calibragem dos parâmetros se deu manualmente, com $\delta = 0.2$, $t_{max} = 1000$, $sp_{min} = 800$ e $\tau_{nov} = 0.01$.

Com três regras geradas, em várias simulações, obteve-se aproximadamente 94% de acerto. A Figura 11 mostra as regras, numeradas no canto esquerdo, e as gaussianas de cada variável fuzzy. Em amarelo, as gaussianas das variáveis de entrada e em azul, as de saída. Essa mesma figura já é acoplada de um exemplo, cujo vetor de entrada está em negrito na parte superior da figura. Observa-se, para a força de disparo da regra, por ser determinada pela conjunção multiplicativa dos termos antecedentes, que basta um dos termos antecedentes ter grau de pertinência zero, que independente do peso da regra, a força de disparo será nula, ou seja, o subconjunto fuzzy resultante para as variáveis de saída da regra, não terá influência nenhuma na fase de agregação. Na fase de defuzzificação, a variável de saída que apresentar o maior valor *crisp*, é a classe do vetor de entrada. Esse valor *crisp* é indicado no canto inferior esquerdo da mesma figura. Deste modo, pode-se concluir que *versicolor* é a classe do vetor deste exemplo.

Como visto, a INFGMN é uma técnica conexionista muito promissora, principalmente por oferecer regras do tipo Mamdani-Larsen. Entretanto resultados diferentes podem ser obtidos, dependendo da forma como são calibrados os parâmetros. Por exemplo, ao utilizar valores mais altos para t_{max} , o tempo de vida útil de um componente se torna maior, o que poderia implicar em uma redução na remoção de componentes (regras fuzzy). A existência de mais regras no FIS obtido o tornaria mais preciso, ao passo que diminuiria a interpretabilidade da base de regras.

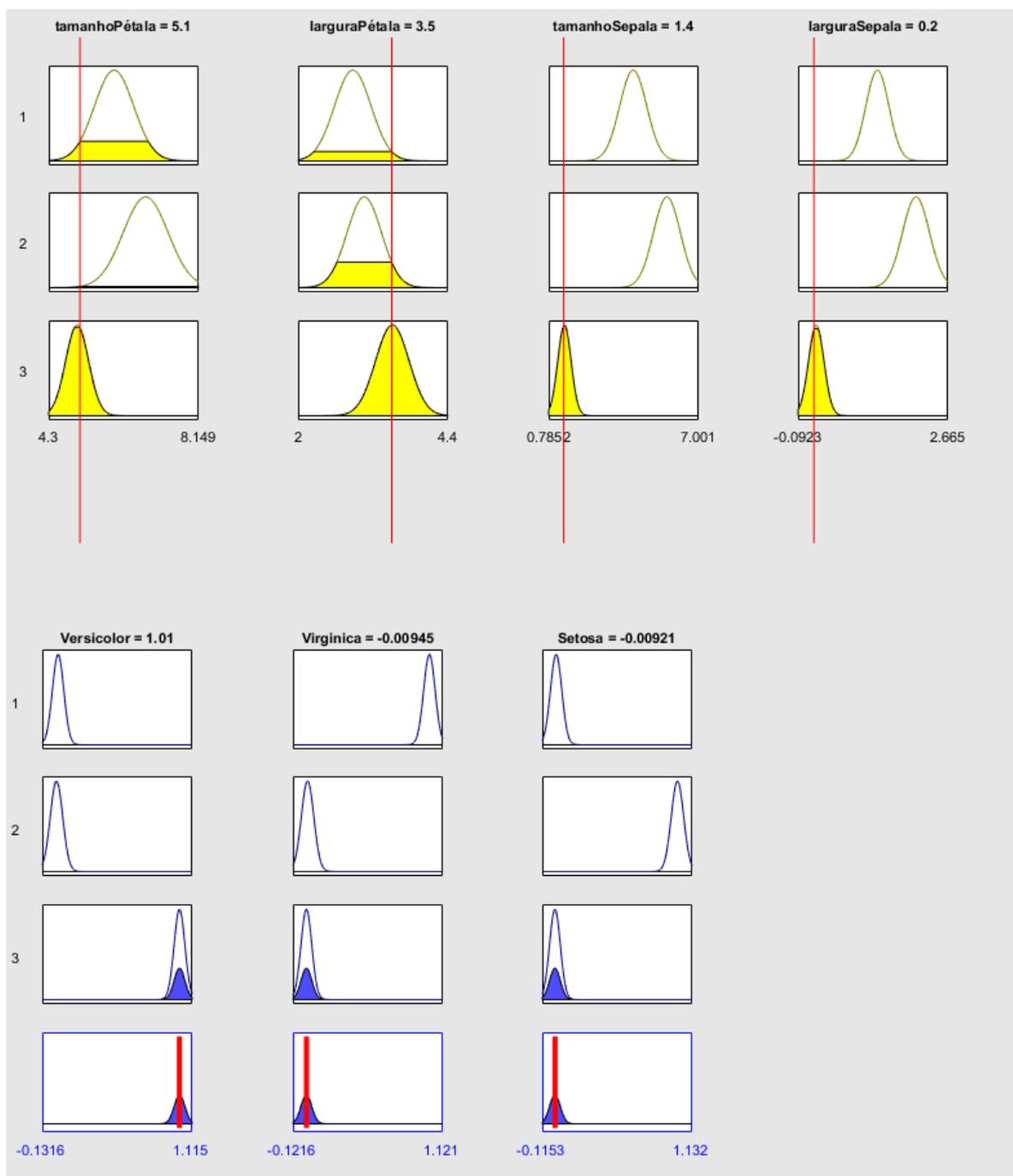


Figura 11 – Conjuntos fuzzy gaussianos para a base de dados da íris.

4 Revisão de Literatura

Neste capítulo abordaremos muito brevemente quatro trabalhos, os quais utilizaram lógica fuzzy, para simular o comportamento de multidões. O primeiro deles é um trabalho de conclusão de curso de graduação, *Modelo para Simulação de Multidões Utilizando Lógica Nebulosa*, realizado por Nasser (2011), e o segundo a ser discutido é a dissertação de mestrado de Sent (2016), *Proposta de Modelo Fuzzy do Modelo de Forças Sociais para Simulação de Pedestres em Situação de Evacuação*. Já os dois últimos trabalhos, são artigos acadêmicos, sendo o primeiro intitulado *Multi-agent system for simulating human behavior in egress simulations*, escrito por Sharma e Singh (2006), o qual utiliza lógica fuzzy, somado ao *Social Force Model* e *Geometric Model* em um simulador chamado *AvatarSimModel*. Já o outro trabalho é *Fuzzy Logic-Based Model That Incorporates Personality Traits for Heterogeneous Pedestrians*, por Xue et al. (2017), que combinam lógica fuzzy com um sistema multi-agente para reproduzir traços de personalidade dos agentes, como imprecisão nas decisões tomadas por eles.

4.1 *Modelo para Simulação de Multidões Utilizando Lógica Nebulosa*

Nasser (2011), em seu trabalho apresenta um modelo baseado em lógica fuzzy que, por meio de uma base regras desenvolvida, realiza simulações de evacuações de agentes em um *software* denominado *NetLogo*. Trata-se de um único FIS, visando substituir as equações do modelo de Helbing, Helbing et al. (1998), apresentadas já aqui, e também de outros modelos que o estendem, como *A Perceptive Model for Virtual Agents in Crowds* proposto por Cony et al. (2007).

O autor descreve o modelo como sendo uma abordagem que visa simular a percepção visual do agente, em relação a agentes e obstáculos vizinhos presentes em uma região próxima a ele, levando em conta, o ângulo e a distância. Essa percepção é dinamicamente atualizada conforme decorre a simulação. Um outro modelo é o de *Flocking*, que sincroniza o comportamento de um agente ao seu grupo de agentes, afim de evitar colisões com obstáculos e outros agentes do ambiente.

Como já dito, utiliza-se apenas um FIS para a tomada de decisão de cada agente, sendo inicialmente invocado em dois momentos alternativos. O primeiro deles sempre é relativo ao desvio de obstáculo, no qual busca-se detectar um deles na área de percepção do indivíduo da simulação. Caso haja essa detecção, então esse FIS é chamado, tendo como entradas o ângulo e a distância com este obstáculo e como saída o ajuste na direção do

agente em graus. A alternativa da invocação deste mesmo FIS ocorre quando a premissa acima é falsa, ou seja, na ausência de um obstáculo na percepção do agente.

Nessa situação alternativa é levado em consideração o comportamento de agrupamento, o qual consiste em verificar a existência de outros agentes no cone de visão deste e, em caso afirmativo, então o FIS calcula o afastamento em graus em direção oposta deste agente em relação aos outros, levando em conta a média da distância e o ângulo com a posição dos outros agentes. Uma vez feito isso, o cone do agente é atualizado e o FIS é utilizado para alinhar a direção do agente com o grupo, a partir da média do ângulo pelo qual os vizinhos se movem.

Por último, novamente ocorre a atualização do campo de visão do agente, para que se possa utilizar novamente o FIS, que determina quanto o agente deve se direcionar ao centro do grupo, dada a diferença entre o ângulo que ele se move atualmente e o ângulo com o centro do grupo e distância média com a posição de membro do grupo.

Abaixo, as duas gravuras da Figura 12 mostram respectivamente os conjuntos fuzzy e as regras de inferência fuzzy do tipo Mamdani, propostos por Nasser (2011). Neste FIS, a combinação dos conjuntos de uma regra se dá pelo operador *OR* e a fase de defuzzificação se dá pelo método da centroide.

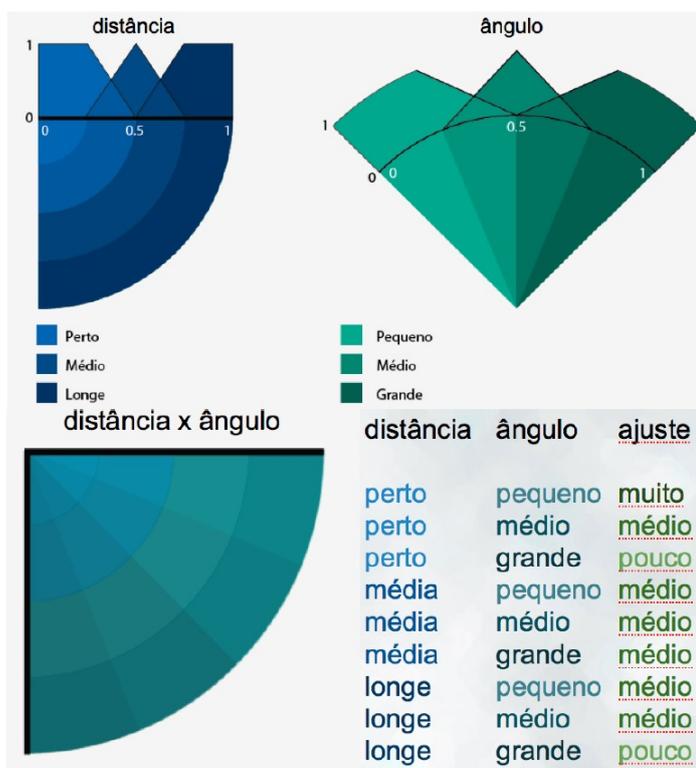


Figura 12 – Conjuntos fuzzy e na imagem abaixo, a base de regras (Nasser, 2011).

Por fim, Nasser (2011), faz experimentos em quatro cenários sugeridos na bibliografia: em corredor com alargamento, em cruzamento, em um centro de cidade e em um

ambiente sem obstáculos. Os resultados, de acordo com o autor, são muito próximos dos modelos originais, conseguindo reproduzir colisões em casos que não podem ser evitadas.

4.2 *Proposta de Modelo Fuzzy de Forças Sociais para Simulação de Pedestres em Situação de Evacuação*

Diferente do trabalho anterior, essa dissertação de mestrado trabalha mais a fundo com o modelo de Helbing and Johanson (2010). O objetivo principal da dissertação de Sent (2016) é melhorar a interpretabilidade do modelo de Helbing, a partir de um novo modelo chamado Modelo Fuzzy de Forças Sociais, o qual substitui as equações do modelo de forças sociais por conjuntos de regras linguísticas do tipo Mamdani e conjuntos fuzzy com a finalidade de simular as evacuações de multidões e seus fenômenos decorrentes. Com a finalidade de não elevar muito o custo computacional, Sent (2016), modela para cada componente de força do SFM um ou mais FIS, compostos por uma baixa quantidade de regras. Sobre a implantação das funções de pertinência, elas são calibradas empiricamente (Sent, 2016).

Já as regras são concebidas a partir do estudo da relação entre variáveis e os resultados produzidos das equações, onde frequentes ajustes são feitos nas funções de pertinência e nas regras, até que se aproximem dos resultados originais. Ao fim deste processo, o autor substitui as equações do SFM, pelos FIS gerados.

No total três são os componentes modelados, respeitando o sistema métrico, para cada componente são considerados parâmetros como posição, velocidade, distância e ângulo, conforme ilustra a Figura 13. O primeiro componente é a Força de Desejo, a qual têm dois FIS, cada um com cinco regras, na qual o primeiro é para determinar a intensidade da força necessária para que se alcance a velocidade desejada, a partir da razão entre velocidade atual e a desejada, cuja diferença é inversamente proporcional a força. O segundo FIS serve para realizar o ajuste da direção desejada de um agente, a partir do seu ângulo atual em relação a saída.

O segundo componente é a Força de Obstáculo, a qual só tem efeito em distâncias menores que 30 centímetros. Ela é composta por apenas um FIS de duas regras e considera como entrada a distância do agente para o obstáculo mais próximo, gerando como saída, a intensidade da força, sendo essa inversamente proporcional a essa distância. Como terceiro componente, a Força Social é decomposta em três FIS. O primeiro, com três regras, é referente ao ajuste do ângulo, levando em conta velocidade relativa do pedestre. O segundo, composto de quatro regras, serve para determinar a intensidade da Força Social sobre a mudança de direção do agente, levando em conta a distância com um pedestre vizinho, o ângulo com o mesmo e a velocidade relativa. Destaque-se que quanto mais próximo estiver o vizinho, maiores são o desconforto do agente e a intensidade dessa



Figura 13 – Diagrama para implementação do Modelo Fuzzy de Forças Sociais (Sent, 2016).

força. Por último, duas regras determinam a Força de Repulsão com o agente vizinho, a partir do ângulo e da velocidade relativa.

A implementação do trabalho de Sent (2016) se dá em um simulador de pedestres desenvolvido pelo mesmo autor em linguagem Julia (Sent, 2014). Em uma análise qualitativa, verifica-se, em dois cenários, um de edificação e outro com um corredor que se abre e fecha, os fenômenos de formação de filas e o outro de “Quanto Mais Rápido Mais Devagar”.

Por último, na análise quantitativa, em um cenário aberto com uma saída e variando o número de pedestres em diversas simulações aleatórias, são percebidas algumas variações em relação ao modelo original, como na velocidade média e distância média percorrida pelos agentes. De toda forma, é constatado uma boa similaridade entre o modelo de Sent (2016), com o de Helbing.

4.3 *Multi-agent system for simulating human behavior in egress simulations*

O comportamento humano em situações críticas se torna imprevisível, a medida que os indivíduos ficam emocionalmente instáveis. Assim, Sharma e Singh (2006), buscam em seu trabalho, tratar esta incerteza comportamental tornando-a interpretável, a partir do uso da lógica fuzzy, a qual é parametrizada sobre variadas condições de estresse, raiva, pânico e visibilidade do local. Todas essas informações são modeladas por meio de um simulador chamado *AvatarSim* escrito em Java (Sharma, 2004). Ele implementa várias técnicas de inteligência artificial, entre elas a abordagem multi-agente, isto é, os agentes agem de forma autônoma e singular e suas ações e objetivos são determinados pelas interações entre eles e o ambiente.

O *AvatarSim*, segundo Sharma e Singh (2006), pode ser usado para estudar

parâmetros de segurança em locais fechados, a partir de estatísticas obtidas em simulações. Esse simulador, apresentado pela Figura 14, é dividido em três módulos. O primeiro deles é *Social Force Model*, é utilizado para simular a repulsão entre os pedestres, principalmente reproduzindo os efeitos em “Quanto-mais-rápido-mais-devagar” em saídas estreitas, onde o entupimento ocorre. O segundo módulo é o *Geometric Model*, cuja finalidade é planejar para cada agente uma rota, sem colisão com os objetos do cenários, através da sua velocidade e posição atuais. O último módulo é o de lógica fuzzy, através de valores linguísticos para condições psicológicas, ambientais e sensoriais, a partir dos quais são montadas regras que determinam a consequência do comportamento. Sharma e Singh (2006), dão um exemplo de uma regra fuzzy, que se o pânico é alto e o estresse é médio, então um comportamento lento é apresentado.

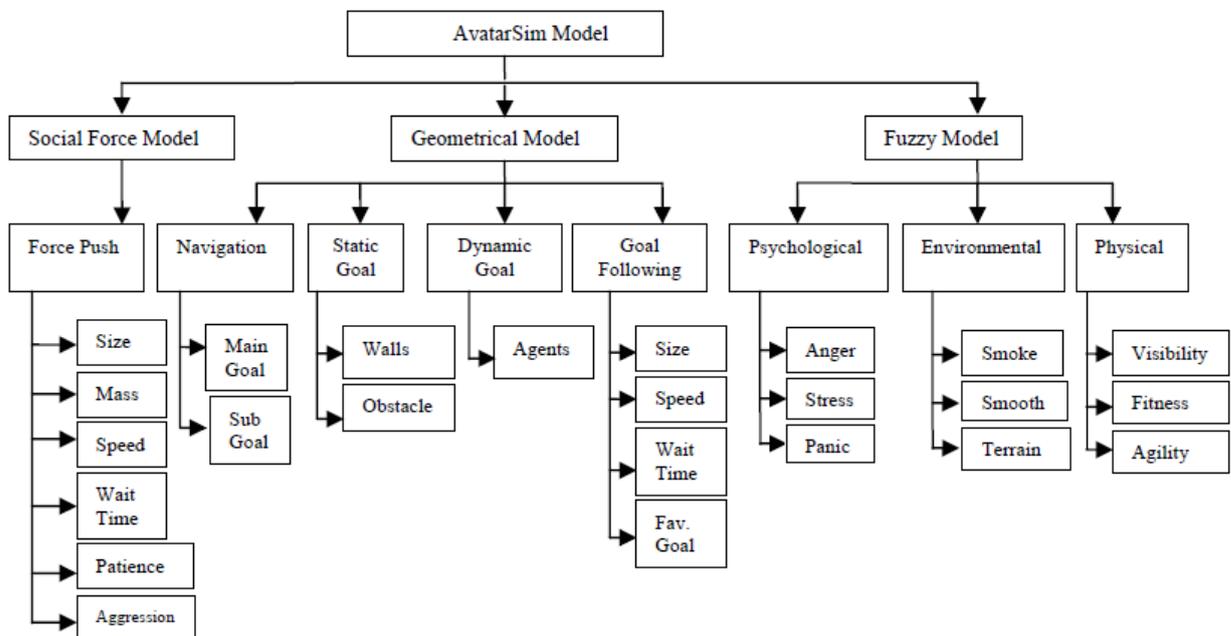


Figura 14 – Módulos do AvatarSim (Sharma and Singh, 2006).

Para experimentação e validação, os dados obtidos empiricamente através das gravações de simulações em tempo real são comparados com as simulações no *AvatarSim*, calibradas com os mesmos parâmetros das simulações gravadas. O local da simulação é uma sala de aula na *Wayne State University*, equipada com câmeras de filmagem, duas saídas e uma falsa saída. Os experimentos gravados, são realizados com 50 estudantes que foram entrevistados, de forma a ajudar na estimativa da calibragem de parâmetros para as variáveis fuzzy, as quais também são observadas pelos vídeos pós-experimentação.

Os resultados com o *AvatarSim* são bem acurados, demonstrando a influência das sensações humanas, como baixa visibilidade e estresse, por conta do atolamento de indivíduos na saída. Por fim, Sharma e Singh (2006), sugerem a adição de características físicas, como novas variáveis fuzzy, mais especificamente, idade e gênero. Por último, sugere

rem que redes neuro-fuzzy, sejam capazes de aprender estes conjuntos de dados coletados e a partir disso gerar sistemas de inferência fuzzy.

