

CURSO DE CIÊNCIA DA COMPUTAÇÃO

Ana Júlia Fontoura

**SIMULAÇÃO MULTIAGENTES EM ÁREAS ATINGIDAS POR  
TERREMOTOS COM FOCOS DE INCÊNDIO**

Santa Cruz do Sul

2016

Ana Júlia Fontoura

**SIMULAÇÃO MULTIAGENTES EM ÁREAS ATINGIDAS POR  
TERREMOTOS COM FOCOS DE INCÊNDIO**

Trabalho de Conclusão apresentado ao  
Curso de Ciência da Computação da  
Universidade de Santa Cruz do Sul, para  
obtenção do título de Bacharela em Ciência da  
Computação.

Orientadora: Prof<sup>ª</sup>. Dr<sup>ª</sup>. Rejane Frozza

Santa Cruz do Sul

2016



## **AGRADECIMENTOS**

Primeiramente gostaria de agradecer a meus pais Geraldo e Margarete por tudo o que me ensinaram durante a vida, pelo incentivo em todos os momentos e pela compreensão pelos finais de semana em casa, sem vocês não teria chegado aqui.

A meu noivo Tauame, por ter me apoiado e motivado ao longo deste trabalho, me dando forças para seguir em frente, mesmo ficando algumas vezes em segundo plano. Essa conquista é tanto minha quanto sua.

Também gostaria de agradecer a orientação da amiga e professora Rejane Frozza, que acompanhou e apoiou o desenvolvimento deste trabalho, contribuindo com dicas e conselhos.

Enfim, agradeço a todos que contribuíram ao longo do curso e deste trabalho.

*Em meus anos mais jovens e vulneráveis  
meu pai me deu um conselho que ficou na minha cabeça até hoje.  
“Sempre que tiver vontade de criticar alguém”, me disse,  
“lembre-se de que nem todas as pessoas neste mundo tiveram as vantagens que você teve.”*  
(FITZGERALD, F. S. *O Grande Gatsby*)

## RESUMO

O cenário criado por um terremoto apresenta muitos desafios para equipes de resgate, uma vez que desabamentos, interrupções no fornecimento de água e energia, incêndios, entre outros, podem ocorrer. O rompimento de tubulações de gás pode gerar incêndios em múltiplas localidades, podendo se espalhar pelas vizinhanças. Assim, o objetivo da pesquisa é desenvolver um Sistema Multiagentes composto por agentes inteligentes que devem combater múltiplos incêndios em um ambiente urbano simulando uma situação pós-terremoto, a fim de reduzir o dano total causado por incêndios, ao final da simulação. Para realizar esta tarefa, o ambiente é explorado dividindo o mapa em setores, através da técnica de Clusterização, com o algoritmo *K-means*, e a decisão de qual incêndio cada agente combaterá é baseada em RNA (Redes Neurais Artificiais), que avalia como entradas os fatores intensidade do incêndio, material e área do prédio em chamas, número de bombeiros combatendo o incêndio, distância entre o incêndio e o hidrante mais próximo, distância entre o bombeiro e o incêndio, coeficiente de inflamabilidade médio dos materiais dos vizinhos sem incêndio do prédio em chamas. Para avaliar os resultados da estratégia proposta, foram realizados 10 pares de simulações, cada par composto por uma simulação utilizando bombeiros com RNA e outro utilizando bombeiros simples. A RNA identificou fatores relevantes para a tomada de decisão em relação aos focos de incêndio a serem combatidos pelos agentes bombeiros, com uma exploração efetiva do ambiente pelo método de clusterização.

Palavras chave: Redes Neurais Artificiais, Sistemas Multiagentes, Clusterização, Algoritmo *K-means*, Combate a Incêndios, Terremotos.

## **ABSTRACT**

The scenario created by an earthquake presents many challenges for rescue teams, such as landslides, disruptions in the supply of water and energy, and others, may occur. Disruption of gas pipes can cause fires in multiple locations that can spread to neighboring buildings. This paper presents a development proposal for a Multiagent System composed of intelligent agents able to fight multiple fires in an urban environment simulating a post-earthquake situation. The objective of this system is to reduce the overall fire damage at the end of the simulation. To accomplish this task, the environment is explored by dividing the map into sectors, through K-means clustering, and the decision of which fire each agent will combat is based on ANN (Artificial Neural Networks), which evaluates the following inputs: fire intensity, building material and floor area, number of firefighters fighting the fire, distance between the fire and the closest hydrant, distance between the firefighter and the fire, average inflammability coefficient of the building materials of the fireless neighbors of the fire. To evaluate the proposed strategy, 10 simulation pairs were performed, each pair composed of a simulation using firefighters with ANN and another with simple ones. The ANN identified relevant factors for decision making in regards to the fires to be fought by the firefighters, with an effective environment exploration by the clustering method.

Key words: Artificial Neural Networks, Multiagent Systems, Clustering, K-means Algorithm, Fire Fighting, Earthquakes.

## LISTA DE FIGURAS

Figura 1 – Modelo básico de um agente reativo.....	29
Figura 2 - Modelo básico de um agente cognitivo .....	29
Figura 3 - Modelo de um agente híbrido .....	30
Figura 4 – Modelo de um agente BDI .....	31
Figura 5 – Neurônio biológico.....	37
Figura 6 – Modelo matemático do neurônio de McCulloch e Pitts.....	37
Figura 7 – Exemplo de uma RNA com 3 camadas.....	38
Figura 8 – Etapas de desenvolvimento da pesquisa.....	54
Figura 9 – Arquitetura do sistema multiagentes .....	55
Figura 10 – Materiais dos prédios .....	56
Figura 11 – Fluxograma da clusterização com o algoritmo K-means .....	57
Figura 12 – Funcionamento do Gerador de Incêndios.....	61
Figura 13 – Fluxograma do Core.....	62
Figura 14 – Raio de visão dos agentes.....	63
Figura 15 – Fluxograma de execução do algoritmo A*.....	64
Figura 16 – Arquitetura do agente Explorador .....	65
Figura 17 – Fluxograma do comportamento do agente Explorador .....	66
Figura 18 – Arquitetura do agente bombeiro.....	67
Figura 19 – Fluxograma do estado “Indo para setor” do bombeiro.....	69
Figura 20 – Fluxograma do estado “Indo para incêndio” do bombeiro.....	70
Figura 21 – Fluxograma do estado “Combatendo incêndio” do bombeiro .....	70



Figura 22 – Fluxograma do estado “Indo para hidrante” do bombeiro .....	70
Figura 23 – Fluxograma do estado “Enchendo tanque” do bombeiro.....	71
Figura 24 – Arquitetura da RNA .....	72
Figura 25 – Interface gráfica.....	76
Figura 26 – Intensidades do incêndio de 0 (mais à esquerda) até 6 (mais à direita) ....	76
Figura 27 – Janela exibida ao final de cada simulação.....	77
Figura 28 – Estado “Indo para incêndio” do bombeiro simples .....	78
Figura 29 – Porcentagem de área queimada do mapa ao final de cada simulação .....	79
Figura 30 – Total de incêndios apagados em cada simulação .....	79
Figura 31 – Número de incêndios existentes ao final de cada simulação.....	80
Figura 32 – Intensidade média dos incêndios selecionados .....	81
Figura 33 – Coeficiente de inflamabilidade médio do material do prédio dos incêndios selecionados.....	81
Figura 34 – Coeficiente de inflamabilidade médio do material dos vizinhos sem incêndio dos incêndios selecionados .....	82
Figura 35 – Área média do prédio dos incêndios selecionados.....	82
Figura 36 – Nº de bombeiros combatendo os incêndios selecionados em média.....	83
Figura 37 – Distância média entre o incêndio selecionado e o bombeiro .....	84
Figura 38 – Distância média entre o incêndio selecionado e o hidrante mais próximo .....	84
Figura 39 – Dados armazenados a cada ciclo no relatório JSON.....	85
Figura 40 – Dados dos incêndios selecionados no relatório JSON .....	86

## LISTA DE TABELAS E QUADROS

Tabela 1 - Desastres naturais ocorridos entre 2001 e 2015.....	14
Tabela 2 - Coeficientes de inflamabilidade dos materiais.....	58
Tabela 3 - Distribuição de intensidade dos incêndios iniciais.....	59
Quadro 1 – Classificação de desastres conforme intensidade .....	18
Quadro 2 – Intensidade de terremotos conforme a Escala Mercalli Modificada .....	22
Quadro 3 – Quadro comparativo dos trabalhos relacionados – SMAs em Desastres Naturais .....	47
Quadro 4 – Quadro comparativo dos trabalhos relacionados de Aprendizagem em Sistemas Multiagentes .....	51

## LISTA DE ABREVIATURAS

<b>AAMAS</b>	<i>Autonomous Agents and Multiagent Systems</i>
<b>BDI</b>	<i>Belief-Desire-Intention</i>
<b>CAPES</b>	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
<b>DAARTS</b>	<i>Decision Assistant Agent in Real Time Simulation</i>
<b>ECP</b>	Estado de Calamidade Pública
<b>HAC</b>	<i>Human-Agent-Collectives</i>
<b>IA</b>	Inteligência Artificial
<b>IAD</b>	Inteligência Artificial Distribuída
<b>IEEE</b>	<i>Institute of Electrical and Electronics Engineers</i>
<b>PIB</b>	Produto Interno Bruto
<b>RNA</b>	Rede Neural Artificial
<b>SE</b>	Situação de Emergência
<b>SMA</b>	Sistema MultiAgentes
<b>UNISC</b>	Universidade de Santa Cruz do Sul

## SUMÁRIO

1 INTRODUÇÃO .....	14
2 FUNDAMENTAÇÃO TEÓRICA.....	17
2.1 Desastres Naturais .....	17
2.1.1 Classificação dos desastres.....	17
2.1.1.1 Intensidade.....	18
2.1.1.2 Evolução.....	19
2.1.1.3 Origem.....	19
2.1.1.4 Duração .....	20
2.1.2 Gerenciamento de desastres naturais.....	20
2.1.3 Terremotos.....	21
2.1.3.1 Magnitude X Intensidade .....	22
2.1.3.2 Efeitos secundários.....	24
2.1.3.3 Maiores Terremotos da História.....	25
2.2 Sistemas Multiagentes – SMAs .....	25
2.2.1 Agentes .....	26
2.2.1.1 Características.....	27
2.2.1.2 Tipos de Agentes .....	27
2.2.1.3 Arquitetura de Agentes .....	28
2.2.3 Sistemas multiagentes .....	32
2.2.3.1 Tipos de interação entre agentes.....	32
2.3 Aprendizagem em Sistemas Multiagentes .....	33
2.3.1 Redes Neurais Artificiais.....	36
2.3.2 Clusterização <i>K-means</i> .....	39
2.4 Trabalhos Relacionados de Sistemas Multiagentes para Desastres Naturais .....	40
2.4.1 A partitioning-based task allocation strategy for Police Multi-Agents.....	40
2.4.2 Coordination of Agents in the RoboCup Rescue: A Partial Global approach .....	41
2.4.3 Disaster Management in Real Time Simulation Using Machine Learning..	42
2.4.4 HAC-ER: A Disaster Response System based on Human-Agent Collectives .....	42
2.4.5 Human-agent collaboration for disaster response.....	43
2.4.6 RoboCup Rescue 2013 LTI Agent Rescue Team Description.....	44
2.4.7 Quadro Comparativo dos SMAs para Desastres Naturais.....	46
2.5 Trabalhos Relacionados à Aprendizagem em Sistemas Multiagentes.....	50
2.5.1 Utilizando Agentes Autônomos com Aprendizado para Identificação e Combate de Incêndios em Áreas Florestais .....	50
2.5.2 Simulação Virtual da Evolução de Estratégias e do Controle Inteligente em Sistemas Multi-Robóticos .....	50
2.5.3 Quadro Comparativo .....	51

2.6 Considerações .....	52
<b>3 SISTEMA MULTIAGENTE PARA COMBATE A INCÊNDIOS .....</b>	<b>53</b>
3.1 Metodologia da Pesquisa .....	53
3.2 Arquitetura do sistema multiagentes .....	55
3.2.1 Mapa .....	55
3.2.2 Gerador de Incêndios .....	57
3.2.3 Core .....	61
3.2.4 Agentes .....	62
3.2.4.1 Exploradores .....	65
3.2.4.2 Bombeiros .....	66
3.2.5 RNA .....	71
3.2.5.1 Geração dos dados para o treinamento da RNA .....	72
3.2.5.2 Treinamento da RNA .....	73
3.3 Interface .....	74
3.4 Validação .....	77
3.5 Considerações .....	86
<b>4 CONCLUSÃO .....</b>	<b>88</b>
<b>REFERÊNCIAS .....</b>	<b>90</b>

## 1 INTRODUÇÃO

Devido principalmente ao grande número de desastres naturais ocorrendo em todo o mundo, o gerenciamento destes tem se tornado uma área importante a ser explorada (SINGH, MODANWAL e BASAK, 2011). Desde o início do século XXI, mais de um milhão de pessoas já morreram em decorrência de desastres naturais, sendo que destas, mais de 700 mil foram vítimas de terremotos, como pode ser observado na Tabela 1 (GUHA-SUPIR, BELOW e HOYOIS, 2015). Além disto, terremotos são a segunda maior causa de desabrigados e feridos, apesar de não fazerem parte dos desastres mais frequentes.

**Tabela 1 - Desastres naturais ocorridos entre 2001 e 2015.**

<b>Tipo de desastre</b>	<b>Ocorrências</b>	<b>Mortos</b>	<b>Feridos</b>	<b>Desabrigados</b>	<b>Dano total (US\$<b>x</b>1000)</b>
<b>Terremoto</b>	406	712.491	1.372.126	11.935.770	489.677.191,00
<b>Tempestade</b>	1.477	189.239	304.310	5.638.395	716.856.102,00
<b>Temperaturas extremas</b>	320	156.151	1.961.382	238.247	42.074.034,00
<b>Epidemia</b>	630	81.770	419.301	0	0,00
<b>Enchente</b>	2.391	80.973	260.640	15.244.100	382.387.806,00
<b>Seca</b>	229	21.079	0	0	69.415.243,00
<b>Deslizamento de terra</b>	265	12.681	2.370	273.056	2.262.785,00
<b>Incêndio florestal</b>	150	864	5.176	57.885	27.506.954,00
<b>Atividade vulcânica</b>	83	665	1.141	116.000	465.655,00

Fonte: (GUHA-SUPIR, BELOW e HOYOIS, 2015)

Estratégias de alocação de recursos apropriadas são necessárias para agir em resposta aos efeitos dos desastres (BASAK, MODANWAL e MAZUMDAR, 2011). Segundo Farinelli *et al.* (2003), esta tarefa impõe alguns desafios secundários, uma vez que diferentes desastres, como incêndios, danos aos sistemas de transporte e comunicações, entre outros, podem

acontecer, tornando difícil realizar um planejamento exato do que será feito em resposta ao desastre.

Assim, os desastres podem ser tratados de duas maneiras distintas: A primeira é evitando a sua ocorrência; a segunda é por meio do estabelecimento prévio de políticas e estratégias de gerenciamento de desastres, quando possível (SILVA, NARDIN e SICHMAN, 2012). Como a maioria dos desastres não pode ser evitada, uma das possibilidades é o desenvolvimento de simulações com agentes inteligentes, como forma de resposta ao desastre a partir de parâmetros relacionados ao comportamento do desastre e das pessoas envolvidas.

O resgate de vítimas em desastres naturais é um problema social de alta relevância. Segundo Singh, Modanwal e Basak (2011), as equipes de resgate agem frequentemente em ambientes perigosos e cheios de incertezas. Por meio da implementação de agentes inteligentes, contribui-se para que se minimize a utilização de humanos e animais em equipes de combate a desastres naturais. Esta redução é ainda mais importante no caso de terremotos, onde frequentemente as estruturas das edificações estão abaladas, impondo um risco iminente à vida de todos os envolvidos na ação de resgate.

Além da preservação das vidas envolvidas nos esforços de salvamento, a utilização de agentes inteligentes permite aumentar a eficiência das equipes. Pois, segundo Kitano e Tadokoro (2001), a magnitude de um desastre, e a velocidade na qual o sistema precisa mudar de acordo com a situação, está muito além da capacidade humana de planejamento em um ambiente extremamente hostil. Portanto, pode-se reduzir o tempo de salvamento, e conseqüentemente, o número de vítimas fatais.

Na literatura, observa-se a existência de trabalhos, tanto na área de aprendizagem e evolução de agentes, como na área de simulação de ambientes de desastres naturais. A fim de se obter resultados relevantes na área de resgate em desastres naturais, este trabalho utiliza conceitos de aprendizagem em agentes, aplicando-os em um ambiente representando uma situação pós-terremoto. Com isto, pretende-se aumentar a eficiência das equipes de resgate, além de expor menos vidas aos riscos impostos pelo ambiente. Para tanto, são estudadas as Redes Neurais Artificiais para serem utilizadas como estratégia de aprendizado para a ação dos agentes no cenário de terremoto, já que são baseadas em aprendizado supervisionado e não supervisionado.

Define-se como problema de pesquisa: “Como Redes Neurais Artificiais e Clusterização podem ser utilizadas para melhorar a tomada de decisão de agentes em um cenário multiagentes de simulação de terremotos?”.

O objetivo principal é desenvolver uma estratégia de aprendizagem, baseada em redes neurais artificiais e clusterização, em agentes inteligentes para atuação em situações de calamidade, com foco em áreas atingidas por terremotos. Para a validação desta estratégia, foi desenvolvido um simulador próprio para analisar o comportamento dos agentes em um cenário pós terremoto com focos de incêndio. Não se optou pelo uso de um simulador já existente para que fosse possível desenvolver as características específicas nos agentes com a estratégia baseada em RNA. A rede neural artificial foi usada para auxiliar a tomada de decisão do agente bombeiro e a clusterização para a divisão do terreno/mapa, reduzindo o escopo de atuação dos agentes.

Sendo assim, pretende-se atingir os seguintes objetivos específicos:

- Analisar outros trabalhos na área de resgate em desastres utilizando agentes inteligentes.
- Verificar o uso de aprendizagem em agentes para atuarem em cenário de terremoto.
- Desenvolver o ambiente simulando uma situação pós-terremoto.
- Desenvolver agentes inteligentes que atuam em um sistema multiagentes simulando uma área atingida por terremoto. Os agentes devem ser capazes de lidar com as incertezas deste tipo de ambiente.
- Validar a atuação dos agentes desenvolvidos para o cenário de terremoto simulado.

No capítulo 2, é apresentada a fundamentação teórica, descrevendo os principais conceitos sobre desastres naturais, sistemas multiagentes, aprendizagem em sistemas multiagentes e, por fim, os trabalhos relacionados. O capítulo 3 descreve a metodologia, o desenvolvimento do trabalho e a validação. Por fim, são apresentadas as conclusões.



## **2 FUNDAMENTAÇÃO TEÓRICA**

A revisão de literatura abordada nesta pesquisa está dividida em quatro tópicos: Desastres naturais, Sistemas Multiagentes, Aprendizagem em Sistemas Multiagentes e, trabalhos relacionados.

### **2.1 Desastres Naturais**

Estatísticas no mundo todo revelam um aumento no número de desastres naturais e seu impacto nas últimas décadas. Dentre as possíveis causas para este aumento, estão a degradação do solo, más condições socioeconômicas e mudança do clima (GENCER, 2013).

Os desastres naturais podem ser conceituados, de forma simplificada, como o resultado do impacto de um fenômeno natural extremo ou intenso sobre um sistema social, causando sérios danos e prejuízos que excedem a capacidade dos afetados em conviver com o impacto (TOBIN e MONTZ, 1997) *apud* (MARCELINO, 2008). Segundo Matsuno e Tadokoro (2004), o estudo de desastres e tecnologias, incluindo sismologia, arquitetura, engenharia civil e geologia contribuem para a mitigação de desastres.

Nas subseções a seguir são demonstradas algumas classificações de desastres e como realizar o seu gerenciamento, e por fim, são abordados alguns conceitos relativos a terremotos.

#### **2.1.1 Classificação dos desastres**

O termo “desastre natural” é utilizado para descrever uma vasta gama de eventos, muitas vezes, com características completamente distintas. Uma seca, por exemplo, tem poucas coisas em comum com um furacão, porém, ambos podem causar impactos socioeconômicos de mesma grandeza. Portanto, quaisquer tentativas de descrever as características de um desastre devem ser vagas o suficiente para comportar a amplitude de eventos existentes.

Em um esforço para diferenciar as ocorrências de desastres naturais entre si, são descritas a seguir algumas classificações, conforme Tominaga, Santoro e Amaral (2012), Kobiyama *et al.* (2006), e Castro (1999).

### 2.1.1.1 Intensidade

O quadro 1 descreve situações criadas por diferentes intensidades de desastres. Nesta classificação baseada no Produto Interno Bruto (PIB) da região atingida, proposta por Castro (1999), os desastres com níveis I e II podem ser superados pela região atingida sem a necessidade de recursos ou ajuda externa. Desastres a partir do nível III dependem da ajuda dos governos estaduais e/ou federal, sendo que em situações de nível IV pode ser necessária intervenção internacional.

**Quadro 1 – Classificação de desastres conforme intensidade**

<b>Nível</b>	<b>Intensidade</b>	<b>Situação</b>
<b>I</b>	Desastre de pequeno porte, onde os impactos causados são pouco importantes e os prejuízos pouco vultosos. <b>(Prejuízo <math>\leq</math> 5% PIB municipal)</b>	Facilmente superável com os recursos do município.
<b>II</b>	De média intensidade, onde os impactos são de alguma importância e os prejuízos são significativos, embora não sejam vultosos. <b>(5% &lt; Prejuízo <math>\leq</math> 10% PIB)</b>	Superável pelo município, desde que envolva uma mobilização e administração especial.
<b>III</b>	De grande intensidade, com danos importantes e prejuízos vultosos. <b>(10% &lt; Prejuízo <math>\leq</math> 30% PIB)</b>	A situação de normalidade pode ser restabelecida com recursos locais, desde que complementados com recursos estaduais e federais. <b>(Situação de Emergência – SE)</b>

IV	Com impactos muito significativos e prejuízos muito vultosos. <b>(Prejuízo &gt; 30% PIB)</b>	Não é suportável pelo município, sem que receba ajuda externa. Eventualmente necessita de ajuda internacional. <b>(Estado de Calamidade Pública – ECP)</b>
----	---	---

Fonte: Adaptado de Kobiyama *et al.* (2006)

### 2.1.1.2 Evolução

Quanto à evolução, os desastres podem ser classificados em “súbitos”, “graduais” ou “somação de efeito parcial”. Segundo Castro (1999), desastres súbitos são aqueles caracterizados pela velocidade repentina em que acontecem, como é o caso em enchentes relâmpagos, tornados ou terremotos. Desastres graduais podem ser descritos como desastres que evoluem mais lentamente de forma progressiva. Neste tipo de classificação, se enquadram as enchentes graduais e secas. Os desastres com somação de efeitos parciais são formados pelo conjunto de pequenos eventos que sozinhos não caracterizam desastres, mas que como um todo pode ser tratado como um desastre. Servem como exemplos para esta classificação inúmeros acidentes de trânsito semelhantes ao longo de um ano.

### 2.1.1.3 Origem

Em relação à origem ou ao agente causador, um desastre pode ser classificado como “natural” ou “humano” (TOMINAGA, SANTORO e AMARAL, 2012). O desastre natural é aquele causado por fenômenos e desequilíbrios da natureza em grandes proporções que acontecem independente de ação humana. Estes desastres podem, ou não, ser agravados pela intervenção do homem na natureza. Um desastre humano (ou antropogênico) é aquele resultante diretamente da ação ou omissão do homem em relação às situações.

Existem ainda classificações secundárias, como por exemplo, desastres originados por dinâmica interna e externa da Terra. Terremotos, maremotos e vulcanismo são exemplos de eventos com origem interna, enquanto tempestade, tornados e inundações são exemplos com origem externa.

#### **2.1.1.4 Duração**

Desastres naturais podem ser classificados em “episódicos” e “crônicos” quanto a sua duração (KOBAYAMA *et al.*, 2006). Geralmente os desastres episódicos, como terremotos, enchentes e vulcões chamam mais atenção devido aos grandes impactos causados por estes eventos. Porém, desastres crônicos, como erosão do solo por exemplo, podem causar prejuízos ambientais e inclusive econômicos no longo prazo. Além disso, em alguns casos, estes podem desencadear outros desastres, como deslizamentos de terra, por exemplo.

#### **2.1.2 Gerenciamento de desastres naturais**

A maioria dos desastres naturais é inevitável. Seus impactos, porém, podem ser mitigados através de políticas de prevenção e conscientização da população. Segundo Tominaga, Santoro e Amaral (2012), o gerenciamento de desastres naturais é um dos instrumentos de gestão urbana, que integrado a outras políticas públicas, tem finalidade de reduzir, prevenir e controlar o risco de desastres na sociedade.

O gerenciamento de um desastre pode ser dividido em oito etapas, segundo Cardona (1996):

- **Prevenção:** A melhor forma de combater um desastre é evitar a sua ocorrência. Estudos sobre causas de desastres, obras de proteção e controle, análise das características dos desastres são ótimas ferramentas para a prevenção de deslizamentos, incêndios, inundações, vazamentos de produtos químicos, entre outros. Infelizmente, alguns fenômenos, como terremotos, furacões e erupções vulcânicas, por exemplo, não podem ser prevenidos, e, neste caso, adotam-se as próximas etapas.
- **Mitigação:** Nos casos onde não é possível prevenir os desastres, ou quando os eventos são pouco frequentes que não justifique investimento em prevenção, a etapa de Mitigação é a mais adequada para reduzir o número de vítimas e danos causados. Nestes casos, são tomadas ações que reduzam os impactos causados pelos desastres.
- **Preparação:** Quando as etapas de Prevenção e Mitigação são insuficientes para eliminar a chance de ocorrência do desastre, são necessárias ações preparatórias. Nesta

etapa, preocupa-se em estruturar a resposta para as emergências iminentes, reduzindo a possibilidade de vítimas fatais.

