

Evolução de Estratégias e Controle Inteligente em Sistemas Multi-Robóticos Robustos

Gustavo Pessin¹, Fernando Osório² (Orientador)

¹ UNIC – Primavera do Leste, Av Guterres 241, Primavera do Leste, MT, Brasil

² USP – Universidade de São Paulo / ICMC - SSC - São Carlos, SP, Brasil
pessin@gmail.com, fosorio@icmc.usp.br

Resumo. Este trabalho está relacionado com a aplicação de técnicas de Inteligência Artificial no desenvolvimento de um Sistema Multi-Agente Robótico, o Robombeiros¹, aplicado ao problema da monitoração e combate a incêndios em áreas florestais. O objetivo macro é evoluir estratégias de formação de equipes de combate a incêndio (unidade de controle) e criar métodos robustos de navegação em agentes robóticos (unidades de combate), considerando um ambiente virtual de simulação realística. O ambiente virtual 3D suporta uma série de características fundamentais para a simulação realística da operação, como terrenos irregulares, processos naturais e restrições físicas na criação e uso de robôs móveis. A operação multi-agente depende essencialmente de duas etapas: planejamento e ação. No planejamento, usamos Algoritmos Genéticos (AG) para evoluir estratégias de posicionamento de atuação dos robôs bombeiros. Para a ação foram criados robôs de combate a incêndios fisicamente simulados, sendo as informações sensoriais de cada robô (*e.g.* GPS, bússola, sonar) usadas na entrada de uma Rede Neural Artificial (RNA). Esta RNA que controla os atuadores do veículo permite a navegação robusta com desvio de obstáculos. Os resultados de diversos experimentos demonstram que a RNA é capaz de controlar de modo plenamente satisfatório os robôs móveis e, que o uso de AGs permite definir uma estratégia de combate a incêndios, obtendo posições muito satisfatórias de ataque ao foco do incêndio. O sistema multi-agente proposto pode vir a ter um papel muito importante no planejamento e execução de operações reais de combate a incêndios florestais, como foi demonstrado pelo sucesso das simulações executadas pela equipe de robôs bombeiros.

Palavras-chave: Sistema Multi-Agente, Rede Neural Artificial, Algoritmos Genéticos, Simulação, Ambiente Virtual 3D, Robôs Móveis Autônomos.

Gustavo Pessin foi aluno de mestrado na UNISINOS/RS, concluindo o curso em fevereiro de 2008. A cópia da dissertação pode ser obtida em: <http://pessin.googlepages.com/disspessin2008.pdf>

1 Introdução

Com a evolução das pesquisas em robótica, cada vez mais os robôs estão se tornando complexos em termos físicos. A grande variedade de estudos em morfologia robótica tem desenvolvido variações de robôs dotados de diversos meios de locomoção (*e.g.* pernas, rodas, esteiras). Em paralelo a este desenvolvimento temos a evolução constante de uma gama extremamente grande de sensores (*e.g.* sistemas de visualização, posicionamento, detecção de obstáculos). O desenvolvimento de algoritmos e técnicas para coordenar estes conjuntos físicos em um ambiente dinâmico é um desafio extremamente complexo [1]. Dotar robôs autônomos de capacidade de

¹ Código-fonte disponível em <http://pessin.googlepages.com>

raciocínio inteligente e de interação com o meio em que estão inseridos é uma área de pesquisa que tem atraído a atenção de um grande número de pesquisadores [2].

Existem diversas áreas onde a habilidade de um único agente não é suficiente ou eficiente para a realização de uma tarefa, em alguns destes casos, como patrulhamento, vigilância, resgate ou exploração o mais indicado é a aplicação de sistemas multi-robóticos. Sistemas multi-robóticos são sistemas onde robôs autônomos trabalham cooperativamente a fim de cumprir uma missão, podendo existir interação entre os robôs ou não [4]. O trabalho apresentado neste artigo envolve o uso de uma variedade de tecnologias de diferentes campos; a construção do sistema Robombeiros serviu como uma boa ferramenta de estudo para avaliar a eficiência da arquitetura multi-agente, da estratégia cooperativa, da fusão de sensores e da modelagem robótica. A tarefa em si poderá ser generalizada para outras atividades práticas como acidentes nucleares ou desastres ambientais. O uso de técnicas de aprendizado de máquina permitiu obter agentes que suportam melhor um ambiente dinâmico, tornando o sistema mais flexível e autônomo possível.