4.4 Fuzzy Logic-Based Model That Incorporates Personality Traits for Heterogeneous Pedestrians

Situações reais de evacuações nem sempre são tão precisas como equações matemáticas, isto é, a capacidade do raciocínio humano ao tomar uma decisão é muito mais complexa e às vezes imprecisa, assumindo a existência da personalidade única do indivíduo. De forma a representar isso tudo, Xue et al. (2017), incorporam um sistema de inferência fuzzy com regras Mamdani, que relaciona qualidades psicológicas com a capacidade de tomada decisão por um agente autônomo, em um sistema multi-agentes. Assim a proposta de Xue et al. (2017), faz parte da categoria de modelos microscópicos em simulações de pedestres.

Cada agente autônomo é modelado como mostra a Figura 15, ou seja, a partir de suas percepções, localização e obtenção dos dados de obstáculos e agentes próximos, e de seus traços de personalidade. O agente é capaz de raciocinar sobre alguns parâmetros, como distância dos vizinhos e tomar decisões, ou seja, ações, implicando em sua forma de movimentar. As personalidades de cada agente são programadas através do modelo OCEAN, cuja sigla corresponde a cinco traços de personalidade: *Openness*, *Conscientiousness*, *Extraversion*, *Agreeableness* e *Neuroticism*. Cada aspecto de personalidade é representado como um vetor, que é valorado de -100 a 100, no qual valores negativos refletem o inverso do que a característica diz e valores positivos intensificam esse aspecto. No caso do zero, este é um valor neutro para ela. Por exemplo, *Neuroticism* teria como valores negativos, neutros e positivos, respectivamente: calma, ansiedade moderada e nervosismo.

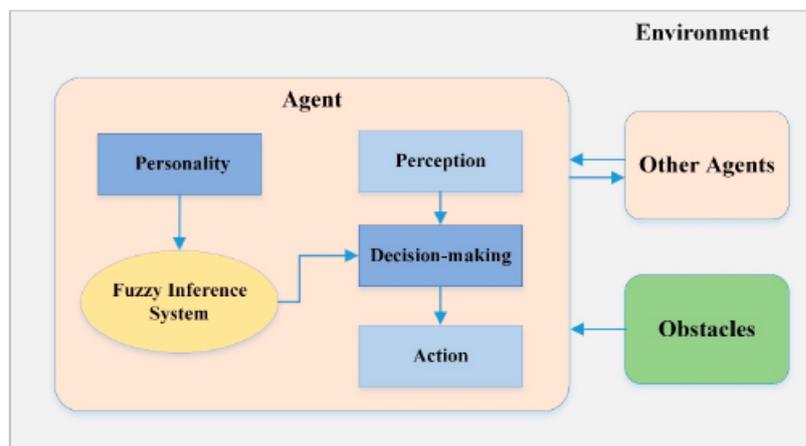


Figura 15 – Modelagem dos agentes (Xiu et al., 2017).

Conforme mostra a Figura 16, os vetores do OCEAN são as entradas do FIS, em que cada variável fuzzy está associada a uma característica OCEAN e têm três conjuntos fuzzy: negativo, neutro e positivo. A saída do FIS são variáveis de tomada de decisão, cujos valores obtidos na defuzzificação influem na ação do agente. Também cada variável de saída possui três conjuntos fuzzy. Para aumentar a interpretabilidade e reduzir o número total de regras, cada uma delas é composta por duas variáveis de entrada e uma variável fuzzy de saída. Em todas as regras elaboradas, os termos antecedentes são ligados pelo operador *AND* e ao todo 45 regras são propostas, sendo 9 delas por variável fuzzy de saída.

	Name	Universe of Discourse	Fuzzy Sets	Membership Function
Inputs	Each factor of OCEAN	[-100, 100]	Negative	Trapezoidal
			Neutral	Trapezoidal
			Positive	Trapezoidal
Outputs	NeighborDist	[3, 30]	Near	Trapezoidal
			Moderate	Trapezoidal
			Far	Trapezoidal
	MaxNeighbors	[1, 100]	Small	Trapezoidal
			Medium	Trapezoidal
			Large	Trapezoidal
	TimeHorizon	[1, 30]	Short	Trapezoidal
			Moderate	Triangular
			Long	Trapezoidal
	Radius	[0.3, 2.0]	Small	Trapezoidal
			Medium	Triangular
			Large	Trapezoidal
PrefVelocity	[1.2, 2.2]	Slow	Triangular	
		Moderate	Triangular	
		Fast	Trapezoidal	

Figura 16 – Entradas e saídas dos FIS proposto (Xiu et al., 2017).

As simulações se dão numa aplicação desenvolvida para Visual C++ e são executadas sobre cenários com passagens estreitas e saída única, composta por 200 agentes detentores de diferentes personalidades. Os resultados mostram os diferentes efeitos da personalidade sobre a trajetória executada pelos indivíduos. Por exemplo, agentes com maior intensidade positiva de *Extraversion* se movem mais rapidamente sobre a multidão e conseguem passar com mais facilidade em passagens estreitas, do que agentes com intensidade negativa desta variável. Outra observação é que quando a maioria dos agentes se encontram muito nervosos, ou seja, com *Neuroticism* muito alto, eles replicam o efeito “Quanto-mais-rápido-mais-devagar” e atrasam a saída.

5 Implementação

Toda implementação original do *Social Force Model* e da substituição da determinação de suas forças por redes neuro-fuzzy do tipo INFGMN para fins de experimento ocorre sobre um *software* em MATLAB chamado PANIC SIMULATOR (Schmidt and Späh, 2014), que foi desenvolvido por Julian Schmidt e Alexander Späh. Este capítulo apresenta os princípios básicos do funcionamento do software, o modelo proposto de forças que são determinadas pela INFGMN, discutindo os critérios adotados na captura de dados, de treino e as regras geradas. A Figura 17 resume a abordagem adotada no desenvolvimento prático deste trabalho.

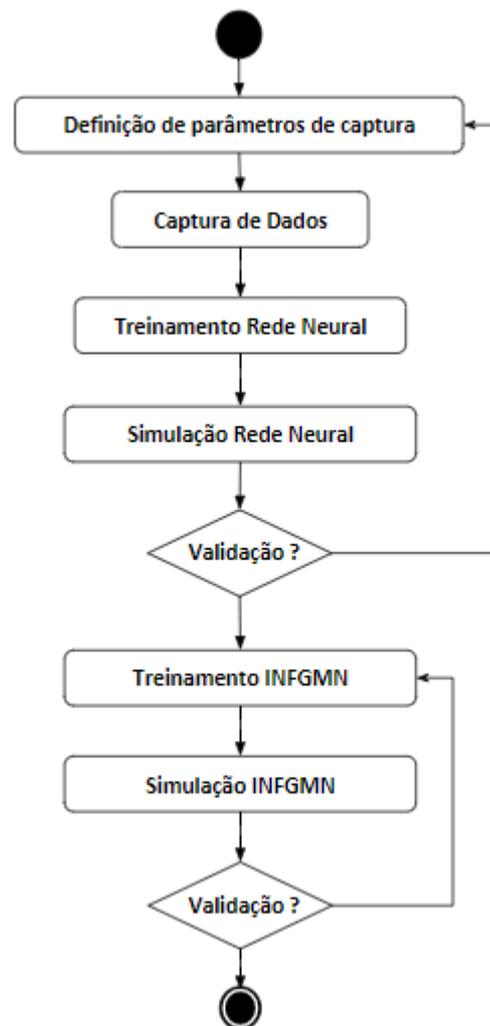


Figura 17 – Diagrama de atividades para a implementação de simulações com a INFGMN.

Uma vez compreendida a relação entre as variáveis do SFM, os dados são capturados e treinados com redes neurais *multi-layer feedforward* com algoritmo *backpropagation*. Simulações são feitas e conforme deficiências vão sendo encontradas, busca-se novamente

executar o simulador original, capturar estes dados nestas situações, treinar e simular a rede novamente. Esse processo é feito exaustivamente até conseguir reproduzir com sucesso os experimentos propostos no Capítulo 6. Então, com o mesmo conjunto de dados para cada eixo de força, utilizado para treinar a MLP, pretendemos treinar a INFGMN. Nessa etapa final, os parâmetros da INFGMN são constantemente calibrados, até conseguir uma precisão de regressão razoável, buscando obter o menor número de regras possível.

5.1 Software PANIC SIMULATOR

O PANIC SIMULATOR é um simulador desenvolvido sobre o MATLAB e que, conforme já discutido, implementa o *Social Force Model*. O simulador trabalha com agentes inseridos em um ambiente 2D e tem funcionalidades na tela principal de controle como pausar, parar e reiniciar, que podem ser vistas pela Figura 18. Nesta figura, o botão verde no canto superior direito serve para começar e pausar a simulação. Abaixo, há o botão para gravar a simulação atual e, por último, o botão de reiniciar a simulação. Ainda no topo é possível visualizar o tempo atual da simulação. Além disso, existe a possibilidade de configurar os parâmetros das equações, como velocidade, comprimento e largura da arena, número de agentes, colunas, paredes, saída, bem como posicioná-los onde bem desejarmos.

A Figura 19 mostra as duas telas principais de configuração: a primeira delas à esquerda, permite inserir os parâmetros das equações. Já na tela da direita, destacamos os botões e seus respectivos ícones no topo da tela: uma seta que permite posicionar ou remover um agente ou obstáculo (tal como alterar suas propriedades), um botão verde que permite adicionar agentes, e dois azuis realizando a mesma operação em relação a uma parede e uma coluna. Dos últimos três botões, dois são utilizados para inserir múltiplas colunas e o botão final para inserir a saída.

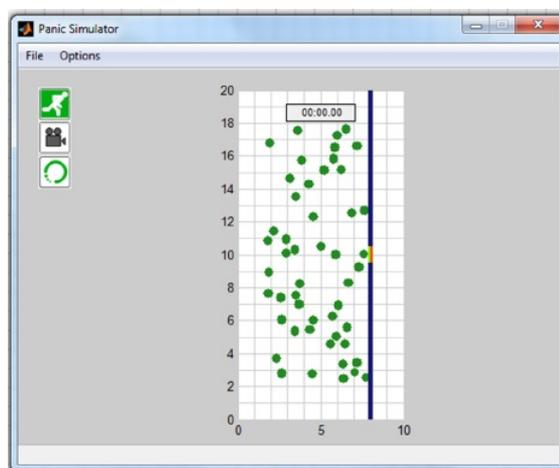


Figura 18 – Tela principal do PANIC SIMULATOR.

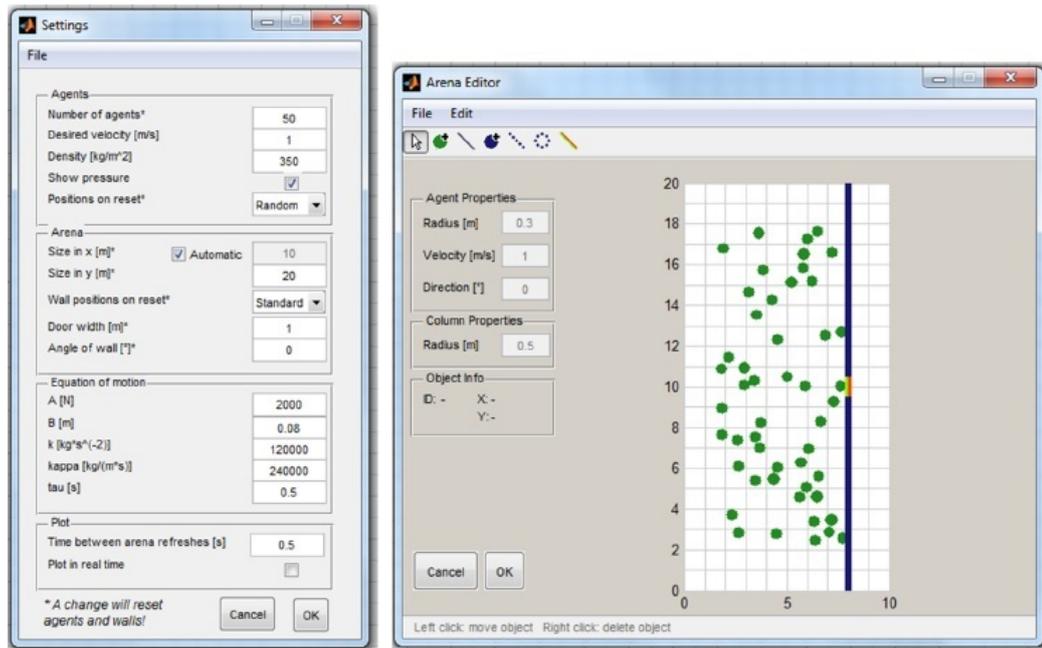


Figura 19 – Opções de configuração de parâmetros das equações e de selecionar o número de agentes.

O objetivo em cada simulação é fazer que com todos os agentes consigam sair da arena, interagindo entre si e com os obstáculos inseridos no ambiente, seguindo as especificações das equações do *Social Force Model*. A Figura 20 identifica quem são os agentes, paredes, colunas e a saída. Ao fim da simulação, um relatório aparece na tela informando o tempo de término na simulação, que corresponde ao tempo de saída do último agente, e em qual instante cada agente conseguiu sair.

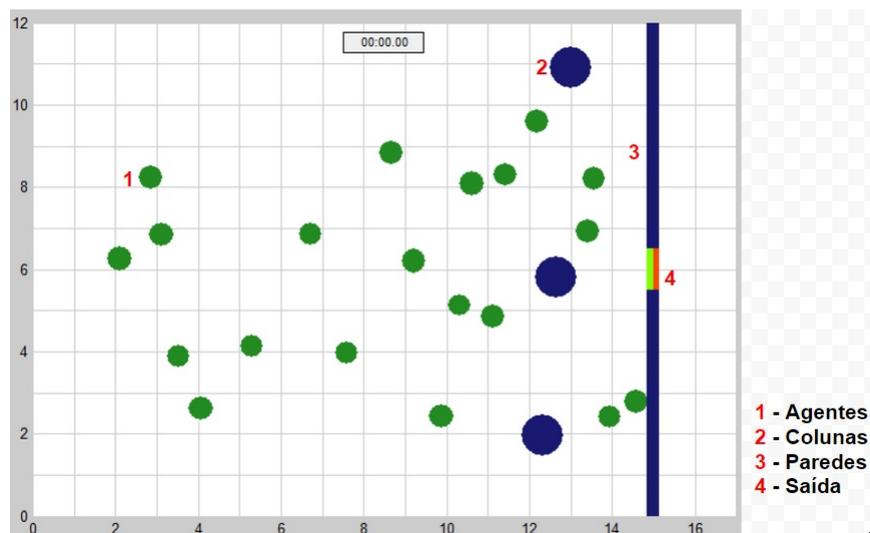


Figura 20 – Componentes físicos de uma simulação, onde sua numeração identifica o tipo do componente.

5.1.1 Eixos de força no PANIC SIMULATOR

Neste simulador, as forças exercidas sobre os agentes são individualizadas para abscissa e a ordenada do plano cartesiano. A Figura 21 exemplifica o sentido e a direção das forças do SFM na arena do PANIC SIMULATOR. Em verde, está a intensidade positiva da força para um agente e em vermelho, o valor negativo para a mesma. Essas sinalizações são exclusivamente dependentes da arena. Por exemplo, se a saída na figura estivesse à esquerda, ao invés da direita, para alcançar ela seria necessário que o valor da força de desejo no eixo- x fosse negativo para o agente.

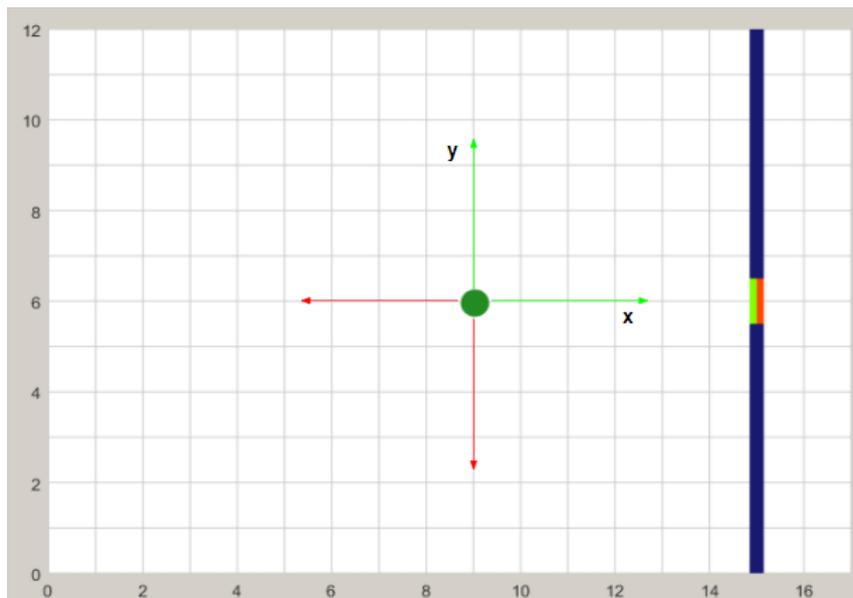


Figura 21 – Direção e sentido das forças do SFM no PANIC SIMULATOR.

5.2 Modelo proposto

A proposta de implementação prática deste trabalho consiste em substituir os componentes da equação (2.3), ignorando a computação das forças de compressão do corpo e atrito deslizante, por duas INFGMN, em que cada uma determina a força exercida sobre um agente em um dado eixo cartesiano. A INFGMN calcula essa força a partir das sensações locais do agente, como se estivesse sendo observado por uma câmera, ou seja, sua velocidade, direção em relação a saída, distâncias e direções de seus quatro agentes mais próximos e o mesmo para o obstáculo mais próximo.

O motivo para isso é que através de observações das simulações originais, foi constatado que são as entidades da simulação que possuem maior influência no cálculo da força total de um agente. Ainda, neste modelo, são mantidas as equações (2.1) e (2.2), para calcular as velocidades dos agentes e, conseqüentemente, a posição do agente no próximo instante de tempo da simulação. Em que, a velocidade no eixo- x é dependente da força no mesmo eixo e isto também ocorre para determinar a velocidade na abscissa.

Abaixo as equações especificam como a força é determinada para um agente α , por meio da INFGMN:

$$\vec{f}_{\alpha_x} = \text{fuzzyForça}_{\alpha_x}(\vec{e}_{\alpha_x}^0, \vec{v}_{\alpha_x}, \Delta\vec{d}_{\alpha\beta_1}, \Delta\vec{d}_{\alpha\beta_2}, \Delta\vec{d}_{\alpha\beta_3}, \Delta\vec{d}_{\alpha\beta_4}, \vec{n}_{\alpha\beta_1_x}, \vec{n}_{\alpha\beta_2_x}, \vec{n}_{\alpha\beta_3_x}, \vec{n}_{\alpha\beta_4_x}, \Delta\vec{d}_{\alpha i}, \vec{n}_{\alpha i_x}) \quad (5.1)$$

$$\vec{f}_{\alpha_y} = \text{fuzzyForça}_{\alpha_y}(\vec{e}_{\alpha_y}^0, \vec{v}_{\alpha_y}, \Delta\vec{d}_{\alpha\beta_1}, \Delta\vec{d}_{\alpha\beta_2}, \Delta\vec{d}_{\alpha\beta_3}, \Delta\vec{d}_{\alpha\beta_4}, \vec{n}_{\alpha\beta_1_y}, \vec{n}_{\alpha\beta_2_y}, \vec{n}_{\alpha\beta_3_y}, \vec{n}_{\alpha\beta_4_y}, \Delta\vec{d}_{\alpha i}, \vec{n}_{\alpha i_y}) \quad (5.2)$$

A primeira equação determina a força exercida no eixo- x e a segunda no eixo- y . Já, v_{α_x} e v_{α_y} são respectivamente a velocidade atual de α no eixo- x e no eixo- y . $\vec{e}_{\alpha_x}^0$ e $\vec{e}_{\alpha_y}^0$ são as direções desejadas para o eixo- x e eixo- y , e por serem implementadas por eixo do plano cartesiano, sua compreensão torna-se bem mais simples, diferindo-se da equação (2.4). Abaixo essa versão simplificada é mostrada:

$$\vec{e}_{\alpha_x}^0 = \frac{\vec{r}_{\alpha x} - \vec{r}_{ex}}{d_{\alpha e}} \quad (5.3)$$

$$\vec{e}_{\alpha_y}^0 = \frac{\vec{r}_{\alpha y} - \vec{r}_{ey}}{d_{\alpha e}} \quad (5.4)$$

Nas equações demonstradas acima $\vec{r}_{\alpha x} - \vec{r}_{ex}$ e $\vec{r}_{\alpha y} - \vec{r}_{ey}$, denotam respectivamente a distância entre as abcissas do agente e da saída, e também das ordenadas entre eles. Por último, $d_{\alpha e}$ é a distância euclidiana entre as duas entidades.