- **Alerta:** Para os desastres em que é possível alertar a sua ocorrência com qualquer antecedência, devem ser realizados investimentos em redes de monitoramento para que seja possível emitir avisos de alerta à população.
- **Resposta:** Todas as etapas demonstradas anteriormente descrevem o gerenciamento pré-desastre. A etapa de Resposta ocorre durante, e imediatamente após, um desastre. Nesta etapa, são executadas as ações planejadas durante a etapa de Preparação e quaisquer ações imediatas que visem salvar o maior número de vidas possíveis, reduzir o sofrimento da população e proteger os bens materiais.
- **Reabilitação:** Uma vez superadas as emergências decorrentes do evento, iniciam-se os esforços de restauração dos serviços básicos. Esta etapa é fundamental para as etapas seguintes de Reconstrução e Desenvolvimento.
- **Reconstrução e Desenvolvimento:** Nesta etapa, iniciam-se a recuperação dos bens materiais, da infraestrutura e das atividades econômicas interrompidas devido ao desastre. Assim, busca-se restaurar as rotinas usuais da região atingida.

### **2.1.3 Terremotos**

Terremotos podem ter impactos devastadores em uma questão de segundos. Segundo Guha-Supir e Voz (2011), sua natureza imprevisível e o potencial de impacto fazem deles um dos desastres mais letais existentes, tomando em média, 27.000 vidas por ano no mundo todo desde 1990. Mais de 1.4 milhões de terremotos acontecem todos os anos ao redor do planeta, uma média de quase 4.000 por dia. Se algum deles atinge centros urbanos com alta densidade populacional ou comunidades onde as construções não são resistentes a terremotos, criam-se potenciais para grandes desastres, principalmente no caso de terremotos maiores.

Na próxima subseção são explicadas as diferenças entre magnitude e intensidade de terremotos, são descritos os eventos desencadeados durante um terremoto, e finalmente, são listados os maiores terremotos registrados.

### 2.1.3.1 Magnitude X Intensidade

Os termos “magnitude” e “intensidade”, dentro da Sismologia, segundo Paulay e Priestley (1992) são utilizados para descrever características distintas de um terremoto. Magnitude é uma grandeza que descreve a energia desprendida durante um evento sísmológico, e normalmente é medida através da escala Richter. Criada por Charles Francis Richter, esta escala foi concebida em 1935 com o objetivo de quantificar a energia de cada terremoto. Porém, ao invés de tentar medir diretamente a fonte de energia do terremoto, Richter decidiu criar uma escala empírica baseada na maior amplitude de onda medida por um sísmógrafo durante um terremoto (WALLACE, 1990). A Equação 1 demonstra o cálculo da magnitude proposto por Richter, onde  $M_L$  representa a magnitude,  $A$  representa a maior amplitude medida no sísmógrafo durante o terremoto e  $A_0$  representa uma função empírica que depende apenas da distância  $\Delta$  entre o sísmógrafo e o epicentro.

$$M_L = \log_{10} A - \log_{10} A_0(\Delta) = \log_{10} \left( \frac{A}{A_0(\Delta)} \right)$$

(Equação 1)

A intensidade do terremoto varia conforme a região, pois é uma medida do impacto sentido, e não da energia liberada. Normalmente, utiliza-se a escala Mercalli Modificada, para medir a intensidade do sismo. Esta escala é baseada na descrição dos danos, como pode ser visto no Quadro 2.

As escalas de magnitude e intensidade não são diretamente relacionadas, pois um terremoto com determinada magnitude pode causar efeitos com intensidades distintas, dependendo, entre outros fatores, do tipo de terreno e da distância da superfície onde o terremoto ocorreu.

**Quadro 2 – Intensidade de terremotos conforme a Escala Mercalli Modificada**

<b>Intensidade</b>	<b>Tremor</b>	<b>Descrição/Dano</b>
<b>I</b>	Não sentido	Não sentido, exceto por alguns, em condições especialmente favoráveis.
<b>II</b>	Fraco	Sentido apenas por poucas pessoas em repouso, especialmente nos andares mais altos de edifícios.

<b>III</b>	Fraco	Bastante perceptível por pessoas em ambientes internos. Muitas pessoas não reconhecem como um terremoto, as vibrações são semelhantes às de um caminhão passando por perto.
<b>IV</b>	Leve	Sentido por muitos em ambientes internos, e por poucos em ambientes externos durante o dia. À noite, alguns são acordados. Paredes estralam, sensação como se um caminhão pesado tivesse atingido o prédio.
<b>V</b>	Moderado	Sentido por praticamente todos, muitos são acordados. Algumas louças e janelas quebram. Objetos instáveis tombam. Relógios de pêndulo podem parar.
<b>VI</b>	Forte	Sentido por todos, muitos se assustam. Alguns móveis pesados se movem, pedaços de gesso ou reboco caem. Dano leve.
<b>VII</b>	Muito forte	Dano insignificante em edificações com bom projeto e bem construídas. Dano leve a moderado em edificações comuns bem construídas. Dano considerável em construções de baixa qualidade ou com projeto mal feito. Algumas chaminés se quebram.
<b>VIII</b>	Severo	Dano leve em edificações especialmente desenhadas. Dano considerável, com colapso parcial em edificações comuns. Grandes danos em edificações de baixa qualidade. Queda de chaminés, colunas, monumentos, muros. Móveis pesados são tombados.
<b>IX</b>	Violento	Dano considerável em edificações especialmente desenhadas. Estruturas bem desenhadas ficam fora de prumo. Grandes danos, com colapso parcial em edificações comuns. Edificações deslocadas das fundações.
<b>X</b>	Extremo	Algumas estruturas de madeira bem construídas são destruídas. A maioria das edificações e estruturas de alvenaria destruídas junto com suas fundações. Trilhos são curvados.

Fonte: (USGS, 1989)

### 2.1.3.2 Efeitos secundários

O maior risco para a população não é o tremor de terra em si, mas sim o colapso de estruturas, rompimentos de linhas de gás e energia, entre outros que podem ser causados pelo tremor de terra. A seguir, são listados os eventos que podem ser desencadeados por terremotos, segundo UWI (2016):

- Colapso de edificações, muros, pontes, queda de móveis e objetos, vidros de janelas e espelhos quebrados: Detritos criados pelo colapso de construções e queda de objetos pesados são a principal ameaça às vidas humanas durante um terremoto. Vidros de janelas de edifícios podem ser estilhaçados também, apresentando um grande risco para os pedestres. Além disso, tremores secundários podem resultar no colapso total de estruturas que foram abaladas pelo tremor principal.
- Riscos de incêndio: Rompimento de linhas de gás, derramamento de líquidos inflamáveis e produtos químicos podem causar incêndios e explosões de grandes proporções. Além disso, também é agravante o rompimento de linhas de abastecimento de água, dificultando o combate aos incêndios.
- Queda de linhas elétricas: Postes e linhas elétricas podem ser derrubados, expondo cabos eletrificados, criando riscos de incêndios e eletrocuções.
- Deslizamentos de terra: Durante um terremoto, grandes rochedos podem se desprender de encostas, e deslizamentos de terra também podem acontecer, soterrando moradores destas regiões.
- Enchentes causadas por rompimento de represa: Nas regiões próximas a barragens e represas, existe o risco de rachaduras ou rompimento das estruturas, causando enchentes relâmpagos.
- *Tsunamis*: Alguns terremotos em regiões costeiras podem causar *tsunamis*, podendo causar destruições maiores que o próprio terremoto em regiões planas.
- Liquefação: Quando sedimentos contendo grandes quantidades de água são submetidos a tremores prolongados, a pressão da água pode fazer com que o sedimento perca toda a força coesiva, e passe a se comportar como um líquido. Neste caso, construções podem afundar ou tombar, e tanques ou outras cavidades podem vir à superfície.



### 2.1.3.3 Maiores Terremotos da História

Em USGS (2016), foi compilada uma lista dos maiores terremotos registrados. Embora a lista seja baseada na magnitude dos terremotos, estes não são os que causaram maiores danos ou vítimas, pois a maioria ocorreu em regiões distantes dos grandes centros. A seguir são listados os cinco maiores eventos:

1. Chile, 22 de maio de 1960: O maior terremoto já registrado atingiu magnitude 9.5, matando aproximadamente 1.655 pessoas e desabrigando mais de 2 milhões. O terremoto gerou ainda *tsunamis*, causando mortes e destruição no Havaí, Japão e Filipinas. É estimado que a zona de ruptura atingiu cerca 1.000Km de comprimento.
2. Grande terremoto do Alaska, 28 de março de 1964: Com magnitude 9.2, foi o maior terremoto a atingir o Alaska. Devido à localização remota, o terremoto sozinho causou apenas 9 mortes. O maior número de vítimas fatais (122) foi causado pelo *tsunami* criado ao longo da costa do Golfo do Alaska, em cidades na costa do Canadá e no Havaí.
3. Costa oeste da Sumatra do Norte, 26 de dezembro de 2004: O terremoto de magnitude 9.1 criou o *tsunami* com maior número de vítimas fatais na história. Mais de 227.000 morreram ou desapareceram, e mais de 1.7 milhões de pessoas foram desabrigadas pelo terremoto e *tsunami* subsequente em 14 países do sudoeste asiático e leste africano.
4. Próximo à costa de Honshu, Japão, 11 de março de 2011: O terremoto de magnitude 9 causou grande destruição no Japão, com mais de 20.000 mortos e desaparecidos. O terremoto também foi marcado pelos danos e destruição de mais de 300.000 edificações, inclusive causando o desastre nuclear de Fukushima.
5. Kamchatka, 04 de novembro de 1952: Este terremoto teve sua magnitude recalculada recentemente, e foi reestabelecido que sua magnitude de fato é 9.0, ao invés do valor de 8.2 originalmente. O terremoto gerou um *tsunami* atingindo o Havaí com danos materiais superiores a US\$800.000,00, porém sem vítimas fatais.

## 2.2 Sistemas Multiagentes – SMAs

A Inteligência Artificial Distribuída (IAD) surgiu no início dos anos 80, e posteriormente tornou-se um domínio de pesquisa bem estabelecido (SICHTMAN,

DEMAZEAU e BOISSIER, 1992). Segundo Prado *et al.* (1993), um sistema de IAD pode ser visto como um grupo de entidades inteligentes, chamadas de agentes, que interagem entre si.

Uma das subáreas da IAD, a de Sistemas Multiagentes, representa uma forma de analisar, projetar e implementar sistemas de *software* complexos, sendo alvo de intensa atividade de investigação e desenvolvimento (RIBEIRO, 2001). Este tipo de tecnologia tem sido utilizado em variadas aplicações como, por exemplo, comércio eletrônico, gerência de redes, interfaces adaptáveis, robótica, resposta ao desastre e sistemas de controle de tráfego.

Este trabalho propõe a utilização de Sistemas Multiagentes no auxílio ao combate de incêndios em terremotos, com a elaboração de uma estratégia baseada em aprendizagem para os agentes realizarem suas ações de forma mais efetiva, por meio de tomada de decisão que resulte na menor área queimada possível.

As próximas seções apresentam uma breve explicação dos conceitos de Agentes e Sistemas Multiagentes.

### 2.2.1 Agentes

Na literatura encontram-se diversas definições para agentes. Para Wooldridge (2002), um agente é um sistema computacional independente capaz de tomar decisões visando satisfazer seus objetivos, sem que o agente seja informado explicitamente em qualquer momento. Ainda, Ferber e Gasser (1991) *apud* Sichman, Demazeau e Boissier (1992) definem que um agente é uma entidade real ou virtual, imersa em um ambiente sobre o qual é capaz de agir, que dispõe de uma capacidade de percepção e de representação parcial deste ambiente, que pode se comunicar com outros agentes e que possui um comportamento autônomo, consequência de suas observações, de seu conhecimento e das suas interações com os outros agentes. Russell e Norvig (2010) definem ainda, que um agente é tudo que percebe o ambiente através de sensores e age sobre este mesmo ambiente através de atuadores.

Apesar de não haver uma definição universalmente aceita de agentes, se observa alguns pontos de concordância entre diversos autores, como por exemplo: autonomia, percepção e interação com o ambiente e com outros agentes.

Na próxima seção serão descritas as principais características de agentes, encontradas na literatura.

### 2.2.1.1 Características

Agentes podem possuir características distintas dependendo do domínio de aplicação. A seguir são demonstradas algumas destas, conforme Wooldridge e Jennings (1995), Franklin e Graesser (1996) e Goodwin (1993):

- **Autonomia:** O agente é capaz de operar independente de intervenção direta de humanos ou de outros agentes, tendo algum tipo de controle sobre suas ações e estado interno.
- **Habilidade Social:** Indica a capacidade de interagir com outros agentes ou humanos através de alguma linguagem pré-definida.
- **Reatividade:** O agente percebe o ambiente onde está inserido e responde rapidamente às mudanças que ocorrem nele.
- **Pró-atividade:** Determina a capacidade do agente de tomar a iniciativa visando seus objetivos, mesmo que não haja mudanças no ambiente ou estímulos externos.
- **Adaptabilidade:** Demonstra a capacidade de aprendizagem baseada nas experiências anteriores.
- **Mobilidade:** O agente pode se transportar de uma máquina para outra.
- **Caráter:** O agente demonstra “personalidade” e/ou estados emocionais de forma convincente.
- **Previsão:** O modelo de funcionamento do ambiente mantido pelo agente é suficientemente preciso para que o agente possa prever como concluir sua tarefa.
- **Interpretação:** Indica a capacidade do agente de interpretar as leituras de seus sensores.
- **Racionalidade:** O agente é capaz de decidir quais comandos ou ações são necessários para o cumprimento de seus objetivos.

### 2.2.1.2 Tipos de Agentes

Segundo Rezende (2003), baseados nas características existentes em agentes, eles podem ser classificados segundo os seguintes eixos:

- **Eixo Cognitivo:** Em relação ao eixo cognitivo, um agente pode ser classificado como Agente Reativo, ou como Agente Cognitivo. Um Agente Reativo age baseado em um modelo de reações aos estímulos provocados pelo ambiente. Um Agente

Cognitivo é mais sofisticado, suas ações são baseadas em um modelo racional de tomada de decisão, podendo também possuir um modelo de representação interna do ambiente e dos outros agentes.

- Eixo de Foco: Em relação ao eixo de foco, um agente pode possuir foco Estrutural e enfatizar a representação física, ou visual, de humanos ou o que quer esteja representando. O agente pode também possuir similaridades comportamentais em relação ao indivíduo que representa, neste caso, tendo foco Comportamental.
- Eixo de Atuação: Um agente pode atuar de forma Isolada ou de forma Social. A forma de atuação é definida pelo fato de os agentes interagirem, ou não, com outros agentes.
- Eixo Ambiental: Em relação ao eixo ambiental, um agente pode atuar em um ambiente *Desktop*, neste caso sendo chamado de Agente de *Desktop*, ou sobre uma rede, neste caso sendo chamado de Agente de *Internet*.

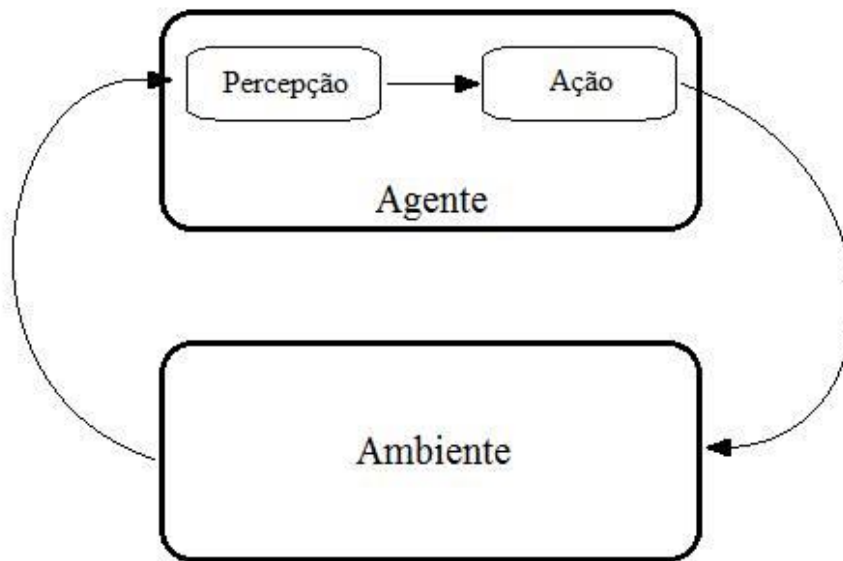
### 2.2.1.3 Arquitetura de Agentes

Segundo Weiss (1999), a arquitetura de um agente, assim como em engenharia de *software*, pode ser dividida em módulos, cada um responsável por uma funcionalidade específica. O conjunto destes módulos, e as interações entre eles, descreve como os dados recebidos do ambiente, aliados ao estado interno do agente, determinam as ações a serem tomadas.

Segundo Wooldridge (2002) e Jennings, Sycara e Wooldridge (1998), existem diferentes tipos de arquitetura que definem o funcionamento interno de um agente. Os autores citam quatro tipos:

- Agentes Reativos: Este tipo de arquitetura baseia-se na ideia de que um comportamento inteligente emerge a partir da interação de diversos comportamentos mais simples. Os agentes com esta arquitetura executam apenas ações como resposta aos estímulos do ambiente, havendo um mapeamento direto entre a percepção e ação tomada, como pode ser observado na Figura 1.

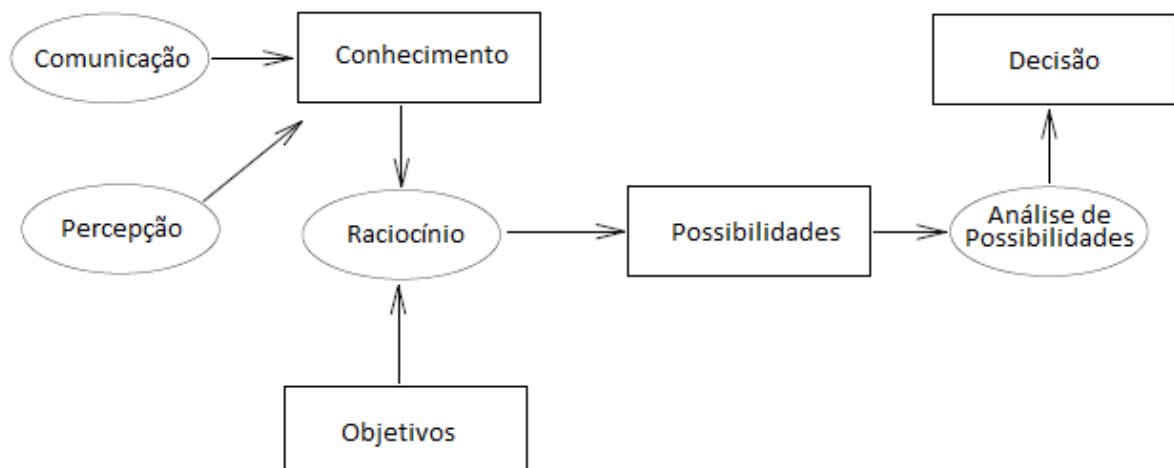
Figura 1 – Modelo básico de um agente reativo



Fonte: (WOOLDRIDGE, 2002)

- **Agentes Deliberativos:** Nesta arquitetura, como pode ser visto na Figura 2, o agente possui explicitamente um modelo simbólico do ambiente (o Conhecimento), baseado nas entradas de dados, que podem ser na forma de Percepção ou de Comunicação com outros agentes. O agente utiliza o Raciocínio para unir os seus Objetivos com o conhecimento adquirido, e planejar as formas de chegar aos seus objetivos (Possibilidades). Finalmente, analisando as possibilidades, o agente chega a uma decisão sobre qual ação tomará (SICHMAN, DEMAZEAU e BOISSIER, 1992).

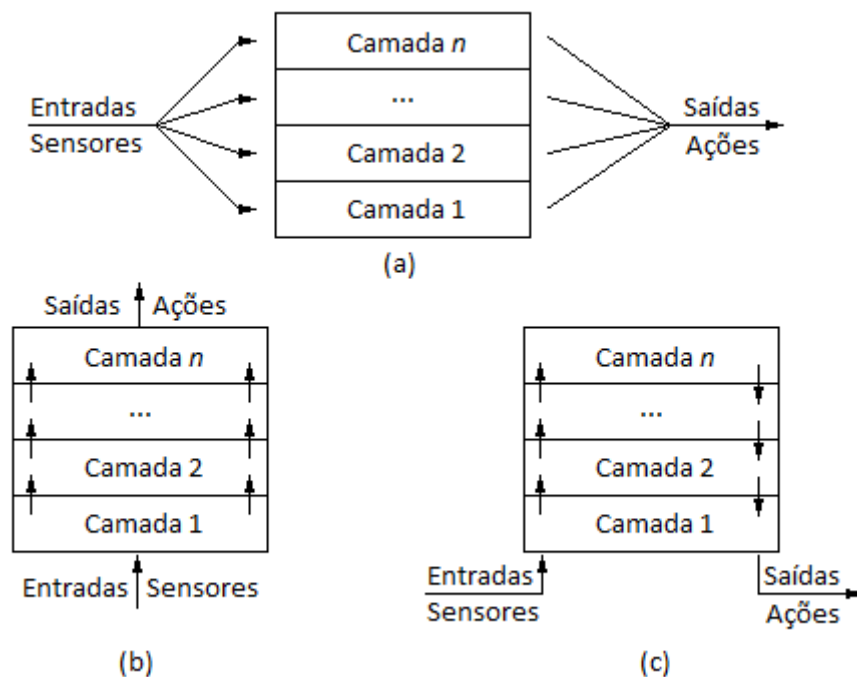
Figura 2 - Modelo básico de um agente cognitivo



Fonte: (SICHMAN, DEMAZEAU e BOISSIER, 1992)

- **Agentes Híbridos:** Nos casos onde é desejável que o agente tenha características reativas e proativas, pode-se utilizar uma abordagem híbrida. O processamento das informações, neste caso, é dividido em camadas contendo arquiteturas tanto reativas quanto racionais. Normalmente há pelo menos duas camadas, uma com comportamento reativo e a outra com comportamento proativo, não havendo restrições sobre a utilização de mais camadas. Na Figura 3(a) pode-se observar um exemplo de arquitetura híbrida horizontal, onde as entradas e as saídas são conectadas diretamente a todas as camadas, de forma que cada camada age como se fosse um agente, sugerindo ações a serem tomadas. Nas Figuras 3(b) e (c), pode-se observar duas formas de arquitetura híbrida vertical, onde as entradas e saídas estão conectadas a no máximo uma camada. Na primeira, os dados são passados de camada para camada até chegar à última, onde são geradas as ações resultantes. Na segunda arquitetura, assim como na primeira, os dados são passados de camada para camada, até chegar à última camada, porém os resultados retornam a camada inicial, onde são geradas as ações. O fluxo de informação nesta segunda arquitetura possui similaridades com uma organização, onde os dados trafegam de baixo para cima, até os níveis mais altos, onde são enviadas as ordens de cima para baixo.

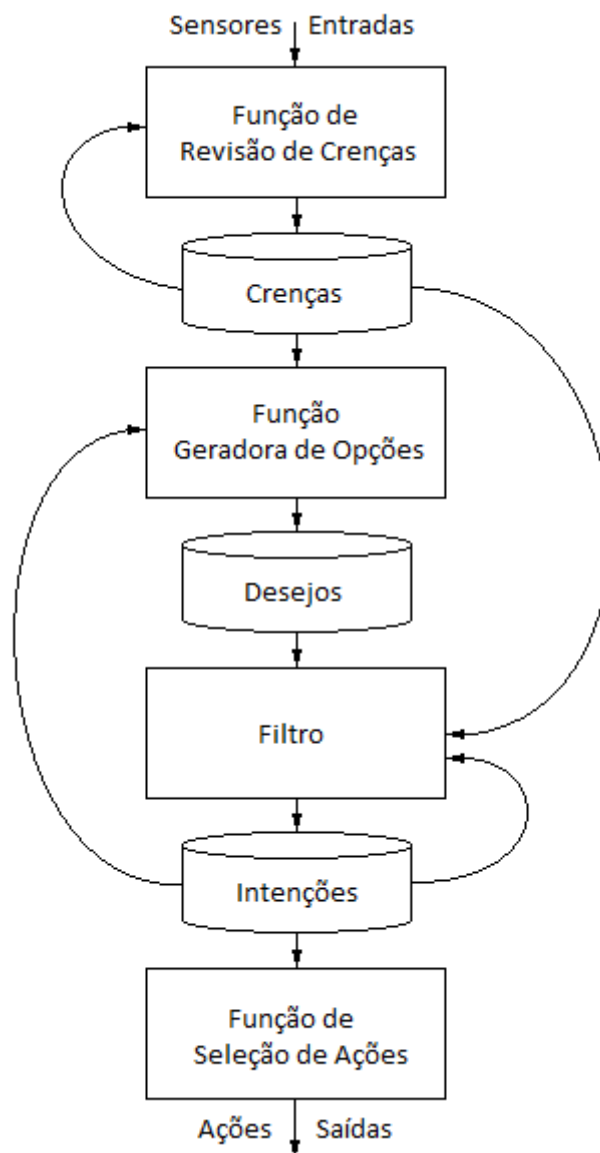
Figura 3 – Modelo de um agente híbrido



Fonte: (WOOLDRIDGE, 2002)

- Agentes *Belief-Desire-Intention* (BDI): São agentes caracterizados por um “estado mental” composto de três componentes, conforme a Figura 4: Crenças, Desejos e Intenções. As crenças representam as informações que o agente possui sobre o ambiente. Os desejos representam os objetivos que o agente almeja dentre as opções disponíveis. As intenções representam os objetivos que o agente tenha escolhido, e que já tiveram recursos dedicados para a sua concretização. O agente atualiza suas crenças constantemente com base nas percepções do ambiente, decidindo quais opções estão disponíveis e filtrando estas opções para determinar suas intenções, e agir de acordo com elas.