Para que seja possível a implementação física real, o sistema multi-agente que propomos deve ser projetado, desenvolvido e testado anteriormente em ambientes de simulação realísticos. Ambientes com terrenos 3D gerados com a biblioteca de programação *Open Scene Graph* (OSG) [18] junto com a biblioteca de programação *Demeter* [19] permitem combinar o mapa de elevação juntamente com uma determinada distribuição de vegetação, criando assim um terreno bastante realístico [13]. Além disso, o uso da biblioteca de programação *Open Dynamics Engine* (ODE) [17] permite implementar atributos físicos (*e.g.* atrito, fricção, gravidade, colisão) tornando o sistema ainda mais realístico.

Neste artigo apresentamos na Seção 2 uma pequena conceituação teórica de simulação e modelagem, abordando robótica e operações reais de identificação e combate de incêndios florestais. A Seção 3 apresenta conceitos de Aprendizado de Máquina e de técnicas de IA. Na Seção 4 descrevemos o ambiente desenvolvido, a operação multi-agente de identificação e combate de incêndios, o desenvolvimento, a aplicação e a avaliação das técnicas de IA propostas. Finalizamos apresentando as conclusões e perspectivas do trabalho realizado.

2 Simulação e Modelagem

Experimentos em robótica móvel podem ser realizados de duas formas: diretos em um robô real ou em um robô simulado em um ambiente virtual realístico [11]. Usualmente, experimentos em robótica móvel utilizando um robô real demandam um enorme investimento de tempo e de recursos financeiros. Para que seja possível a implementação física real, o sistema multi-agente que propomos deve ser projetado, desenvolvido e testado anteriormente em ambientes de simulação realísticos. A simulação de sistemas robóticos é especialmente necessária para robôs caros, grandes, ou frágeis [1], sendo uma ferramenta extremamente poderosa para agilizar o ciclo de desenvolvimento de sistemas de controle robóticos eliminando desperdício de recursos, tanto financeiros como computacionais. Para que uma simulação seja útil, entretanto, ele deve capturar características importantes do mundo físico, onde o termo *importantes* tem relação ao problema em questão [1].

A fim de melhor entender como proceder no combate a incêndios florestais e assim planejar as estratégias a serem implementadas nos agentes autônomos, foi realizado um amplo estudo sobre técnicas reais de operação, combustíveis florestais e modelos de propagação de incêndios em florestas. Este estudo está detalhado em [20] e [23]. Portanto, neste trabalho foi implementado um sistema de simulação física bastante realista, que busca modelar e simular: (i) a propagação de incêndios em florestas; (ii) o planejamento de trajetórias e estratégias de combate a incêndios por parte de uma brigada robótica; (iii) a navegação autônoma e robusta dos robôs inseridos no ambiente simulado (evitando colisões e buscando atingir seus objetivos).

3 Aprendizado de Máquina

Aprendizado de Máquina (*Machine Learning* - ML) é uma área da Inteligência Artificial que tem como objetivo desenvolver técnicas computacionais de aprendizado e de aquisição de conhecimentos [5]. Essas técnicas devem exibir um comportamento inteligente e realizar tarefas complexas com um nível de competência equivalente ou superior ao de um especialista humano [6]. Para a construção do sistema proposto neste artigo utilizamos Redes Neurais Artificiais e Algoritmos Genéticos, assim, descrevemos sucintamente suas características a seguir.