Ainda, nas equações (5.1) e (5.2), $\Delta d_{\alpha\beta}$ e $\Delta d_{\alpha i}$ é um valor de distância dado pelas respectivas somas: $r_{\alpha} + r_{\beta} - d_{\alpha\beta}$ e $r_{\alpha} + r_i - d_{\alpha i}$, que representa a soma dos raios de ambas de entidades, subtraindo a distância em relação ao centro da massa entre ambas entidades. Já $\vec{n}_{\alpha\beta_x}$, $\vec{n}_{\alpha\beta_y}$, $\vec{n}_{\alpha i_x}$ e $\vec{n}_{\alpha i_y}$ são as direções normalizadas dos agentes, com as entidades em um dado eixo e obtidas de forma análoga as equações (5.3) e (5.4) onde, nesse caso ao invés da saída, seria o agente ou obstáculo.

5.2.1 Captura de dados

A partir de simulações com o *Social Force Model* são montados dois conjuntos de dados, um relativo ao eixo- x e outro relativo ao eixo- y . Para o primeiro conjunto são coletados, como entrada, os parâmetros da equação (5.1) e como saída, a soma total de forças do *Social Force Model* para esse mesmo eixo. Este procedimento também é aplicado ao conjunto de dados do eixo- y , possuindo como entradas as variáveis da equação (5.2) e como saída a soma total de forças desse eixo.

As simulações para essa coleta consideram um total de 10 a 20 agentes, estando todos espalhados aleatoriamente em um cenário de doze metros de largura, dezessete metros de comprimento, o qual é o padrão do software. Em algumas simulações, colunas também são espalhadas no cenário, afim de aumentar a precisão na detecção de colisões. Para os demais parâmetros das equações do *Social Force Model*, são mantidas as configurações padrão do simulador, como a Tabela 1 mostra. Os valores para as constantes A , B , τ são os utilizados em experimentos feitos por Helbing (2000).

Para a massa e o raio dos agentes, o simulador gera valores aleatórios, entre 70 a 80 kg para a massa e entre 0.25 e 0.3 metros para o raio. No que se refere a velocidade desejada, esse valor é muito variável em experimentos. Entretanto, Bernadini et al. (2014), mencionam que observando os dados obtidos em um vídeo de evacuações urbanas, a velocidade média dos pedestres é no máximo de 4.0 m/s, com desvio padrão de 0.3 m/s. Assim, 4.0 m/s é o valor adotado.

Tabela 1 – Parâmetros do PANIC SIMULATOR

Nome do Parâmetro	Valor
Largura da arena	12 m
Comprimento da arena	17 m
Velocidade Desejada	4.0 m/s
A	2000
B	0.08 m
τ	0.5 s
Raio dos Pedestres	[0.25 m, 0.3 m]
Raio das Colunas	0.5 m
Massa	[70, 80] Kg

Configuração de parâmetros para simulações utilizando o *Social Force Model* no PANIC SIMULATOR e que coletam os dados para as variáveis do conjunto de treinamento da MLP e INFGMN.

5.2.2 Treinamento na MLP

Para avaliar os conjuntos de dados, primeiramente é utilizada uma MLP para cada eixo de força, da mesma forma que nas equações (5.1) e (5.2). Em ambas as redes neurais, os dados são pré-processados, conforme Li et al. (2000) recomendam. Neste caso, como é conhecida a natureza e relacionamento entre as variáveis, por meio do estudo das equações do SFM, e como os dados são precisamente capturados por um software simulador, sem risco de inconsistência numérica para as variáveis capturadas, basta, apenas, efetuar uma normalização para cada variável. Como a discrepância dos valores é muito alta, normaliza-se esses atributos entre -1 e 1, aplicando a técnica *min-max*, a qual tem a sua fórmula descrita abaixo:

$$\mathcal{X}_d^* = \left(\frac{X_d - \min_d}{\max_d - \min_d} \right) \cdot (\text{novoMax} - \text{novoMin}) + \text{novoMin} \quad (5.5)$$

Neste caso, \mathcal{X}_d^* , é o valor normalizado para o d -ésimo elemento do vetor \mathcal{X} , max_d é o maior valor encontrado na d -ésima dimensão do conjunto de vetores a se normalizar, min_d é o menor valor encontrado na d -ésima dimensão desse mesmo conjunto. Já, $novoMin = -1$, $novoMax = +1$, ou seja, a faixa de valores para qual se deseja normalizar.

A MLP utilizada tem em sua camada de entrada, neurônios associados as variáveis de entrada das equações (5.1) ou (5.2) e a saída constituída apenas pela força. Para a única camada intermediária são utilizados 30 neurônios e a função de treinamento é o de gradiente descendente, com regularização bayesiana. São treinados 85% de cada conjunto de treinamento, com os 15% restantes para testes. Ao fim do treinamento, tem-se uma regressão muito próxima de 1. A Figura 22, mostra a regressão em ambos os conjunto de dados, considerando 100% deles, ou seja, para treino e testes. Nessa figura, para cada ponto, a ordenada corresponde a força obtida pela rede e a abcissa o seu valor no conjunto de treinamento. Também consegue-se uma *performance* de média da soma dos erros quadrados na casa de 10^{-7} e gradiente de 10^{-9} . Como o capítulo de experimentos mostra, esta rede neural simulou adequadamente os fenômenos esperados.

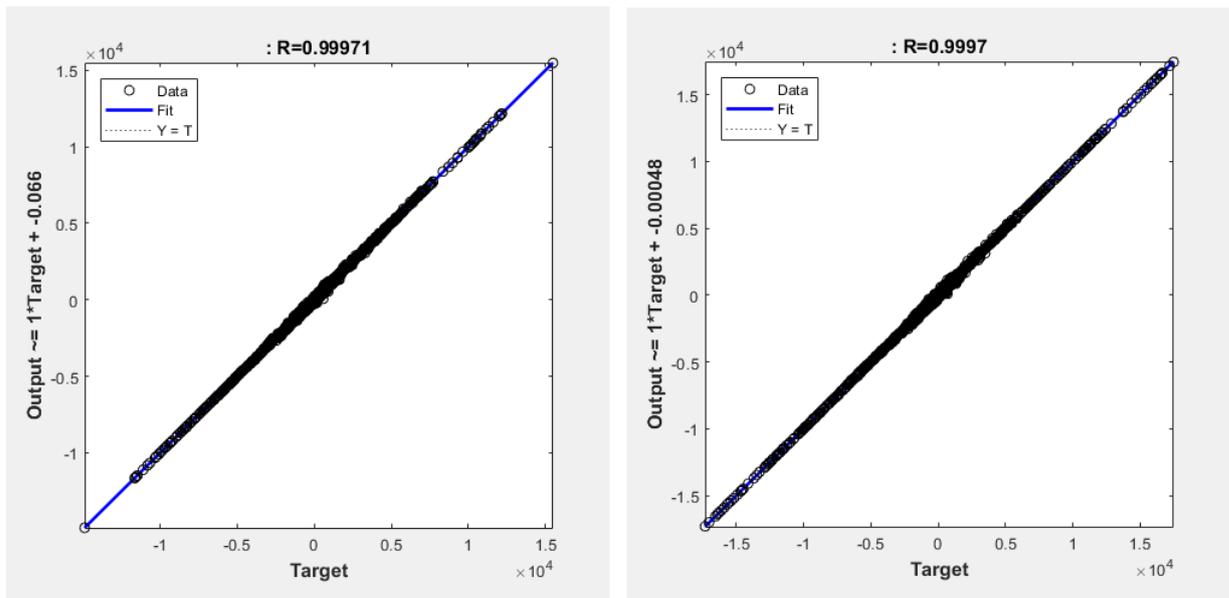


Figura 22 – Regressão da MLP para os conjuntos de treinamento da força no eixo- x e para o eixo- y .

5.2.3 Treinamento na INFGMN

O treinamento da INFGMN ocorre com os mesmos conjuntos de dados obtidos para treinar a MLP. Também o critério de pré-processamento dos conjuntos é o mesmo da MLP. Adicionalmente, cada conjunto de dados precisa ser manualmente embaralhado, visto que a biblioteca da INFGMN não realiza isso e a rede é sensível a ordem de entrada dos dados. Ao todo, 100% do conjunto de dados para cada eixo de força são treinados. Os parâmetros adotados são: $\delta = 0.001$, $t_{max} = 10000$, $sp_{min} = 1$ e $\tau_{nov} = 0.22$. Eles são manualmente calibrados, até conseguir obter uma boa relação de regressão, embora existam pequenas

exceções indicadas pelos pontos mais distantes da linha de regressão, como tudo mostra a Figura 23. É possível melhorar a precisão da regressão até 0.99, aumentando o valor t_{max} para 20000, porém o número de regras seria o dobrado e isso afetaria severamente o desempenho da aplicação, o que tornaria os experimentos impraticáveis. O motivo é que com $t_{max} = 10000$, obteve-se um resultado muito desapontador em relação a alta quantidade de regras geradas e que são discutidas a seguir.

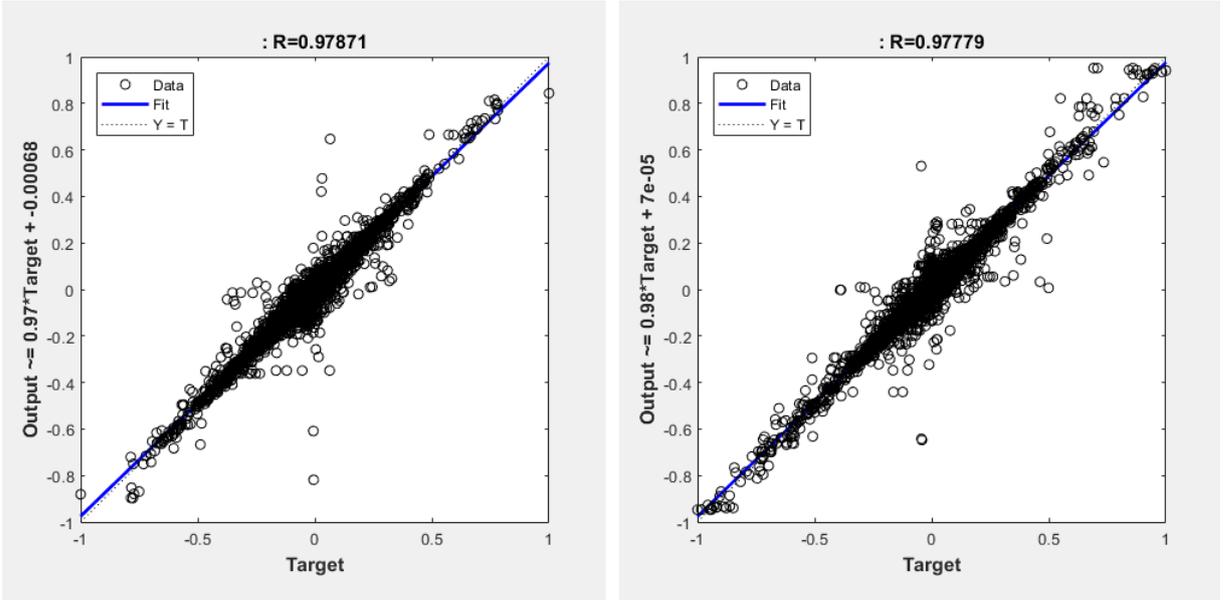


Figura 23 – Regressão da INFGMN para os conjuntos de treinamento da força no eixo- x e para o eixo- y .

5.2.4 Base de regras obtida

Para o FIS da força no eixo- x , são obtidas 13839 regras e para o eixo- y , 12963 regras. Isso não torna a resolução do problema interpretável, pois implica que cada variável fuzzy de cada FIS tem 13839 e 12963 conjuntos fuzzy, para respectivamente eixo- x e eixo- y . Abaixo são mostrados os formatos das regras criadas:

$$\mathcal{R}_1^x: \text{IF } e_x^0 \text{ is } \phi_1 \wedge v_x \text{ is } \phi_1 \wedge \Delta d_{\alpha\beta 1} \text{ is } \phi_1 \dots \wedge \Delta d_{\alpha\beta 4} \text{ is } \phi_1 \wedge \vec{n}_{\alpha\beta 1_x} \text{ is } \phi_1 \wedge \dots \wedge \vec{n}_{\alpha\beta 4_x} \text{ is } \phi_1 \wedge \Delta d_{\alpha i} \text{ is } \phi_1 \wedge \vec{n}_{\alpha i_x} \text{ is } \phi_1 \text{ THEN } fuzzyForça_{\alpha_x} \text{ is } \phi_1$$

$$\mathcal{R}_2^x \text{ IF } e_x^0 \text{ is } \phi_2 \wedge v_x \text{ is } \phi_2 \wedge \Delta d_{\alpha\beta 1} \text{ is } \phi_2 \dots \wedge \Delta d_{\alpha\beta 4} \text{ is } \phi_2 \wedge \vec{n}_{\alpha\beta 1_x} \text{ is } \phi_2 \wedge \dots \wedge \vec{n}_{\alpha\beta 4_x} \text{ is } \phi_2 \wedge \Delta d_{\alpha i} \text{ is } \phi_2 \wedge \vec{n}_{\alpha i_x} \text{ is } \phi_2 \text{ THEN } fuzzyForça_{\alpha_x} \text{ is } \phi_2$$

...

$$\mathcal{R}_{13839}^x: \text{IF } e_x^0 \text{ is } \phi_{13089} \wedge v_y \text{ is } \phi_{13089} \wedge \Delta d_{\alpha\beta 1} \text{ is } \phi_{13089} \dots \wedge \Delta d_{\alpha\beta 4} \text{ is } \phi_{13089} \wedge \vec{n}_{\alpha\beta 1_x} \text{ is } \phi_{13089} \wedge \dots \wedge \vec{n}_{\alpha\beta 4_x} \text{ is } \phi_{13089} \wedge \Delta d_{\alpha i} \text{ is } \phi_{13089} \wedge \vec{n}_{\alpha i_x} \text{ is } \phi_{13089} \text{ THEN } fuzzyForça_{\alpha_x} \text{ is } \phi_{13089}$$

$$\mathcal{R}_1^y: \text{IF } e_y^0 \text{ is } \phi_1 \wedge v_x \text{ is } \phi_1 \wedge \Delta d_{\alpha\beta 1} \text{ is } \phi_1 \dots \wedge \Delta d_{\alpha\beta 4} \text{ is } \phi_1 \wedge \vec{n}_{\alpha\beta 1_y} \text{ is } \phi_1 \wedge \dots \wedge \vec{n}_{\alpha\beta 4_y} \text{ is } \phi_1 \wedge \Delta d_{\alpha i} \text{ is } \phi_1 \wedge \vec{n}_{\alpha i_y} \text{ is } \phi_1 \text{ THEN } fuzzyForça_{\alpha_y} \text{ is } \phi_1$$

\mathcal{R}_2^y : IF e_y^0 is $\phi_2 \wedge v_y$ is $\phi_2 \wedge \Delta d_{\alpha\beta 1}$ is $\phi_2 \dots \wedge \Delta d_{\alpha\beta 4}$ is $\phi_2 \wedge \vec{n}_{\alpha\beta 1_y}$ is $\phi_2 \wedge \dots \wedge \vec{n}_{\alpha\beta 4_x}$ is $\phi_2 \wedge \Delta d_{\alpha i}$ is $\phi_2 \wedge \vec{n}_{\alpha i_y}$ is ϕ_2 THEN $fuzzyForça_{\alpha_y}$ is ϕ_2

...

\mathcal{R}_{12963}^y : IF e_y^0 is $\phi_{12963} \wedge v_y$ is $\phi_{12963} \wedge \Delta d_{\alpha\beta 1}$ is $\phi_{12963} \dots \wedge \Delta d_{\alpha\beta 4}$ is $\phi_{12963} \wedge \vec{n}_{\alpha\beta 1_y}$ is $\phi_{12963} \wedge \dots \wedge \vec{n}_{\alpha\beta 4_y}$ is $\phi_{12963} \wedge \Delta d_{\alpha i}$ is $\phi_{12963} \wedge \vec{n}_{\alpha i_y}$ is ϕ_{12963} THEN $fuzzyForça_{\alpha_y}$ is ϕ_{12963}

Sabe-se que o número de regras em um sistema fuzzy cresce exponencialmente na mesma proporção em que mais variáveis fuzzy são adicionadas ao sistema. Por isso, mesmo considerando o uso de outra rede neuro-fuzzy, embora muito menos interpretável por ter regras Takagi-Sugeno, como a ANFIS, não surtiria efeito significativo na redução de regras, pois há dezenas de variáveis de entrada. Uma possibilidade para reduzir o número destes conjuntos seria reduzir o número de entradas, considerando apenas um agente vizinho mais próximo.

Neste caso, duas consequências seriam previsíveis: a primeira delas é que a regressão ficaria muito mais imprecisa, pois as técnicas conexionistas não saberiam diferenciar, quando há mais que um agente próximo. Ou seja, poderia se ter intensidade de forças completamente diferentes para um vizinho posicionado no mesmo lugar e mesma direção, além de não contemplar as sensações locais de um agente, se o mesmo se encontrar cercado por outros vizinhos. Em segundo, embora menos regras fossem geradas, ainda haveria uma quantidade absurda delas, pois no mínimo seriam consideradas 6 entradas, como velocidade, distâncias relativas a um pedestre vizinho e ao obstáculo mais próximos, e direções em relação a saída, pedestre e obstáculo mais próximo.

Uma alternativa é fazer, como Sent (2016), e tratar separadamente os três componentes forças, o que em seu trabalho gerou pouquíssimas regras, embora as tenha elaborado manualmente. De toda forma, menos variáveis seriam necessárias para cada sistema de inferência fuzzy: duas para a força de desejo e para obstáculo, e no mínimo duas para a força social, ou seja, dependendo do número de pedestres próximos a ser considerado, haveria o risco dessa última força conter um número alto de regras.

Isso sem contar, que o propósito desta pesquisa é gerar uma força exercida sobre o agente em cada eixo e ignorar toda fundamentação teórica e matemática complicada, que envolve entender o que cada componente de força realiza, pois existem somas de forças, constantes, exponenciações com as distâncias e por assim, se sucede. Ou seja, não haveria a abstração dos componentes de forças em apenas uma única, de tal forma que isso representasse a resposta de um agente à seus estímulos sensoriais físicos em um instante de tempo.

6 Experimentos e Resultados

De forma a validar esses conjuntos de dados são comparadas simulações do SFM com a MLP, para que posteriormente a INFGMN seja treinada com esses mesmos conjuntos de dados e, então, esteja sujeita a comparações com os outros dois modelos, observando se os fenômenos típicos de comportamentos de evacuação aparecem na nossa simulação. Cada comparação é composta por uma análise de duas etapas. A primeira, qualitativa, busca observar os fenômenos em multidões. Já a segunda etapa é uma análise quantitativa, em que coleta-se dados estatísticos das simulações, como as médias referente às forças medidas, o tempo de duração e o tempo médio em que cada agente sai do ambiente da simulação.

Em ambas as análises, dois cenários são utilizados, mas com uma quantidade de agentes variando, em dependência com a análise feita e da técnica conexionista aplicada. Isso se faz necessário, uma vez que a INFGMN gerou mais de 12 mil regras para cada FIS, deixando o desempenho do simulador extremamente lento. O primeiro cenário, à esquerda na Figura 24, é o cenário padrão do simulador, onde os agentes estão em uma arena com 12 metros de largura e 17 metros de comprimento (12x17 metros), disputando a única saída, sendo ela cercada de duas paredes (obstáculos). O segundo cenário, à direita da Figura 24, tem dimensões de 20x35 metros e várias colunas e obstáculos, espalhadas no cenário e os agentes afastados o máximo possível da saída, de forma a observar melhor seus movimentos a uma distância maior.