Figura 4 – Modelo de um agente BDI



Fonte: (WEISS, 1999)

### 2.2.3 Sistemas multiagentes

Sistemas multiagentes (SMA) são ideais para representar problemas que possuem múltiplos métodos de solução, múltiplas perspectivas e/ou múltiplas entidades solucionadoras de problemas. Tais sistemas possuem as vantagens tradicionais de solução de problemas distribuídos e concorrentes, mas também com a vantagem de padrões sofisticados de interação (JENNINGS, SYCARA e WOOLDRIDGE, 1998).

Segundo Rezende (2003), os problemas abordados em SMA são atividades de um conjunto de agentes autônomos em um universo multiagentes. Autônomos quer dizer, neste contexto, que os agentes têm uma existência própria e possuem objetivos próprios a atingir. Como não existe um problema prévio a ser solucionado, o foco de pesquisa reside nos modelos para conceber agentes, suas organizações e interações de modo genérico, para que possam ser instanciadas num caso particular quando um determinado problema é colocado para a sociedade de agentes e esta deve resolvê-lo. Ou seja, se concebe os meios através dos quais é possível assegurar que tais agentes irão cooperar uns com os outros para resolver um problema que seja colocado à sociedade.

#### 2.2.3.1 Tipos de interação entre agentes

Segundo Huhns e Stephens (1999) os tipos de interação entre agentes podem ser divididos em coordenação, cooperação, negociação e conflito.

- **Coordenação:** As ações de múltiplos agentes precisam ser coordenadas, porque existem dependências entre as ações dos mesmos. Existe uma necessidade de se adequar a restrições globais, e nenhum agente sozinho possui a competência, recursos, ou informações para atingir os objetivos do sistema (HUHNS e STEPHENS, 1999). Um exemplo de coordenação pode ser a troca de informações sobre quais ações cada agente está executando, para garantir que não exista redundância na resolução de problemas.
- **Cooperação:** Uma estratégia básica compartilhada por muitos protocolos de cooperação é decompor e distribuir as tarefas. Esta estratégia de “Dividir para conquistar” pode reduzir a complexidade do objetivo a ser cumprido, uma vez que subtarefas menores necessitam agentes mais simples e com menos recursos para resolvê-las (HUHNS e STEPHENS, 1999). No entanto, deve-se ter cautela, pois há



possibilidade de existirem interações entre as subtarefas e conflitos entre os agentes. A decomposição de tarefas pode ser realizada pelo projetista do sistema, de forma que a divisão do trabalho é programada durante a fase de implementação. Esta decomposição pode ser realizada espacialmente, baseada no *layout* das fontes de informação no ambiente ou pontos de decisão.

- **Negociação:** É um processo pelo qual dois ou mais agentes tomam uma decisão em conjunto, cada um tentando atingir seus próprios objetivos. Inicialmente, os agentes comunicam seus interesses, que podem conflitar, e então tentam chegar a um acordo através de concessões ou buscando alternativas (HUHNS e STEPHENS, 1999). Os principais aspectos a serem considerados em uma negociação são a linguagem utilizada pelos agentes, o protocolo seguido por cada agente conforme eles negociam, e os processos de tomada de decisão utilizados pelos agentes para determinar suas posições, concessões e critérios para acordos.
- **Conflito:** A interação entre agentes com objetivos distintos pode iniciar conflitos, portanto, a resolução de conflitos é essencial para o comportamento coordenado entre agentes. Não existe uma única estratégia adequada para todos os ambientes. Barber *et al.* (1998) cita algumas estratégias que podem ser utilizadas dependendo da situação, como, por exemplo votação, convenção de prioridades, árbitros, relaxamento de restrições, entre outros.

Mas também, se observa nos SMAs outras formas de interação, como: Comunicação (troca de informação entre agentes), Colaboração (agentes auxiliam outros com objetivos diferentes, como uma troca de favores), Organização (estrutura dos agentes no ambiente multiagentes), Planejamento (definição de planos em conjunto), Interferência Social (ações de um agente interferindo de forma positiva ou negativa nas ações dos outros agentes no ambiente).

### **2.3 Aprendizagem em Sistemas Multiagentes**

Um ponto chave do comportamento inteligente é a capacidade de aprender, e devido a isto, muitos esforços foram concentrados neste assunto, nas disciplinas que estudam inteligência e racionalidade (SHOHAM e LEYTON-BROWN, 2010).

Inteligência implica certo nível de autonomia, que, por sua vez, depende da habilidade de realizar decisões independentemente. Assim, agentes necessitam de ferramentas apropriadas para tomar tais decisões. Na maioria dos domínios dinâmicos, é impossível para o

projetista prever todas as situações nas quais o agente pode se encontrar. Portanto, o agente precisa ser capaz de se adaptar a novos ambientes. Conseqüentemente, o aprendizado é uma parte crucial da autonomia e, assim, deve ser foco de pesquisas na área de agentes e multiagentes (ALONSO *et al.*, 2001).

O problema de aprendizagem pode ser formulado como o aprendizado de uma função *estado-ação* ou *estado-ação-valor* que representa a melhor ação que um agente deveria tomar, baseado em um histórico de recompensas e nas ações de outros agentes do sistema (PARKES e UNGAR, 1997). Os autores afirmam ainda que o estado de um ambiente deve modelar todas as informações que o agente utiliza para adaptar a sua estratégia.

De certa forma, o aprendizado de agentes em SMA pode ser visto como um subdomínio do aprendizado de máquina. Pesquisas em SMA com este tipo de abordagem, geralmente aplicam algoritmos de aprendizado focados em agentes isolados e tratam outros agentes como parte do ambiente. Como resultado nestas pesquisas, o aprendizado multiagentes é visto como uma propriedade emergente (ALONSO *et al.*, 2001).

Existem duas categorizações principais relativas à forma do aprendizado em SMA: Aprendizado Centralizado e Aprendizado Descentralizado (WEISS, 1999). Em um mesmo sistema, podem coexistir agentes aprendendo isoladamente (centralizado) e grupos de agentes interagindo para realizar o aprendizado (descentralizado). O autor afirma ainda que a centralização e descentralização podem ser consideradas duas representações de aprendizado em SMA, que por sua vez se estende por diversos aspectos, como os listados a seguir:

- Grau de descentralização: É representado pela distribuição e paralelismo presente no aprendizado. Um dos extremos deste aspecto pode ser representado por um agente que realiza sozinho todas as atividades de aprendizado de forma sequencial, e o outro extremo é um sistema no qual as tarefas de aprendizado são executadas por todos os agentes de forma distribuída e paralela.
- Características de interação: Conforme Weiss (1999) essas características são: nível de interação (varia desde observações simples, até negociações e diálogos com complexas trocas de informação); duração da interação (curta ou longa); frequência de interações (baixa ou alta); padronização de interação (varia desde completamente desestruturada até estritamente hierárquica); variação das interações (imutável ou mutável).
- Características de envolvimento: O envolvimento pode ser dividido em duas características, relevância e papel no aprendizado. Em relação à relevância, o envolvimento de um agente pode ser necessário para que o objetivo do sistema seja

atingido, ou pode ser que não haja a necessidade de envolvimento. Quanto ao papel executado no aprendizado, um agente pode ser genérico (tendo a capacidade de desempenhar qualquer papel) ou pode ser especialista (com capacidade para executar apenas um tipo de tarefa).

- Características relacionadas aos objetivos: Sobre os objetivos, o autor divide também em duas características: o tipo de melhoramento que se espera obter com o aprendizado, e a compatibilidade dos objetivos dos agentes. Sobre o tipo de melhoramento, pode-se buscar melhorar as habilidades de um único agente, ou pode-se buscar melhorar o trabalho em equipe de múltiplos agentes (através de melhor comunicação, negociação ou coordenação, por exemplo). Em relação à compatibilidade, os objetivos podem ser conflituosos ou podem ser complementares.

O aspecto do método de aprendizagem, conforme descrito por Michalski, Carbonell e Mitchell (1983), pode ser separado nas seguintes categorias, iniciando pela mais simples e avançando até a mais complexa:

- Aprendizagem direta: Neste tipo de aprendizagem o conhecimento é programado diretamente, e não há inferência em relação às informações que são apresentadas ao agente.
- Aprendizagem por instrução: Neste tipo de aprendizagem surge a figura do professor, responsável por apresentar as informações para o sistema de maneira que o mesmo possa assimilá-las. É necessário algum nível de inferência por parte do sistema que realiza o aprendizado, mas o maior ônus ainda é do professor que precisa organizar o conteúdo de maneira apresentável.
- Aprendizagem por analogia: Através de analogias, um sistema pode utilizar experiências anteriores, semelhantes às que estão sendo apresentadas, para assimilar o novo conhecimento. Um motorista, por exemplo, que nunca dirigiu caminhões mas sabe dirigir automóveis comuns, pode adaptar e transformar os conhecimentos que já tinha à nova situação apresentada. Este tipo de aprendizagem exige um grau maior de inferência em relação à aprendizagem por instrução e direta.
- Aprendizagem a partir de exemplos: Nesta categoria de aprendizagem, são apresentados ao sistema apenas exemplos e contraexemplos do conhecimento que deve ser assimilado, e fica a cargo do sistema inferir sobre hipóteses que expliquem os exemplos de forma consistente. Técnicas de aprendizagem por reforço e aprendizagem supervisionada normalmente adotam este tipo de aprendizagem.

- Aprendizagem por observação: Fica a cargo do sistema identificar qualquer padrão nos dados apresentados a ele, sem qualquer interferência externa (por professor ou ambiente). Pode ser dividida em duas subcategorias, aprendizagem por observação passiva ou por experimentação ativa. Na primeira, o sistema não interfere no ambiente para realizar o aprendizado, enquanto na segunda o sistema pode criar perturbações no ambiente para observar os resultados de suas perturbações. O aprendizado não-supervisionado, utiliza este tipo de aprendizagem.

Quanto à forma de realimentação de dados que o sistema, ou agente, recebe em relação ao aprendizado que executa, o processo de aprendizado pode ser classificado, de três maneiras segundo Weiss (1999):

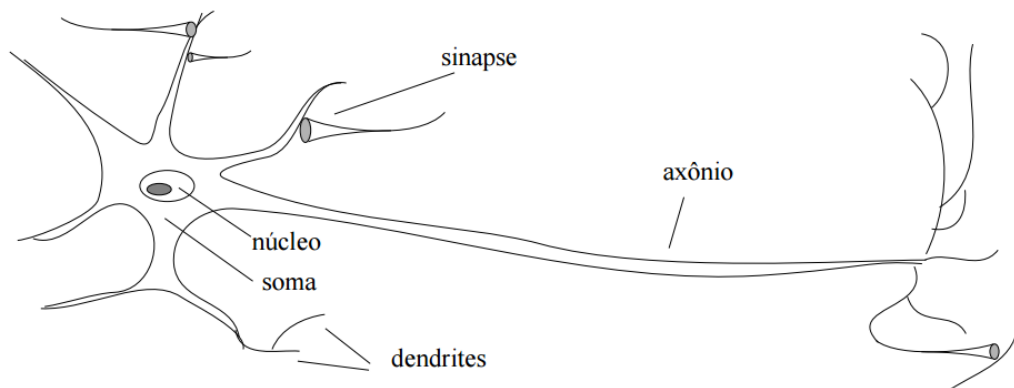
- Aprendizado supervisionado: Um “professor” fornece as entradas e saídas esperadas para o sistema, e o objetivo é criar uma política genérica que ligue as entradas às saídas, de forma que em situações semelhantes, o sistema será capaz de gerar as saídas desejadas.
- Aprendizado não-supervisionado: São fornecidas para o sistema apenas as entradas, deixando a cargo do sistema procurar por padrões ou estruturas nos dados recebidos.
- Aprendizado por reforço: Neste tipo de aprendizado, o sistema interage com um ambiente dinâmico, tentando descobrir as ações necessárias para atingir objetivos (que podem ser representados através de recompensas).

### **2.3.1 Redes Neurais Artificiais**

Muitas técnicas de inteligência artificial buscam inspiração na natureza, como é o caso dos Algoritmos Evolutivos, Otimização baseada em colônia de formigas, Otimização baseada em enxame de partículas, Aprendizado por Reforço, entre outros. Da mesma forma, as Redes Neurais Artificiais (RNA) são inspiradas no funcionamento dos neurônios do cérebro humano.

O modelo genérico de um neurônio pode ser visualizado na Figura 5. Cada neurônio possui um corpo, ou soma, onde fica o núcleo da célula. Ligado ao corpo, estão os dendritos e o axônio. Os dendritos recebem estímulos de outros neurônios através de sinapses e repassam estes estímulos para o axônio. Os estímulos recebidos pelos dendritos chegam ao axônio, e caso o nível de estímulo ultrapasse um certo nível, o axônio transmite este sinal aos outros neurônios conectados às extremidades do axônio.

Figura 5 – Neurônio biológico.

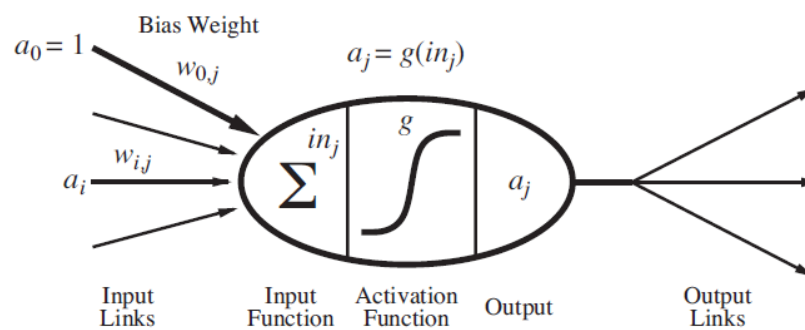


Fonte: (RAUBER, 1997)

A partir deste modelo simplista do neurônio, muitos pesquisadores passaram a replicar este comportamento através de modelos matemáticos. Segundo Russel e Norvig (2010), o primeiro modelo matemático de um neurônio (Figura 6) foi proposto por McCulloch e Pitts em 1943. Este modelo de neurônio é utilizado até hoje no estudo de Redes Neurais Artificiais.

O neurônio é conectado a diversas entradas  $a_i$ , cada uma com um peso  $w_i$  associado a elas. O somatório do produto das entradas e pesos  $a_i w_i$  é repassado para a função de ativação  $g$  que determina a saída  $a_j$  do neurônio.

Figura 6 – Modelo matemático do neurônio de McCulloch e Pitts.



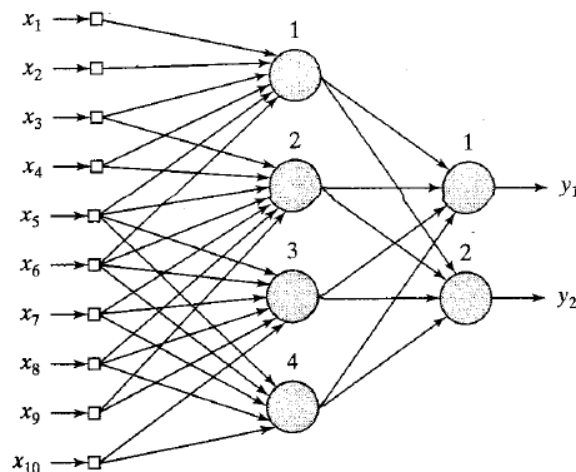
Fonte: (RUSSEL e NORVIG, 2010)

A função de ativação, segundo Rauber (1997), pode assumir muitas formas, porém as três mais utilizadas são descritas a seguir:

- Função linear: Esta é a função de ativação mais simples, pois a saída  $y$  é apenas uma função linear na forma  $f(x) = ax$ .
- Função escada: Esta função, mais tradicional, gera uma saída binária com valor “1” apenas caso o valor da entrada ultrapasse um certo limite  $a$ . Assim, esta função assume a forma  $f(x) = 1, se x \geq a; f(x) = 0, se x < a$ .
- Função sigmoide: Esta função é semelhante à função escada, porém possui uma transição suave entre os valores 0 e 1. A função sigmoide assume a forma  $f(x) = \frac{1}{1+e^{-x}}$  e é uma das mais utilizadas em redes com retro propagação de erro, por ter como característica o fato de ser derivável.

Uma RNA é formada quando múltiplos neurônios são conectados, como pode ser observado na Figura 7. Os neurônios são conectados em camadas, e uma RNA deve possuir no mínimo 2 camadas, uma para entrada de dados e outra para saída. Porém uma rede pode possuir qualquer número de camadas entre as de entrada e saída, sendo estas chamadas de camadas escondidas. A RNA de exemplo na Figura 7 possui uma camada de entrada com 10 neurônios, uma camada oculta com 4 neurônios, e uma camada de saída com 2 neurônios. Quanto maior o número de camadas utilizadas, e também o número de neurônios por camada, maior é a capacidade de aprendizado da rede.

Figura 7 – Exemplo de uma RNA com 3 camadas.



Fonte: (HAYKIN, 1999)

Em cada camada intermediária e na camada de entrada é adicionado um neurônio extra, chamado de *bias*, ou viés, com valor de entrada fixo em 1. Segundo Haykin (1999), o *bias* serve meramente para deslocar o limiar de decisão em relação à origem.

Durante a fase de aprendizado, são apresentados exemplos de entradas e saídas esperadas para a rede, e os pesos da rede são ajustados para que as saídas estejam adequadas às esperadas (HAYKIN, 1999). O ajuste necessário em cada peso é baseado no somatório das diferenças entre a saída esperada e a saída de fato da rede, ou seja, o somatório dos erros. Em uma rede com apenas duas camadas (uma de saída e outra de entrada), o ajuste dos pesos é simples. Porém, em redes com três ou mais camadas, além de calcular o erro entre a saída da rede e a camada anterior, é necessário determinar o erro entre as camadas intermediárias e suas respectivas camadas anteriores. Para isso, Russel e Norvig (2010) descrevem o algoritmo de *error backpropagation*, ou retropropagação de erro, cujo objetivo é calcular o erro das camadas intermediárias a partir do erro calculado na saída da rede neural. Considerando a existência de duas camadas  $j$  e  $k$ , onde  $j$  é a camada anterior à camada  $k$ , o cálculo do erro na camada  $j$   $\Delta_j$  é dado pela Equação 2, onde  $g'(in_j)$  é a derivada da função de ativação na camada  $j$ ,  $w_{j,k}$  é o peso entre os neurônios das camadas  $j$  e  $k$ , e  $\Delta_k$  é o erro calculado na camada  $k$ . O algoritmo recebe o nome de retropropagação de erro, pois inicialmente o erro é calculado na camada de saída e depois é retropropagado camada após camada (utilizando a Equação 2) até a primeira camada intermediária.

$$\Delta_j = g'(in_j) \sum_k w_{j,k} \Delta_k$$

(Equação 2)

### 2.3.2 Clusterização *K-means*

A prática de classificar objetos de acordo com suas similaridades é a base de grande parte dos trabalhos científicos, e além disso, a organização dos dados em grupos é uma das formas fundamentais de entendimento e aprendizagem (JAIN e DUBES, 1988). As técnicas de clusterização têm como objetivo buscar padrões em dados desordenados, classificando e organizando os objetos de acordo com suas semelhanças, distância ou densidade em um espaço multidimensional (JAIN e DUBES, 1988). Segundo Sanches (2003), um dos algoritmos mais frequentemente utilizados para clusterização é o *K-means*, proposto por James MacQueen em 1967, onde o usuário/especialista é responsável por definir o número de partições, ou *clusters*, a serem encontradas.

O algoritmo *K-means* inicialmente seleciona aleatoriamente  $k$  centroides, que são os pontos considerados o centro de cada *cluster*. A seguir, a lista de todos os objetos é iterada, e cada objeto passa a pertencer ao *cluster* cujo centroide é o mais próximo de si. Depois que todos os objetos são atribuídos aos *clusters*, o centro de cada cluster é recalculado através do cálculo do ponto médio entre todos os objetos do *cluster*. Assim, tem-se um novo centroide para cada *cluster* e o processo de iterar por todos os objetos é reiniciado. Este processo se repete até que os centroides não mudem mais de posição ou até que algum limite de iterações pré-estabelecido seja atingido.

## 2.4 Trabalhos Relacionados de Sistemas Multiagentes para Desastres Naturais

Nesta seção, é apresentada uma análise comparativa das principais características de alguns trabalhos selecionados, dedicados ao estudo de Sistemas Multiagentes em combate a desastres naturais.

### 2.4.1 A partitioning-based task allocation strategy for Police Multi-Agents

O trabalho de Liang, Yang e Deng (2014) propôs a utilização da técnica de clusterização com o algoritmo *K-means* para dividir áreas do mapa (relativo aos locais atingidos) entre grupos de agentes do tipo *Police Force* na *Robocup Rescue*<sup>1</sup>. Devido às alterações que podem ocorrer ao longo da simulação, foi desenvolvido um algoritmo que permite a troca temporária ou permanente dos agentes para outra área. Os algoritmos propostos foram aplicados na solução da equipe *Apollo-rescue* para a *Robocup Rescue Simulation System*. O foco de aplicação é na área de recuperação de terremotos, mais especificamente na liberação de ruas bloqueadas, para que as equipes de bombeiros e ambulâncias possam trabalhar com maior eficiência. As abordagens propostas no trabalho foram testadas na plataforma *server-v1.1* da *Robocup Rescue*, disponibilizada junto ao repositório de códigos da mesma. A forma de avaliação adotada foi a execução da solução 10 vezes em cada um dos 5 mapas adotados, calculando a pontuação média em cada mapa. A pontuação utilizada é a do simulador, e considera a quantidade de sobreviventes (com o percentual de pontos de vida restantes) e o percentual de área construída que não foram consumidas pelas chamas. Os mapas têm características distintas, como tamanho, número de edifícios, número de civis, entre outras. A repetição das execuções foi necessária para mitigar

---

<sup>1</sup> *ROBOCUP RESCUE*: <http://www.robocuprescue.org/>



os efeitos da aleatoriedade na performance dos agentes. Os resultados atingidos foram satisfatórios, segundo os autores. Para análise, foram comparadas as médias das pontuações em cada mapa com as pontuações da mesma equipe utilizando os algoritmos adotados no ano anterior da competição. Em todos os mapas houve melhoria no desempenho da equipe. Além disso, a equipe ficou em 3º lugar no ano em que participaram da competição com este trabalho.

#### **2.4.2 Coordination of Agents in the RoboCup Rescue: A Partial Global approach**

O objetivo do trabalho, desenvolvido por Pereira, Nardin e Sichman (2011), é propor uma abordagem de coordenação dos agentes da *Robocup Rescue* de maneira que o maior número de vítimas seja salva e o menor número de construções seja consumida por incêndios. O trabalho propõe uma abordagem de alocação de tarefas baseada em Planejamento Global Parcial para coordenar agentes na *Robocup Rescue*. Esta abordagem considera a existência de informações locais e globais, sendo que os agentes armazenam as informações locais e os centros (agentes coordenadores) mantêm uma visão consolidada de todos os agentes. O trabalho realiza um estudo sobre o simulador da *Robocup Rescue*, descrevendo as propriedades e comportamentos de cada tipo de objeto, capacidades de cada tipo de agente, e, finalmente, os tipos de mensagens trocadas entre os agentes. Para avaliar o algoritmo proposto, foram executadas simulações em 4 mapas: *VC*, *Foligno*, *Kobe*, *Kobe4*. Os resultados obtidos foram comparados aos agentes de exemplo da *Robocup Rescue*. Em todos os mapas foram utilizados os mesmos parâmetros para ambos os algoritmos. Cada algoritmo foi executado 30 vezes em cada mapa e a forma de avaliação foi a pontuação do simulador após 300 ciclos. Esta pontuação considera a quantidade de sobreviventes (com o percentual de pontos de vida restantes) e o percentual de área construída que não foram consumidas pelas chamas. Para comparar os resultados, foi utilizado um teste estatístico, sendo que o algoritmo proposto foi superior ao utilizado nos algoritmos de exemplo da *Robocup Rescue*, em 3 dos 4 mapas, considerando a pontuação descrita anteriormente. Como trabalhos futuros, o autor sugere melhorar a estratégia proposta, distribuindo os agentes no mapa conforme o número de civis e focos de incêndio em cada região. Além disso, o autor também pretende implementar agentes capazes de propagar informações e resgatar vítimas de forma totalmente descentralizada, independente dos centros (agentes coordenadores).

### 2.4.3 Disaster Management in Real Time Simulation Using Machine Learning

O trabalho, apresentado em Khouj *et al.* (2011), tem como objetivo criar e treinar previamente um agente capaz de agir em cenários de emergência, medindo a capacidade operacional e disponibilidade de recursos nas infraestruturas mais críticas e redirecionando os recursos. O foco de aplicação do trabalho está em gerenciar os recursos disponíveis em uma situação de desastre, a fim de aumentar o número de vítimas do desastre que receberão alta nos hospitais ou postos de atendimento. A técnica desenvolvida se baseia em um assistente de tomada de decisão em tempo real, chamado de *Decision Assistant Agent in Real Time Simulation* (DAARTS), que controla quatro tipos de infraestruturas: hospitais, estações de energia elétrica, estações de bombeamento de água, e pontos de encontro. O agente avalia a condição física e quantidade de recursos disponíveis em cada estrutura para redirecionar estes recursos de forma que o maior número de pacientes receba alta nos hospitais. Para escolher a forma de distribuição dos recursos, é utilizado um algoritmo de *Q-Learning*, que é um algoritmo de aprendizagem. O agente proposto foi implementado na plataforma *I2Sim* do *MATLAB*. Todas as simulações realizadas também foram executadas no *I2Sim*. Para avaliar o desempenho do algoritmo foram executadas mais de 500 simulações com diferentes valores para o coeficiente de aprendizagem, coeficiente de desconto e coeficiente de exploração (parâmetros usados pelo algoritmo de aprendizado *Q-Learning*). Depois de realizadas as simulações, o agente DAARTS foi capaz de identificar um conjunto de parâmetros que resulta na alta de 1.512 pacientes. Para critério de comparação, um operador humano inexperiente foi capaz de obter 762 altas distribuindo os recursos manualmente. Em trabalhos futuros, os autores sugerem endereçar cenários mais realistas, com maior número de elementos de infraestrutura. Nestes cenários, a tabela de consultas do *Q-Learning* cresce rapidamente, de forma que o algoritmo proposto não é computacionalmente viável. Como solução, a proposta é utilizar um algoritmo com redes neurais artificiais para obter melhorias.