Uma técnica de IA muito utilizada no controle de navegação de sistemas reativos [12] é a de Redes Neurais Artificiais (RNA). RNAs são sistemas paralelos distribuídos, compostos por unidades de processamento simples que calculam determinadas funções matemáticas, normalmente não-lineares [9]. Matematicamente, RNAs são aproximadores universais, que realizam mapeamentos em espaços de funções multi-variáveis [3]. A capacidade de aprender e generalizar das RNAs é um dos seus maiores atrativos. O treinamento de uma RNA deve envolver várias rodadas de simulação, iniciando os pesos de forma aleatória. Outra questão importante é o grau de generalização, sendo medido usualmente através do uso de uma base de validação usada em paralelo à base de treinamento. O *Stuttgart Neural Network Simulator* (SNNS) [16] foi o simulador de RNAs adotado neste trabalho, possuindo um grande número de algoritmos de aprendizado e recursos de simulação. Um aplicativo do SNNS, o SNNS2C, permite a conversão de uma RNA em código em C, que pode ser inserido em outras aplicações.

Na tarefa de controle de incêndios, uma questão importante tem relação com o posicionamento dos agentes para atuação (visando bloquear a propagação do fogo); de acordo com as capacidades de atuação de cada agente, as condições climáticas (vento, chuva), a topografia e a vegetação, diferentes formações podem ser sugeridas. Estas formações, quando sugeridas por um especialista, devido à complexidade do problema, podem não levar em conta todas as variáveis. Uma técnica de IA indicada para estes casos é a de Algoritmos Genéticos (AG) [7], [8]. AGs são técnicas de otimização global que empregam estratégia de busca paralela e estruturada, embora aleatória, direcionada a busca de pontos de aptidão [9]. Permitem assim a realização de busca multi-critério em um espaço multidimensional, e por serem não supervisionados, não necessitam de nenhuma base de informação de antemão, onde se corretamente utilizados são capazes de escapar de mínimos locais [15]. A GALib (<http://lancet.mit.edu/ga>), adotada neste trabalho, é uma das mais completas e eficientes bibliotecas para programação de AGs.

Um amplo estudo sobre trabalhos relacionados com o foco deste artigo (*e.g.* robôs aplicados ao problema de operação em ambientes hostis, controle robótico, evolução de estratégias, formação de grupos, formas de comunicação) pode ser visto em [23].

4 Operação de Identificação e Combate

O modelo de operação multi-agente para identificação e combate a incêndios foi inicialmente descrito em [20]. Este modelo teve um protótipo 2D desenvolvido com a biblioteca SDL, sendo a operação multi-agente completamente baseada em regras. O modelo de controle baseado em RNA foi inicialmente descrito em [28], aplicado satisfatoriamente em um protótipo 2D. A proposta inicial da RNA aplicada no controle inteligente de veículos autônomos fisicamente simulados (para ação), usando um protótipo 3D com chão plano desenvolvido com a biblioteca ODE pode ser vista em [25] e [29]; posteriormente esta RNA foi aplicada em um protótipo 3D com chão irregular, desenvolvido com as bibliotecas ODE, OSG e Demeter. A proposta do modelo de operação (arquitetura) usando GA e RNA foi inicialmente descrita em [24] e [26]. O modelo de AG para evolução de estratégia (planejamento), desenvolvido com a GALib, foi inicialmente descrito em [27]. O ambiente virtual completo está detalhado em [22] e [23].

A operação multi-agente depende essencialmente de duas etapas: **planejamento** e **ação**. Foram implementados protótipos específicos para cada etapa, desenvolvidos em C++. O protótipo onde é realizada a ação dos robôs bombeiros, controlados por RNAs pode ser visto na Fig. 1(b). Este protótipo tridimensional usa a biblioteca OSG, responsável pela saída gráfica do protótipo, a biblioteca Demeter, responsável pelo terreno irregular e a biblioteca ODE, responsável pelo realismo físico, tanto da morfologia robótica como da colisão entre os objetos presentes no ambiente (*e.g.* robôs, árvores, inclinação de terreno). O uso da biblioteca ODE permite que os robôs simulados fisicamente respeitem questões como gravidade, inércia e atrito. Por exemplo, mantendo uma força f constante nos motores lineares (torque) um veículo terá velocidade v em regiões planas, em regiões de declive terá v maior e em aclives terá v menor. O protótipo onde é realizado o planejamento usa um Algoritmo Genético; não necessita obrigatoriamente de visualização, porém está implementado com a possibilidade de uma saída gráfica 2D desenvolvida com SDL, como mostra a Fig. 4(b). A integração entre os protótipos se dá através de um arquivo texto (simulando a comunicação com os agentes), após realizar a evolução, o **planejador** cria um arquivo com as posições de atuação que é lido na inicialização do módulo de **ação no combate**. A Fig. 1(a) apresenta o diagrama da operação multi-agente.