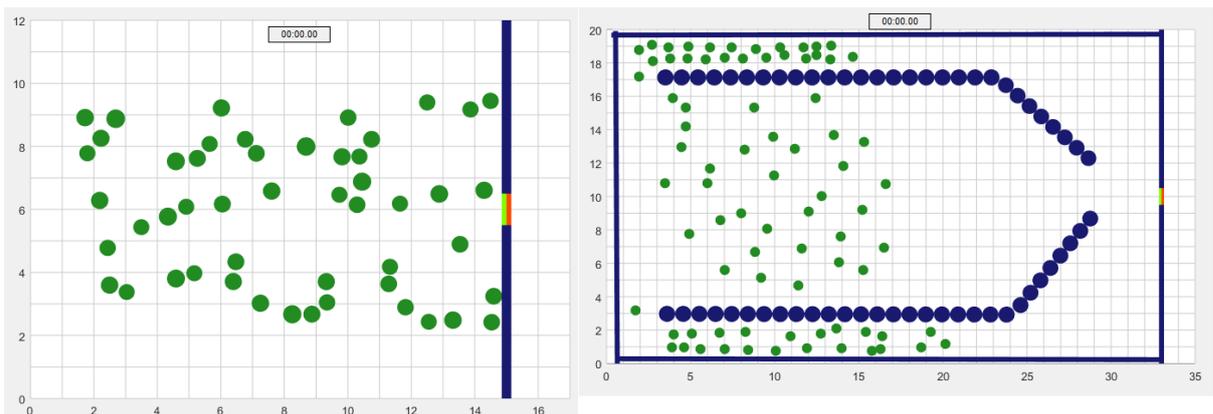


Figura 24 – Cenários de simulação do PANIC SIMULATOR para INFGMN e MLP.

A tabela abaixo indica quais parâmetros do *Social Force Model* são utilizados nos experimentos e quais são ignorados, dependendo do modelo simulado. A justificativa do uso destes valores para a maior parte dos parâmetros é a mesma fornecida no capítulo anterior. Entretanto, para a obtenção da velocidade, o valor da massa foi uniformizado e segundo, Bernadini et al. (2014), é utilizado 80 kg para adultos em simulações de

pedestres.

Tabela 2 – Parâmetros para experimentos

Parâmetro		Modelo		
Nome	Valor	SFM	MLP	INFGMN
Velocidade Desejada	4.0 m/s	sim	não	não
A	2000	sim	não	não
B	0.08 m	sim	não	não
τ	0.5 s	sim	sim	sim
Raio dos Pedestres	[0.25, 0.3] m	sim	sim	sim
Raio das Colunas	0.5 m	sim	sim	sim
Massa	80 Kg	sim	sim	sim

Configuração de parâmetros para simulações do PANIC SIMULATOR, utilizando o *Social Force Model*, redes neurais do tipo MLP e redes neuro-fuzzy do tipo INFGMN.

6.1 Análise qualitativa

Para ilustrar o potencial dos conjuntos de dados coletados, utilizou-se muito mais agentes para os experimentos com a MLP do que para INFGMN. Assim, as simulações na MLP utilizam 50 agentes, a quantidade padrão do simulador para o experimento no primeiro cenário e 80 agentes no segundo. Para a INFGMN, os dois cenários são simulados com 10 agentes.

As gravuras da Figura 24 representam o tempo zero em ambos o cenários com a MLP. No primeiro cenário, a Figura 25 mostra três momentos importantes da simulação e, sequencialmente, ordenados de *a* até *c*. Neste local é possível observar os pedestres se dirigindo para a saída (força de desejo), a repulsão entre os agentes vizinhos, e também a repulsão com os obstáculos, o que respectivamente impede os agentes de ocuparem um mesmo espaço e de atravessarem as paredes. Entre os fenômenos capturados, as duas últimas gravuras destacam o “Quanto-mais-rápido-mais-devagar” e parcialmente o “Congelamento-por-aquecimento”, visto que sem as forças de compressão e deslize, os indivíduos diminuem drasticamente suas velocidades, mas não ficam completamente paralisados no gargalo.

Já o segundo experimento com a MLP, apresentado na Figura 26, contém sete momentos ordenadas de *a* até *f*. Os primeiros três momentos, demonstram a atuação da força de repulsão com as colunas do cenário e a “Formação de Faixas” em um mesmo sentido, havendo uma faixa na parte de baixo e outras duas na parte de cima da arena. Além disso, visível entre a segunda e quarta gravura, há um cruzamento entre três fluxos de pedestres, ocorrendo assim o fenômeno da “Oscilação de fluxos em gargalos”, onde esses três se alternam na única passagem disponível, a qual é a saída. Também o “Quanto-mais-rápido-mais-devagar”, se faz presente nas três últimas gravuras, em conjunto com o

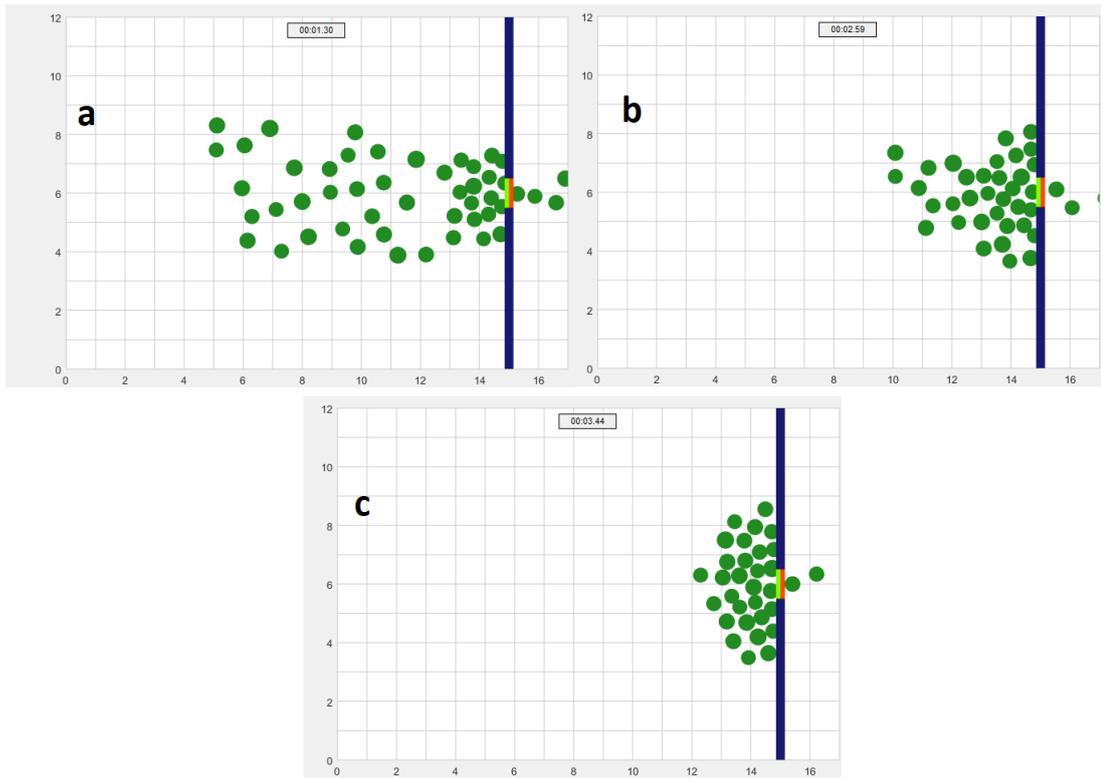


Figura 25 – Sequência de simulação para a MLP, no primeiro cenário e ordenada da gravura *a* até *c*.

“Congelamento-por-aquecimento”, que se dá por haver uma quantidade muito maior de pedestres no gargalo, o que permite visualizar a ideia de que eles ficam em um estado sólido. Assim, nas duas últimas gravuras, é possível verificar que os pedestres que estão no meio do atolamento e aqueles próximos às paredes da saída, estão paralisados.

Utilizando a INFGMN, os resultados ficam aquém do esperado, conferindo-se isso pelas Figuras 27, no cenário 1, e 28, para o cenário 2. Logo, essa técnica é incapaz de reproduzir quaisquer fenômenos em multidões, sequer conseguindo simular as forças de repulsão, ou seja, os agentes atravessam as paredes, como consta na segunda gravura das Figuras 27 e 28, e também não se repelem entre si, ignorando a existência de outros agentes.

Ainda assim, os agentes seguem para o sentido da arena de onde se localiza a saída e, em alguns casos do primeiro cenário, como nas duas últimas ilustrações da Figura 27, é possível observar alguns indivíduos atingirem exatamente a posição de onde se encontra a saída, reproduzindo assim a Força de Desejo. Dois possíveis motivos para isso, é que a maior parte do conjunto de treinamento é obtido em um cenário muito próximo à esse, e que a INFGMN responde melhor para agentes dos quais seus vizinhos e obstáculos próximos estejam mais distantes.

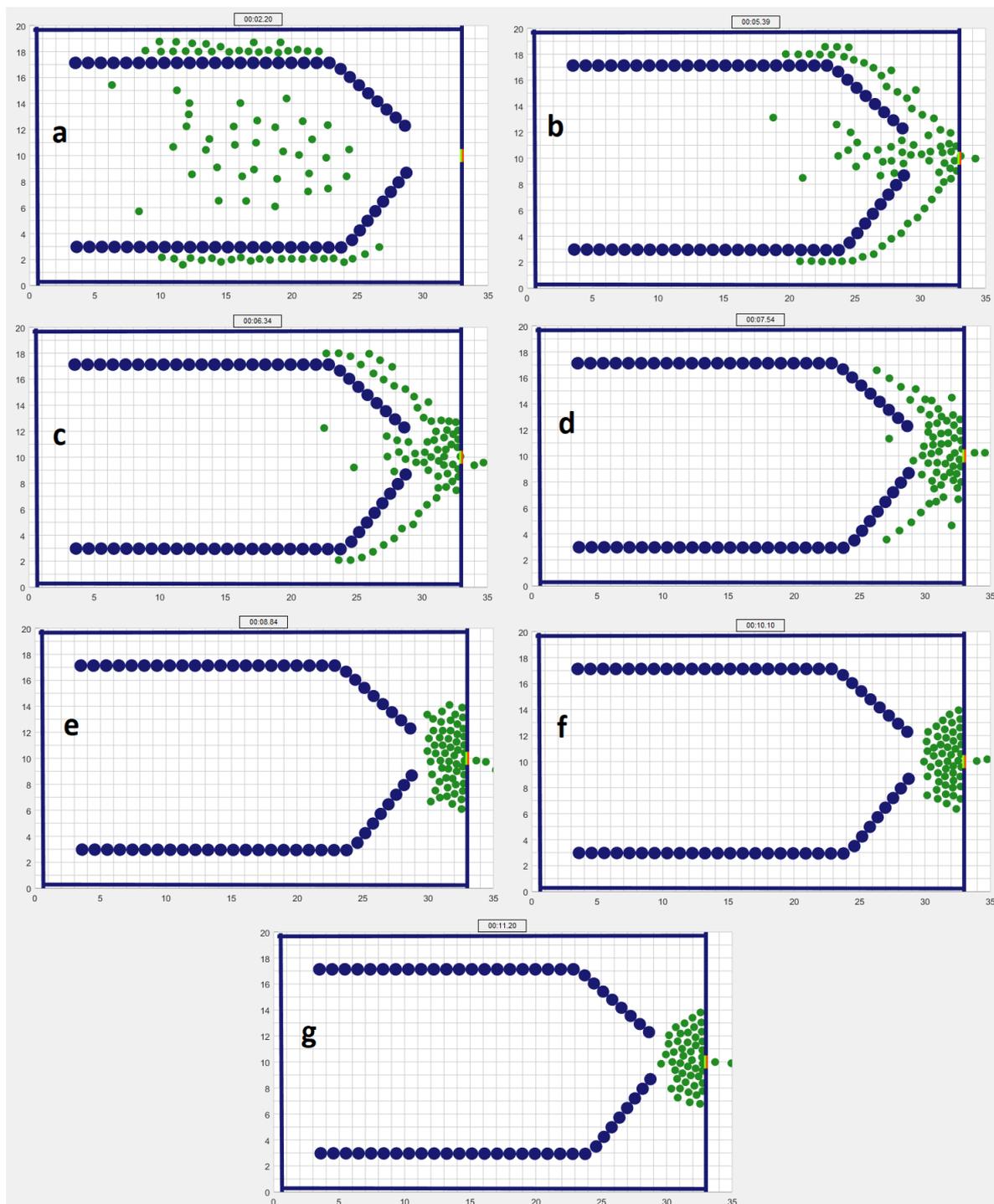


Figura 26 – Sequência de simulação para a MLP, no segundo cenário e ordenada da gravura *a* até *f*.

6.2 Análise quantitativa

Para a última etapa de análise são simulados 10 agentes para o SFM, MLP e a INFGMN, sendo que em todos os três casos, os agentes estão exatamente na mesma posição para um dado cenário. O posicionamento nessa simulação não é importante, desde que respeite o intervalo das variáveis no conjunto de treinamento. A Figura 29 indica esse posicionamento nos dois cenários.

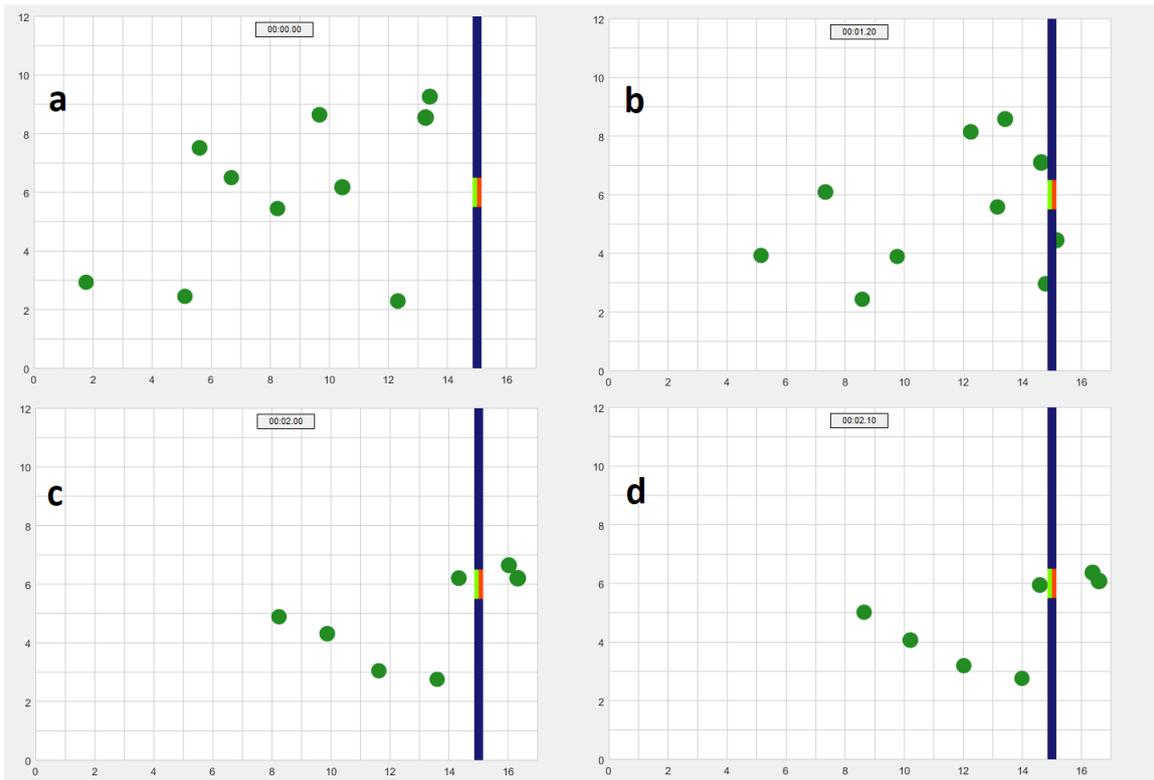


Figura 27 – Sequência de simulação para a INFGMN, no segundo cenário e ordenada da gravura *a* até *d*.

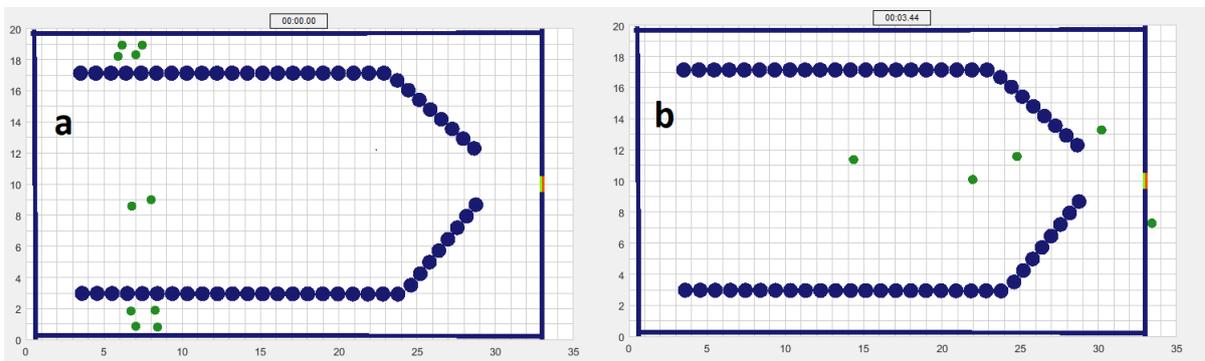


Figura 28 – Sequência de simulação para a INFGMN, no segundo cenário, ordenada de *a* até *b*.

As Figuras 30 e 31 mostram a média da força obtida para os 10 agentes em cada passo, ou seja, para cada invocação do SFM ou MLP ou INFGMN. O número de passos em cada simulação varia, dependendo da intensidade de força obtida pelos agentes. Visto isso, é possível averiguar uma boa similaridade entre as médias obtidas pelo SFM e o MLP, principalmente para forças do eixo-*x*, o que por consequência permite reproduzir mais fielmente as forças e fenômenos do SFM.

Em contrapartida, no caso da INFGMN, ela apresenta uma similaridade numérica muito baixa com o SFM e contém grandes picos de força, fazendo com que os agentes saiam muito antes do previsto. Isso é perceptível principalmente no segundo cenário, onde com intensidade de forças muito altas nos dois eixos, os agentes iniciam a simulação realizando movimentos completamente descoordenados, para que depois, havendo um certo ajuste

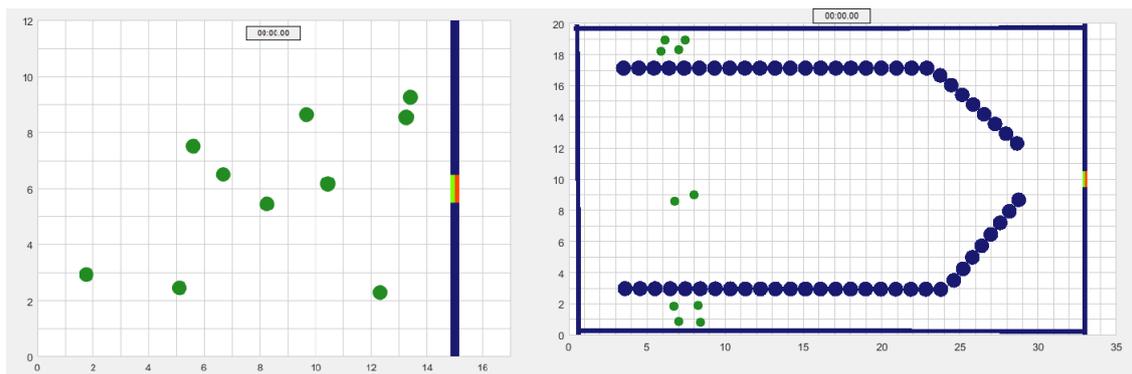


Figura 29 – Posicionamento dos agentes para a coleta de dados estatísticos, para o cenário 1 e 2.

na força do eixo- x , consigam sair do cenário, embora geralmente saiam pelas laterais da arena.