### 2.4.4 HAC-ER: A Disaster Response System based on Human-Agent Collectives

O trabalho proposto por Ramchurn *et al.* (2015) é um sistema que coleta informações sobre desastres através de redes sociais, organizações confiáveis, e veículos aéreos não tripulados (*drones*), e realiza uma avaliação destas informações, a fim de auxiliar humanos, com foco de aplicação em recuperação de áreas atingidas por desastres. É demonstrada, então, a integração de agentes e humanos em equipes chamadas de *Human-Agent-Collectives*

(HAC). Ao acontecer um desastre, informações são coletadas em redes sociais e organizações confiáveis, para determinar os locais que precisam de mais atenção. Para realizar a análise destas informações, são utilizadas duas técnicas de aprendizagem de máquina: Processo Gaussiano e Combinação de classificadores Bayesianos independentes. Depois da análise, é criado um mapa de calor com os locais onde existe a maior probabilidade de desastre. Assim, são lançados *drones* nos locais de maior interesse para que sejam coletadas imagens, a fim de se averiguar a gravidade da situação. O controle dos *drones* pode ser transferido entre o comandante e os seus subalternos, através das interfaces desenvolvidas. Baseado nas informações disponíveis, um agente coordenador, utilizando um Processo de Decisão de Markov baseado em sistemas multiagentes, aliado a um coordenador humano, cria uma estratégia de distribuição de equipes de primeiros socorros. O agente coordenador constantemente reavalia a situação no ambiente para detectar novos eventos e realocar as equipes de resgate. O coordenador humano pode realizar alterações nas estratégias propostas pelo agente coordenador, a fim de adequar as mesmas à realidade da sua equipe. Para lidar com a possibilidade das informações coletadas estarem incorretas, os autores desenvolveram um subsistema que controla o fluxo de informações e tarefas dentro do sistema. Desta forma, caso um conjunto de tarefas seja alocado, baseado em uma informação errada, todas as equipes alocadas recebem notificações assim que o erro é percebido. Assim, pode-se realocar as equipes para os locais onde realmente existem problemas. O sistema concebido foi apresentado a mais de 100 usuários, sendo validado com equipes de resgate reais. Os testes realizados demonstram que a ferramenta possui melhor desempenho quando os usuários possuem maior controle sobre a autonomia. Nos trabalhos futuros, os autores sugerem aplicar o sistema em exercícios de treinamento em resposta a desastres, além de avaliar os mecanismos de interação em situações *in loco*.

#### **2.4.5 Human-agent collaboration for disaster response**

A pesquisa apresentada em Ramchurn *et al.* (2016) analisa um ambiente de desastre como incerto e dinâmico, já que novas tarefas e perigos podem aparecer no local. O trabalho desenvolvido refere-se ao uso de técnicas de decisão para solucionar problemas de alocação de tarefas por meio de planejamento baseado em agentes, no qual os humanos são guiados pelos agentes na execução das tarefas. Os primeiros a chegarem ao local do desastre são os médicos, as equipes de segurança e de resgate, a fim de salvar vidas; são pessoas com diferentes habilidades e experiências. As dificuldades ocorrem no sentido de determinar a

melhor sequência de tarefas a serem executadas e como estas tarefas serão alocadas. Para auxiliar nestas situações, há algoritmos e mecanismos de coordenação de equipes desenvolvidos, mas que não consideram a questão de incertezas no ambiente nem as habilidades das pessoas das equipes. Por isso, a importância de realizar simulações de desastres com soluções que envolvam a interação entre humanos e agentes. Os autores definiram um cenário de simulação para avaliar o planejamento baseado em agentes que auxilia a coordenação das equipes de resgate. Este cenário refere-se a um desastre no qual um satélite, que utiliza combustível radioativo, cai em uma zona urbana, causando danos materiais e atingindo as pessoas; e as partículas radioativas podem contaminar alimentos e pessoas. O algoritmo desenvolvido detecta as incertezas no cenário, referentes ao ambiente (como focos de incêndio, inundação, motins que podem acontecer), realiza a alocação de tarefas, de modo que o tempo de conclusão da tarefa diminua, e planeja rotas para livrar as equipes da exposição à radiação. O problema de coordenação é representado pelo processo de decisão Markoviano multiagente, a fim de contribuir na compreensão de como humanos e agentes podem colaborar no planejamento das tarefas em caso de desastres. O cenário foi implementado no formato de um jogo e contou com 24 participantes (17 homens e 7 mulheres) para avaliação. Os participantes sentiram a necessidade de aperfeiçoar o plano de alocação de tarefas, para que as pessoas da simulação não fossem encobertas pela nuvem radioativa. Os autores sugerem que para efetivar a colaboração coletiva entre humanos e agentes, é preciso adaptabilidade (por exemplo, em relação à rejeição de tarefas atribuídas, novas restrições do ambiente, novas tarefas que surgem); interação (troca de mensagens simples entre agentes e humanos, com atribuição direta de tarefas); autonomia (em relação à escolha dos planos e replanejamento).

#### **2.4.6 RoboCup Rescue 2013 LTI Agent Rescue Team Description**

No trabalho de Barroso *et al.* (2013), são demonstradas as estratégias da equipe *LTI Agent Rescue* para a competição *Robocup Rescue* de 2013. O objetivo do trabalho é resolver os principais problemas de exploração do mapa e coordenação de agentes nas estratégias adotadas durante a competição de 2011. A coordenação dos agentes adotada é completamente distribuída, não havendo utilização dos agentes centrais, para evitar o ponto único de falha e atraso na propagação de informações, decorrentes da utilização deste tipo de agente. Os agentes trocam informações através de canais de rádio, e a partir destas informações e das observações realizadas, constroem um modelo do mundo (contendo o estado de cada objeto

no mapa, como por exemplo posição de agentes e prédios consumidos por chamadas) e populam uma base de tarefas (incluindo também quais agentes estão trabalhando em cada tarefa). Para evitar o congestionamento dos canais de rádio, todas as mensagens são comprimidas através da codificação adaptativa de Huffman. Caso os canais de rádio fiquem congestionados, os agentes passam a transmitir de forma intercalada, de forma que em ciclos pares, apenas os agentes com *ID* par transmitem, e vice-versa. Para a exploração, os autores adotam uma estratégia de particionamento simples, sem levar em consideração a distribuição dos prédios, dividindo o mapa em *n* setores iguais, onde *n* é o número de agentes policiais. São descritos também melhoramentos nas estratégias dos três tipos de agentes móveis existentes no simulador da *Robocup Rescue*: Policial, Ambulância e Bombeiro. A cada policial é atribuído um setor, assim evita-se que múltiplos policiais trabalhem no mesmo bloqueio (um bloqueio não é removido mais rapidamente se houverem mais policiais trabalhando nele). No início da simulação, cada policial se move até o refúgio mais próximo, liberando os bloqueios encontrados no caminho, e depois movem-se até o seu setor e liberam os bloqueios do setor. A ambulância verifica sua base de tarefas e começa a executar a tarefa com o melhor retorno esperado, dando preferência para socorrer vítimas mais próximas e com maior vida restante. Em relação aos bombeiros, os autores destacam que estes precisam agir rapidamente, assim que os focos de incêndio surgem, de forma que a estratégia de exploração dos policiais é fundamental para realizar a descoberta de novos incêndios. Para a escolha de qual incêndio o bombeiro deve apagar, escolhe-se o incêndio com a maior recompensa esperada, com base na Equação 3.

$$\left( \frac{1}{\text{areaPredio}} * \text{intensidadeIncendio} \right) + \frac{1}{\sqrt{\text{distancia}}} + \text{numeroVizinhosSemIncendio}$$

(Equação 3)

Os resultados obtidos pela técnica de exploração demonstraram ligeira melhora em relação ao trabalho anterior após testes realizados nos mapas *Berlin*, *Istanbul*, *Kobe*, *Paris* e *VC*. Os mapas *VC* e *Berlim* demonstraram ser mais adequados ao particionamento com setores retangulares. Em trabalhos futuros, os autores sugerem que o particionamento seja realizado com base nos refúgios, que os policiais realizem a liberação dos caminhos das ambulâncias de forma preventiva (com base nas tarefas das ambulâncias), e realizar melhorias no cálculo de recompensa da ambulância.

### 2.4.7 Quadro Comparativo dos SMAs para Desastres Naturais

Esta seção apresenta um quadro comparativo (Quadro 3) dos trabalhos descritos nas seções anteriores.

No Quadro 3, pode-se observar que quase todos os trabalhos implementados para a *Robocup Rescue* utilizaram técnicas de particionamento do mapa para lidar com a exploração do cenário. Isso demonstra que uma abordagem válida para resolver a exploração do problema de resgate é o particionamento do mapa. Dentre os trabalhos que utilizaram o simulador da *Robocup Rescue*, pode-se perceber que o aprendizado não foi utilizado no processo de tomada de decisão das tarefas de resgate. Apesar dos trabalhos que não utilizaram o simulador apresentarem diferentes métodos de aprendizado, os trabalhos de Ramchurn *et al.* (2015) e Ramchurn *et al.* (2016) abrem espaço para o envolvimento de humanos nas tarefas de resgate. No trabalho de Barroso *et al.* (2013) destaca-se como interessante a avaliação dos incêndios no ambiente através de uma heurística, e no trabalho de Liang, Yang e Deng (2014) destaca-se a utilização do método de clusterização com o algoritmo *K-means* para particionar o mapa em regiões de maneira mais igualitária em relação ao método criado por Barroso *et al.* (2013).

**Quadro 3 – Quadro comparativo dos trabalhos relacionados – SMAs em Desastres Naturais**

Trabalhos Relacionados	Objetivo	Ambiente Utilizado	Tipo de Desastre	Técnica de Aprendizado Utilizada	Outras Técnicas	Resultados
(LIANG, YANG e DENG, 2014)	Distribuir Policiais de forma eficiente para desbloquear ruas, e permitir o trabalho de Ambulâncias e Bombeiros.	Simulador <i>Robocup Rescue</i> .	Terremoto.	Clusterização com o algoritmo <i>K-Means</i> .	Não utiliza.	A estratégia trouxe melhorias no desempenho dos agentes policiais.
(PEREIRA, NARDIN e SICHMAN, 2011)	Coordenar agentes de forma que o maior número de vítimas seja salva e o menor número de construções seja consumida por incêndios.	Simulador <i>Robocup Rescue</i> .	Terremoto.	Não utiliza.	Planejamento Global Parcial.	Apresenta resultados superior aos agentes de exemplo da <i>Robocup Rescue</i> em 3 de 4 mapas utilizados

						durante a validação.
(KHOUJ <i>et al.</i> , 2011)	Criar e treinar previamente um agente para gerenciar a distribuição de recursos.	Simulador <i>I2Sim</i> no <i>MatLab</i> .	Qualquer tipo de desastre.	<i>Q-Learning</i> .	Não utiliza.	O algoritmo obteve desempenho superior ao de um operador humano na distribuição de recursos.
(RAMCHURN <i>et al.</i> , 2015)	Auxiliar humanos durante situações de emergência, através de coleta, avaliação e gerência de informações, envio de <i>drones</i> e resgate.	Ambiente real	Qualquer tipo de desastre.	Combinação de classificadores Bayesianos independentes	Processo Gaussiano. Processo de Decisão de Markov.	Foi validado com equipes de resgate reais. O desempenho do sistema é maior quando os usuários possuem maior autonomia sobre as



						decisões tomadas.
(RAMCHURN <i>et al.</i> , 2016)	Solucionar problemas de alocação de tarefas por meio de planejamento baseado em agentes.	Ambiente real.	Qualquer tipo de desastre	Aprendizado baseado na coordenação multiagentes.	Processo de decisão Markoviano multiagente	Foi avaliado por 24 participantes, que sugeriram melhorias no plano de alocação de tarefas.
(BARROSO <i>et al.</i> , 2013)	Resolver os principais problemas de exploração do mapa e coordenação de agentes da equipe <i>LTI Agent Rescue</i> da <i>Robocup Rescue</i> .	Simulador <i>Robocup Rescue</i>	Terremoto	Não utiliza.	Particionamento do mapa. Heurísticas para avaliar bloqueios, vítimas e incêndios.	Houve ligeira melhoria no desempenho da estratégia de exploração em relação à estratégia desenvolvida anteriormente.

## **2.5 Trabalhos Relacionados à Aprendizagem em Sistemas Multiagentes**

Nesta seção, é apresentada uma análise comparativa das principais características de alguns trabalhos selecionados, dedicados ao estudo de Aprendizagem em Sistemas Multiagentes.

### **2.5.1 Utilizando Agentes Autônomos com Aprendizado para Identificação e Combate de Incêndios em Áreas Florestais**

O trabalho de Pessin *et al.* (2007) propõe um Sistema Multiagente heterogêneo com o objetivo de combater incêndios florestais sem o envolvimento de humanos. A equipe é composta por três tipos de agentes: Agente monitor, que detecta os locais onde existem incêndios, baseados em imagens de satélites; Agente de combate, que deve se mover até o incêndio e criar aceiros para combatê-lo; e Agente líder, responsável por repassar as informações do agente monitor para os agentes de combate. Para a comunicação dos agentes é utilizado um sistema de Quadro-negro, onde as mensagens vão do agente monitor para o agente líder, e do agente líder para os agentes de combate. Para evitar a mata mais densa, o agente de combate utiliza uma RNA que indica o ângulo em que o agente deve se mover para chegar até o incêndio. Para validar os agentes propostos, os autores desenvolveram um ambiente de simulação de incêndios florestais.

### **2.5.2 Simulação Virtual da Evolução de Estratégias e do Controle Inteligente em Sistemas Multi-Robóticos**

O trabalho de Pessin, Osório e Musse (2008) tem como objetivo projetar um Sistema Multiagente, formado por robôs controlados, agentes inteligentes que utilizam Redes Neurais Artificiais e Algoritmos Genéticos para determinar a melhor forma de combater incêndios florestais através da técnica de combate indireto por abertura de aceiros. Além disso, o trabalho também contempla a criação de um simulador de incêndios florestais para avaliar o desempenho dos agentes criados. Para determinar a melhor abordagem para combater o incêndio (posicionamento dos aceiros), foi utilizado um Algoritmo Genético que busca minimizar o tamanho do aceiro e também a área total queimada. Para a navegação dos agentes, foi utilizada uma RNA que aprendeu como os motores do robô devem ser acionados baseada na leitura dos sensores disponíveis, a fim de evitar colisões e se mover até o objetivo.

### 2.5.3 Quadro Comparativo

Esta seção apresenta um quadro comparativo (Quadro 4) dos trabalhos descritos nas seções anteriores.

**Quadro 4 – Quadro comparativo dos trabalhos relacionados de Aprendizagem em Sistemas Multiagentes**

Trabalhos Relacionados	Objetivo	Aplicação	Técnica de Aprendizagem Utilizada	Outras técnicas
Pessin <i>et al.</i> (2007)	Combater incêndios florestais através de robôs autônomos.	Sistemas Multiagentes e Combate a incêndios florestais.	Redes Neurais Artificiais (Algoritmo de Aprendizagem <i>Backpropagation</i> ).	Não utiliza.
Pessin, Osório e Musse (2008)	Combater incêndios florestais através de robôs autônomos.	Sistemas Multiagentes e Combate a incêndios florestais.	Redes Neurais Artificiais (Algoritmo de Aprendizagem <i>Backpropagation</i> ).	Algoritmos Genéticos

No Quadro 4, pode-se observar que em ambos os trabalhos a navegação dos agentes foi determinada pelo uso de uma RNA. Porém, no trabalho de Pessin *et al.* (2007) existe um maior nível de abstração no uso da RNA, uma vez que ela é utilizada para indicar a direção que o agente deve se mover para evitar a vegetação mais densa. No trabalho de Pessin, Osório e Musse (2008) foi reduzido o nível de abstração, atribuindo à RNA a tarefa de acionar os motores do agente diretamente, ao invés de apenas indicar a melhor direção. Além disso, no segundo trabalho os autores implementaram um algoritmo genético para escolher a melhor disposição dos aceiros para o combate dos incêndios.

## 2.6 Considerações

Neste capítulo foram realizados estudos sobre desastres naturais, demonstrando como são classificados e de que maneira podem ser gerenciados. Especificamente sobre terremotos, foram demonstrados seus impactos e os efeitos secundários, listando também os maiores terremotos já registrados. Percebeu-se que um dos maiores riscos depois de ocorrer um terremoto, além dos desabamentos, são os incêndios que surgem e se espalham de forma imprevisível, representando um desafio para as equipes de resgate.

Para a representação de equipes de resgate, foram estudados agentes, descrevendo suas características, tipos de agentes e suas arquiteturas. Sobre sistemas multiagentes, foram explicados os tipos de interação entre agentes.

Devido a atuação dos agentes em ambientes incertos, um estudo sobre a aprendizagem em sistemas multiagentes se mostrou necessário. Para isso, foram estudadas redes neurais artificiais e clusterização, mais especificamente o método *k-means*.

Finalmente, foram levantados trabalhos relacionados nas áreas de sistemas multiagentes em desastres naturais, e aprendizagem em sistemas multiagentes, realizando um levantamento das técnicas utilizadas e resultados atingidos.

Após o estudo das características dos trabalhos na área de sistemas multiagentes em desastres naturais, conclui-se que o *k-means* se mostrou adequado para a divisão do mapa em setores, facilitando a exploração dos agentes, pois considera a distribuição dos prédios no mapa para realizar a divisão. Em relação aos trabalhos de aprendizagem em sistemas multiagentes, concluiu-se que a RNA é uma forma de aprendizado menos custosa que outros métodos de aprendizado e eficiente.

### 3 SISTEMA MULTIAGENTE PARA COMBATE A INCÊNDIOS

Neste capítulo é apresentada a metodologia e a descrição de todas as etapas do desenvolvimento: Modelo de agente bombeiro inteligente, baseado nos trabalhos de Barroso *et al.* (2013) e Liang, Yang e Deng (2014), para combate a incêndios, e do simulador de situação pós-terremoto onde a atuação dos agentes é validada. Para dar suporte ao agente bombeiro na identificação de focos de incêndio, também foi desenvolvido um agente explorador. São apresentadas as estratégias baseadas em redes neurais artificiais, clusterização e o cenário de atuação dos agentes.

#### 3.1 Metodologia da Pesquisa

A pesquisa é de caráter exploratório, já que são analisados trabalhos existentes e seus resultados e, a partir disso, são descritas as características envolvidas no domínio de aplicação (GIL, 2002).

Quanto à coleta de dados, a pesquisa é de caráter bibliográfico, com busca de trabalhos em publicações de congressos (AAMAS), e no portal de periódicos da CAPES e da IEEE.

Quanto aos procedimentos de coleta, é estudo de caso, com o desenvolvimento de simulações de terremotos, com a montagem dos cenários envolvendo as principais consequências de um terremoto, como incêndios, e analisando os resultados, a partir da atuação dos agentes inteligentes no combate a incêndios (YIN, 2015).

A pesquisa é de caráter quantitativo quanto às variáveis utilizadas. Serão avaliadas a quantidade de focos de incêndios combatidos e área de prédios destruída por focos de incêndios. Também qualitativa quanto à avaliação da estratégia de aprendizagem utilizada, neste caso, redes neurais artificiais e clusterização.

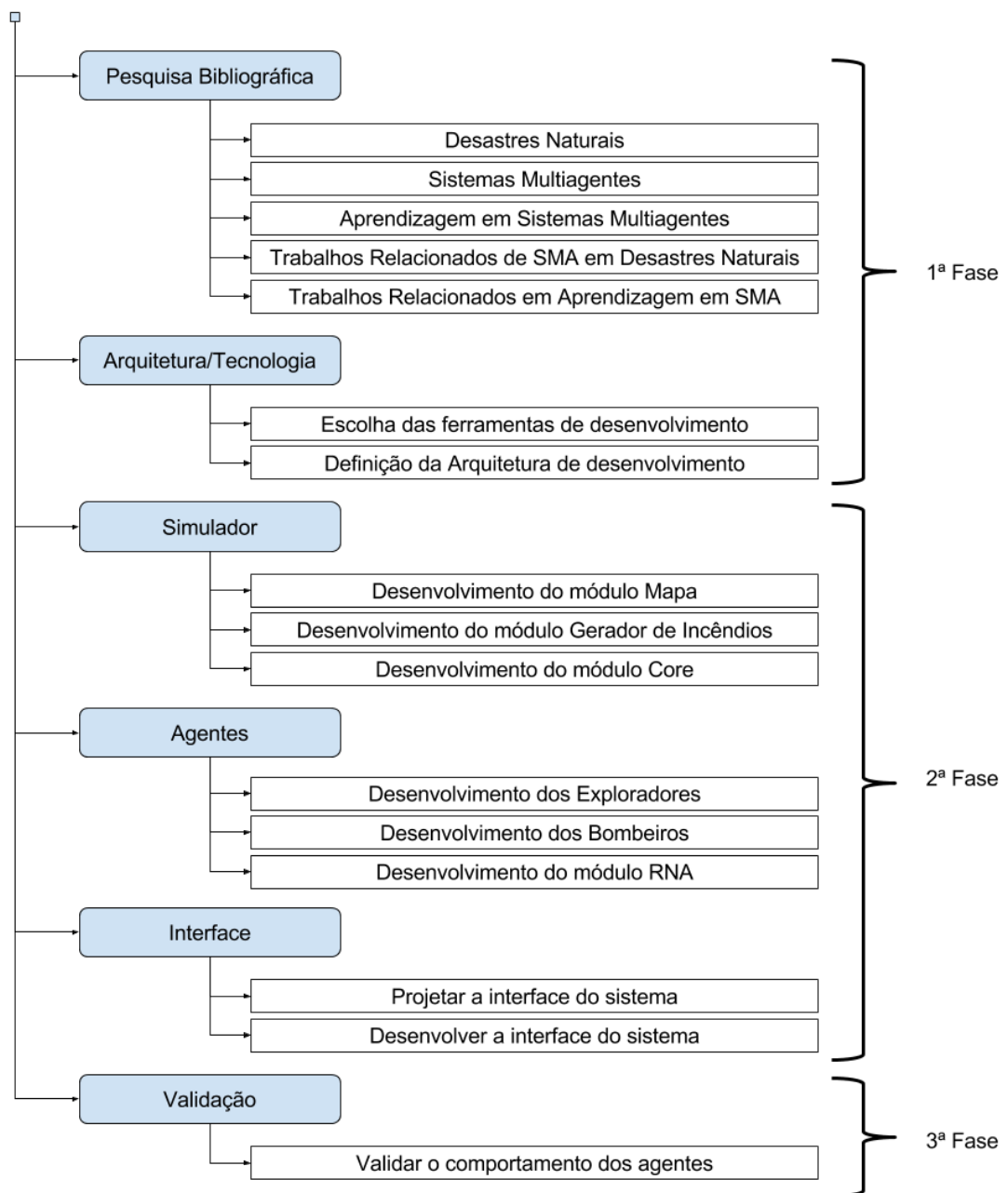
A Figura 8 apresenta as etapas de desenvolvimento da pesquisa, que serão detalhadas nas próximas seções. A 1ª fase foi dividida em 2 partes. Na 1ª parte foi realizado um levantamento bibliográfico sobre desastres naturais, sistemas multiagentes e aprendizagem em SMA, depois foram levantados os trabalhos relacionados sobre a utilização de SMA em desastres naturais, e sobre aprendizagem em SMA. Na 2ª parte, a partir do levantamento bibliográfico e dos trabalhos relacionados, foi definida a arquitetura do sistema. Finalmente, foram definidas as ferramentas a serem utilizadas durante a próxima fase.

Depois, na 2ª fase foi iniciado o desenvolvimento, que foi dividido em 3 partes: Simulador, Agentes e Interface. A parte do simulador consiste na implementação do ambiente

(implementação dos módulos Gerador de Incêndios, Mapa e Core) onde os agentes estão inseridos. A parte dos agentes consiste na implementação da percepção, modelo de mundo, tomada de decisão dos agentes e ação, e no caso dos bombeiros, o módulo de RNA e o treinamento deste. Finalmente, na parte da interface foi desenvolvida uma interface para que seja possível visualizar o comportamento dos agentes no ambiente.

Na 3ª fase foram exibidos os resultados e foi realizada a validação do comportamento observado nos bombeiros com e sem a RNA.

Figura 8 – Etapas de desenvolvimento da pesquisa

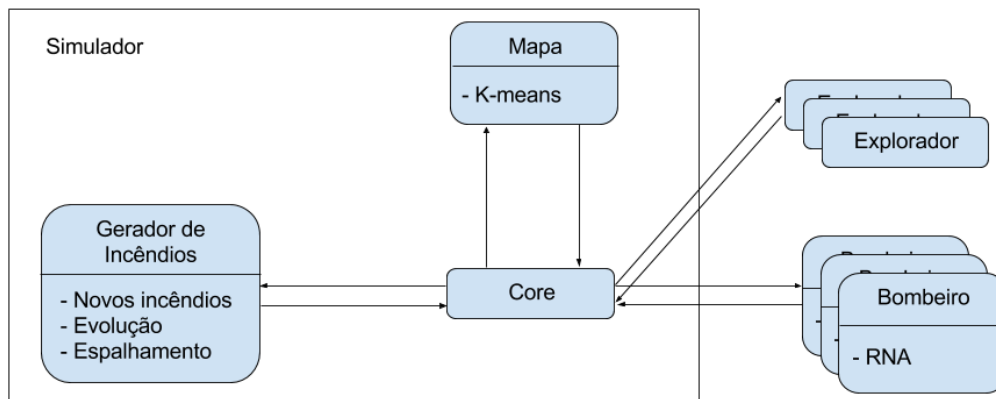


Fonte: (AUTORAS, 2016)

### 3.2 Arquitetura do sistema multiagentes

A arquitetura do sistema multiagentes se divide em duas partes: os agentes e o simulador. O simulador é composto por três módulos: Gerador de Incêndios, Mapa e *Core*, enquanto os agentes são divididos em Bombeiros e Exploradores, cada um com sua própria arquitetura. Uma visão geral da arquitetura e do fluxo de informação entre seus componentes pode ser observada na Figura 9.