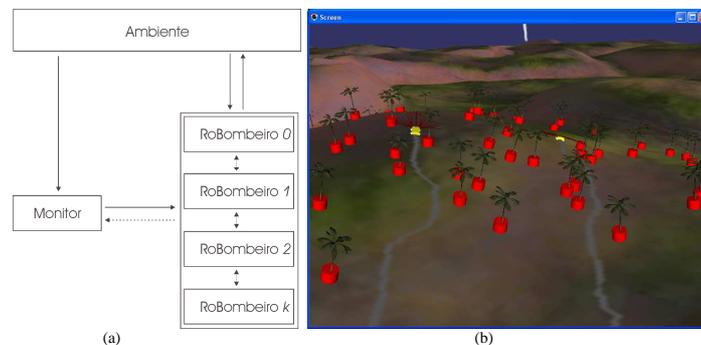


Fig. 1. (a) Diagrama da operação multi-agente; (b) Ambiente de simulação desenvolvido com OSG, ODE e Demeter apresentando dois veículos em deslocamento.

Para a simulação da vegetação e correta propagação do fogo, existe uma matriz oculta sob o terreno. Esta matriz possui, para cada área do terreno, o tipo de vegetação presente, assim, considerando orientação do vento, intensidade do vento e tipo de vegetação de uma área, podemos construir a simulação de propagação do fogo. A velocidade de propagação respeita dados do modelo retirados de [10]. A propagação do fogo busca simular a forma como este se propagaria em um ambiente real, integrando as informações de vegetação, de topografia e de comportamento de propagação. A criação dos mapas teve como base cartas topográficas e o mapa de modelos de combustíveis florestais, que podem ser vistos em [20]. No ambiente desenvolvido, cada árvore existe como um modelo OSG e como um cilindro ODE para teste de colisões. Os sensores percebem os cilindros (Fig. 1(b)) que são posicionados junto às árvores, permitindo assim uma reação do sistema de navegação do robô a fim de evitar colisões. O ambiente de simulação permite optar entre uma exibição gráfica mais realista ou simplificada.

Usamos o ambiente para simular a seguinte operação: um agente monitor (satélite) monitora o terreno da área florestal, ao identificar uma área com foco de incêndio, ativa o módulo de evolução de estratégias (detalhado na Subseção 4.2). Após obter as coordenadas de atuação através do AG, o agente monitor envia mensagens aos agentes de combate informando suas posições de atuação (coordenada GPS). O comportamento dos agentes de combate é híbrido (possuem um plano para execução desta navegação, além de possuir um sistema de reação sensorial-motora), deslocando-se em direção a posição de seu objetivo específico desviando de obstáculos. O método de combate de incêndio simulado é o método indireto [14]. Os agentes de

combate simulados são motoniveladoras que tem como finalidade cercar o foco de incêndio e criar um aceiro (área livre de vegetação onde o fogo se extingue pela falta de combustível). Esta operação pode ser entendida com a Fig. 4. Quanto ao controle de posicionamento, um sensor do tipo GPS é simulado em cada robô. Em experimentos realizados com um *GPS Garmin Etrex* (www.garmin.com) obtivemos um erro médio de $18,6m$. Considerando que cada agente possui seu próprio GPS, o tratamento deste erro é crucial na criação dos aceiros [20], de modo que não ocorram brechas na criação de uma zona de bloqueio ao fogo. O erro médio deste sensor é usado de modo a criar uma margem de segurança (somado as posições estimadas) na criação do aceiro.

4.1 Morfologia dos Robôs Móveis

Os robôs móveis foram desenvolvidos com a biblioteca de simulação de corpos rígidos articulados ODE. As Fig. 3(b) e 3(c) apresentam o veículo desenvolvido. Dada a existência de restrições físicas, a única maneira de controlar este veículo é com a aplicação de forças em seus dois motores simulados, que são: um motor angular (para o giro do volante) e um motor linear (para o torque). Além do GPS, responsável pela obtenção da localização, cada robô possui também uma bússola, necessária para a obtenção da orientação do veículo e cinco sensores de distância (sonares).