As Tabelas 3 e 4 mostram para cada modelo, a média total das forças obtidas na simulação a um dado eixo e o seu respectivo desvio padrão. Todos os valores foram arredondados para as duas primeiras casas decimais. Como resultado, novamente destaca-se uma proporção numérica muito similar entre as estatísticas obtidas pela MLP e a SFM, demonstrando a alta capacidade regressão da MLP. Apesar da INFGMN ter médias bem mais distantes dos outros dois modelos, o seu desvio padrão mostra-se parcialmente compatível com eles, principalmente no cenário 2.

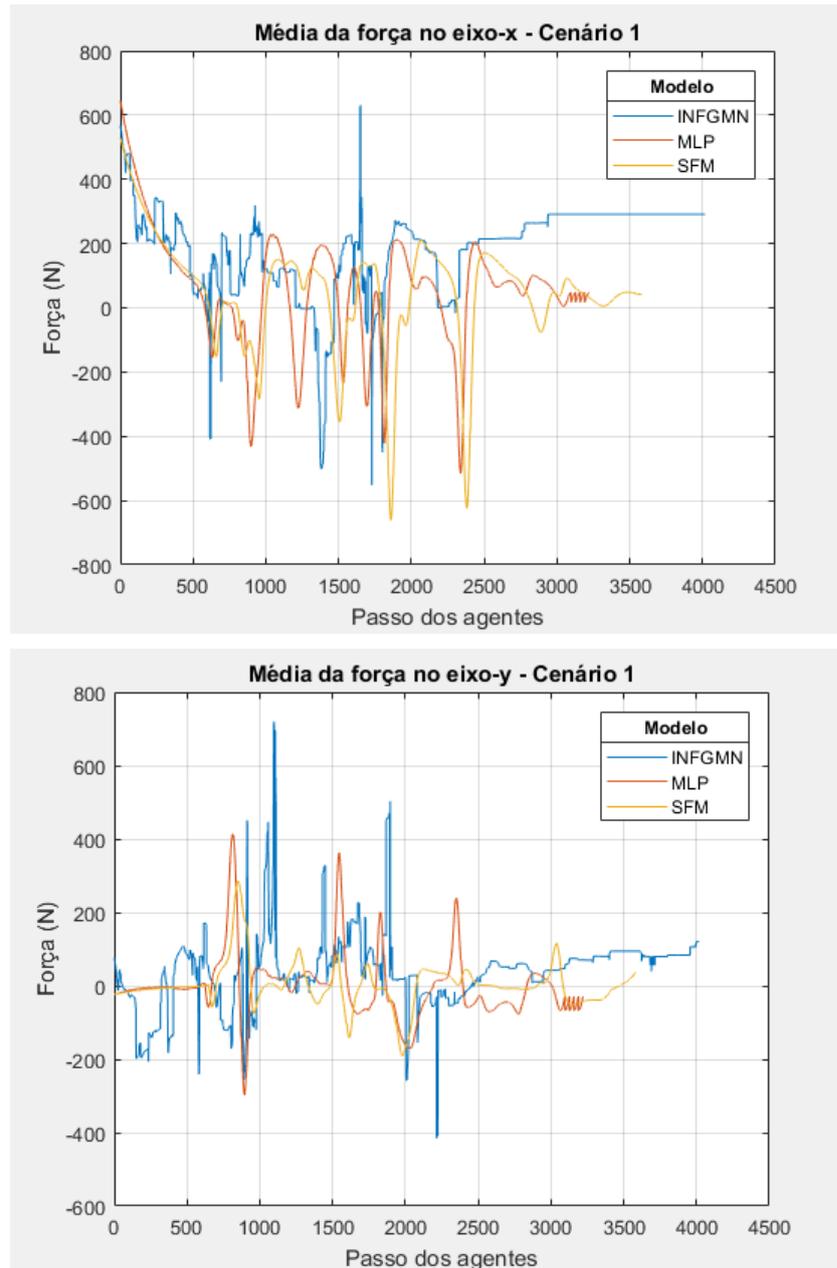
Tabela 3 – Resultados das forças para o cenário 1

Medida	SFM	MLP	INFGMN
Média da força (eixo- x)	49.66 N	58.0 N	186.73 N
Média da força (eixo- y)	2.24 N	-1.45 N	39.32 N
Desvio padrão da força (eixo- x)	588.82 N	663.57 N	355.66 N
Desvio padrão da força (eixo- y)	420.52 N	502.32 N	347.63 N

Tabela 4 – Resultados das forças para o cenário 2

Medida	SFM	MLP	INFGMN
Média da força (eixo- x)	16.33 N	18.50 N	262.73 N
Média da força (eixo- y)	0.40 N	0.80 N	21.70 N
Desvio padrão da força (eixo- x)	504.25 N	413.49 N	514.09 N
Desvio padrão da força (eixo- y)	518.19 N	598.12 N	484.30 N

A seguir, os dois gráficos de barra, 32 e 33, expõem respectivamente para o primeiro e segundo cenários, os tempos de saída de cada agente em uma simulação com SFM, MLP e INFGMN. As abcissas desse gráfico indicam o número de agentes que saíram até o instante de tempo fornecido pelas ordenadas. Observa-se ali que, por consequência da boa regressão obtida pela MLP, o tempo de saídas dos agentes para MLP e SFM são muito próximos, o que demonstra que os conjuntos de selecionados são realmente muito bons. Para a INFGMN, mesmo com fraca aproximação numérica na obtenção das forças,

Figura 30 – Média das forças no cenário 1 para o eixo- x e eixo- y

os tempos de saída foram muito próximos aos da MLP e SFM. Isso ocorre porque a maior parte do conjunto de treinamento é obtido em um cenário próximo do primeiro.

Depois, as Tabelas 5 e 6, indicam para cada cenário em cada modelo, a duração da simulação em segundos que equivale ao tempo de saída do último agente, a média de tempo da saída dos agentes e seu respectivo desvio padrão. Todos os valores são arredondados para as duas primeiras casas decimais. A partir disso, nota-se uma boa aproximação na duração de simulação do cenário 1, com os três modelos e uma grande diferença para INFGMN no cenário 2, onde por possuir grandes pico de força, a saída dos agentes torna-se precipitada e a duração da simulação mais curta.

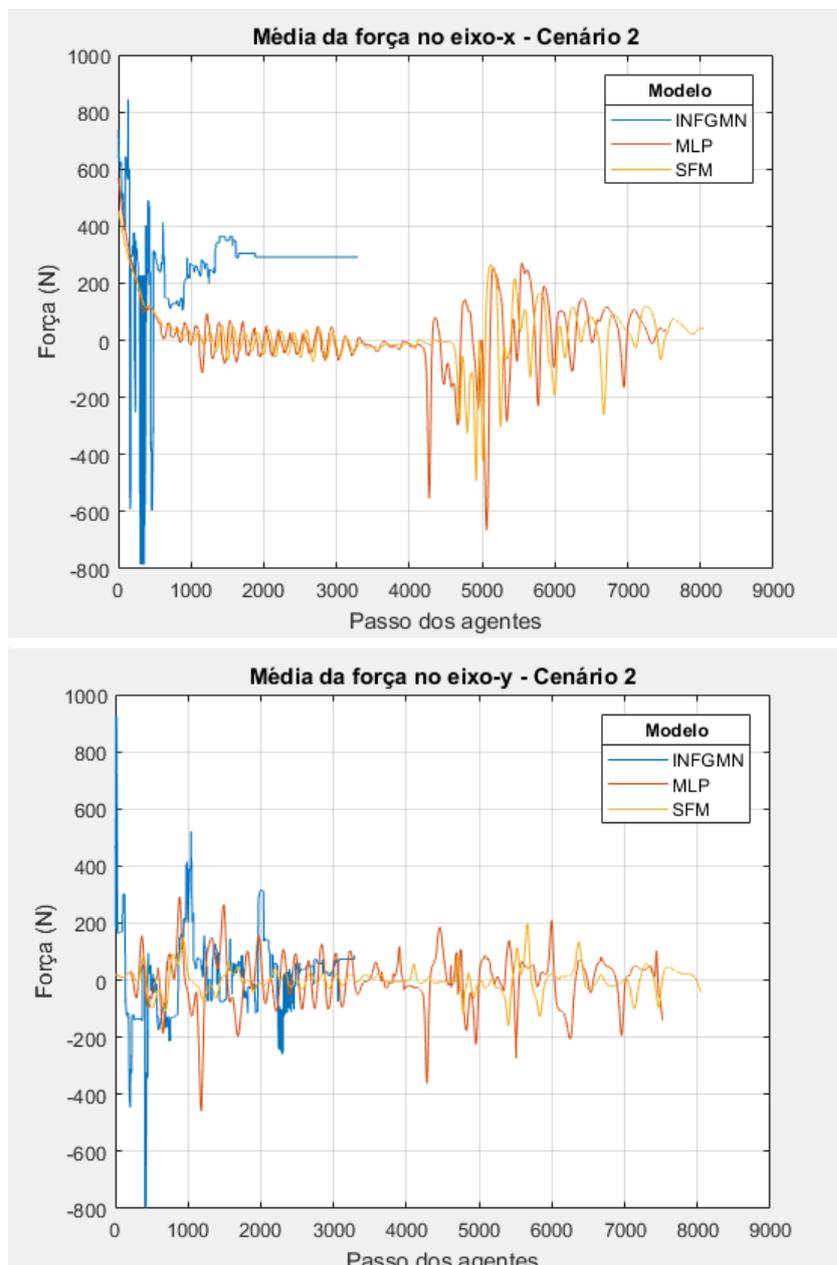


Figura 31 – Média das forças no cenário 2 para o eixo-*x* e eixo-*y*

Tabela 5 – Tempos para o cenário 1

Medida	SFM	MLP	INFGMN
Duração da simulação	5.75 s	4.95 s	6.43 s
Média dos agentes	2.59 s	3.08 s	3.29 s
Desvio padrão	1.60 s	1.16 s	1.39 s

Tabela 6 – Tempos para o cenário 2

Medida	SFM	MLP	INFGMN
Duração da simulação	12.94 s	12.06 s	5.10 s
Média dos agentes	9.51 s	9.43 s	3.80 s
Desvio padrão	2.20 s	1.74 s	0.83 s

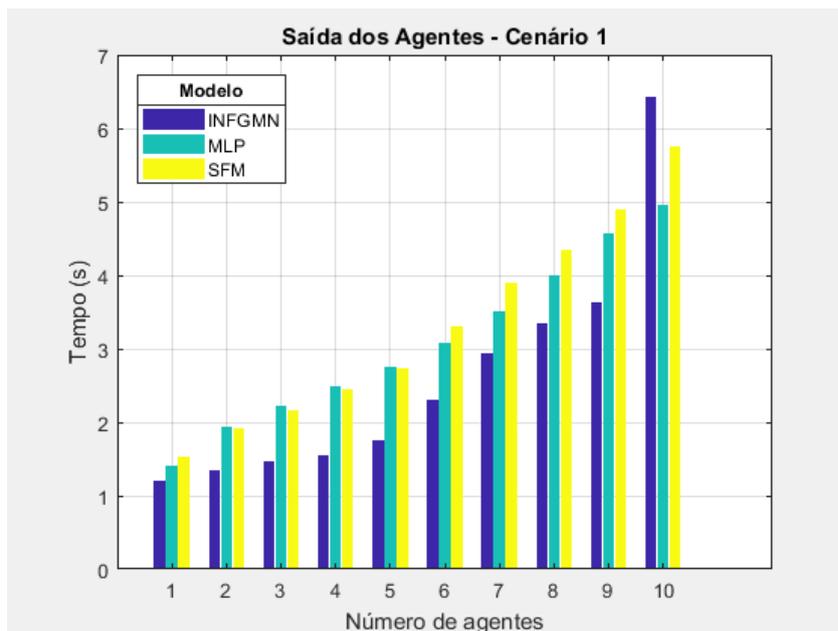


Figura 32 – Tempos de saída para os agentes no cenário 1.

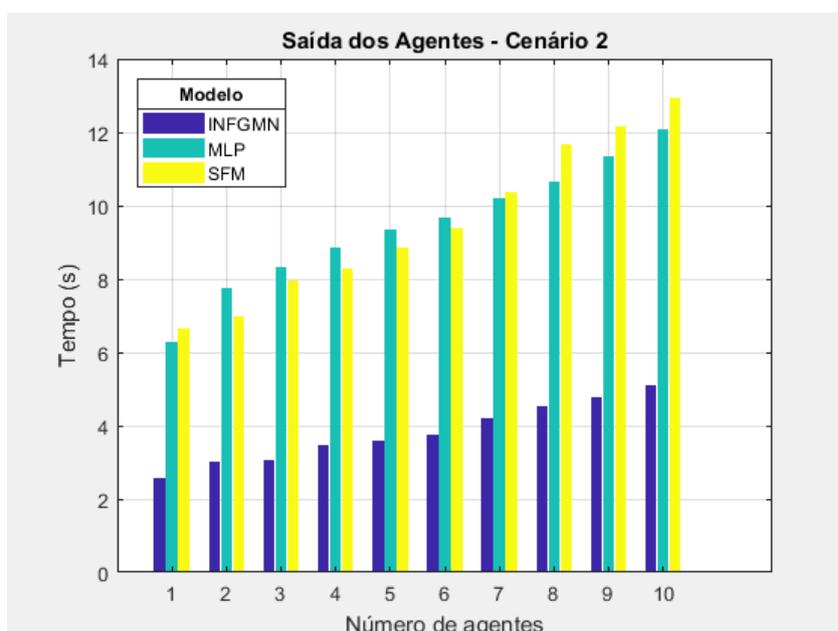


Figura 33 – Tempos de saída para os agentes no cenário 2.

6.3 Considerações finais

Em todas as análises efetuadas é possível concluir que os conjuntos de dados selecionados são suficientemente bons. Uma vez que uma técnica conexionista adequada para resolver problemas de regressão multivariada, como a MLP, consigam uma reprodução fiel da maioria dos fenômenos descritos e também apresentem boa aproximação numérica com o modelo do Helbing sob as restrições já especificadas. Por outro lado, a INFGMN, mesmo conseguindo em sua fase de treinamento uma boa taxa de regressão, nos experimentos práticos não se tem sucesso em reproduzir adequadamente os fenômenos em multidões.

Uma possível hipótese para isso, é que ao lidar com o dilema da precisão-interpretabilidade, o uso de regras Mamdani-Larsen implicam em menor precisão ao simular diferentes evacuações de multidões, pois o método da centroide não oferece valores diferenciados para situações extremas em evacuações, como a repulsão entre pedestres e de pedestres com obstáculos.

Segundo, Diab e Saade (2004), o método da centroide não consegue obter valores *crisp*, que alcancem os limites do conjunto fuzzy de saída agregado. Isso pode acabar influenciando bastante, se o valor da centroide corresponder a um grau de pertinência baixo nesse conjunto agregado. Ross (2009) comenta essa última situação, como a plausibilidade do valor *crisp*, sendo este um critério para medir a qualidade do método de defuzzificação.

A obtenção de um valor não adequado para um eixo de força de um agente possui um efeito cumulativo ao longo da simulação, pois a velocidade desse agente é atualizada com um valor desconhecido ao intervalo dessa variável. Em virtude disso, como ela é uma das entradas do FIS nesse eixo de força, o próximo cálculo produz novamente um valor errado, pois na etapa de fuzzificação, o grau de pertinência dessa nova velocidade é de zero, ou tão próximo quanto em todas as regras. Isso impede a ativação das regras e faz com que o valor *crisp* seja pouco plausível. Por fim, é o motivo pelo qual o agente reproduz movimentos descoordenados durante a simulação.

Além disso, são dezenas de variáveis a serem treinadas, o que requer um volume de dados exponencialmente maior, afim de englobar o máximo de combinações possíveis entre os valores das variáveis. A justificativa é que a INFGMN seria mais eficaz em agrupar os dados em menos conjuntos fuzzy de saída e para que os valores *crisp* obtidos pelo método da centroide reproduzissem adequadamente os comportamentos de pedestres nas inúmeras situações de evacuações. O fato de haver mais de doze mil conjuntos fuzzy (componentes de mistura gaussianos) em ambos FIS, implica que o conjunto de dados não é suficientemente correlacionado para a INFGMN.

Sendo assim, o uso eficiente dessa rede neuro-fuzzy nesses experimentos implicaria em fazer uma outra abordagem na etapa de treinamento. Uma possibilidade para conseguir um melhor agrupamento é tentar especializar o conjunto de treinamento para um cenário específico, restringindo o intervalo de valores que as variáveis possam ter, favorecendo a correlação entre elas. Isso otimizaria a precisão na etapa de defuzzificação e evitaria a criação excessiva de componentes de mistura gaussianos, tornando o desempenho da aplicação mais satisfatório. Entretanto, isso em nada contribuiria para tornar este problema mais interpretável, dado que o número de entradas permaneceria igual.

7 Conclusão

Essa pesquisa buscou analisar a possibilidade de que o modelo de forças de Helbing pudesse ser substituído por redes neuro-fuzzy que gerassem regras do tipo Mamdani-Larsen, como a *Incremental Neuro-Fuzzy Gaussian Mixture Network*. Uma revisão sobre o *Social Force Model* foi efetuada e constatou-se que ao desconsiderar as forças de compressão do corpo e atrito deslizante, seria possível ter um modelo mais simples e com precisão próxima da original. Além disso, as redes neuro-fuzzy foram apresentadas a partir de uma breve introdução sobre lógica fuzzy e redes neurais artificiais, como a MLP. Por fim, apresentou-se a arquitetura da INFGMN, demonstrando o seu potencial com um pequeno exemplo de classificação do conjunto da íris.

Muitos dos trabalhos que buscaram melhorar a interpretabilidade do *Social Force Model* o fizeram através da criação de sistemas de inferência fuzzy, embora ainda mantivessem parte do modelo original de Helbing. Nesta pesquisa, foram dois os sistemas de inferência fuzzy obtidos: um para a determinação de forças nas ordenadas e outro para abcissas. Cada sistema recebeu de entrada doze variáveis fuzzy, que representam sensações dos agentes no ambiente de evacuação, tais como suas velocidades e direções em relação a saída, distâncias e direções dos quatro pedestres mais próximos e do obstáculo mais próximo.

O processo de implementação prática foi realizado seguinte forma: montagem dos conjuntos de dados, através de um simulador que implementa o *Social Force Model*, treinamento com a MLP, validação do modelo através da observações de replicações de fenômenos e da regressão satisfatória em simulações. Por fim, esses mesmos conjuntos de dados são treinados com a INFGMN, cujos parâmetros são calibrados manualmente. O número de regras obtido foi extremamente alto e nada interpretável para um especialista do domínio, totalizando mais de doze mil regras por FIS. Esse número se justifica, pois, o crescimento no número de regras é exponencialmente proporcional ao número de variáveis fuzzy presentes na modelagem do sistema.

Ao fim, duas análises foram realizadas, comparando os desempenhos da MLP, INFGMN com o *Social Force Model*, onde foi possível concluir que o conjunto de dados obtido é realmente bom, pois a MLP conseguiu uma excelente regressão numérica com o SFM e também uma reprodução de fenômenos muito próxima da real. Entretanto, para os sistemas de inferência fuzzy gerados pela INFGMN, mesmo com dezenas de milhares de regras, não foi possível reproduzir os fenômenos em multidões e as forças obtidas tiveram valores muito distantes das obtidas pelo SFM.

Assim, essa pesquisa sugere a necessidade de uma outra abordagem na montagem

desses conjuntos de treinamento multivariados para INFGMN. Por exemplo, levar em conta apenas cenários de evacuações específicos, de tal forma que o intervalo de valores dessas variáveis seja menor. Isso favoreceria o agrupamento desses dados de entrada em termos linguísticos de saída, o que reduziria o número de conjuntos fuzzy gerados, por haver menos componentes de mistura gaussianas, ao passo que o método da centroide seria mais preciso em resoluções de problemas como este.