Figura 9 – Arquitetura do sistema multiagentes



Fonte: (AUTORAS, 2016)

O tempo na simulação é controlado pelo *core*, e é dividido em intervalos discretos, ou ciclos. Em cada ciclo são realizados cálculos por parte do gerador de incêndio para determinar as mudanças que ocorrem no ambiente, e os agentes determinam suas ações, a fim de mitigar o dano causado pelos incêndios. Os agentes interagem com o *core* para realizar suas ações.

#### 3.2.1 Mapa

O módulo Mapa é uma estrutura de dados baseada em matriz, onde cada célula armazena diversos objetos com seus respectivos atributos:

- Incêndio: Posição x, Posição y, Intensidade, Energia.
- Agente bombeiro: Posição x, Posição y, Capacidade do tanque, Nível de água, Ação atual, Setor Atribuído, Velocidade de combate de incêndio, Velocidade de deslocamento, Velocidade de reabastecimento, Alcance do jato de água.

- Agente explorador: Posição x, Posição y, Ação atual, Setor atribuído, Velocidade de deslocamento.
- Prédio: Posição x, Posição y, Material, Área total, Área queimada.
- Rua: Posição x, Posição y.
- Hidrante: Posição x, Posição y.

A criação dos mapas foi realizada através da ferramenta *Tiled Map Editor*. Esta ferramenta é utilizada para criação de mapas baseados em células, permitindo a exportação de arquivos no formato JSON, CSV, Javascript e LUA. Para a utilização dos mapas neste cenário multiagentes, foi adotado o formato JSON, que é flexível o suficiente para incluir atributos em cada célula, como tipo da célula e material (caso seja um prédio), por exemplo, e mais simples de importar para objetos Java do que as outras opções disponíveis.

Cada prédio do mapa pode ser feito de um dos seguintes materiais: Alvenaria, Alvenaria e Madeira, ou Madeira. Cada um destes materiais tem impactos diferentes na evolução e espalhamento dos incêndios, conforme será explicado na próxima subseção. Os diferentes prédios com seus respectivos materiais podem ser observados na Figura 10.

Figura 10 – Materiais dos prédios



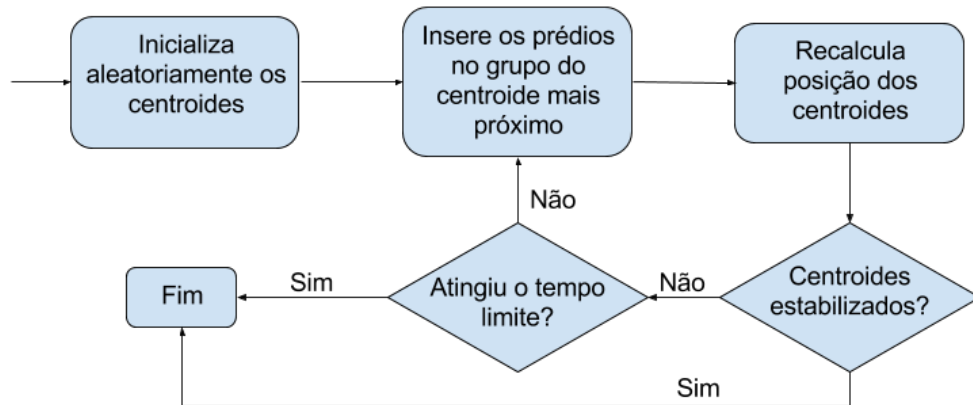
Fonte: (AUTORAS, 2016)

Além de armazenar estas informações, o Mapa também é responsável por distribuir os prédios em setores através da clusterização com o algoritmo *K-means*. Esta divisão tem o objetivo de tornar mais eficiente a exploração do mapa, por parte dos agentes. Na interface é definido o número de setores desejados, e para cada setor é gerado um ponto aleatório no mapa, chamado de centroide. Depois, cada prédio do mapa é atribuído ao setor cujo centroide é o mais próximo de si. Depois que todos os prédios são atribuídos a um setor, a posição de cada centroide é recalculada tomando a média de todas as posições dos prédios de seu setor.



Se qualquer centroide mudar de lugar, o processo de atribuir os prédios aos setores do centroide mais próximo é repetido. A repetição continua até que os centroides parem de mudar de lugar, ou até que o tempo limite de 5 segundos seja atingido. Um fluxograma deste processo pode ser observado na Figura 11.

Figura 11 – Fluxograma da clusterização com o algoritmo K-means



Fonte: (AUTORAS, 2016)

### 3.2.2 Gerador de Incêndios

O Gerador de Incêndios é responsável por controlar a evolução, espalhamento e surgimento dos incêndios da simulação. Ele também calcula, a cada ciclo da simulação, a área que deve ser queimada em cada prédio com incêndio. Apesar de o gerador de incêndios ser responsável pelo cálculo da área queimada, é o *core* que acessa o mapa e “queima” os prédios, conforme informado pelo Gerador de Incêndios.

Cada incêndio possui, além de sua localização, uma intensidade e energia:

- **Intensidade:** Influencia o dano que o incêndio causa no prédio a cada ciclo. Esta é a única informação do incêndio visível aos agentes da simulação, e é um valor inteiro podendo variar de 0 (Extremamente Baixa) até 6 (Extremamente Alta).
- **Energia:** Indica o quão difícil será apagar o incêndio. No primeiro ciclo do surgimento de um incêndio, ela é determinada pela intensidade inicial. Nos ciclos posteriores ao do surgimento, na etapa de evolução, é a energia que é incrementada, e posteriormente baseada nessa energia, a intensidade é

recalculada. A Energia permite que um incêndio evolua sua intensidade de forma suave a cada ciclo, e também possibilita realizar ajustes mais finos na velocidade que os bombeiros apagam os incêndios, em comparação com utilizar apenas a intensidade. Quando um incêndio atinge a intensidade máxima (Extremamente Alta), a energia para de crescer.

Existem quatro fatores que podem ser definidos pelo usuário na interface: fator de evolução, fator de espalhamento, fator de surgimento e fator de velocidade de queima.

- O fator de evolução (que pode variar de 0 até 1) determina diretamente a velocidade que a energia dos incêndios cresce, e, conseqüentemente, a velocidade com que cada incêndio mudará de intensidade.
- O fator de espalhamento (que pode variar de 0 até 1), aliado à intensidade do incêndio e ao coeficiente de inflamabilidade do material do prédio (conforme a Tabela 2), determina a chance de um incêndio espalhar-se para algum prédio vizinho sem incêndio.
- O fator de surgimento (que pode variar de 0 até 1) determina diretamente a chance de um incêndio com intensidade extremamente baixa surgir em um local aleatório do mapa.
- O fator de velocidade de queima (que pode ser qualquer valor maior que 0) define o quanto a intensidade do incêndio irá influenciar na área queimada por um incêndio a cada ciclo.

**Tabela 2 – Coeficientes de inflamabilidade dos materiais**

<b>Material</b>	<b>Coeficiente de inflamabilidade</b>
Alvenaria	0,5
Alvenaria e Madeira	1
Madeira	2

Fonte: (AUTORAS, 2016)

No início da simulação, o Gerador de Incêndios cria os incêndios iniciais em lugares aleatórios. O número de incêndios criados inicialmente é determinado pelo usuário na interface da simulação. Estes incêndios iniciais têm sua intensidade determinada

aleatoriamente, com probabilidade maior de intensidades mais baixas serem escolhidas, conforme a Tabela 3. Quando um incêndio surge, independente do ciclo, sua energia é inicializada conforme a Equação 4 de inicialização da energia do incêndio, onde a intensidade do incêndio é somada em 1, para evitar que a energia de um incêndio seja 0 em sua intensidade mínima. A área do prédio também é levada em consideração no cálculo da energia para que incêndios em prédios com áreas maiores sejam mais difíceis de serem apagados.

**Tabela 3 – Distribuição de intensidade dos incêndios iniciais.**

<b>Intensidade</b>	<b>Probabilidade de escolha</b>
0 (Extremamente Baixa)	50,78125%
1 (Muito Baixa)	25%
2 (Baixa)	12,5%
3 (Média)	6,25%
4 (Alta)	3,125%
5 (Muito Alta)	1,5625%
6 (Extremamente Alta)	0,78125%

Fonte: (AUTORAS, 2016)

$$energia = (1 + intensidade) * areaPredio$$

(Equação 4)

Depois, nos ciclos subsequentes à sua criação, os incêndios são evoluídos, aumentando sua energia (conforme Equação 5 de evolução da energia do incêndio) e corrigindo a sua intensidade, de acordo com a Equação 6.

$$energia = energia + fatorEvolucao * areaPredio * fatorAleatorio * coeficienteInflamabilidadeMaterial$$

(Equação 5)

$$intensidade = \left\lfloor \frac{energia}{areaPredio} - 1 \right\rfloor$$

(Equação 6)

O cálculo da energia incrementada no incêndio a cada ciclo (descrito na Equação 5), leva em consideração a área do prédio para tornar mais difícil combater incêndios em prédios grandes (assim eles terão mais energia). O coeficiente de inflamabilidade do material também é levado em consideração, de maneira que incêndios em prédios de alvenaria sejam mais fáceis de combater do que incêndios em prédios de madeira. O coeficiente de inflamabilidade pode fazer com que a energia ganha por ciclo aumente duas vezes mais rápido do que normalmente (caso seja madeira) ou pode fazer com que a energia ganha a cada ciclo seja a metade do normal (caso seja alvenaria). O fator aleatório serve apenas para gerar indeterminismo na simulação, representando a natureza caótica de um incêndio. O fator de evolução serve como um controle para o usuário do sistema poder definir a velocidade que o incêndio vai crescer.

Além disso, os incêndios existentes podem se espalhar para os vizinhos, dependendo da chance determinada na interface de simulação pelo fator de espalhamento, conforme Equação 7. Caso todos os vizinhos diretos do incêndio já estejam em chamas ou queimados, o incêndio pode se espalhar para algum vizinho do outro lado da rua (caso este incêndio seja adjacente a uma rua).

*Chance de espalhamento em %*

$$= \frac{\textit{intensidadeAtual}}{\textit{intensidadeMáxima}} * \textit{fatorEspalhamento} \\ * \textit{coeficienteInflamabilidadeMaterial} * 100$$

(Equação 7)

O cálculo da chance de espalhamento do incêndio demonstrado na Equação 7, considera a relação entre a intensidade atual do incêndio e a intensidade máxima. Quanto maior a intensidade do incêndio, maior é a chance de o incêndio se espalhar. Além disso, outros fatores que influenciam nesta chance são o fator de espalhamento, definido pelo usuário na interface (quanto maior o seu valor, maior a chance), e o coeficiente de inflamabilidade do material do prédio, que pode fazer com que a chance caia pela metade, permaneça a mesma, ou dobre, dependendo do material. A multiplicação pelo escalar 100 serve apenas para a representação do resultado em porcentagem.

Finalmente, a cada ciclo, baseado no fator de surgimento, o gerador de incêndios determina se um incêndio com intensidade mínima (e energia definida conforme Equação 4) deve surgir em algum lugar aleatório no mapa.

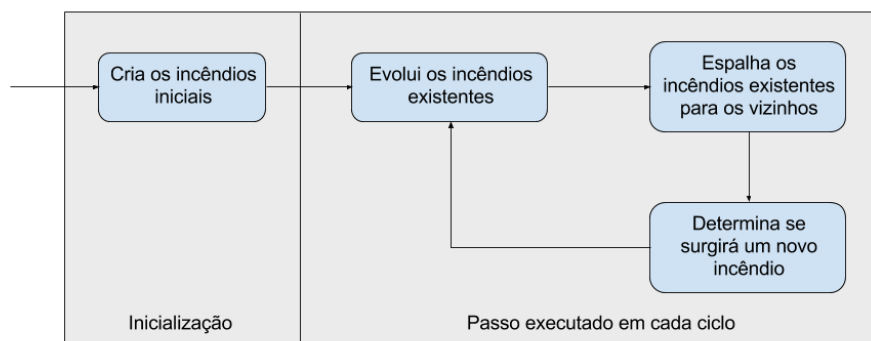
Em cada ciclo, todos os prédios que estão em chamas têm sua área queimada incrementada conforme a Equação 8.

$$areaQueimada = (intensidadeIncendio + 1) * fatorVelocidadeQueima$$

(Equação 8)

O funcionamento geral do Gerador de Incêndios pode ser observado na Figura 12.

Figura 12 – Funcionamento do Gerador de Incêndios



Fonte: (AUTORAS, 2016)

### 3.2.3 Core

O módulo Core é responsável pela passagem de informação entre os outros módulos e entre os agentes. O *Core* controla toda a simulação, realizando chamadas ao Gerador de Incêndios e aos Agentes a cada ciclo e inclusive realizando chamadas de função na interface gráfica da simulação para exibir o estado atual da simulação.

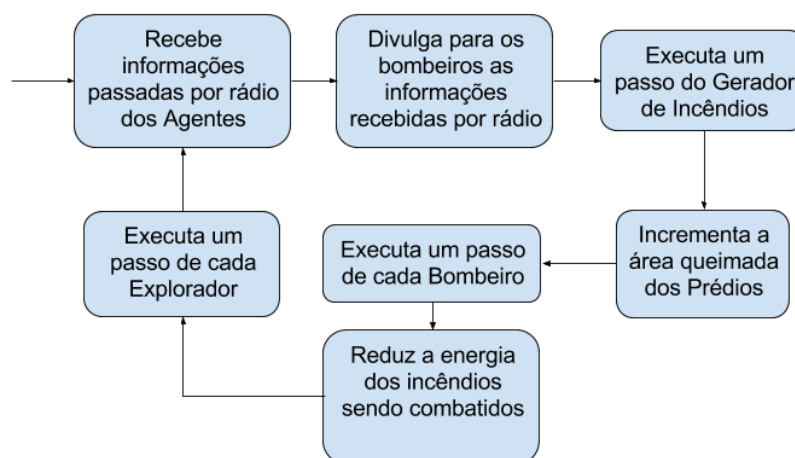
Em cada ciclo, o *Core* executa as seguintes etapas:

1. Coleta as informações passadas por rádio pelos agentes exploradores e bombeiros.
2. Divulga para os bombeiros as informações passadas por rádio no primeiro passo. Isso faz com que as informações de rádio que os bombeiros recebem estejam sempre defasadas em um ciclo, uma vez que nos passos 3 e 4 os incêndios e prédios não estão mais no mesmo estado em que estavam quando a informação foi transmitida.

3. Executa um passo do Gerador de Incêndios, fazendo com que os incêndios existentes evoluam e se espalhem, e eventualmente criando um incêndio em local aleatório.
4. Incrementa a área queimada dos prédios, e caso toda a área do prédio seja consumida, remove o incêndio e o prédio passa a ser “queimado” e não pode mais ser alvo de incêndios.
5. Executa um passo de cada bombeiro. Em um passo, cada bombeiro pode mover-se, ou combater um incêndio ou encher o tanque em um hidrante. Caso um bombeiro esteja combatendo um incêndio, a energia do incêndio é decrementada.
6. Executa um passo de cada explorador. Em um passo, cada explorador pode apenas mover-se pelo seu setor a procura de novos incêndios.

O fluxograma de funcionamento do *Core* pode ser observado na Figura 13.

Figura 13 – Fluxograma do Core



Fonte: (AUTORAS, 2016)

### 3.2.4 Agentes

Os agentes da simulação têm como objetivo mitigar os danos causados pelos incêndios. Para isso, cada agente pode observar as células do mapa em um raio de 5 células ao redor dele, e reportar por rádio as informações obtidas. O raio de visão do agente pode ser observado na Figura 14. As informações de rádio chegam com um ciclo de atraso para os outros agentes, pois não seria realista se todos os agentes soubessem de tudo o que os outros

sabem instantaneamente. Dentre as informações passadas por rádio estão: ação atual do agente, os prédios avistados, e a posição e intensidade dos incêndios. A comunicação via rádio acontece através do módulo *Core*, que a cada ciclo recebe as informações baseadas no que cada agente está enxergando, e unifica todas estas informações em um objeto (removendo informações duplicadas) e o repassa para os agentes no final do ciclo. Desde o início da simulação, todos os agentes conhecem a arquitetura do mapa (localização das ruas, dos prédios e dos hidrantes), porém desconhecem os locais que possuem incêndios.

Figura 14 – Raio de visão dos agentes

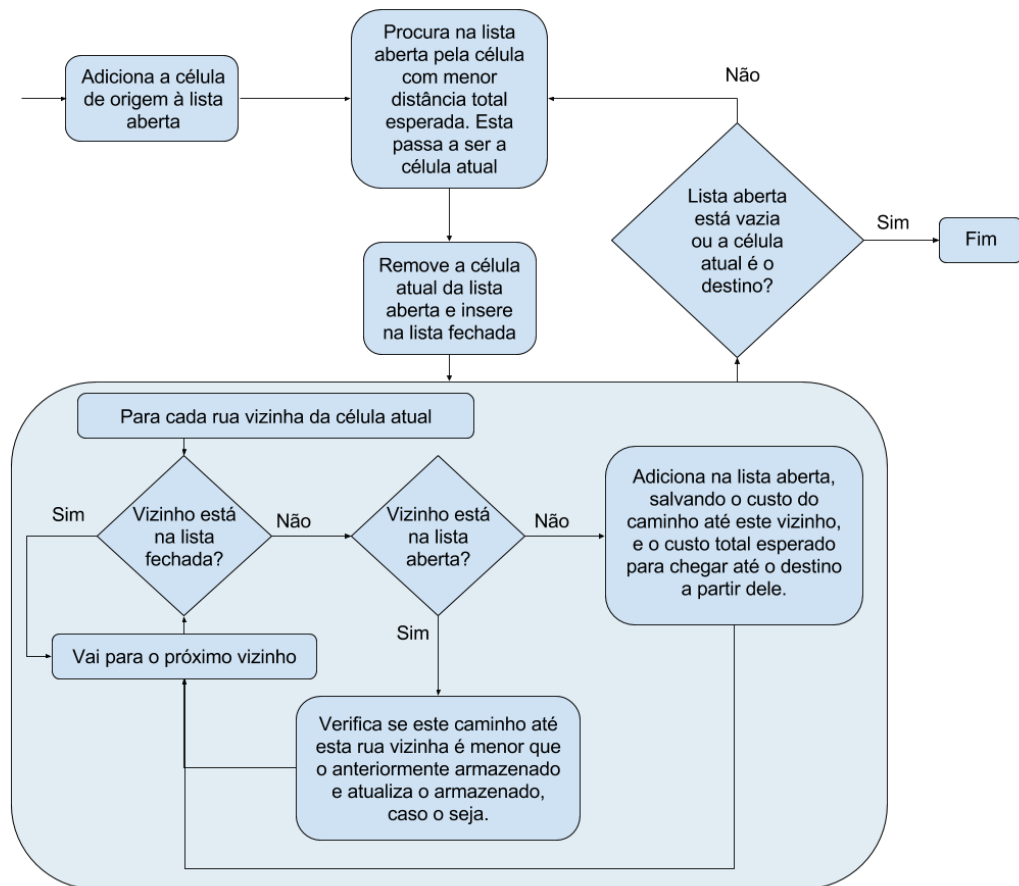


Fonte: (AUTORAS, 2016)

Foram definidos dois tipos de agentes: Exploradores e Bombeiros. Essa divisão é fundamental para a eficiência dos bombeiros, uma vez que estes podem focar o seu tempo no combate a incêndios, deixando a tarefa de exploração do mapa para os exploradores. A cada ciclo um agente pode realizar uma ação e reportar tudo o que observou via rádio. A única ação disponível para o agente explorador é mover-se, enquanto o bombeiro pode combater incêndio e encher o tanque de água, além da ação de movimento.

Para mover-se até o seu destino, cada agente utiliza o algoritmo A\* para a escolha da rota, que conforme Russel e Norvig (2010) é simples, eficiente e o mais popular método para planejamento de rotas. O A\* recebe como parâmetro o ponto de origem (posição do agente) e ponto de destino (alvo do agente), e então realiza os passos descritos na Figura 15.

Figura 15 – Fluxograma de execução do algoritmo A\*



Fonte: (AUTORAS, 2016)

O algoritmo expande as ruas vizinhas iterativamente a partir da célula de origem (posição do agente), sempre dando preferência para as ruas que possuem a menor distância total esperada até o destino (partindo da origem e passando pela célula), com base na heurística da distância de Manhattan (definida como a soma dos módulos das diferenças das coordenadas  $x$  e  $y$  dos pontos em questão), conforme definida em Sinwar e Kaushik (2014). A cada rua que é expandida, suas células vizinhas do tipo “Rua” são incluídas na lista de células a serem visitadas (chamada de lista aberta) e cada uma destas células vizinhas armazenam como célula “pai” a célula atual. Cada rua que é visitada é removida da lista aberta e incluída na lista fechada. O algoritmo termina a execução quando a rua sendo visitada é a célula de destino, e neste momento o caminho todo é reconstruído com base na célula “pai” armazenada em cada célula.

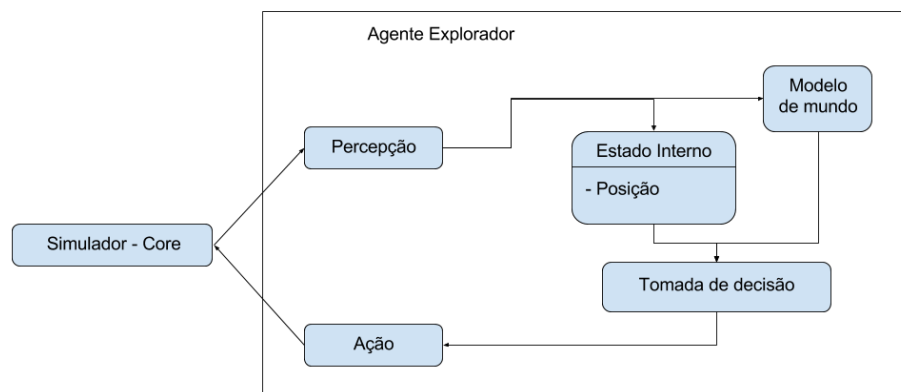
A seguir o comportamento de ambos os agentes é descrito.



### 3.2.4.1 Exploradores

O explorador é um agente reativo, com maior velocidade de deslocamento que o bombeiro, podendo mover-se a 5 células por ciclo. Sua única tarefa é reportar tudo o que observa em cada ciclo. Sua arquitetura pode ser observada na Figura 16. O explorador armazena no Modelo de mundo as observações obtidas pela Percepção. Com base nas informações do Modelo de mundo e do estado interno, o explorador decide para qual direção mover-se, tomando esta ação em seguida.

Figura 16 – Arquitetura do agente Explorador



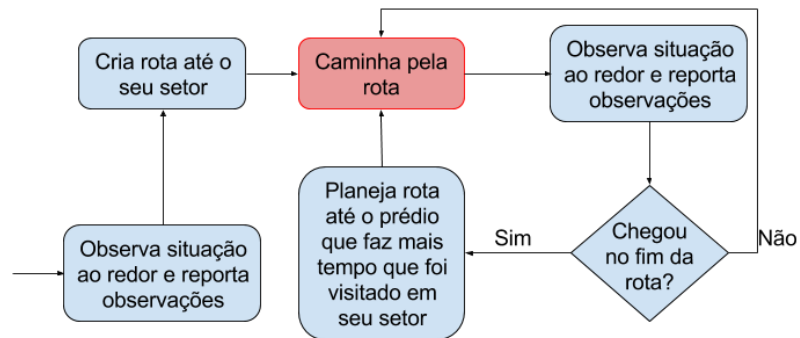
Fonte: (AUTORAS, 2016)

Os exploradores possuem os seguintes atributos:

- **Posição:** Coordenada nos eixos  $x$  e  $y$  dentro do mapa.
- **Ação atual:** Este objeto contém qual a ação que o explorador está executando no momento (que é sempre se movendo), e as informações sobre esta ação (destino e o caminho planejado até o destino).
- **Velocidade de deslocamento:** Cada explorador é capaz de percorrer 5 células do mapa em um ciclo.
- **Setor:** O setor para o qual este explorador foi atribuído.

Cada explorador é atribuído a um setor no início da simulação e se move até ele, reportando tudo o que vê. Depois que chega ao seu setor, ele fica o resto da simulação indo até os pontos dentro do setor que faz mais tempo que foram visitados, buscando por novos incêndios que possam ter surgido. O fluxograma que descreve o comportamento do Explorador, com a ação destacada em vermelho, pode ser observado na Figura 17.