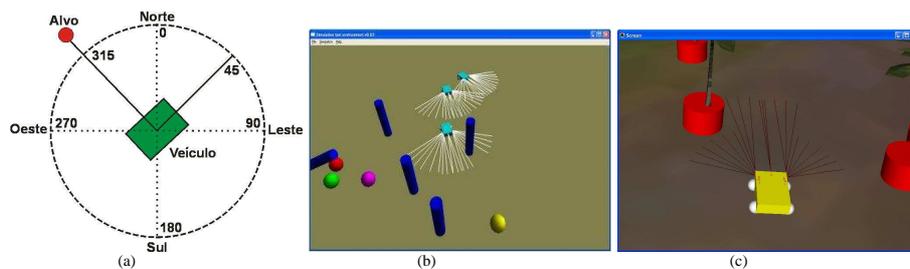


Fig. 2. Sensores desenvolvidos: (a) Bússola e GPS, responsáveis pela orientação, localização e obtenção do azimute; (b) e (c) Sensores de distância.

O azimute (ângulo para o alvo) é obtido a partir da posição atual (GPS) e da posição do alvo (recebida por mensagem), como mostra a Fig. 2(a). Os sensores de distância são sonares simulados, apresentando as características de capacidade de medir distâncias entre $15cm$ e $11m$, como o *Polaroid 6500* (www.senscomp.com).

4.2 Evolução de Estratégias de Posicionamento

O mecanismo de planejamento usa um AG para definir as posições iniciais e finais de atuação de cada robô no combate ao incêndio. Considerando que os agentes de combate são motoniveladoras que tem como finalidade criar um aceiro, necessitamos que o AG retorne as seguintes informações: ângulo inicial e final e raio inicial e final para cada robô, ambos em relação ao ponto inicial do foco de incêndio. Esta operação pode ser entendida com a Fig. 4. O genoma desenvolvido pode ser visto na Tabela 1. Neste, estão presentes informações de todo o grupo envolvido, assim, o tamanho do genoma é dependente da quantidade de agentes no sistema. Realizamos simulações considerando a existência de 4 agentes de combate.

O *fitness* desenvolvido acumula os seguintes valores finais de cada simulação: (i) Total de área queimada: busca minimizar a área queimada; (ii) Total de área com aceiro: busca minimizar a área de trabalho dos robôs, e; (iii) Erro médio absoluto: busca minimizar a diferença entre a média geral de aceiros úteis em relação ao aceiro útil de cada indivíduo, assim, o tamanho das

áreas de trabalho de cada robô tende a se equalizar. Buscamos, na simulação, minimizar o valor do *fitness*. A Fig. 3 apresenta resultados das simulações com diversos tamanhos de população; podemos ver o *fitness* diminuindo com o passar das gerações.

Tabela 1. Genoma dos indivíduos.

Gene	Função	Valor Mínimo	Valor Máximo
0	Posição inicial do agente 0	0,000°	360,000°
1	Posição final do agente 0 e inicial do agente 1	0,000°	360,000°
2	Posição final do agente 1 e inicial do agente 2	0,000°	360,000°
3	Posição final do agente 2 e inicial do agente 3	0,000°	360,000°
4	Posição final do agente 3	0,000°	360,000°
5	Raio inicial do agente 0	0m	100m
6	Raio final do agente 0 e inicial do agente 1	0m	100m
7	Raio final do agente 1 e inicial do agente 2	0m	100m
8	Raio final do agente 2 e inicial do agente 3	0m	100m
9	Raio final do agente 3	0m	100m