7.1 Trabalhos Futuros

Para trabalhos futuros sugere-se ignorar as sensações locais dos agentes e adotar uma abordagem que foque mais na interpretabilidade dos componentes do *Social Force Model* do que em sua unidade. Como exemplo, quebrar o SFM nos três componentes de força, gerando uma INFGMN para cada um, por eixo de força e comparar com outra rede neuro-fuzzy, como a ANFIS e com o modelo de Sent (2016). Um outro caminho totalmente diferente seria rever o modelo de Helbing de forma a pesquisar a possibilidade de estados anteriores dos agentes ter influência em seus próximos movimentos, como por exemplo sua força anterior. Para isso, poderia-se utilizar redes neurais recorrentes e, se houver uma redução significativa de variáveis, então estudar a possibilidade de uma nova pesquisa com a INFGMN.

Referências

- BERNARDINI, G.; D'ORAZIO, M.; QUAGLIARINI, E.; SPALAZZI, L. An agent-based model for earthquake pedestrians' evacuation simulation in urban scenarios. In: *Transportation Research Procedia*. [S.l.: s.n.], 2014. v. 2, p. 255–263.
- CONY, C. A.; BICHO, A. de L.; JUNG, C. R.; MAGALHÃES, L. P.; MUSSE, S. R. A perceptive model for virtual agents in crowds. p. 141–150, jun. 2007.
- DHEERU, D.; TANISKIDOU, E. K. *UCI Machine Learning Repository*. 2017. Disponível em: <http://archive.ics.uci.edu/ml>.
- DIAB, H.; SAADE, J. J. Defuzzification methods and new techniques for fuzzy controllers. In: *Iranian Journal of Electrical and Computer Engineering*. [S.l.: s.n.], 2004. v. 3, p. 161–174.
- FISHER, R. A. The use of multiple measurements in taxonomic problems. *Annals Eugen.*, v. 7, p. 179–188, 1936.
- GAN, M.-T.; HANMANDLU, M.; TAN, A. H. From a gaussian mixture model to additive fuzzy systems. *IEEE Transactions on Fuzzy Systems*, v. 13, n. 3, p. 303–316, June 2005. ISSN 1063-6706.
- HELBING, D.; FARKAS, I.; VICSEK, T. Simulating dynamical features of escape panic. *Nature*, v. 407, p. 487–490, 2000.
- HELBING, D.; JOHANSSON, A. Pedestrian, crowd and evacuation dynamics. In: *Encyclopedia of Complexity and Systems Science*. [S.l.: s.n.], 2010. v. 16, p. 697–716.
- HELBING, D.; MOLNÁR, P. Social force model for pedestrian dynamics. *Phys. Rev. E*, American Physical Society, v. 51, p. 4282–4286, 1995.
- HENDERSON, L. On the fluid mechanics of human crowd motion. In: *Transportation Res.* [S.l.: s.n.], 1974. v. 8, p. 509–515.
- HENDERSON, L. F. The Statistics of Crowd Fluids. *Nature*, Nature Publishing Group, v. 229, n. 5284, p. 381–383, fev. 1971. ISSN 0028-0836.
- JANG, J. S. R. Anfis: adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, v. 23, n. 3, p. 665–685, May 1993. ISSN 0018-9472.
- KASABOV, N. Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, v. 31, n. 6, p. 902–918, Dec 2001. ISSN 1083-4419.
- KASABOV, N. *Evolving Connectionist Systems: The Knowledge Engineering Approach*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006. ISBN 1846283450.
- KASABOV, N. K.; SONG, Q. Denfis: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *IEEE Transactions on Fuzzy Systems*, v. 10, n. 2, p. 144–154, Apr 2002. ISSN 1063-6706.

- LI, H.; PHILIP, C. C. L.; HUANG, H.-P. *Fuzzy Neural Intelligent Systems: Mathematical Foundation and the Applications in Engineering*. 1st. ed. Boca Raton, FL, USA: CRC Press, Inc., 2000. ISBN 0849323606.
- LIN, C.-T.; LEE, C. S. G. *Neural Fuzzy Systems: A Neuro-fuzzy Synergism to Intelligent Systems*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1996. ISBN 0-13-235169-2.
- MAZZUTTI, T.; ROISENBERG, M.; FILHO, P. José de F. Infgmn – incremental neuro-fuzzy gaussian mixture network. In: *Expert Systems with Applications*. Tarrytown, NY, USA: Pergamon Press, Inc., 2017. v. 89, p. 160–178.
- NASSER, T. Z. *Modelo para Simulação de Multidões Utilizando Lógica Nebulosa*. Monografia (Graduação) — Universidade Federal de Santa Catarina, Florianópolis, SC, 2011.
- RODRIGUES, D. G.; MOURA, G.; JACINTO, C. M. C.; FILHO, P. J. de F.; ROISENBERG, M. Generating interpretable mamdani-type fuzzy rules using a neuro-fuzzy system based on radial basis functions. In: *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. [S.l.: s.n.], 2014. p. 1352–1359. ISSN 1098-7584.
- ROSS, T. *Fuzzy Logic with Engineering Applications*. Wiley, 2009. ISBN 9780470748510. Disponível em: https://books.google.com.br/books?id=nhz1f9j6_SMC.
- SCHMIDT, J.; SPÄH, A. PANIC SIMULATOR. 2014. Disponível em: <https://github.com/julianrschmidt/panicSimulator>.
- SENT, A. *Proposta de Modelo Fuzzy do Modelo de Forças Sociais para Simulação de Pedestres em Situação de Evacuação*. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, Florianópolis, SC, 2016.
- SENT, A. D. Fuzzy Social Force Model. 2014. Disponível em: <https://github.com/altieresdelsent/FSFM>.
- SENT, A. D.; ROISENBERG, M.; FILHO, P. J. de F. Simulation of crowd behavior using fuzzy social force model. In: *2015 Winter Simulation Conference (WSC)*. [S.l.: s.n.], 2015. p. 3901–3912.
- SHARMA, S.; SINGH, H. Multi-agent system for simulating human behavior in egress simulations. In: *Proceedings of the NAACSOS, Annual Conference of the North American Association for Computational Social and Organizational Sciences*. [S.l.: s.n.], 2006.
- SHARMA, S. C. An object oriented approach to simulating agent-based behavior. In: . [s.n.], 2004. Disponível em: <http://www.cs.bowiestate.edu/sharad/sim/index.html>.
- XUE, Z.; DONG, Q.; FAN, X.; JIN, Q.; JIAN, H.; LIU, J. Fuzzy logic-based model that incorporates personality traits for heterogeneous pedestrians. *Symmetry*, v. 9, n. 10, 2017. ISSN 2073-8994. Disponível em: <http://www.mdpi.com/2073-8994/9/10/239>.
- YU, D.; DENG, L. Automatic speech recognition: A deep learning approach. In: _____. [S.l.]: Springer Publishing Company, Incorporated, 2014. cap. Chapter 2: Gaussian Mixture Models, p. 13–21. ISBN 978-1-4471-5779-3.

ZADEH, L. Fuzzy sets. *Information and Control*, v. 8, n. 3, p. 338 – 353, 1965.
ISSN 0019-9958. Disponível em: <http://www.sciencedirect.com/science/article/pii/S001999586590241X>.

APÊNDICE A – Artigo

Pesquisa em simulação de evacuação de multidões utilizando *Incremental Neuro-Fuzzy Gaussian Mixture Network*

Nathan Frois Pereira Paiva¹

¹CTC – INE – Universidade Federal de Santa Catarina (UFSC)
Florianópolis – SC – Brasil

nathanfrois@gmail.com

Abstract. *Modeling and simulating human behavior in crowds is essential to understanding what happens to individuals in panic situations. Helbing has developed a model called textit Social Force Model which represents these behaviors through differential equations. In this model, each individual is a particle with a mass inserted in an environment, where there are forces of attraction and repulsion between particles and entities of that area. Thus, this research aims to use simulated evacuation behaviors through the Social Force Model, which serve as the basis for the training of neuro-fuzzy systems, so that they can be interpreted as fuzzy sets and rules of an inference system fuzzy on these sets.*

Resumo. *Modelar e simular o comportamento humano quando em multidões é essencial para compreender o que ocorre com os indivíduos em situações de pânico. Helbing elaborou um modelo chamado Social Force Model que representa esses comportamentos através de equações diferenciais. Neste modelo, cada indivíduo é uma partícula com uma massa inserida em um ambiente, onde existem forças de atração e repulsão entre partículas e entidades dessa área. Assim sendo, essa pesquisa visa utilizar comportamentos de evacuação simulados através do Social Force Model, os quais servem de base para o treinamento de redes neuro-fuzzy, para que possam ser interpretados como conjuntos fuzzy e regras de um sistema de inferência fuzzy sobre estes conjuntos.*

1. Introdução

Modelar e simular o comportamento humano quando em multidões é essencial para compreender o que ocorre com os indivíduos em situações de pânico. Isto é fundamental para que um arquiteto possa, por exemplo, projetar um espaço fechado da maneira mais segura possível, evitando congestionamento em evacuações, promovendo uma circulação mais confortável e o rápido deslocamento dos pedestres até a saída do local.

Atualmente existem duas abordagens aceitas para modelar e simular o comportamento humano: a macroscópica e a microscópica [Helbing and Molnár, 1995]. Embora computacionalmente mais complexa, a segunda é uma modelagem que oferece maior precisão. Dentro do modelo microscópico existe o *Social Force Model* o qual esse trabalho se baseia e de acordo com, Sent and Roisenberg et al. [2015], é o que melhor equilibra confiabilidade e desempenho.

O princípio da proposta de Helbing é representar o comportamento dessas pessoas numa evacuação de forma análoga à propagação de partículas [Helbing and Molnár,

1995]. Ou seja, cada indivíduo é uma partícula com uma massa, inserido em um ambiente, onde existe um tipo de força que atrai a partícula para o ponto de saída, outra que repele o pedestre de obstáculos próximos e também outra que repele pessoas próximas e tudo isso é explicado por equações matemáticas de difícil compreensão.

Assim sendo, essa pesquisa visa utilizar comportamentos de evacuação simulados através do *Social Force Model*, os quais servem de base para o treinamento de redes neuro-fuzzy. Depois de treinados, estes comportamentos podem ser interpretados como conjuntos fuzzy e regras de um sistema de inferência fuzzy sobre estes conjuntos.

Durante o artigo são apresentadas fundamentações teóricas sobre *Social Force Model* e a rede neuro-fuzzy utilizada para o fim dessa pesquisa. A quarta seção discute como a pesquisa é desenvolvida em um simulador, que implementa o *Social Force Model*. Na quinta seção, são apresentados os experimentos com os conjuntos de dados obtidos. Comparando o desempenho entre o modelo original, a INFGMN e a *multi-layer perceptron* (MLP). A última seção conclui o trabalho, sugerindo pesquisas futuras.

2. Social Force Model

Considere um pedestre α inserido em um ambiente qualquer. Segundo, Helbing e Johansson [2010] e Helbing et al. [2000], podemos descrever a força de aceleração deste pedestre α a partir da seguinte equação:

$$\vec{f}_\alpha(t) = \frac{m_\alpha}{\tau_\alpha} (v_\alpha^0 \vec{e}_\alpha^0 - \vec{v}_\alpha) + \sum_{\alpha \neq \beta} \vec{f}_{\alpha\beta}(t) + \sum_i \vec{f}_{\alpha i}(t) \quad (1)$$

Em que $\frac{m_\alpha}{\tau_\alpha} (v_\alpha^0 \vec{e}_\alpha^0 - \vec{v}_\alpha)$ denota o cálculo força de desejo para α . O segundo termo desta equação, denota soma de forças sociais no instante de tempo t , entre α e cada outro indivíduo β presente na simulação. O terceiro termo da equação, determina também uma soma forças de obstáculos para o mesmo instante t , entre α e cada obstáculo i .

2.1. Força de Desejo

No primeiro termo da Equação (1), temos que cada indivíduo α se move com uma velocidade desejada v_α^0 , para uma direção desejada \vec{e}_α^0 , ao qual temos um certo tempo de aceleração τ_α , com $v_\alpha^0 \vec{e}_\alpha^0$ tendendo a v_α^0 e sob esse valor subtrai-se a velocidade atual \vec{v}_α . Essa direção desejada toma como base o local que o pedestre α tem como objetivo final denominado por \vec{r}_α^0 , a qual se deseja alcançar da maneira mais confortável e rápida possível.

2.2. Força Social

Seja o componente da soma $\vec{f}_{\alpha\beta}(t)$ da Equação (1), ele é um somatório da força de repulsão entre α e cada outro indivíduo β entre uma lista de indivíduos [Helbing and Johansson, 2010]. A repulsão entre os pedestres é causada pela alta proximidade física entre eles. Abaixo, a equação de Força Social é demonstrada:

$$\vec{f}_{\alpha\beta}(t) = \{A \exp^{(r_{\alpha\beta} - d_{\alpha\beta})/B} + k\Theta(r_{\alpha\beta} - d_{\alpha\beta})\} \times \vec{n}_{\alpha\beta} + \kappa\Theta(r_{\alpha\beta} - d_{\alpha\beta}) \Delta v_{\beta\alpha}^t \vec{t}_{\alpha\beta} \quad (2)$$

Onde $d_{\alpha\beta} = \|\vec{r}_\alpha - \vec{r}_\beta\|$, ou seja a distância vetorial entre a posição do pedestre α e o pedestre β . Assim sendo $A \exp^{(r_{\alpha\beta} - d_{\alpha\beta})/B} \times \vec{n}_{\alpha\beta}$ o alcance da força repulsiva, $k\Theta(r_{\alpha\beta} -$

$d_{\alpha\beta}) \times \vec{n}_{\alpha\beta}$ a força do corpo que contrai a compressão do corpo, e $k\Theta(r_{\alpha\beta} - d_{\alpha\beta})\Delta v_{\beta\alpha}^t \vec{t}_{\alpha\beta}$ uma "força de atrito deslizante" impedindo o movimento tangencial relativo. A função Θ retorna o próprio parâmetro, $\Theta(r_{\alpha\beta} - d_{\alpha\beta}) = r_{\alpha\beta} - d_{\alpha\beta}$, se o resultado for ≥ 0 , caso contrário retorna zero.

2.3. Força de Obstáculo

De acordo com Helbing e Johansson [2010], se trocarmos β por i obteríamos a mesma equação da Força Social, para a Força de Obstáculo. A diferença é que como os obstáculos estão em repouso, ou seja, $\vec{v}_i = 0$, então apenas a velocidade do agente α é considerada.

Esse modelo de força descrito nessa seção é o que o simulador utiliza, entretanto as forças de atrito deslizante e a força de compressão para pedestres e obstáculos, serão desconsideradas nesta pesquisa. A razão principal é que a matemática é extremamente complexa, necessitando-se de muitas variáveis para que uma técnica conexionista reproduzisse estes eventos com uma precisão minimamente aceitável, além de tornar o modelo pouquíssimo interpretável. Ainda assim é possível reproduzir as forças de repulsão.

2.4. Fenômenos em multidões e evacuações

O *Social Force Model* é um modelo capaz de reproduzir fielmente fenômenos que ocorrem em multidões na vida real. Assim sendo, A "Formação de Faixas" é um exemplo de padrão de movimento contendo filas de indivíduos em uma direção uniforme por onde o fluxo passa. Já, a "Oscilação de Fluxos em Gargalos" (*Oscillatory Flows at Bottlenecks*), é quando há um cruzamento de fluxos de pedestres no mesmo sentido e sobre uma passagem muito estreita, formando um gargalo. Ocorre-se então uma alternância de passagem entre os fluxos, onde um fluxo que exercer uma pressão maior sobre o outro, ocupa a passagem momentaneamente.

Uma evacuação de multidões ocorre quando tem-se um ambiente com uma extrema densidade de pedestres e eles estão sob estresse psicológico devido a necessidade de sair do local o mais rápido possível. Nesse contexto é comum ocorrer a destruição das faixas, com os pedestres ficando parados temporariamente. Esse fenômeno tem o nome de "Congelamento por Aquecimento" (*Freezing by Heat*) [Helbing and Johansson, 2010]. Também, quando o fluxo de entrada em um gargalo é maior que o de saída, os indivíduos ficam passivos a maiores interações físicas e de efeitos de atrito, causando lentidão na evacuação. O fenômeno para isso é chamado de "Quanto Mais Rápido, Mais Devagar" (*Faster-Is-Slower Effect*) [Helbing and Johansson, 2010].

3. Redes Neuro-Fuzzy

As redes neuro-fuzzy podem ser compreendidas como uma combinação das características de sistemas fuzzy e de redes neurais. A integração desses sistemas em um só, permite com que as redes neurais provenham a capacidade de aprendizagem para sistemas fuzzy e estes permitam com que as redes neurais possam ser entendidas como regras IF-THEN de alto nível. Assim, Kasabov [2006], estabelece que os sistemas de inferência neuro-fuzzy consistem em um conjunto de regras e um método de inferência que são incorporados ou combinados com uma estrutura conexionista para uma melhor adaptação.

3.1. INFGMN (*Incremental Neuro-Fuzzy Gaussian Mixture Network*)

A INFGMN (*Incremental Neuro-Fuzzy Gaussian Mixture Network*) é um sistema neuro-fuzzy proposto por Mazzutti et al. [2017], e têm como característica: o aprendizado incremental permitindo à rede aprender constantemente, sem as necessidades de retreinar todo o conjunto de dados e de gerar um novo conjunto de regras, cada vez que um novo padrão é apresentado a rede. A INFGMN também lida com o dilema da acurácia-interpretabilidade, equilibrando o número de regras obtido com a precisão do sistema. Ainda em relação ao conjunto de regras Mamdani-Larsen formado na INFGMN, elas são obtidas a partir de um modelo de misturas gaussianas.

3.1.1. GMM (*Gaussian Mixture Model*)

Mazzutti et al. [2017], mencionam que a função de densidade de probabilidade de misturas gaussianas (pdf - *probability density function*) para uma variável aleatória ou para o vetor de entrada da INFGMN, \mathcal{X} , de dimensão D , é definida por:

$$G(\mathcal{X}; p, u, C) = \sum_{j=1}^J p(j) \mathcal{N}^D(\mathcal{X}; \mu_j, C_j) \quad (3)$$

Onde $p(j)$, $\forall j \in [1, J]$, é a j -ésima proporção de mistura independente entre J componentes, ou probabilidades *a priori*. Em que a soma dessas proporções equivale sempre a 1. Nesta mesma equação acima, C é uma matriz de covariância. Por último, $\mathcal{N}^D(\mathcal{X}; \mu_j, C_j)$, é a densidade multivariada normal para a j -ésima componente de mistura.

3.1.2. IGMM (*Incremental Gaussian Mixture Model*)

O IGMM (*Incremental Gaussian Mixture Model*) é a versão incremental do GMM. Assim IGMM utiliza as mesmas equações apresentadas anteriormente, só que com alguns parâmetros adicionais. Com isso $p(\mathcal{X}|j)$, é a probabilidade de que o vetor de entrada, \mathcal{X} , pertença ao j -ésimo componente de mistura. Ela é calculada pela densidade multivariada normal e quando esse número for inferior a uma fração, que envolve um valor configurável, chamado de limiar de verossimilhança, τ_{nov} . Então, diz-se, que \mathcal{X} é rejeitado por esse j -ésimo componente de mistura.

Quando \mathcal{X} é rejeitado para todos os J componentes, então cria-se novo componente, j^* , onde inicialmente o vetor de médias, $u_{j^*} = \mathcal{X}$, e a matriz de covariância, C_{j^*} , é equivalente a matriz identidade multiplicada por uma constante personalizável δ , da qual sugere-se um valor baixo, como $\frac{1}{100}$. Essa constante é também multiplicada junto é variância total de cada d -ésima dimensão do vetor de entrada.

Segundo, Mazzutti et al. [2017], para a atualização das distribuições do IGMM, são definidas as seguintes equações:

$$p(j|\mathcal{X}) = \frac{p(\mathcal{X}|j)p(j)}{\sum_{j=1}^J p(\mathcal{X}|j)p(j)} \quad (4)$$

$$sp_j^* = sp_j + p(j|\mathcal{X}), \forall j \in J \quad (5)$$

$$w_j = \frac{sp_j^*}{p(j|\mathcal{X})}, \forall j \in J \quad (6)$$

$$\mu_j^{d*} = \mu_j + w_j(\mathcal{X} - \mu_j^d), \forall j \in J, \forall d \in D \quad (7)$$

$$C_j^{d*} = C_j^d - (\mu_j^{d*} - \mu_j^d)(\mu_j^{d*} - \mu_j^d)' + w_j [(\mathcal{X} - \mu_j^{d*})(\mathcal{X} - \mu_j^{d*})'], \forall j \in J, \forall d \in D \quad (8)$$

$$p(j)^* = \frac{sp_j^*}{\sum_{q=1}^J sp_q}, \forall j \in J \quad (9)$$

Onde, sp_j e sp_j^* , são respectivamente a probabilidade *a posteriori* acumulada do j -ésimo componente e o seu novo valor atualizado. p^* , μ^* , C^* são os outros parâmetros atualizados.