Figura 17 – Fluxograma do comportamento do agente Explorador



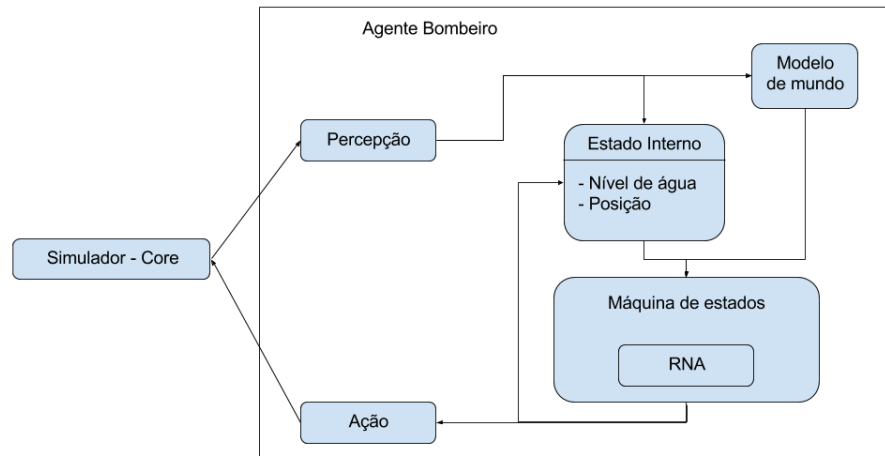
Fonte: (AUTORAS, 2016)

### 3.2.4.2 Bombeiros

O bombeiro é um agente cognitivo, com inteligência baseada em RNA (Redes Neurais Artificiais). Sua principal tarefa é combater incêndios, porém, assim como o explorador, ele também reporta para os outros agentes tudo o que observa.

A arquitetura proposta para o agente bombeiro pode ser observada na Figura 18. O agente recebe informações do ambiente e o resultado de suas ações através do módulo de Percepção. As percepções são utilizadas para atualizar o Modelo do mundo e a posição do agente no Estado interno. O Modelo de mundo armazena todas as informações que o agente possui sobre o ambiente, como locais com incêndios e quais incêndios os outros agentes possuem como alvo, por exemplo. O Estado interno armazena os atributos do agente, que são descritos em detalhe a seguir. O bombeiro possui uma máquina de estados que controla o seu comportamento, e tem seus estados e transições descritos a seguir. As informações do Modelo do mundo e Estado interno são acessadas pela máquina de estados para determinar o próximo estado (e conseqüentemente, próxima ação) do agente.

Figura 18 – Arquitetura do agente bombeiro



Fonte: (AUTORAS, 2016)

Os bombeiros possuem os seguintes atributos:

- Posição: Coordenada nos eixos  $x$  e  $y$  dentro do mapa.
- Capacidade do tanque: Determina a quantidade de água que o bombeiro pode carregar e possui o valor 10.000 litros para todos os bombeiros.
- Nível de água: Indica a quantidade atual de água do bombeiro. Todos os bombeiros começam com o tanque cheio e consomem 500 litros por ciclo para combater os incêndios.
- Velocidade de reabastecimento: Indica quantos litros por ciclo o bombeiro pode reabastecer no hidrante, e possui valor de 2.500 litros por ciclo.
- Ação atual: Este objeto contém qual a ação que o bombeiro está executando no momento, e algumas informações sobre esta ação. Se o bombeiro estiver se movendo, armazena o caminho planejado e o incêndio alvo, caso o bombeiro esteja indo para um incêndio.
- Velocidade de combate de incêndio: Indica a quantidade de energia removida de um incêndio sendo combatido por este bombeiro a cada ciclo, e possui valor de 50 unidades de energia por ciclo.
- Velocidade de deslocamento: Cada bombeiro é capaz de percorrer 3 células do mapa em um ciclo.
- Alcance do jato de água: Indica a que distância o bombeiro pode combater o incêndio e possui o valor de 2 células de distância.
- Setor: O setor ao qual este bombeiro pertence.

Para combater incêndios, os bombeiros consomem a água que fica armazenada em seu tanque, que inicialmente está cheio. Para reabastecer o tanque de água, o bombeiro move-se até o hidrante mais próximo, onde após alguns ciclos, o tanque é completamente preenchido. A seleção de qual incêndio o bombeiro irá combater é realizada por uma RNA com aprendizagem supervisionada, que será descrita posteriormente.

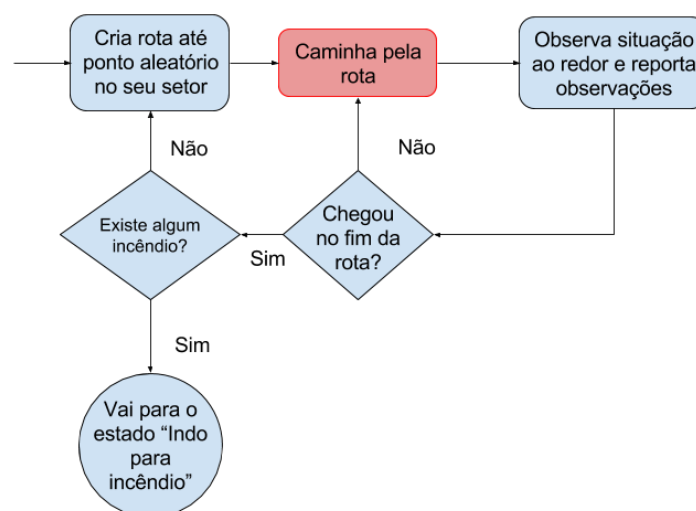
O comportamento do bombeiro é controlado por uma máquina de estados, onde cada estado está diretamente ligado à ação que o bombeiro está realizando. Porém, deve-se ressaltar que não necessariamente o bombeiro vai percorrer todo o fluxograma de um estado em apenas um ciclo. O que determina o final do ciclo é quando o bombeiro atinge um nodo vermelho do fluxograma, que representa uma ação tomada. Os fluxogramas de cada estado podem ser visualizados (com as ações destacadas em vermelho) nas Figuras 19, 20, 21, 22 e 23, e a descrição deles é realizada a seguir:

- Indo para setor: Neste estado, o bombeiro planeja a rota até um ponto aleatório dentro do setor atribuído a ele. Ao longo do caminho, até este ponto, o bombeiro reporta aos outros agentes tudo o que observa. Depois que chega ao ponto definido anteriormente, ele verifica se existe algum incêndio no mapa inteiro, e caso exista, muda para o estado “Indo para incêndio”. Caso não exista incêndio, seleciona um novo ponto aleatório dentro do setor e move-se até ele, repetindo o processo anterior.
- Indo para incêndio: Quando o bombeiro entra neste estado, a primeira coisa que é verificada é se o bombeiro já possui um incêndio selecionado previamente (caso ele esteja vindo do estado “Enchendo o tanque” e já estava combatendo um incêndio antes). Caso já exista um incêndio selecionado, é verificado se o incêndio ainda existe e o bombeiro planeja sua rota até o incêndio. Caso o bombeiro não possua um incêndio selecionado ou este incêndio não exista mais, um novo incêndio é selecionado. Para selecionar o novo incêndio, ele verifica inicialmente se sabe de algum incêndio dentro do seu setor, e depois caso não exista, verifica se existe algum fora de seu setor. Em última instância, caso não exista incêndio que o bombeiro saiba, ele realiza exploração mudando para o estado “Indo para setor”. Uma vez identificado um ou mais incêndios, o bombeiro utiliza a RNA para dar notas aos incêndios que ele conhece. O bombeiro seleciona o incêndio com maior nota dentro de seu setor (ou no mapa inteiro, caso o setor esteja sem incêndios conhecidos) e cria uma rota até o incêndio. Depois disso, o bombeiro basicamente se move até o

incêndio, sendo que a cada ciclo ele reporta tudo o que observa e verifica se o incêndio ainda existe (através das informações recebidas por rádio, ou pela própria observação, caso esteja próximo o suficiente). Caso o incêndio não exista mais (por ter sido apagado por outros bombeiros, ou por ter consumido o prédio), o bombeiro realiza novamente o processo de seleção de incêndios descrito anteriormente. Caso o bombeiro chegue ao incêndio e ele ainda exista, muda para o estado “Combatendo incêndio”.

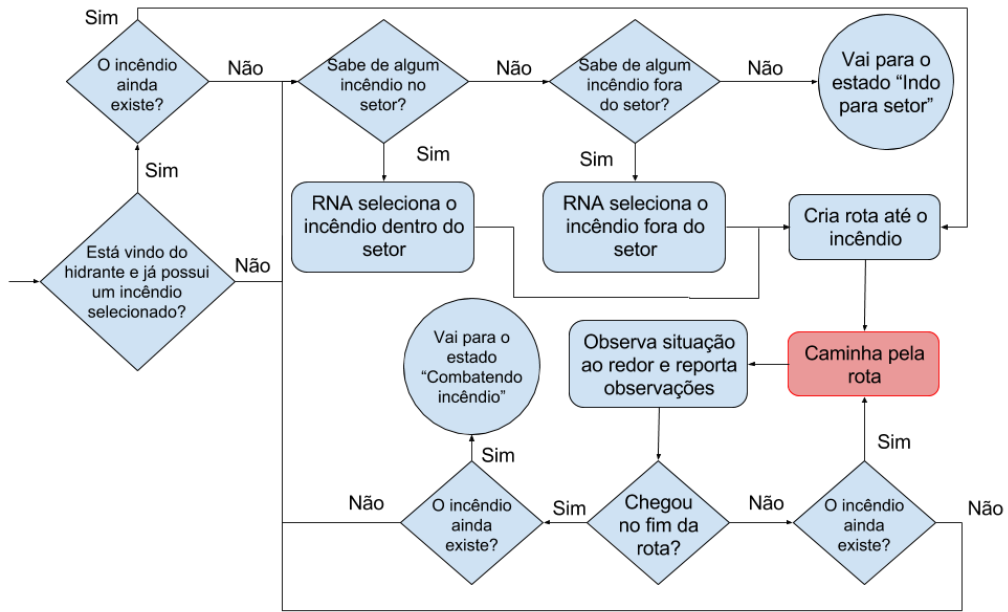
- **Combatendo incêndio:** Neste estado o agente combate ao incêndio até que a água em seu tanque se esgote ou que o incêndio seja extinto. No caso de a água acabar, muda para o estado “Indo para hidrante”. Caso o incêndio tenha sido extinto, muda para o estado “Indo para incêndio”, que seleciona um novo incêndio.
- **Indo para hidrante:** Neste estado o bombeiro seleciona o hidrante mais próximo e cria uma rota até ele, caminhando pela rota até o hidrante, reportando o que observa, e ao chegar ao hidrante, muda para o estado “Enchendo tanque”.
- **Enchendo tanque:** Neste estado, o agente enche o tanque até que o nível de água chegue ao limite máximo, e depois volta a apagar o incêndio anterior, caso ele ainda exista, mudando para o estado “Indo para incêndio”.

Figura 19 – Fluxograma do estado “Indo para setor” do bombeiro



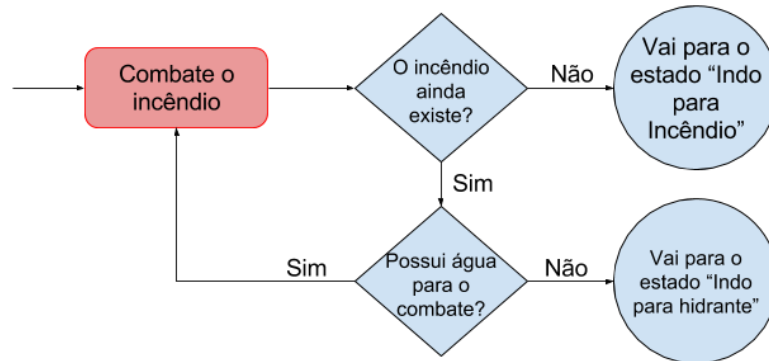
Fonte: (AUTORAS, 2016)

Figura 20 – Fluxograma do estado “Indo para incêndio” do bombeiro



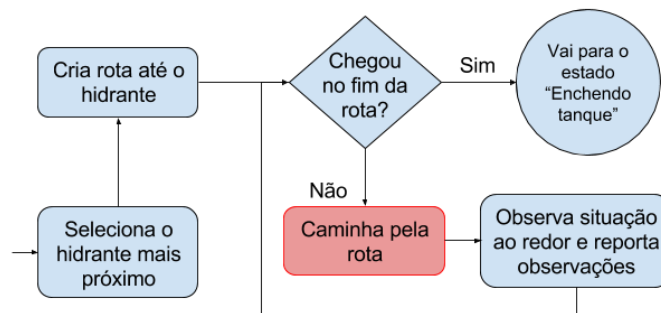
Fonte: (AUTORAS, 2016)

Figura 21 – Fluxograma do estado “Combatendo incêndio” do bombeiro



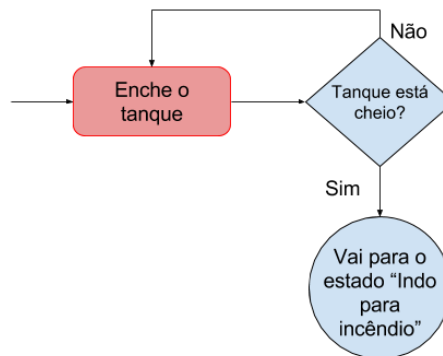
Fonte: (AUTORAS, 2016)

Figura 22 – Fluxograma do estado “Indo para hidrante” do bombeiro



Fonte: (AUTORAS, 2016)

Figura 23 – Fluxograma do estado “Enchendo tanque” do bombeiro



Fonte: (AUTORAS, 2016)

### 3.2.5 RNA

O módulo de RNA está presente nos agentes bombeiros e é responsável pela tomada de decisão do agente em relação a qual incêndio deve ser combatido por este. Para a implementação da RNA foi utilizada a biblioteca de código aberto Neuroph<sup>2</sup>. Esta biblioteca permite a criação de RNAs com diferentes arquiteturas, funções de transferências, métodos de aprendizagem, métodos de cálculo de erro e critérios de parada. Para este trabalho foi criada uma rede *Multilayer Perceptron* com 1 camada oculta, utilizando como método de aprendizagem o *Backpropagation* e função de transferência sigmóide. São utilizados 7 neurônios de entrada e 1 para a saída. Cada entrada e saída é representada por uma variável normalizada do tipo *double* que pode variar de 0 até 1.

Para realizar a tomada de decisão de qual incêndio será combatido, a RNA recebe como entradas os principais fatores que determinam a importância de um incêndio: intensidade do incêndio (0 até 6), coeficiente de inflamabilidade do material do prédio em chamas (0,5 até 2), área do prédio em chamas (20 até 1600), número de bombeiros apagando ou indo apagar o incêndio (0 até 7), distância do hidrante até o incêndio (0 até 40 células), distância do bombeiro até o incêndio (0 até 40 células) e coeficiente de inflamabilidade médio dos materiais dos vizinhos sem incêndio do prédio em chamas (0,5 até 2); e gera como saída uma nota (0 até 30) que representa o tempo economizado e a área salva esperada no caso do agente apagar este incêndio. Para a saída da rede foi estabelecido um limite de valor 30, pois através da observação de 10000 simulações de treinamento, observou-se que apesar de alguns valores discrepantes, a maioria das saídas esteve abaixo deste valor. Estes fatores foram

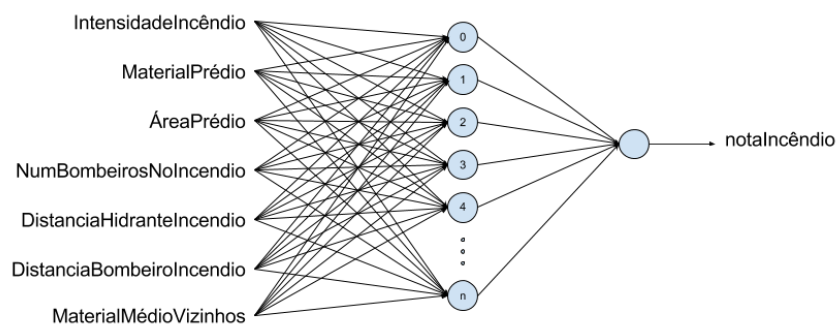
<sup>2</sup> NEUROPH: <http://neuroph.sourceforge.net/>

definidos levando em consideração os aspectos que determinam o dano que um incêndio pode causar, seja diretamente sobre o prédio que ele está, ou indiretamente através do espalhamento para prédios vizinhos. Também, são as características que auxiliam a tomada de decisão pelos bombeiros de quais focos de incêndio priorizar para combater.

Cada uma das entradas e a saída da rede precisam ser normalizadas antes de serem passadas para a RNA. Para realizar esta normalização cada entrada e a saída é dividida pelo seu maior valor possível, obtendo assim um valor entre 0 e 1.

Além dos 7 neurônios de entrada e 1 de saída descritos, a rede possui uma camada oculta com número de neurônios configurável pelo usuário na interface. A Figura 24 demonstra a arquitetura desta rede neural.

Figura 24 – Arquitetura da RNA



Fonte: (AUTORAS, 2016)

Porém, antes de ser utilizado pelo agente para realizar tomadas de decisão, este módulo precisa ser treinado com exemplos de situações de incêndios e o resultado destas situações. Pode-se dividir o treinamento em duas etapas: Geração dos dados para o treinamento da RNA e Treinamento da RNA.

### 3.2.5.1 Geração dos dados para o treinamento da RNA

Para treinar uma RNA, é necessário criar um conjunto de pares de entradas e saídas de exemplo para a rede aprender o comportamento desejado. Para obter valores de exemplo para a saída da RNA, são realizadas duas simulações com as mesmas condições iniciais, uma com o envolvimento do bombeiro e outra sem. Para obter saída da RNA, é utilizada a Equação 9, obtendo como resultado uma nota que será maior para incêndios que o bombeiro faça maior diferença em relação ao tempo economizado ou à área queimada. Cada simulação iniciará



com apenas um foco de incêndio e não surgirão novos ao longo da simulação, mas o incêndio inicial pode se espalhar. Não são utilizados mais focos de incêndio, pois seria difícil determinar qual o foco responsável pelos danos observados.

$$notaIncendio = \frac{areaQueimadaSemBombeiro * tempoSemBombeiro}{areaQueimadaComBombeiro * tempoComBombeiro}$$

(Equação 9)

O usuário define na interface de simulação o tamanho do conjunto de treinamento que deve ser criado para utilizar na fase de aprendizado. Para cada par de entradas e saídas do conjunto, é criada uma instância do *Core* com 100 ciclos de limite, utilizando um fator de surgimento igual a 0 e para o resto dos parâmetros é utilizado o que for definido na interface pelo usuário. Esta instância do *Core* é executada duas vezes (uma com o bombeiro e outra sem) para determinar a nota do incêndio, sendo que cada execução pode parar antes, caso o(s) incêndio(s) (o incêndio inicial pode se espalhar para os vizinhos) seja(m) apagado(s) antes de atingir o centésimo ciclo.

Uma vez obtidos os valores de nota dos incêndios, os dados de entrada e saída são inseridos no conjunto de treinamento que é normalizado para valores entre 0 e 1 e, então, passados para a RNA realizar o aprendizado.

### 3.2.5.2 Treinamento da RNA

Uma vez instanciada a RNA, são passados para ela os dados de treinamento normalizados, e internamente a biblioteca passa por todos os pares do conjunto de treinamento, propagando o erro e ajustando os pesos da rede. Cada vez que a rede passa por todos os pares do conjunto de treinamento conta como uma iteração de treinamento. A cada iteração de treinamento a biblioteca disponibiliza o erro médio quadrático atual, para que seja possível acompanhar a convergência do aprendizado. Se após 1.000 iterações (ou conforme o usuário determinar na interface), a RNA não atingir um erro médio quadrático menor que 0,01, o treinamento é interrompido e os pesos atribuídos à rede são aqueles que geraram o menor erro durante o treinamento.

### 3.3 Interface

O sistema possui somente uma tela, que pode ser visualizada na Figura 25. A tela pode ser dividida em duas partes: visualização da simulação (na parte esquerda) e controle da simulação (na parte direita). Os pontos destacados na Figura são descritos a seguir:

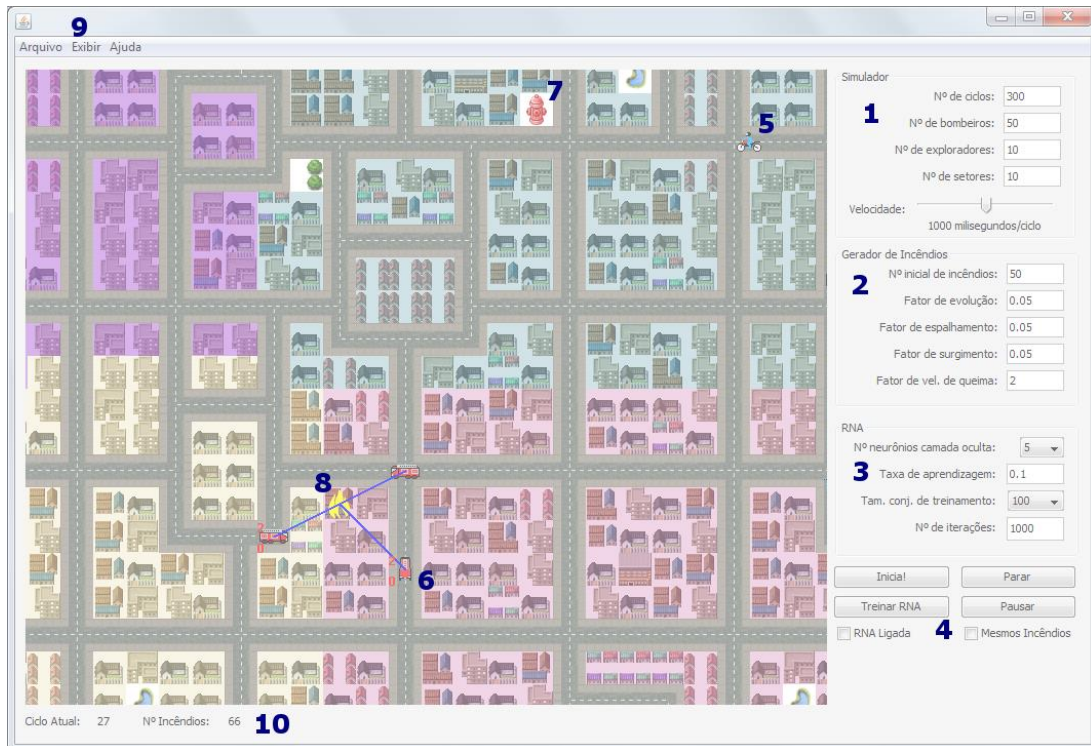
1. Configurações gerais do simulador: Nesta área são definidos o número de ciclos que a simulação executará, número de bombeiros, número de exploradores e número de setores que o mapa possuirá. Além disso, a velocidade de execução da simulação pode ser ajustada (em tempo de execução), sendo possível fazer com que cada ciclo dure desde 2 segundos até o menor tempo possível (executa o mais rápido que o computador conseguir).
2. Configurações do gerador de incêndios: Nesta área são definidos todos os parâmetros que controlam o comportamento dos incêndios e o número inicial de incêndios na simulação. O “Fator de evolução” (entre 0 e 1) afeta a velocidade que os incêndios mudam de intensidade. O “Fator de espalhamento” (entre 0 e 1) afeta a chance dos incêndios se espalharem para seus vizinhos. O “Fator de surgimento” (entre 0 e 1) determina a chance de surgir um incêndio novo em local aleatório no mapa a cada ciclo. O “Fator de velocidade de queima” (qualquer valor positivo) afeta a área queimada por ciclo em cada incêndio.
3. Configurações da RNA: Nesta área podem ser configurados os parâmetros da arquitetura da RNA (como número de neurônios na camada oculta) e do treinamento da RNA. O campo “Taxa de aprendizagem” pode receber valores entre 0 e 1. O campo “Tam. conj. de treinamento” determina o número de pares de entradas e saídas que são gerados para realizar o treinamento da RNA. O “Nº de iterações” determina quantas vezes a RNA irá iterar sobre o conjunto de treinamento durante a etapa de treinamento.
4. Nesta área ficam os botões que comandam a simulação. O botão “Treinar RNA” realiza o treinamento de uma RNA com os parâmetros definidos nos campos da área RNA (3). A RNA irá aprender o comportamento dos incêndios com base no que estiver configurado na área do Gerador de Incêndios (2). O botão “Inicia!” faz com que a simulação inicie com as configurações definidas nos campos anteriores. Os botões “Pausar” e “Parar” param a execução da simulação, com a diferença de que o botão “Pausar” permite a continuação da

simulação posteriormente, pressionando-o novamente. Caso o campo “RNA Ligada” esteja marcado quando o botão “Inicia!” for pressionado, os agentes bombeiros selecionarão o incêndio de forma inteligente, com base na RNA, e caso este campo esteja desmarcado os agentes selecionarão apenas o incêndio mais próximo. Se o campo “Mesmos Incêndios” estiver marcado quando o botão “Inicia!” for pressionado, a simulação é realizada utilizando os mesmos incêndios criados na última simulação realizada.

5. Agente Explorador.
6. Agente Bombeiro. Observa-se que neste caso existem 2 números em vermelho ao lado do bombeiro. O número na parte superior da célula indica o número de bombeiros atualmente naquela célula, e o número na parte inferior indica quantos exploradores estão atualmente nela.
7. Hidrante.
8. Incêndio sendo combatido por 5 bombeiros a partir de 3 células diferentes. As linhas azuis indicam que os bombeiros estão combatendo aquele incêndio. A intensidade do incêndio pode ser identificada através da sua cor, conforme a Figura 26.
9. Menus: No menu “Arquivo” é possível salvar a RNA atual ou carregar RNAs treinadas salvas anteriormente, carregar um novo Mapa. No menu “Exibir” é possível mudar o modo de exibição do mapa para o modo minimalista, removendo as texturas dos prédios para facilitar a visualização dos incêndios da simulação.
10. Nesta área pode-se acompanhar o ciclo atual da simulação e quantos incêndios atualmente estão no mapa.

Para visualizar o mapa de maneira mais ampla ou em detalhe, é possível alterar o nível de *zoom* na área de visualização da simulação através do giro da *mouse wheel*. Além disso é possível navegar pelo mapa clicando e arrastando ele.