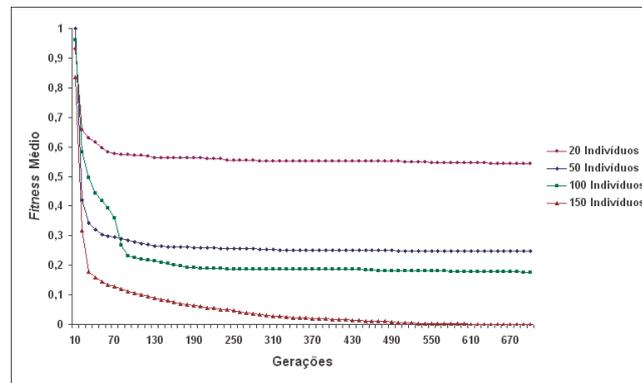


Fig. 3. Evolução do fitness de acordo com o número de gerações para 4 agentes de combate e diferentes quantidades de indivíduos.

O gráfico apresentado na Fig. 3 mostra que o melhor *fitness* é obtido com uma quantidade de indivíduos igual a 150; entre 400 e 700 gerações, a otimização do *fitness* praticamente estabiliza. Os melhores indivíduos resultantes das simulações de evolução de estratégia podem ser vistos na Tabela 2. Podemos ver o resultado satisfatório da aplicação de um destes cromossomos nas Fig. 4(a) e 4(b).

Tabela 2. Cromossomos dos melhores indivíduos resultantes de 3 simulações para 4 agentes; utilizando no AG 150 indivíduos e 700 gerações (genes de 0 a 9).

Simulação	Graus (Genes 0 a 4)					Distâncias (Genes 5 a 9)				
1	225,7	199,2	174,1	155,2	136,3	27,3	30,4	33,9	38,0	35,9
2	224,8	200,7	177,8	160,6	136,6	28,0	29,7	30,5	31,1	33,2
3	134,8	142,6	172,3	196,1	244,8	54,9	42,9	35,8	30,3	26,3

As posições de atuação são calculadas aplicando no cromossomo as Equações 1 e 2. Onde (x_f, y_f) é a coordenada da posição final dos robôs, (x_a, y_a) é a coordenada da posição inicial do incêndio, r_i é o raio (gene 5 a 9) e a_i é o ângulo (gene 0 a 4). O raio, bem como o ângulo, são específicos para cada operação de cada robô (coordenada inicial e final de criação de aceiro, como mostra a Tabela 1).

$$x_f = x_a + r_i \times \cos(a_i) \quad (1)$$

$$y_f = y_a + r_i \times \sin(a_i) \quad (2)$$

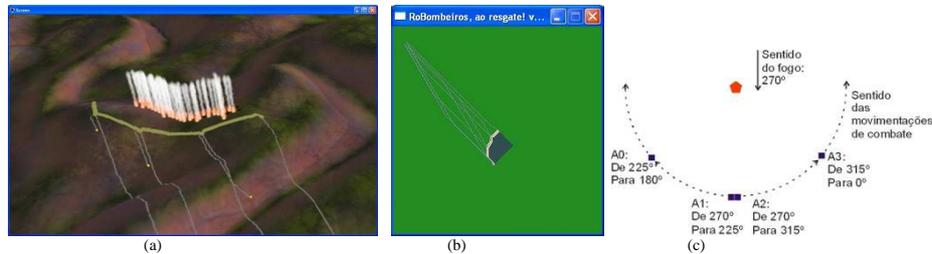


Fig. 4. (a) Resultado do planejamento usando AG com 150 indivíduos e 700 gerações; (b) Mesmo resultado em visualização 2D; (c) Modelo teórico de operação.

4.3 Controle de Navegação

No início da simulação, cada robô recebe uma mensagem do tipo “*desloque-se autonomamente até (x,y)*”. Isso faz com que cada robô inicie a navegação em direção a posição solicitada, sendo controlado apenas pela RNA. O controle inteligente de navegação foi desenvolvido com uma RNA do tipo *Multi Layer Perceptron (MLP)* treinada com o algoritmo de aprendizagem *Resilient Backpropagation (RProp)*. Esta RNA foi desenvolvida e treinada no SNNS. O controle inteligente realizado pela RNA permite a navegação e desvio de obstáculos em um ambiente dinâmico, usando apenas as informações disponíveis localmente, obtidas pelos sensores dos robôs móveis autônomos.