Por último, um dado j -ésimo componente é removido do modelo, sempre que essas duas condições sejam verdades: $t_j > t_{max}$ e $sp_j < sp_{min}$. Isto é, o tempo de vida acumulado do j -ésimo componente não deve ser maior que um tempo máximo estipulado pelo usuário, para mostrar que esse componente é útil ao modelo, sendo isso verificado pela segunda condição, em que sp_{min} é também uma constante personalizável.

3.1.3. Arquitetura da INFGMN

A arquitetura da INFGMN é dividida em cinco camadas, conforme ilustra a Figura 1 e sobre ela observam-se as seguintes notações: \mathcal{IV} , \mathcal{OV} , \mathcal{G} , \mathcal{W} , \mathcal{R} , K e M , que são respectivamente as variáveis fuzzy de entrada (antecedentes das regras fuzzy), as variáveis fuzzy de saída (consequentes das regras fuzzy), o conjunto de J componentes da mistura gaussiana, o conjunto dos pesos das regras, o conjunto de regras, o número de regras e o número de dimensões do vetor de saída.

A INFGMN têm dois modos de operação, o primeiro deles, *learning*, é ativado nas primeiras quatro camadas, seguindo essa ordem, e tem como objetivo gerar e atualizar um (*Fuzzy Inference System* - FIS) do tipo Mamdani-Larsen, a partir dos padrões apresentados na primeira camada. Já o modo *recalling* é pré-condicionado a uma rede já estar parcialmente treinada e é ativado para as três últimas camadas. Em sua ocorrência, tem-se a apresentação de um ou mais padrões sem saída(s) conhecida(s) à rede, para que ela seja responsável por inferir as regras e defuzzificar a saída.

A primeira camada (*Input Layer*) é composta essencialmente pelo vetor de dados da entrada, \mathcal{X} , de dimensão D , em que \mathcal{X}_d é o valor da d -ésima dimensão desse vetor associado ao d -ésimo nodo, entre os D nodos distribuídos nessa camada. Finalmente, todos nodos dessa camada são transmitidos de forma intacta para todos os nós da segunda camada.

A segunda camada (*IGMM layer*) trabalha sobre o IGMM, após o vetor \mathcal{X} ser apresentado a essa camada. Suas funções são: criar e ajustar os parâmetros de cada distribuição gaussiana, de acordo com as equações (4), (5), (6), (7), (8) e (9). Após isso, esses valores são computados pela função, $\mathcal{G}_j(\mathcal{X}; p^*, \mu^*, C^*), \forall j \in [1, J]$, a qual produz como saída o j -ésimo componente de mistura gaussiana atualizado.

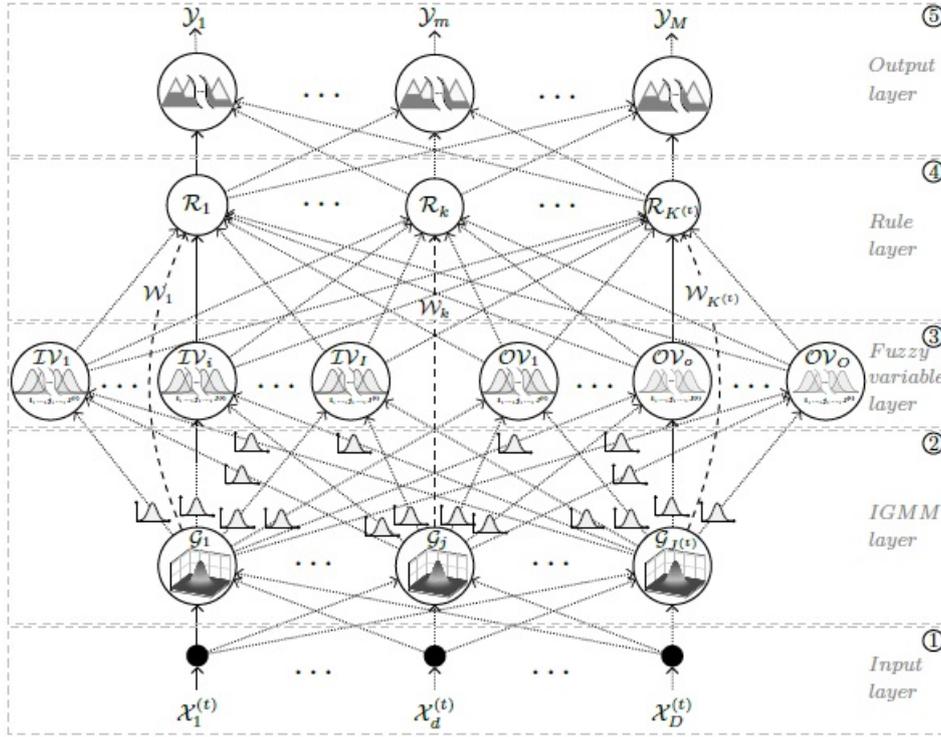


Figura 1. Arquitetura da INFGMN (Mazutti et al., 2017).

A terceira camada (*Fuzzy variable layer*), no modo de operação *learning* compreende a criação e/ou atualização dos antecedentes e consequentes de cada regra fuzzy do tipo Mamdani-Larsen. Esta camada é composta por nós que correspondem às variáveis linguísticas, onde cada variável é associada à alguma d -ésima dimensão de \mathcal{X}_d , $\forall d \in [1, D]$, logo a camada tem D nós. Além disso todas as variáveis possuem a mesma quantidade termos linguísticos, ou seja uma quantidade conjuntos fuzzy gaussianos equivalente ao número de componentes gaussianos. Já o j -ésimo conjunto fuzzy de cada variável é composto das matrizes de covariância e média referente ao j -ésimo componente dessa variável.

Por último, Mazzutti et al. (2017), alega que no modo de operação *recalling*, essa camada tem a função de fuzzificar o vetor de entrada, \mathcal{X} . Ou seja, para \mathcal{X}_i , será executada i -ésima função de pertinência da variável linguística de entrada, \mathcal{IV}_i , definida por: $\phi^{(\mathcal{IV}_i)} = N^1(\mathcal{X}_i, u_{i(j)}, C_{i(j)})$. Já para o cálculo da o -ésima variável linguística de saída, \mathcal{OV}_o , a função de pertinência é fornecida por: $\phi^{(\mathcal{OV}_o)} = \int_{y_o} N^1(\dagger_o, u_{o(j)}, C_{o(j)}) dy_o$.

Na quarta camada (*Rule layer*), para o modo de operação *learning*, as regras fuzzy do tipo Mamdani-Larsen são criadas e/ou atualizadas, por meio das variáveis linguísticas de entrada e saída geradas na terceira camada. Onde o número de regras geradas é igual ao número de componentes de mistura, e portanto, $K = J$. Além disso, cada regra é composta por, I termos antecedentes, os quais são sempre ligados pelo operador *AND*, e O termos consequentes, portanto $D = I + O$. Também associada a cada regra, está \mathcal{W}_k , o peso da k -ésima regra, sendo ele equivalente a j -ésima probabilidade *a priori*.

No modo de operação *recalling*, ocorre a inferência das regras, a partir da força de disparo de cada k -ésima regra, obtida pelo resultado de um produtório entre todos

termos antecedentes dessa regra, multiplicado pelo peso dessa k -ésima regra, \mathcal{W}_k . Obtida as forças de disparo, então uma operação de implicação multiplicativa é incidida sobre os conjuntos fuzzy de saída de todas as regras. O próximo passo é obter um conjunto fuzzy agregado, OV^{*m} , para cada m -ésima variável de saída, $\forall m \in [1, M]$, através de uma agregação aditiva sobre os conjuntos dessa variável e sua função de pertinência agregada, ϕ^{*m} .

A quinta camada (*Output layer*) é ativada apenas no modo *recalling* e tem como função obter um valor *crisp* para cada m -ésima variável de saída fuzzy, a partir da defuzzificação pelo método da centroide, o qual computa para cada m -ésimo conjunto fuzzy agregado, a sua função de pertinência, ϕ^{*m} .

4. Implementação

A implementação original do *Social Force Model* e da substituição da determinação de suas forças por redes neuro-fuzzy do tipo INFGMN para fins de experimento ocorre sobre um software em MATLAB chamado PANIC SIMULATOR (Schmidt and Späh, 2014). O simulador trabalha com agentes inseridos em um ambiente 2D e o objetivo em cada simulação é fazer que com todos agentes consigam sair da arena. A Figura 2 identifica quem são os agentes, paredes, colunas e a saída.

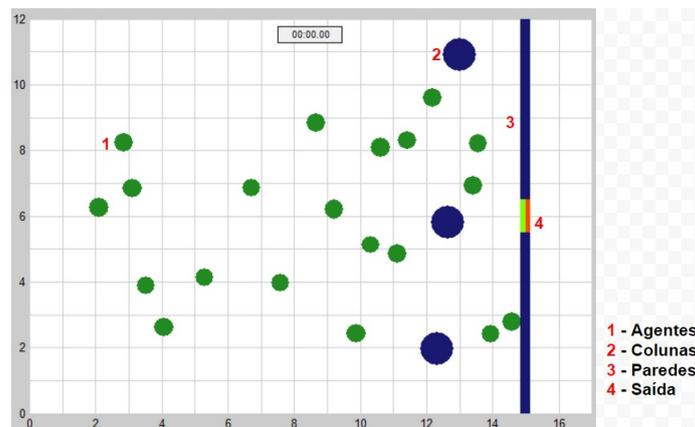


Figura 2. Componentes físicos de uma simulação, onde sua numeração identifica o tipo do componente.

Uma vez compreendida a relação entre as variáveis do SFM, os dados são capturados e treinados com redes neurais *multi-layer feedforward* com algoritmo *backpropagation*. Simulações são feitas e conforme deficiências vão sendo encontradas, busca-se capturar estas situações com o simulador original. Então, a rede é treinada e simulada novamente. Esse processo é feito exaustivamente até conseguir reproduzir com sucesso os experimentos propostos da próxima seção. Posteriormente com o mesmos conjuntos de dados obtidos para treinar a MLP, treina-se com a INFGMN até conseguir uma precisão de regressão razoável e buscando obter o menor número de regras possível.

4.1. Modelo proposto

A proposta de implementação prática deste trabalho, consiste em substituir os componentes da Equação (1), ignorando a computação das forças de compressão do corpo

e atrito deslizante, por duas INFGMN, em que cada uma determina a força exercida sobre um agente em um dado eixo cartesiano. A INFGMN calcula essa força a partir das sensações locais do agente, como distâncias e direções de seus quatro agentes mais próximos e o mesmo para o obstáculo mais próximo. O motivo para isso é que através de observações das simulações originais, foi constatado que são as entidades da simulação que possuem maior influência no cálculo da força total de um agente.

Abaixo as equações especificam como a força é determinada para um agente α , por meio da INFGMN:

$$\vec{f}_{\alpha_x} = \text{fuzzyForça}_{\alpha_x}(\vec{e}_{\alpha_x}^0, \vec{v}_{\alpha_x}, \Delta\vec{d}_{\alpha\beta 1}, \Delta\vec{d}_{\alpha\beta 2}, \Delta\vec{d}_{\alpha\beta 3}, \Delta\vec{d}_{\alpha\beta 4}, \vec{n}_{\alpha\beta 1_x}, \vec{n}_{\alpha\beta 2_x}, \vec{n}_{\alpha\beta 3_x}, \vec{n}_{\alpha\beta 4_x}, \Delta\vec{d}_{\alpha i}, \vec{n}_{\alpha i_x}) \quad (10)$$

$$\vec{f}_{\alpha_y} = \text{fuzzyForça}_{\alpha_y}(\vec{e}_{\alpha_y}^0, \vec{v}_{\alpha_y}, \Delta\vec{d}_{\alpha\beta 1}, \Delta\vec{d}_{\alpha\beta 2}, \Delta\vec{d}_{\alpha\beta 3}, \Delta\vec{d}_{\alpha\beta 4}, \vec{n}_{\alpha\beta 1_y}, \vec{n}_{\alpha\beta 2_y}, \vec{n}_{\alpha\beta 3_y}, \vec{n}_{\alpha\beta 4_y}, \Delta\vec{d}_{\alpha i}, \vec{n}_{\alpha i_y}) \quad (11)$$

A primeira equação determina a força exercida no eixo- x e a segunda no eixo- y . Já, \vec{v}_{α_x} e \vec{v}_{α_y} são respectivamente a velocidade atual de α no eixo- x e no eixo- y . $\vec{e}_{\alpha_x}^0$ e $\vec{e}_{\alpha_y}^0$ são as direções desejadas normalizadas para o eixo- x e eixo- y .

De volta, nas Equações (10) e (11), $\Delta d_{\alpha\beta}$ e $\Delta d_{\alpha i}$ é um valor de distância dada pelas respectivas somas: $r_\alpha + r_\beta - d_{\alpha\beta}$ e $r_\alpha + r_i - d_{\alpha i}$. Quanto mais negativo for esse valor, mais distante os corpos estão um do outro e positivo, quando as entidades se colidem. Já $\vec{n}_{\alpha\beta_x}$, $\vec{n}_{\alpha\beta_y}$, $\vec{n}_{\alpha i_x}$ e $\vec{n}_{\alpha i_y}$ são as direções normalizadas dos agentes com as entidades em um dado eixo.

4.1.1. Captura de dados

A partir de simulações com o *Social Force Model*, são montados dois conjuntos de dados, um relativo ao eixo- x e outro relativo ao eixo- y . Para o primeiro e segundo conjunto são coletados como entrada, os parâmetros das equações (10) e (11). Já a saída para ambos os conjuntos é a força obtida nessas equações.

As simulações para essa coleta se decorriam num total de 10 a 20 agentes, estando todos espalhados aleatoriamente em um cenário de doze metros de largura, dezessete metros de comprimento, o qual é o padrão do software. Também, colunas são adicionadas para aumentar a precisão na detecção de colisões. Para os demais parâmetros das equações do *Social Force Model*, são mantidas as configurações padrão do simulador, como a Tabela 1 mostra.

4.1.2. Treinamento na MLP

Para avaliar os conjuntos de dados, primeiramente é utilizada uma MLP para cada eixo de força, da mesma forma que nas Equações (10) e (11). Em ambas as redes neurais, os dados são pré-processados, conforme Li et al. (2000), recomendam. Neste caso, como é conhecida a natureza e relacionamento entre as variáveis, por meio do estudo das equações do SFM, e como os dados são precisamente capturados por um software simulador, sem risco de inconsistência numérica para as variáveis capturadas, basta, apenas, efetuar uma normalização para cada variável. Como a discrepância dos valores é muito alta, normaliza-se esses atributos entre -1 e 1, aplicando a técnica min-max.

Tabela 1. Parâmetros do PANIC SIMULATOR

Nome do Parâmetro	Valor
Largura da arena	12 m
Comprimento da arena	17 m
Velocidade Desejada	4.0 m/s
A	2000
B	0.08 m
τ	0.5 s
Raio dos Pedestres	[0.25 m, 0.3 m]
Raio das Colunas	0.5 m
Massa	[70, 80] Kg

A MLP utilizada consta de 30 neurônios na única camada intermediária e a função de treinamento é a de gradiente descendente, com regularização bayesiana. São treinados 85% de cada conjunto de treinamento, com 15% restantes para testes. Ao fim do treinamento obteve-se uma regressão de mais de 0.999 em ambos os conjuntos de dados.

4.1.3. Treinamento na INFGMN

O treinamento da INFGMN ocorreu com os mesmos conjuntos de dados obtidos para treinar a MLP. O critério de pré-processamento do conjunto, foi o mesmo utilizado na MLP. Também, cada conjunto de dados foi manualmente embaralhado, visto que a biblioteca da mesma não realiza isso e a rede é sensível a ordem de entrada dos dados. Ao todo, 100% do conjunto de dados para cada eixo de força, foram treinados. Os parâmetros utilizados foram: $\Delta = 0.001$, $t_{max} = 10000$, $sp_{min} = 1$ e $\tau_{nov} = 0.22$. Eles foram manualmente calibrados até conseguir obter uma boa relação de regressão, um total de 0.97 por conjunto de dados. Entretanto apesar da precisão boa na regressão, que é proporcional ao valor adotado em t_{max} , obteve-se um resultado muito desapontador em relação a alta quantidade regras geradas e que são discutidas a seguir.

4.1.4. Base de regras obtida

Para o FIS da força no eixo- x , são obtidas 13839 regras e para o eixo- y , 12963 regras. Isso não torna a resolução do problema interpretável, pois isso implicada que cada variável fuzzy de cada FIS tem 13839 e 12963 conjuntos fuzzy, para respectivamente eixo- x e eixo- y . Abaixo são mostrados os formatos das regras criadas para uma j -ésima regra de cada eixo:

\mathcal{R}_j^x : IF e_x^0 is $\phi_j \wedge v_x$ is $\phi_j \wedge \Delta d_{\alpha\beta 1}$ is $\phi_j \dots \wedge \Delta d_{\alpha\beta 4}$ is $\phi_j \wedge \vec{n}_{\alpha\beta 1_x}$ is $\phi_j \wedge \dots \wedge \vec{n}_{\alpha\beta 4_x}$ is $\phi_j \wedge \Delta d_{\alpha i}$ is $\phi_j \wedge \vec{n}_{\alpha i_x}$ is ϕ_j THEN $fuzzyForca_{\alpha_x}$ is ϕ_j

\mathcal{R}_j^y : IF e_y^0 is $\phi_j \wedge v_y$ is $\phi_j \wedge \Delta d_{\alpha\beta 1}$ is $\phi_j \dots \wedge \Delta d_{\alpha\beta 4}$ is $\phi_j \wedge \vec{n}_{\alpha\beta 1_y}$ is $\phi_j \wedge \dots \wedge \vec{n}_{\alpha\beta 4_y}$ is $\phi_j \wedge \Delta d_{\alpha i}$ is $\phi_j \wedge \vec{n}_{\alpha i_y}$ is ϕ_j THEN $fuzzyForca_{\alpha_y}$ is ϕ_j

Sabe-se que o número de regras em um sistema fuzzy cresce exponencialmente, na mesma proporção em que mais variáveis fuzzy vão sendo adicionadas ao sistema.

Ao considerar um número menor de agentes vizinhos, a regressão ficaria muito mais imprecisa, pois a técnica conexionista não saberia diferenciar, quando há mais que um agente próximo. Ou seja, poderia se ter intensidade de forças completamente diferentes para um vizinho posicionado no mesmo lugar e mesma direção.

Uma alternativa é fazer, como Sent (2016) e tratar separadamente os três componentes forças, o que em seu trabalho gerou pouquíssimas regras, embora as tenha elaborado manualmente. Entretanto, o propósito desta pesquisa é ignorar toda fundamentação teórica e matemática complicada, que envolve entender o que cada componente de força realiza. Ou seja, não haveria a abstração dos componentes de forças em apenas uma única, de tal forma que isso representasse a resposta de um agente à seus estímulos sensoriais físicos em um instante de tempo.

5. Experimentos e Resultados

De forma a validar esses conjuntos de dados são comparadas simulações do SFM com a MLP, para que posteriormente a INFGMN seja treinada com esses mesmos conjuntos de dados e, então, esteja sujeita a comparações com os outros dois modelos. Cada comparação é composta por uma análise de duas etapas. A primeira, qualitativa, busca observar os fenômenos em multidões. Já a segunda etapa é uma análise quantitativa, em que coleta-se dados estatísticos das simulações.

Em ambas as análises, dois cenários são utilizados, mas com uma quantidade de agentes que varia. Isso se faz necessário, uma vez que a INFGMN gerou mais de 12 mil regras para cada FIS, deixando o desempenho do simulador extremamente lento. O primeiro cenário, à esquerda na Figura 3, é o cenário padrão do simulador. O segundo cenário, à direita da Figura 3, tem dimensões de 20x35 metros e várias colunas, obstáculos, espalhadas no cenário e os agentes afastados o máximo possível da saída, de forma a observar melhor seus movimentos a uma distância maior.