Figura 25 – Interface gráfica



Fonte: (AUTORAS, 2016)

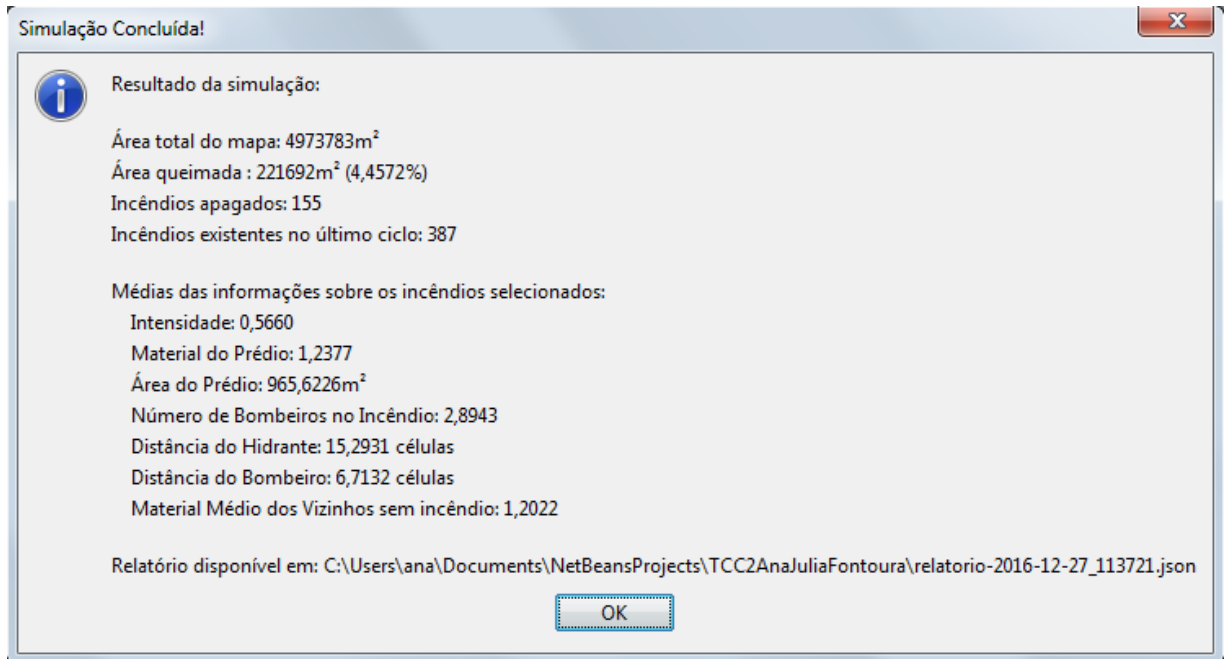
Figura 26 – Intensidades do incêndio de 0 (mais à esquerda) até 6 (mais à direita)



Fonte: (AUTORAS, 2016)

Ao final de cada execução, é exibida uma janela com os resultados da simulação e o endereço do relatório em formato JSON (o formato do relatório é explicado na seção de Validação) gerado por esta simulação. Como pode ser observado na Figura 27, esta janela informa a área total do mapa, área queimada, incêndios apagados e incêndios existentes ao final da simulação, além de exibir uma média das informações dos incêndios selecionados pelos bombeiros.

Figura 27 – Janela exibida ao final de cada simulação

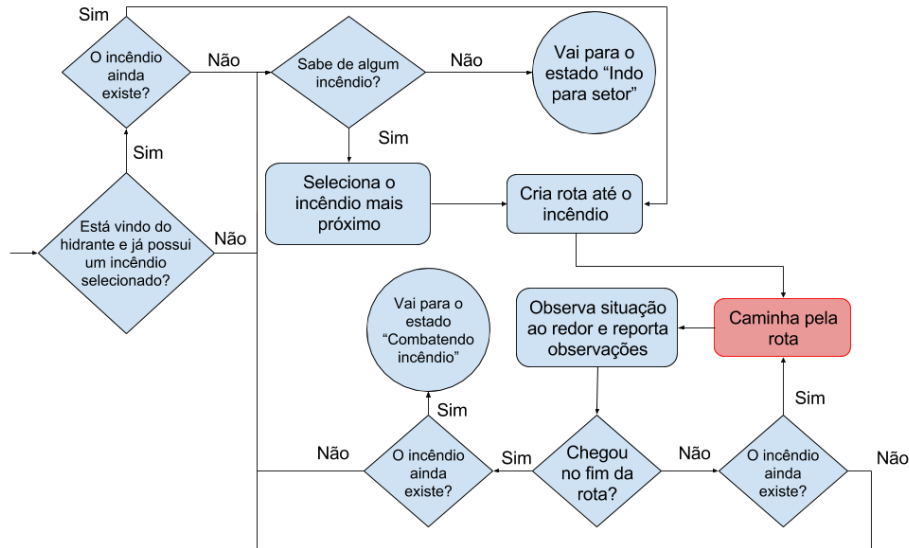


Fonte: (AUTORAS, 2016)

### 3.4 Validação

Para aplicar a etapa de validação do sistema, foi desenvolvido um agente bombeiro mais simples, com estratégia gulosa, para comparar ao inteligente (com RNA). Este agente simples possui a mesma estrutura do agente inteligente, com os mesmos módulos de percepção, modelo de mundo, estado interno e ação, mas com a máquina de estados sem a RNA. Apesar da máquina de estados do agente simples possuir os mesmos estados que a do agente inteligente, o estado “Indo para incêndio” do agente simples seleciona o incêndio com menor distância até o bombeiro, sem levar em consideração a existência de setores definida pelo algoritmo de clusterização de *K-means*. As únicas situações onde os setores são considerados são no início da simulação para distribuir os bombeiros pelo mapa e quando um bombeiro não sabe da existência de incêndio para apagar, o que resulta em um comportamento de exploração dentro de seu setor. O Fluxograma deste estado modificado pode ser observado na Figura 28.

Figura 28 – Estado “Indo para incêndio” do bombeiro simples



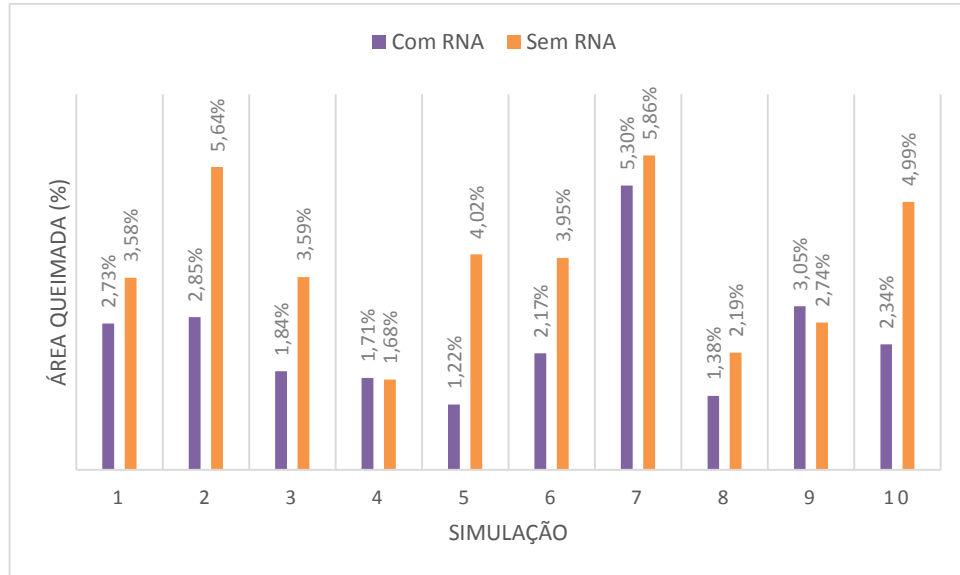
Fonte: (AUTORAS, 2016)

A validação deste trabalho foi realizada com base em 10 pares de simulações, onde cada par é composto por uma simulação utilizando bombeiros com RNA e a outra utilizando bombeiros simples, ambas com os mesmos incêndios iniciais. Foram comparados os resultados de cada par de simulações e foram avaliados o total de área queimada, número de incêndios apagados, número de incêndios existentes ao final da simulação, além das informações sobre os incêndios selecionados. Cada simulação foi realizada com as mesmas configurações: 300 ciclos, 50 bombeiros, 10 exploradores, 10 setores, 50 incêndios iniciais, fator de evolução com valor 0,05, fator de espalhamento com valor 0,05, fator de surgimento com valor 0,05 e fator de velocidade de queima com valor 2. Os agentes com RNA utilizaram uma RNA treinada com 10.000 pares de treinamento, 1.000 iterações de treinamento, taxa de aprendizagem 0,1 e 25 neurônios na camada oculta.

Nas Figuras 29, 30 e 31 são demonstrados os gráficos com os resultados dos 10 pares de simulações. Pode-se observar que em 8 de 10 casos, os bombeiros com RNA foram mais efetivos em reduzir a área queimada pelos incêndios, e em 6 de 10 casos a quantidade de focos de incêndio combatidos pelos agentes com RNA foi maior. Apesar do total de incêndios apagados pelos agentes com e sem RNA ser semelhante em quase todas as simulações, percebe-se que o número de incêndios existentes ao final de cada simulação em geral é bem menor nos casos com RNA. Isto indica que os bombeiros inteligentes realizaram melhores

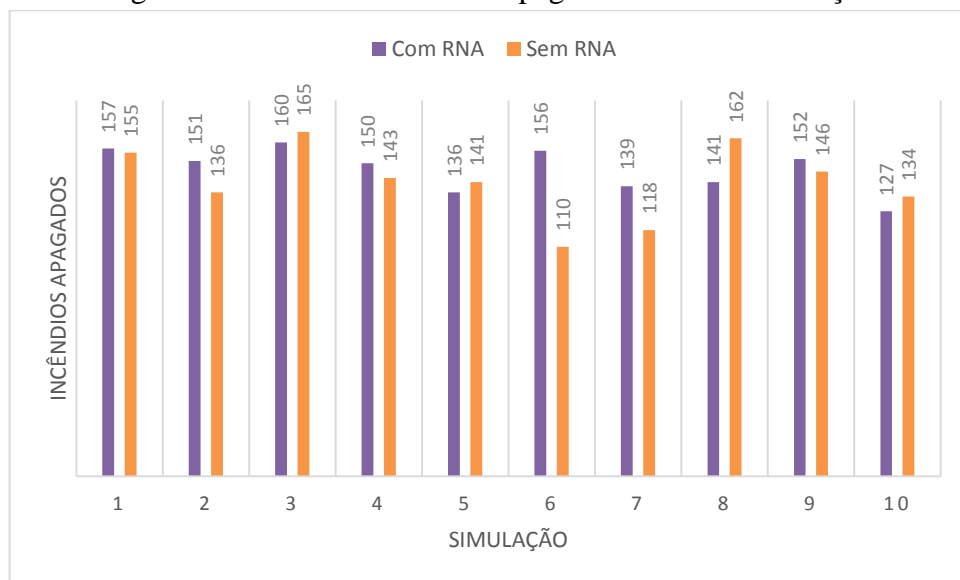
escolhas de alvos para serem apagados, eliminando incêndios que se espalhariam mais facilmente.

Figura 29 – Porcentagem de área queimada do mapa ao final de cada simulação



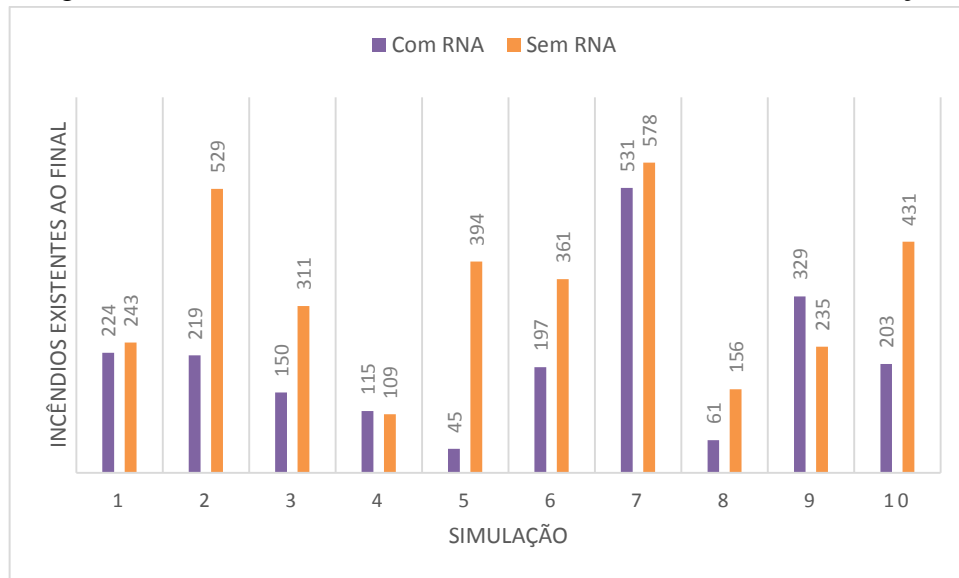
Fonte: (AUTORAS, 2016)

Figura 30 – Total de incêndios apagados em cada simulação



Fonte: (AUTORAS, 2016)

Figura 31 – Número de incêndios existentes ao final de cada simulação

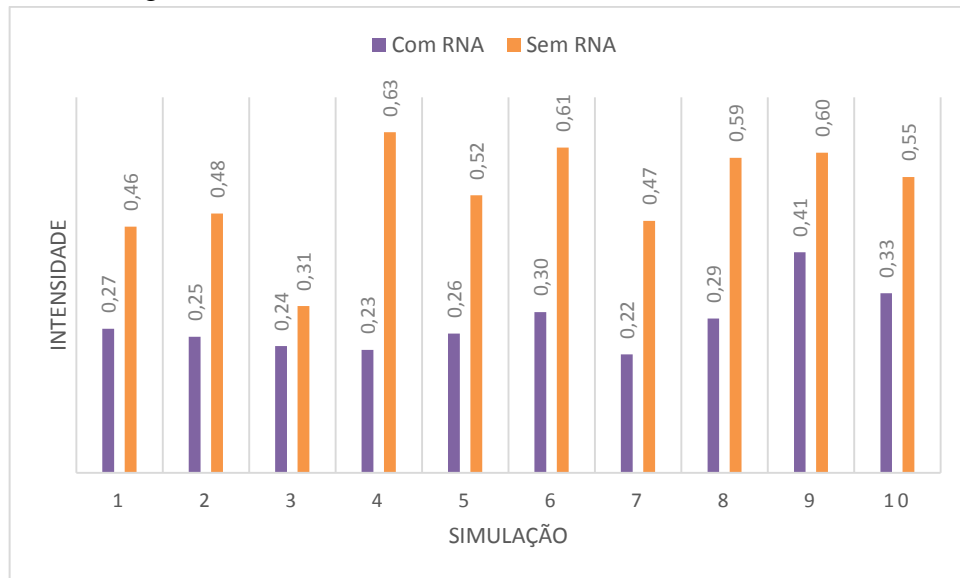


Fonte: (AUTORAS, 2016)

Nos gráficos exibidos nas Figuras 32, 33, 34 e 35 pode-se observar as características que a RNA considerou mais relevante na seleção do incêndio a ser combatido. Em todas as simulações a RNA selecionou em média, incêndios com menor intensidade, em prédios com menor área, material mais inflamável e com vizinhos sem incêndio mais inflamáveis. A preferência por incêndios com baixa intensidade acontece porque incêndios na fase inicial são os mais fáceis de serem extintos, evitando que evoluam e se espalhem causando maiores estragos. Pelo mesmo motivo incêndios em prédios menores são selecionados em média. Foi dada preferência para prédios com materiais mais inflamáveis e vizinhos mais inflamáveis porque estes são os incêndios que evoluem mais rápido e se espalham mais rapidamente, aumentando consequentemente a área queimada.

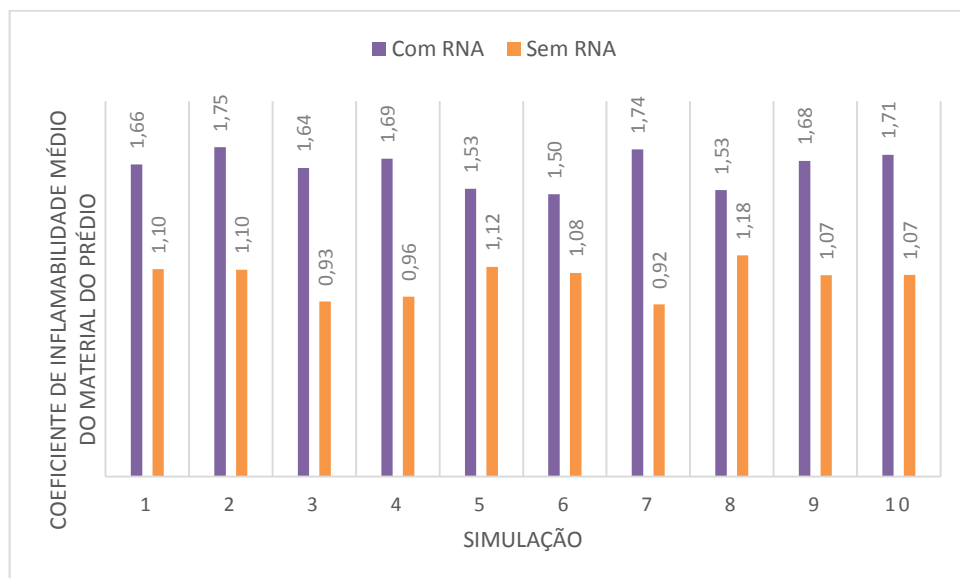


Figura 32 – Intensidade média dos incêndios selecionados



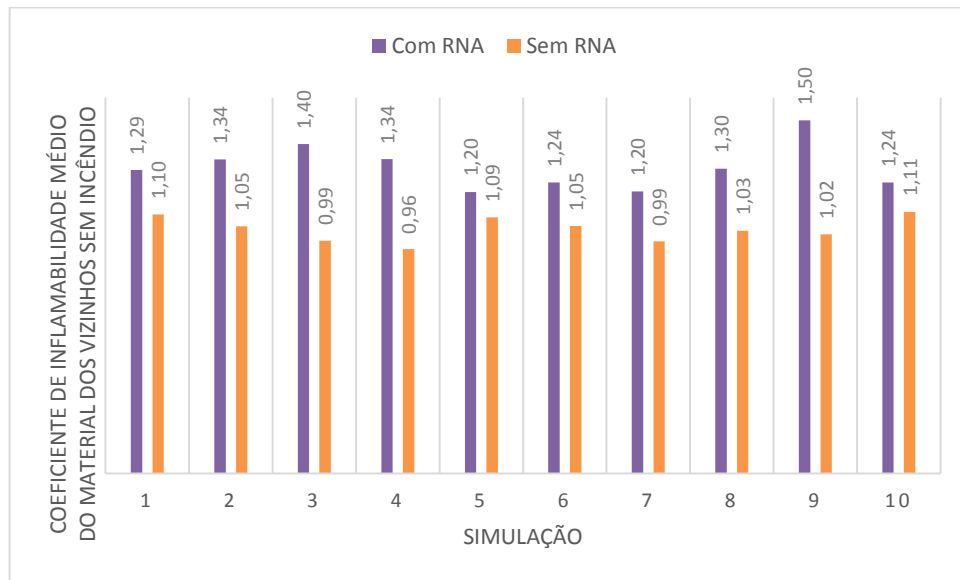
Fonte: (AUTORAS, 2016)

Figura 33 – Coeficiente de inflamabilidade médio do material do prédio dos incêndios selecionados



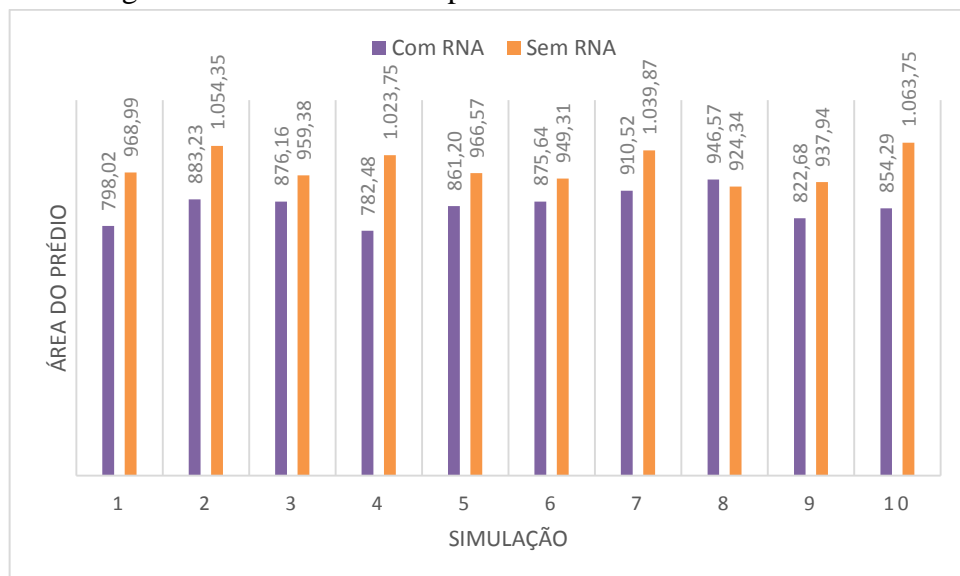
Fonte: (AUTORAS, 2016)

Figura 34 – Coeficiente de inflamabilidade médio do material dos vizinhos sem incêndio dos incêndios selecionados



Fonte: (AUTORAS, 2016)

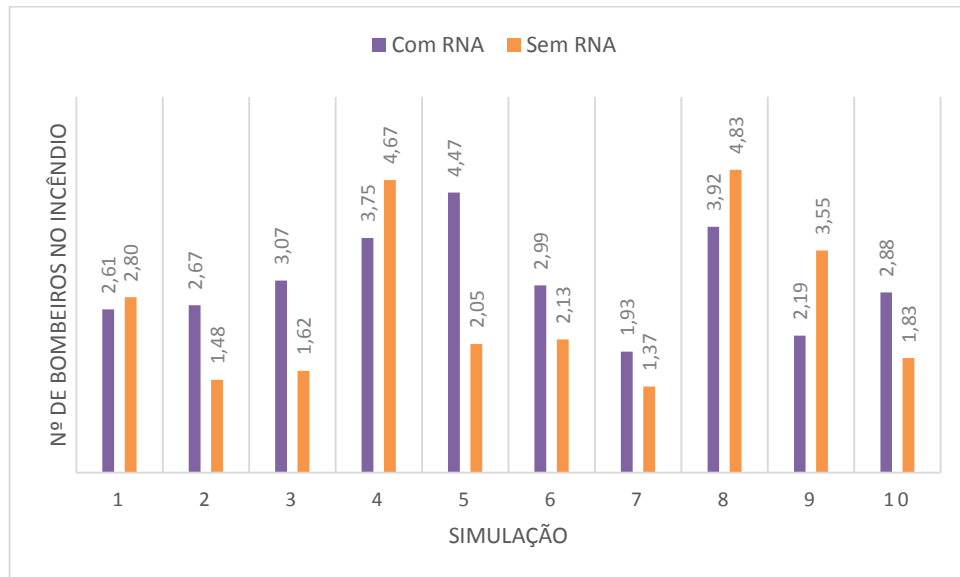
Figura 35 – Área média do prédio dos incêndios selecionados



Fonte: (AUTORAS, 2016)

Comparando os gráficos das Figura 36 e 29, pode-se observar que em 8 de 10 casos, a estratégia que fez com que muitos bombeiros combatessem o mesmo incêndio foi a que obteve menor área do mapa queimada ao final da simulação.

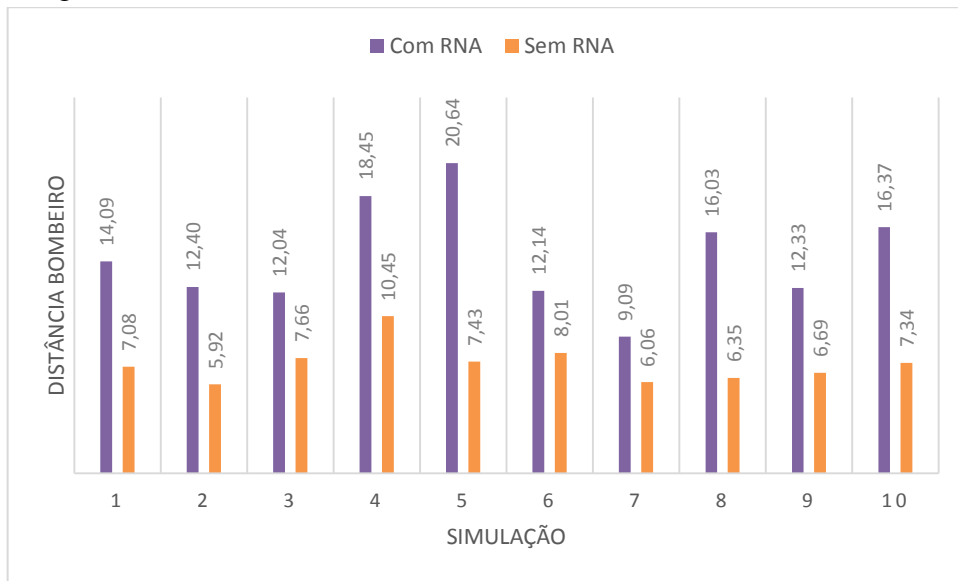
Figura 36 – N° de bombeiros combatendo os incêndios selecionados em média



Fonte: (AUTORAS, 2016)

Na Figura 37 pode-se observar que a RNA deu pouca importância para a distância entre o bombeiro e o incêndio, uma vez que em todas as simulações, a diferença de distâncias foi grande entre a estratégia gulosa do bombeiro simples (sempre selecionando o incêndio com menor distância) e a estratégia da RNA. Considerando os gráficos exibidos anteriormente, nas Figuras 32, 33, 34 e 35, os bombeiros inteligentes deram preferência para as características do incêndio, do prédio e de seus vizinhos ao invés da distância até ele.