A RNA tem como entradas as seguintes informações: (i) Orientação (ângulo) do veículo, em relação ao plano (x,y) , obtido através de uma bússola simulada; (ii) Azimute (ângulo para o alvo) do veículo, obtido através da bússola, do GPS e da mensagem contendo a coordenada do alvo; (iii) Cinco valores de sensores de distância (sonares). As saídas da RNA são: (i) Força a aplicar no motor angular (giro da barra de direção, de -1.5 a 1.5); (ii) Força a aplicar no motor linear (torque, de 0.0 a 6.0). Experiências iniciais mostraram que é imprescindível que o veículo diminua a velocidade em caso de curvas, principalmente quando bruscas. Usamos uma única RNA que controla tanto o giro como o torque, sendo capaz de orientar o veículo e realizar a navegação sem a necessidade de um controle humano ou de uma codificação prévia de um autômato que informe quando desviar de um obstáculo ou como navegar. A base de dados de treinamento foi obtida a partir de um sistema simples baseado em regras. Foram realizadas 32 simulações para obter dados. A base de dados final obtida apresentou 4.985 registros, sendo dividida em 70% para treino e 30% para validação.

Uma vez obtidos os dados para treino e validação, partimos para definição da topologia da RNA. Testamos 6 diferentes topologias de RNA, com 4, 9, 18, 24, 30 e 36 neurônios na camada oculta. A análise e escolha da melhor RNA foi feita através do Erro Médio Absoluto (E_{MAE}) e do Erro Médio Quadrático (E_{MSE}). Realizamos 3 treinos de cada topologia de RNA para análise do E_{MAE} sendo cada treino realizado com uma semente aleatória diferente. O E_{MAE} usando nesta validação foi obtido a partir da análise do resultado dos testes das RNA com 5.000, 10.000, 20.000, 40.000, 60.000, 80.000 e 100.000 ciclos. Conforme apresentado em [21], as RNAs com 4 e 9 neurônios na camada oculta não apresentaram capacidade de aprendizado, já, o treino nas RNAs com 18, 24, 30 e 36 neurônios na camada oculta mostrou que todas estas RNAs apresentam capacidade de aprendizado, sendo a RNA com 24 neurônios na camada oculta a que apresenta os menores erros. No ciclo de treino número 32.500 ocorre a inversão das curvas de

treino e teste (Ponto Ótimo de Generalização [5]). Esta foi a RNA escolhida para ser aplicada no simulador. O detalhamento da avaliação de todas as RNAs está descrito em [21].

Após implementar a RNA selecionada no controle dos robôs móveis, inserindo no sistema de controle o código da RNA, buscamos avaliar se o controle proporcionado pela RNA é eficiente para realizar a navegação entre os pontos iniciais e finais solicitados pelo agente monitor. Assim, realizamos inúmeros experimentos com diferentes quantidades de árvores no ambiente e com diferentes topografias do terreno. Também buscamos avaliar a quantidade de erro (imprecisão) nos sensores e atuadores que a RNA suporta, mantendo a navegação correta.

Dados os pontos iniciais e finais para um grupo de agentes, a navegação no ambiente desenvolvido pode ser vista na Fig. 5. O ambiente foi parametrizado com 4 níveis de ocupação por árvores, calculamos a área de ocupação do total de árvores em relação à área do terreno. As ocupações são aproximadamente 10%; 5%; 2,5% e 0,625%. Resultados serão considerados satisfatórios da seguinte forma: cada veículo deve ser capaz de atravessar uma região simulada representando uma área de cerca de 1km utilizando as diferentes taxas de ocupação por árvores. Os veículos não podem colidir com árvores nem com outros veículos.

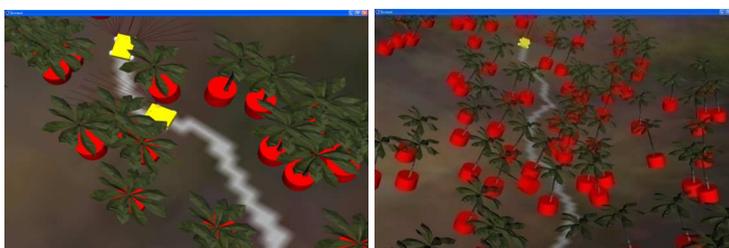


Fig. 5. Imagens de simulação com navegação e desvio satisfatórios, para observação dos caminhos realizados pelos robôs móveis.