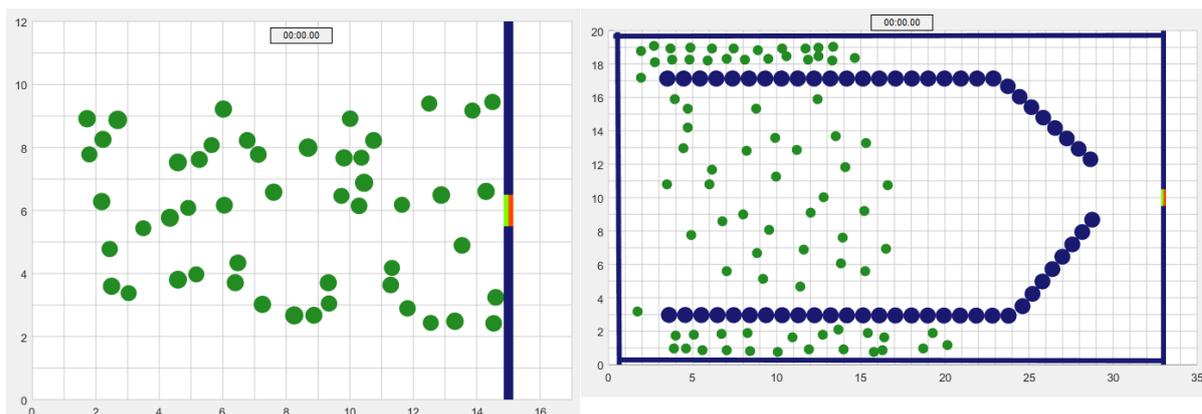


Figura 3. Cenários de simulação do PANIC SIMULADOR para INFGMN e MLP.

5.1. Análise qualitativa

Para ilustrar o potencial dos conjuntos de dados coletados, utilizou-se muitos mais agentes para os experimentos com a MLP, que para INFGMN. Nos experimentos com a MLP, além das forças de desejo e repulsão, constam entre os fenômenos capturados: o “Quanto-mais-rápido-mais-devagar” em ambos cenários e o “Congelamento-por-aquecimento” no segundo cenário. Este último visto no segundo cenário, pois há mais

agentes sendo simulados e os que se encontram no meio do atolamento, ficam paralisados. Também, no segundo cenário foi possível observar a “Formação de Faixas” de pedestres que ficam nas laterais da arena. Posteriormente, essas faixas se cruzam em direção à saída dando forma a “Oscilação de fluxo de gargalo”.

Utilizando a INFGMN, os resultados ficam aquém do esperado. Essa técnica é incapaz de reproduzir quaisquer fenômenos em multidões, sequer conseguindo simular as forças de repulsão, ou seja, os agentes atravessam as paredes e também não se repelem entre si, ignorando a existência de outros agentes. Ainda assim, os agentes seguem para o sentido da arena de onde se localiza a saída e em alguns casos do primeiro cenário é possível observar alguns indivíduos atingirem exatamente a posição de onde se encontra a saída, reproduzindo assim a Força de Desejo. Dois possíveis motivos para isso, é que a maior parte do conjunto de treinamento é obtido em um cenário muito próximo à esse, e que a INFGMN responda melhor para agentes, da qual seus vizinhos e obstáculos próximos estejam mais distantes.

5.2. Análise quantitativa

Para a última etapa de análise, são simulados 10 agentes para o SFM, MLP e a INFGMN, sendo que em todos os três casos, os agentes estão exatamente na mesma posição para um dado cenário.

As Figuras 4 e 5 mostram a média da força obtida para os 10 agentes em cada passo, ou seja, para cada invocação do SFM ou MLP ou INFGMN. O número de passos em cada simulação varia, dependendo da intensidade de força obtida pelos agentes. Visto isso, é possível averiguar uma boa similaridade entre as médias obtidas pelo SFM e o MLP. Em contrapartida, no caso da INFGMN, ela apresenta uma similaridade numérica muito baixa com o SFM e contém grandes picos de força, fazendo com que os agentes saiam muito antes do previsto. Isso é perceptível principalmente no segundo cenário.

A seguir, os dois gráficos de barra, 6 e 7, expõem respectivamente para o primeiro e segundo cenários, os tempos de saída de cada agente em uma simulação com SFM, MLP e INFGMN. As abcissas desse gráfico indicam o número de agentes que saíram até o instante de tempo fornecido pelas ordenadas. Observa-se ali que, por consequência da boa regressão obtida pela MLP, o tempo de saídas dos agentes para MLP e SFM são muito próximos. Para a INFGMN, mesmo com fraca aproximação numérica na obtenção das forças, os tempos de saída foram muito próximos aos da MLP e SFM. Isso ocorre porque a maior parte do conjunto de treinamento é obtido em um cenário próximo do primeiro.

5.3. Considerações finais

Os conjuntos dados selecionados são suficientemente bons, uma vez que a MLP consegue uma reprodução fiel da maioria dos fenômenos descritos. Por outro lado, com a INFGMN, não se tem sucesso em reproduzir adequadamente os fenômenos em multidões. Uma possível hipótese para isso, é que o método da centroide não oferece valores diferenciados para situações de repulsão.

A obtenção de um valor não adequado para um eixo de força de um agente possui um efeito cumulativo ao longo da simulação, pois a velocidade desse agente é atualizada com um valor desconhecido ao intervalo dessa variável. Em virtude disso, na etapa de fuzzificação, o grau de pertinência dessa nova velocidade irá tender à zero. Isso impede a

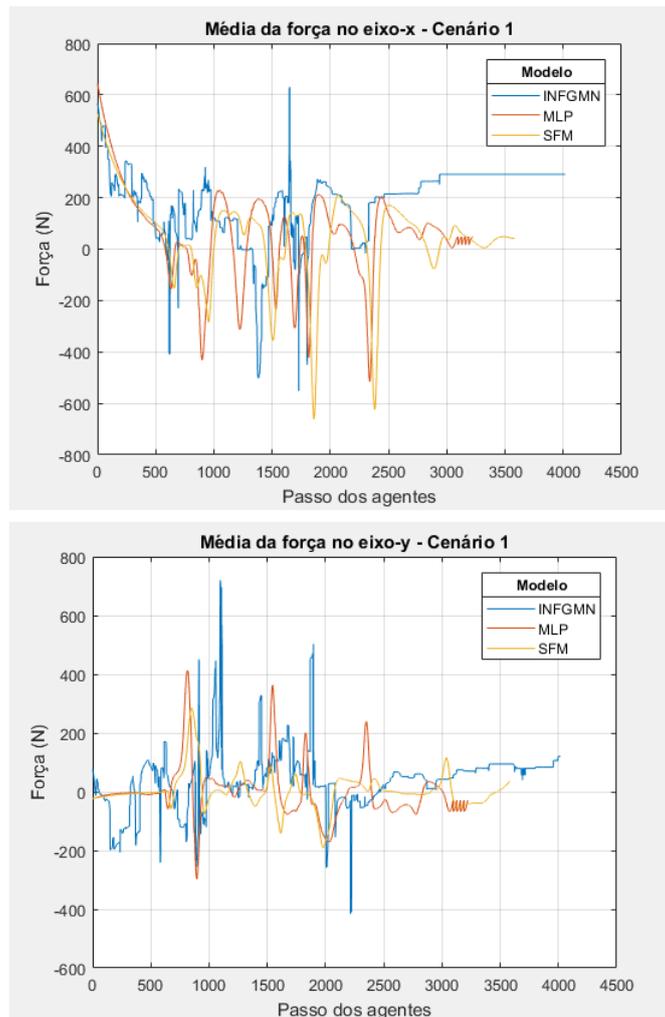


Figura 4. Média das forças no cenário 1 para o eixo- x e eixo- y

ativação das regras e faz com que o valor *crisp* para força seja pouco plausível. Por fim, é o motivo pelo qual o agente reproduz movimentos descoordenados durante a simulação.

O fato de haver mais de doze mil conjuntos fuzzy em ambos FIS, implica que o conjunto de dados não é suficientemente correlacionado para a INFGMN. Uma possibilidade para corrigir isso é tentar especializar o conjunto de treinamento para um cenário específico, restringindo o intervalo de valores que as variáveis possam ter. Isso otimizaria a precisão na etapa de defuzzificação e agruparia os dados em menos componentes de mistura gaussianos, tornando o desempenho da aplicação mais satisfatório. Entretanto, isso em nada contribuiria para tornar este problema mais interpretável, dado que o número de entradas permaneceria igual.

6. Conclusão

Essa pesquisa buscou analisar a possibilidade de que o modelo de forças de Helbing pudesse ser substituído por redes neuro-fuzzy que gerassem regras do tipo Mamdani-Larsen, como a INFGMN. Duas análises foram realizadas, comparando os desempenhos da MLP, INFGMN com o *Social Force Model*. Sendo possível concluir que o conjunto de

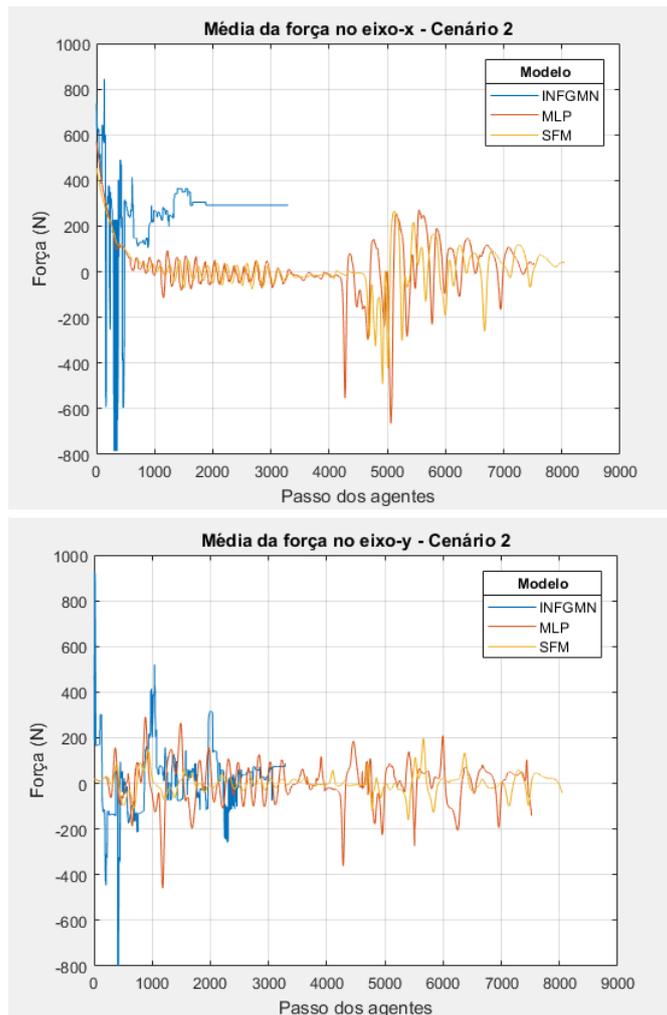


Figura 5. Média das forças no cenário 2 para o eixo- x e eixo- y

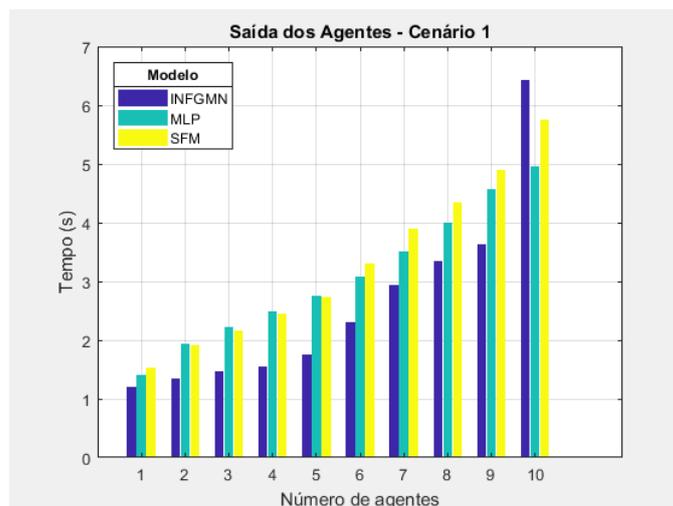


Figura 6. Tempos de saída para os agentes no cenário 1.

dados obtido é realmente bom, pois a MLP conseguiu reproduzir adequadamente os com-

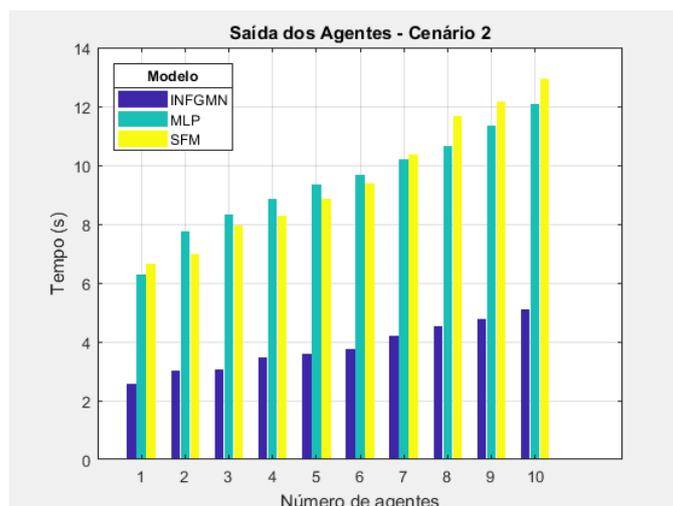


Figura 7. Tempos de saída para os agentes no cenário 2.

portamentos de multidões. Entretanto, para a INFGMN, mesmo com dezenas de milhares de regras, não foi possível reproduzir esses fenômenos e as forças resultantes tiveram valores muito distantes das obtidas pelo SFM.

Assim, essa pesquisa sugere a necessidade de uma outra abordagem na montagem desses conjuntos de treinamento multivariados para INFGMN. Por exemplo, levar em conta apenas cenários de evacuações específicos, de tal forma que o intervalo de valores dessas variáveis seja menor. Isso favoreceria o agrupamento desses dados de entrada em termos linguísticos de saída, o que reduziria o número de conjuntos fuzzy gerados, por haver menos componentes de mistura gaussianas, ao passo que o método da centroide seria mais preciso em resoluções de problemas como este.

Para trabalhos futuros sugere-se ignorar as sensações locais dos agentes e adotar uma abordagem que foque mais na interpretabilidade dos componentes do *Social Force Model* do que em sua unidade. Um caminho diferente, seria rever o modelo de Helbing de forma pesquisar a influência de estados anteriores dos agentes, como sua força, em seus próximos movimentos, reduzindo o número final de variáveis. Para isso poderia-se utilizar redes neurais recorrentes.

Referências

- [Bernardini et al. 2014] Bernardini, G., D’orazio, M., Quagliarini, E., and Spalazzi, L. (2014). An agent-based model for earthquake pedestrians’ evacuation simulation in urban scenarios. In *Transportation Research Procedia*, volume 2, pages 255–263.
- [Cony et al. 2007] Cony, C. A., de Lima Bicho, A., Jung, C. R., Magalhães, L. P., and Musse, S. R. (2007). A perceptive model for virtual agents in crowds. page 141–150.
- [Dheeru and Karra Taniskidou 2017] Dheeru, D. and Karra Taniskidou, E. (2017). UCI machine learning repository.
- [Diab and Saade 2004] Diab, H. and Saade, J. J. (2004). Defuzzification methods and new techniques for fuzzy controllers. In *Iranian Journal of Electrical and Computer Engineering*, volume 3, page 161–174.

- [Fisher 1936] Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals Eugen.*, 7:179–188.
- [Gan et al. 2005] Gan, M.-T., Hanmandlu, M., and Tan, A. H. (2005). From a gaussian mixture model to additive fuzzy systems. *IEEE Transactions on Fuzzy Systems*, 13(3):303–316.
- [Helbing et al. 2000] Helbing, D., Farkas, I., and Vicsek, T. (2000). Simulating dynamical features of escape panic. *Nature*, 407:487–490.
- [Helbing and Johansson 2010] Helbing, D. and Johansson, A. (2010). Pedestrian, crowd and evacuation dynamics. In *Encyclopedia of Complexity and Systems Science*, volume 16, pages 697–716.
- [Helbing and Molnár 1995] Helbing, D. and Molnár, P. (1995). Social force model for pedestrian dynamics. *Phys. Rev. E*, 51:4282–4286.
- [Henderson 1974] Henderson, L. (1974). On the fluid mechanics of human crowd motion. In *Transportation Res.*, volume 8, pages 509–515.
- [Henderson 1971] Henderson, L. F. (1971). The Statistics of Crowd Fluids. *Nature*, 229(5284):381–383.
- [Jang 1993] Jang, J. S. R. (1993). Anfis: adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(3):665–685.
- [Kasabov 2001] Kasabov, N. (2001). Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 31(6):902–918.
- [Kasabov 2006] Kasabov, N. (2006). *Evolving Connectionist Systems: The Knowledge Engineering Approach*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.
- [Kasabov and Song 2002] Kasabov, N. K. and Song, Q. (2002). Denfis: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *IEEE Transactions on Fuzzy Systems*, 10(2):144–154.
- [Li et al. 2000] Li, H., Philip, C. C. L., and Huang, H.-P. (2000). *Fuzzy Neural Intelligent Systems: Mathematical Foundation and the Applications in Engineering*. CRC Press, Inc., Boca Raton, FL, USA, 1st edition.
- [Lin and Lee 1996] Lin, C.-T. and Lee, C. S. G. (1996). *Neural Fuzzy Systems: A Neuro-fuzzy Synergism to Intelligent Systems*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- [Mazzutti et al. 2017] Mazzutti, T., Roisenberg, M., and José de Freitas Filho, P. (2017). Infgmn – incremental neuro-fuzzy gaussian mixture network. In *Expert Systems with Applications*, volume 89, pages 160–178. Pergamon Press, Inc., Tarrytown, NY, USA.
- [Nasser 2011] Nasser, T. Z. (2011). Modelo para simulação de multidões utilizando lógica nebulosa.
- [Rodrigues et al. 2014] Rodrigues, D. G., Moura, G., Jacinto, C. M. C., de Freitas Filho, P. J., and Roisenberg, M. (2014). Generating interpretable mamdani-type fuzzy rules using a neuro-fuzzy system based on radial basis functions. In *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1352–1359.

- [Ross 2009] Ross, T. (2009). *Fuzzy Logic with Engineering Applications*. Wiley.
- [Schmidt and Späh 2014] Schmidt, J. and Späh, A. (2014). PANIC SIMULATOR.
- [Sent 2016] Sent, A. (2016). Proposta de modelo fuzzy do modelo de forças sociais para simulação de pedestres em situação de evacuação. Master's thesis, Universidade Federal de Santa Catarina, Florianópolis, SC.
- [Sent 2014] Sent, A. D. (2014). Fuzzy Social Force Model.
- [Sent et al. 2015] Sent, A. D., Roisenberg, M., and de Freitas Filho, P. J. (2015). Simulation of crowd behavior using fuzzy social force model. In *2015 Winter Simulation Conference (WSC)*, pages 3901–3912.
- [Sharma and Singh 2006] Sharma, S. and Singh, H. (2006). Multi-agent system for simulating human behavior in egress simulations. In *Proceedings of the NAACCSOS, Annual Conference of the North American Association for Computational Social and Organizational Sciences*.
- [Sharma 2004] Sharma, S. C. (2004). An object oriented approach to simulating agent-based behavior.
- [Xue et al. 2017] Xue, Z., Dong, Q., Fan, X., Jin, Q., Jian, H., and Liu, J. (2017). Fuzzy logic-based model that incorporates personality traits for heterogeneous pedestrians. *Symmetry*, 9(10).
- [Yu and Deng 2014] Yu, D. and Deng, L. (2014). *Automatic Speech Recognition: A Deep Learning Approach*, chapter Chapter 2: Gaussian Mixture Models, pages 13–21. Springer Publishing Company, Incorporated.
- [Zadeh 1965] Zadeh, L. (1965). Fuzzy sets. *Information and Control*, 8(3):338 – 353.