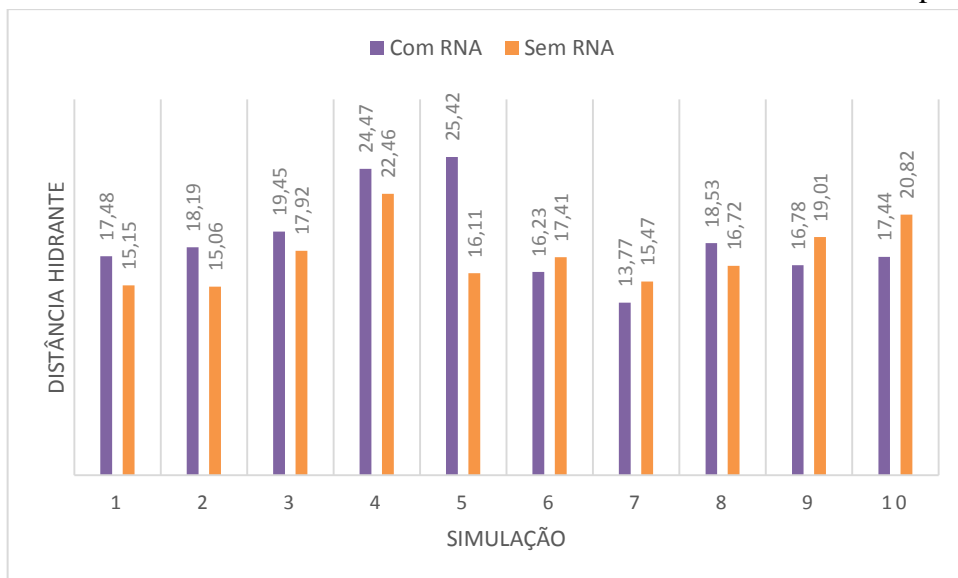
Figura 37 – Distância média entre o incêndio selecionado e o bombeiro



Fonte: (AUTORAS, 2016)

A distância entre o incêndio e o hidrante (Figura 38) parece ser a característica menos relevante dentre os considerados pela RNA, pois ela não apresenta padrão aparente, e ainda em casos extremos como o da simulação 5, onde a área queimada obtida pela estratégia da RNA foi a menor de todas, a distância média entre os incêndios e o hidrante mais próximo foi a maior de todas. Seria esperado que incêndios distantes de hidrantes seriam os que mais penalizariam a área queimada, devido ao tempo perdido durante os reabastecimentos de água.

Figura 38 – Distância média entre o incêndio selecionado e o hidrante mais próximo



Fonte: (AUTORAS, 2016)

Para que seja possível uma análise mais profunda dos resultados, em cada simulação é gerado um arquivo de relatório em formato JSON. Este arquivo possui as configurações utilizadas para realizar a simulação e o estado do ambiente a cada ciclo. Caso tenha sido utilizada uma RNA na simulação, a taxa de aprendizagem e o número de neurônios na camada oculta são armazenados. A cada ciclo são armazenadas a área queimada do mapa e o número de incêndios apagados desde o início da simulação até o momento, e o número de incêndios existentes no ciclo atual, conforme Figura 39. Além disso, caso algum bombeiro tenha selecionado um incêndio, os dados deste são armazenados, como pode ser observado na Figura 40. Estes dados disponíveis no relatório JSON podem ser analisados, em trabalhos futuros, por algoritmos de descoberta de padrões, para identificar relações entre as variáveis e a tomada de decisão dos bombeiros.

Figura 39 – Dados armazenados a cada ciclo no relatório JSON

```
{
  "unixtimestamp": 1478556377880,
  "data": "07-11-2016_20:06:17",
  "areaTotalMapa": 4970920,
  "numCiclos": 300,
  "numBombeiros": 50,
  "numExploradores": 10,
  "numSetores": 10,
  "numInicialDeIncendios": 50,
  "fatorEvolucao": 0.05,
  "fatorEspalhamento": 0.05,
  "fatorSurgimento": 0.05,
  "fatorVelocidadeDeQueima": 2,
  "numNeuroniosCamadaOculta": 25,
  "taxaAprendizado": 0.1,
  "ciclos": [
    {
      "cicloAtual": 1,
      "areaQueimadaAteAgora": 208,
      "incendiosApagadosAteAgora": 0,
      "incendiosExistentesNoMomento": 50,
      "incendiosSelecionadosNesteCiclo": [
        ]
      ]
    }
  ],
}
```

Fonte: (AUTORAS, 2016)

Figura 40 – Dados dos incêndios selecionados no relatório JSON

```

{
  "cicloAtual": 8,
  "areaQueimadaAteAgora": 1762,
  "incendiosApagadosAteAgora": 0,
  "incendiosExistentesNoMomento": 61,
  "incendiosSelecionadosNesteCiclo": [
    {
      "intensidade": 0,
      "materialPredio": 2,
      "areaPredio": 536,
      "numBombeirosNoIncendio": 0,
      "distanciaHidrante": 33,
      "distanciaBombeiro": 20,
      "materialMedioVizinhos": 2
    },
    {
      "intensidade": 3,
      "materialPredio": 2,
      "areaPredio": 956,
      "numBombeirosNoIncendio": 0,
      "distanciaHidrante": 9,
      "distanciaBombeiro": 14,
      "materialMedioVizinhos": 1.25
    }
  ]
}

```

Fonte: (AUTORAS, 2016)

### 3.5 Considerações

Considerando os dados apresentados anteriormente, pode-se concluir que a RNA aprendeu a valorizar as características do incêndio (intensidade) e do local onde ele está (área e material do prédio, e material dos vizinhos diretos). Porém, estas características não são suficientes para determinar em todas as situações qual é o melhor incêndio para ser apagado. Considerar características de uma região mais ampla ao redor do incêndio, e não só dos vizinhos diretos, pode contribuir para melhorar a decisão da RNA, pois um incêndio pode se espalhar para além de seus vizinhos diretos, tornando relevantes as características dos prédios das redondezas.

Em relação ao trabalho de Barroso *et al.* (2013), este trabalho apresentou uma melhoria na estratégia de exploração do mapa, através da sua divisão em setores utilizando a clusterização de *K-means* (inspirada pelo trabalho de Liang, Yang e Deng (2014)), que considera a distribuição dos prédios pelo mapa. Além de melhorar a estratégia de exploração, este trabalho propôs a substituição da heurística utilizada em Barroso *et al.* (2013) por uma RNA para realizar a seleção do incêndio. Após a validação, percebeu-se que o fator de distância, presente na heurística de Barroso *et al.* (2013), que serviu como inspiração para determinar algumas das entradas da RNA, não foi um dos fatores determinantes para a seleção de qual incêndio combater. Além disso, em comparação com o mesmo trabalho, foi identificado um comportamento contrário na RNA em relação à intensidade do incêndio, cuja

ênfase foi na preferência para incêndios menores (mais fáceis de apagar), enquanto a heurística original dava preferência para incêndios com intensidades mais altas. Em relação à área do prédio em chamas, tanto a RNA quanto a heurística deram preferência para prédios com áreas menores.

Após a análise quantitativa (gráficos) e qualitativa, pode-se concluir que a estratégia de seleção de incêndios da RNA, aliada à clusterização de *K-means* para exploração do mapa, se mostrou adequada para o domínio de combate a incêndios em situações de terremotos com multiagentes, pois foi capaz de aprender a importância de cada um dos fatores para realizar a seleção de quais incêndios combater.

## 4 CONCLUSÃO

O trabalho desenvolvido contribui para o trabalho de equipes de resgate durante uma situação pós-terremoto, no que diz respeito à atuação de bombeiros no combate a incêndios gerados pelo rompimento de tubulações de gás e de quedas de linhas de força. Embora o enfoque deste trabalho não tenha sido nas formas de apagar o incêndio em si, sua contribuição para este domínio é relevante, uma vez que a alocação de tarefas possui grande impacto no desempenho das equipes.

Assim, foi apresentada uma estratégia de aprendizagem, baseada em redes neurais artificiais e clusterização, em agentes inteligentes para atuação em áreas atingidas por terremotos, com foco no problema de combate a incêndios urbanos.

Após as análises feitas na seção de validação, concluiu-se que a estratégia aprendida pela RNA resulta em menor área queimada em relação à estratégia gulosa na maioria dos casos. O total de incêndios apagados em média pela RNA é pouco maior do que o da estratégia gulosa. Porém, apesar do número semelhante de incêndios apagados em ambas as estratégias, a RNA claramente selecionou melhor quais incêndios apagar, resultando em um número menor de incêndios existentes ao final da maioria das simulações. Além disso, os agentes bombeiros inteligentes aprenderam a valorizar a intensidade do incêndio, material do prédio, área do prédio e material dos vizinhos, no momento de selecionar qual incêndio combater. Porém, esta estratégia ainda pode ser melhorada, pois houve dois casos em que ela não resultou na menor área queimada. Um parâmetro que pode ser levado em consideração para melhorar este desempenho é a característica da região do incêndio, utilizando mais do que apenas as informações dos vizinhos diretos do incêndio (como foi feito neste trabalho).

Como resposta ao problema de pesquisa “Como Redes Neurais Artificiais e Clusterização podem ser utilizadas para melhorar a tomada de decisão de agentes em um cenário multiagentes de simulação de terremotos?”, o trabalho demonstra que é possível utilizar Redes Neurais Artificiais em agentes inteligentes com o objetivo de selecionar incêndios a serem combatidos. Também foi demonstrado como a exploração do ambiente pode ser auxiliada pela clusterização de *K-means*, através da divisão do mapa em setores.

No Quadro 3, apresentado na seção 2.4.7 do capítulo 2, há uma comparação entre os trabalhos relacionados estudados. Retomando-se a tabela, inclui-se aqui as características da aplicação desenvolvida.



Trabalhos Relacionados	Objetivo	Ambiente Utilizado / Aplicação	Tipo de Desastre	Técnica de Aprendizado Utilizada	Outras Técnicas	Resultados
FONTOURA e FROZZA (2016)	Desenvolver uma estratégia de aprendizagem, baseada em redes neurais artificiais e clusterização, em agentes inteligentes.	Simulador próprio.	Terremoto.	Redes Neurais Artificiais e Clusterização de <i>K-Means</i> .	Não utiliza.	Os agentes bombeiros inteligentes superaram versões simples com algoritmo guloso de seleção de incêndios em 8 de 10 casos.

Como trabalhos futuros, sugerem-se:

- Alterar as entradas da RNA para que sejam analisadas outras características do incêndio, como remover a distância entre o hidrante e o incêndio, e incluir as características da região do mapa onde o incêndio encontra-se (áreas e materiais dos prédios próximos, por exemplo).
- Desenvolver uma aplicação para encontrar padrões nos dados incluídos no relatório de saída de cada simulação, buscando identificar padrões que possam ajudar a desenvolver uma estratégia com melhor desempenho no combate aos incêndios.
- Incluir no simulador desenvolvido, mais funcionalidades para simular um ambiente de terremotos, como bloqueio de ruas, desabamentos, resgate de vítimas e vento, que pode influenciar no espalhamento dos incêndios.
- Realizar comparações com outros simuladores de situações de desastres do tipo terremoto, como o da *Robocup Rescue*, por exemplo, para identificar vantagens ou desvantagens de se utilizar cada um.

## REFERÊNCIAS

- ALONSO, E.; D'INVERNO, M.; KUDENKO, D.; LUCK, M.; NOBLE, J. *Learning in multi-agent systems*. In: The Knowledge Engineering Review. Cambridge University Press. 2001. p. 277-284. Disponível em: <<http://eprints.whiterose.ac.uk/1227/>>, acessado em 11 de março de 2016.
- BARBER, K. S.; LIU, T. H.; GOEL, A.; MARTIN, C. E. *Conflict Representation and Classification in a Domain-Independent Conflict Management Network*. The Laboratory for Intelligent Processes and Systems. The University of Texas at Austin. 1998. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.42.9288&rep=rep1&type=pdf>>, acessado em 21 de janeiro 2016.
- BARROSO, Alan D.; SANTANA, Felipe de C.; LASSANCE, Victor; SILVA, Annibal B. M. da; NARDIN, Luis G.; BRANDÃO, Anarosa A. F.; SICHMAN, Jaime S. *RoboCup Rescue 2013 LTI Agent Rescue Team Description*. 2013. Disponível em <[https://www.ida.liu.se/~TDDD10/papers/rsl-agents/rsl2013\\_submission\\_6.pdf](https://www.ida.liu.se/~TDDD10/papers/rsl-agents/rsl2013_submission_6.pdf)>, acessado em 21 de abril de 2016.
- BASAK, S.; MODANWAL, N.; MAZUMDAR, B. D. *Multi-Agent Based Disaster Management System: A Review*. International Journal of Computer Science and Technology (IJCT), Vol 2, Issue 2. 2011.
- CARDONA, Omar Darío. *El Manejo de Riesgos y los Preparativos para Desastres: Compromiso institucional para mejorar la calidad de vida*. In: Desastres: modelo para armar. p. 128 – 147. La RED. Lima. 1996. Disponível em <[ipcc-wg2.gov/nj-lite\\_download.php?id=6682](http://ipcc-wg2.gov/nj-lite_download.php?id=6682)>, acessado em 18 de janeiro de 2016.
- CASTRO, Antônio Luiz Coimbra de. *Manual de planejamento em defesa civil*. Vol. 1. Ministério da Integração Nacional. Brasília. 1999. Disponível em <[www.defesacivil.mg.gov.br/conteudo/arquivos/manuais/Manuais-de-Defesa-Civil/Manual-PLANEJAMENTO-1.pdf](http://www.defesacivil.mg.gov.br/conteudo/arquivos/manuais/Manuais-de-Defesa-Civil/Manual-PLANEJAMENTO-1.pdf)>, acessado em 16 de janeiro de 2016.
- FARINELLI, A.; GRISETTI, G.; IOCCHI, L.; CASCIO, S. L.; NARDI, D. *Design and Evaluation of Multi Agent Systems for Rescue Operations*. In: International Conference on Intelligent Robots and Systems, IEEE. 2003.
- FRANKLIN, Stan; GRAESSER, Art. *Is it an Agent, or Just a Program? A Taxonomy for Autonomous Agents*. In: Proceedings of the Workshop on Intelligent Agents III, Agent Theories, Architectures, and Languages. Springer-Verlag. Londres. 1996.
- GENCER, E. A. *The Interplay between Urban Development, Vulnerability and Risk Management: A Case Study of the Instabul Metropolitan Area*. Springer. 2013.
- GIL, Antonio Carlos. *Como elaborar projetos de pesquisa*. São Paulo, v. 5, p. 61, 2002.
- GOODWIN, Richard. *Formalizing Properties of Agents*. Carnegie Mellon University. Pittsburgh. 1993. Disponível em: <[www.agent.ai/doc/upload/200302/good93.pdf](http://www.agent.ai/doc/upload/200302/good93.pdf)>, acessado

em 10 de janeiro de 2016.

GUHA-SUPIR, D.; BELOW, R.; HOYOIS, P.; *EM-DAT: International Disaster Database. Université Catholique de Louvain – Brussels – Belgium*. Disponível em <<http://www.emdat.be>>, acessado em 17 de dezembro de 2015.

GUHA-SUPIR, D.; VOS, F. *Human Casualties in Earthquakes*. In: *Advances in Natural and Technological Hazards Research*. Vol. 29. p. 13-24. Springer Science+Business Media. 2011

HAYKIN, Simon; *Neural Networks: A Comprehensive Foundation*. 2ª Ed. Pearson Education. 1999.

HUHNS, Michael N.; STEPHENS, Larry M. *Multiagent Systems and Society of Agents*. In: *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. p. 79-120. The MIT Press. Cambridge. 1999

JAIN, Anil K.; DUBES, Richard C. *Algorithms for Clustering Data*. Prentice Hall. New Jersey. 1988.

JENNINGS, Nicholas R.; SYCARA, Katia; WOOLDRIDGE, Michael. *A Roadmap of Agent Research and Development*. In: *Autonomous Agents and Multi-Agent Systems*. Vol 1. p. 275-306. Kluwer Academic Publishers. Boston. 1998. Disponível em <[www.cs.ox.ac.uk/people/michael.wooldridge/pubs/jaamas98.pdf](http://www.cs.ox.ac.uk/people/michael.wooldridge/pubs/jaamas98.pdf)>, acessado em 21 de janeiro de 2016.

KHOUIJ, Mohammed; LÓPEZ, César; SARKARIA, Sarbjit; MARTI, José. *Disaster Management in Real Time Simulation Using Machine Learning*. In: *2011 24<sup>th</sup> Canadian Conference on Electrical and Computer Engineering*. IEEE. 2011. p. 1507-1510.

KITANO, H.; TADOKORO, S. *RoboCup Rescue A Grand Challenge for Multiagent and Intelligent Systems*. In: *AI Magazine*, Vol. 22, Number 1. AAAI. 2001.

KOBIYAMA, Masato; MENDONÇA, Magaly; MORENO, Davis Anderson; MARCELINO, Isabela P. V. de Oliveira; MARCELINO, Emerson V.; GONÇALVES, Edson F.; BRAZETTI, Leticia Luiza Penteadó; GOERL, Roberto Fabris; MOLLERI, Gustavo, S. Fontes; RUDORFF, Frederico de Moraes. *Prevenção de desastres naturais: conceitos básicos*. Editora Organic Trading. Curitiba. 2006.

LIANG, Zhiwei; YANG, Xiang; DENG, Yao. *A partitioning-based task allocation strategy for Police Multi-Agents*. In: *The 26<sup>th</sup> Chinese Control and Decision Conference*. IEEE. 2014. p. 2124-2128.

MARCELINO, Emerson Vieira. *Desastres Naturais e Geotecnologias: Conceitos Básicos*. INPE. São José dos Campos. 2008. Disponível em <<http://mtc-m16c.sid.inpe.br/col/sid.inpe.br/mtc-m18@80/2008/07.02.16.22/doc/publicacao.pdf>>, acessado em 16 de janeiro de 2016.

MATSUNO, Fumitoshi; TADOKORO, Satoshi. *Rescue Robots and Systems in Japan*. In: *International Conference on Robotics and Biomimetics*. Shenyang. 2004.

- MICHALSKI, Ryszard S.; CARBONELL, Jaime G.; MITCHELL, Tom M. *Machine Learning: An Artificial Intelligence Approach*. TIOGA Publishing Co. Palo Alto. California. 1983.
- PARKES, David C.; UNGAR, Lyle H. *Learning and Adaption in Multiagent Systems*. In: Multiagent learning: Papers from the 1997 AAAI Workshop. AAAI Press. 1997. Disponível em < <http://www.eecs.harvard.edu/econcs/pubs/learning.pdf>>, acessado em 29 de maio de 2016.
- PAULAY, T.; PRIESTLEY, M. J. N. *Seismic Design of Reinforced Concrete and Masonry Buildings*. John Wiley & Sons. 1992.
- PEREIRA, André H.; NARDIN, Luis Gustavo; SICHMAN, Jaime Simão. *Coordination of Agents in the RoboCup Rescue: A Partial Global approach*. In: 2011 Workshop and School of Agent Systems, their Environment and Applications. IEEE. 2011. p. 45-50.
- PESSIN, Gustavo; OSÓRIO, Fernando; MUSSE, Soraia. *Simulação Virtual da Evolução de Estratégias e do Controle Inteligente em Sistemas Multi-Robóticos*. X Symposium of Virtual and Augmented Reality. 2008.
- PESSIN, Gustavo; OSÓRIO, Fernando; MUSSE, Soraia; NONNEMMACHER, Vinícius; FERREIRA, Sandro Souza. *Utilizando Agentes Autônomos com Aprendizado para a Identificação e Combate de Incêndios em Áreas Florestais*. 2007.
- PRADO, J. P. A.; FREITAS, R. L.; GUILHERME, I. R.; NAKAMITI, G. S.; RILLO, Márcio. *Inteligência Artificial Distribuída em Ambientes Reativos*. In: I Simpósio Brasileiro de Automação Inteligente (SBAI), 1993. Disponível em: <[http://fei.edu.br/sbai/SBAI1993/ARTIGOS/I\\_SBAI\\_35.pdf](http://fei.edu.br/sbai/SBAI1993/ARTIGOS/I_SBAI_35.pdf)>, acessado em 09 de janeiro de 2016.
- RAMCHURN, Sarvapali D.; HUYNH, Trung Dong; IKUNO, Yuki; FLANN, Jack; WE, Feng; MOREAU, Luc; JENNINGS, Nicholas R.; FISCHER, Joel E.; JIANG, Wenchao; RODDEN, Tom; SIMPSON, Edwin; REECE, Steven; ROBERTS, Stephen. *HAC-ER: A Disaster Response System Based on Human-Agent Collectives*. In: Proceedings of the 14<sup>th</sup> International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015). 2015. Disponível em: < <http://www.robots.ox.ac.uk/~sjrob/Pubs/fp634-ramchurn.pdf>>, acessado em 12 de março de 2016.
- RAMCHURN, Sarvapali D.; WU, Feng; JIANG, Wenchao; FISCHER, Joel E.; REECE, Steve; ROBERTS, Stephen; RODDEN, Tom; GREENHALGH, Chris; JENNINGS, Nicholas R. *Human-agent collaboration for disaster response*. In: Autonomous Agent and Multi-Agent Systems, 30. Springer. 2016. p. 82-111.
- RAUBER, Thomas Walter. *Redes Neurais Artificiais*. Encontro Regional de Informática. 1997. Disponível em <<http://www.inf.ufes.br/~thomas/pubs/eri98.pdf>>, acessado em 27 de maio de 2016.
- REZENDE, Solange Oliveira. *Sistemas Inteligentes: Fundamentos e Aplicações*. p. 269-294. Editora Manole Ltda. Barueri. 2003.
- RIBEIRO, Paula Clark. *Modelagem e Implementação OO de Sistemas Multi-Agentes*.

Departamento de Informática. Pontifícia Universidade Católica do Rio de Janeiro. 2001. Disponível em: <[ftp://inf.puc-rio.br/pub/docs/theses/01\\_MSc\\_ribeiro.pdf](ftp://inf.puc-rio.br/pub/docs/theses/01_MSc_ribeiro.pdf)>, acessado em 09 de janeiro de 2016.

RUSSEL, Stuart J.; NORVIG, Peter. *Artificial Intelligence: A Modern Approach*. 3<sup>rd</sup> Edition. Pearson, Upper Saddle River, 2010.

SANCHES, Marcelo Kaminski. *Aprendizado de máquina semi-supervisionado: proposta de um algoritmo para rotular exemplos a partir de poucos exemplos rotulados*. Dissertação de Mestrado. USP. 2003. Disponível em <[http://www.teses.usp.br/teses/disponiveis/55/55134/tde-12102003-140536/publico/Dissertacao\\_MKS.pdf](http://www.teses.usp.br/teses/disponiveis/55/55134/tde-12102003-140536/publico/Dissertacao_MKS.pdf)>, acessado em 27 de maio de 2016.

SHOHAM, Yoav; LEYTON-BROWN, Kevin. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. 2010. p. 199-200. Disponível em: <<http://www.masfoundations.org/>>, acessado em 12 de março de 2016.

SICHMAN, Jaime Simão; DEMAZEAU, Yves; BOISSIER Olivier; *When can knowledge-based systems be called agents?* In: IX Simpósio Brasileiro de Inteligência Artificial (SBIA). 1992. Disponível em: <[citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.53.5888&rep=rep1&type=pdf](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.53.5888&rep=rep1&type=pdf)>, acessado em 09 de janeiro de 2016.

SILVA, A. B. M. da; NARDIN, L. G.; SICHMAN, J. S. *Um Método Baseado em Particionamento para Exploração de Ambientes de Desastre*. IX Encontro Nacional de Inteligência Artificial (ENIA), In: Brazilian Conference on Intelligent Systems 2012. 2012.

SINGH, V. K.; MODANWAL, N.; BASAK, S. *MAS Coordination Strategies and their Application in Disaster Management Domain*. 2<sup>nd</sup> International Conference on Intelligent Agent and Multi-Agent Systems (IAMA), IEEE. 2011.

SINWAR, D.; KAUSHIK, R. *Study of Euclidean and Manhattan Distance Metrics using Simple K-Means Clustering*. In: International Journal for Research in Applied Science and Engineering Technology (IJRASET). Vol. 2. Issue 5. 2014. Disponível em: <<http://www.ijraset.com/files/serve.php?FID=467>>, acessado em 01 de novembro de 2016.

TOMINAGA, Lídia Keiko; SANTORO, Jair; AMARAL, Rosângela do. *Desastres Naturais: Conhecer para prevenir*. 2<sup>a</sup> Edição. Instituto Geológico. São Paulo. 2012.

USGS, United States Geological Survey. *Largest Earthquakes in the World Since 1900*. 2016. Disponível em: <[http://earthquake.usgs.gov/earthquakes/world/10\\_largest\\_world.php](http://earthquake.usgs.gov/earthquakes/world/10_largest_world.php)>. Acessado em 20 de janeiro de 2016.

\_\_\_\_\_. United States Geological Survey. *The Severity of an Earthquake*. 1989. Disponível em: <<http://pubs.usgs.gov/gip/earthq4/severitygip.html>>. Acessado em 20 de janeiro de 2016.

UWI. The University of the West Indies. *Hazardous Earthquake Effects*. 2016. Disponível em <<http://www.uwiseismic.com/General.aspx?id=14>>, acessado em 25 de janeiro de 2016.

WALLACE, Robert E. *The San Andreas Fault System, California*. United States Geological

Survey. p. 195-196. United States Government Printing Office. Washington. 1990. Disponível em: <<http://pubs.er.usgs.gov/publication/pp1515>>, acessado em 21 de janeiro de 2016.

WEISS, Gerhard. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. p. 29-78. MIT Press. Cambridge. 1999.

WOOLDRIDGE, Michael. *An Introduction to Multiagent Systems*. John Wiley & Sons, 2002.

WOOLDRIDGE, Michael; JENNINGS, Nicholas R. *Intelligent Agents: Theory and Practice*. In: Knowledge Engineering Review. Vol. 10. Issue 02. p. 115-152. 1995.

YIN, Robert K. *Estudo de Caso: Planejamento e Métodos*. Editora Bookman. 2015.