Os resultados das primeiras simulações, com diferentes tipos de ocupação do ambiente por árvores, podem ser vistos na Tabela 3. Podemos ver nesta Tabela que, para ambientes com ocupação de 5%, ou menos, a RNA foi capaz de realizar a navegação com desvios satisfatórios em 98% e 100% dos casos. Os erros que ocorreram advém da entrada dos veículos em regiões de afunilamento da vegetação; uma forma inicial para tratar este erro seria a aplicação de capacidade de dar ré, que ainda não foi implementada em nosso modelo.

Tabela 3. Resultado das simulações usando a RNA com diferentes ocupações no ambiente.

Quantidade de Simulações de Navegação	Área Ocupada com Árvores	Resultados Satisfatórios
50	10,00%	42 (84%)
50	5,000%	49 (98%)
50	2,500%	49 (98%)
50	0,625%	50 (100%)

A partir das simulações apresentadas na Tabela 3, escolhemos o ambiente com 5% de ocupação com árvores para realizar novas simulações aplicando ruído simulado nos sensores e nos atuadores. A Tabela 4 apresenta os resultados de simulações aplicando ruído simulado; essa simulação de ruído foi feita com a aplicação de um percentual aleatório entre *-ruído* e *+ruído* sobre o valor original do sensor/atuator. Podemos ver que os resultados são 100% satisfatórios com aplicação de até 10% de ruído em todos os sensores e atuadores. A aplicação de 20% de ruído apresentou falhas em desvios simples. No caso da aplicação de até 100% de ruído na leitura de um único sensor de distância (escolhido aleatoriamente a cada ciclo) podemos ver que os

resultados são 100% satisfatórios quando o ruído é aplicado em apenas um sensor; quando aplicado em dois sensores, o controle apresenta falha ao percorrer o trajeto.

Tabela 4. Resultado das simulações com aplicação de ruído.

Quantidade de Simulações de Navegação	Ruído Aplicado nos 5 Sonares	Result. Satisf.	Ruído Aplicado nos 2 Atuadores	Result. Satisf.	Ruído de até 100% em n sensores aleatórios	Result. Satisf.
20	Até 5,0%	20 (100%)	Até 5,00%	20 (100%)	$n=1$	20 (100%)
20	Até 10,0%	20 (100%)	Até 10,0%	20 (100%)	$n=2$	16 (80%)
20	Até 20,0%	18 (90%)	Até 20,0%	18 (90%)		

A Fig. 6 apresenta imagens de simulações com aplicação de ruído de até 100% na leitura de um sensor aleatório, como as apresentadas na Tabela 4. Esta Figura apresenta um veículo com sensores puros (sem interferência) e um veículo com a aplicação de ruído nos sensores. Podemos ver uma trajetória perfeitamente reta do veículo sem ruído nos sensores e a trajetória irregular do veículo com erro simulado. A trajetória torna-se irregular devido ao fato de: (i) o veículo interpretar o ruído nos sensores como um possível aviso de proximidade de colisão; (ii) o valor do comando aplicado no acionamento das rodas também sofre influências do erro simulado.

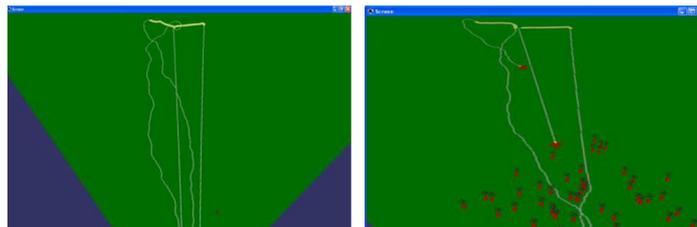


Fig. 6. Navegação de dois robôs. Ambas as imagens apresentam trajetórias de ida e volta; um robô com ruído simulado e outro não.

5 Conclusão

O objetivo deste artigo foi detalhar o modelo, a implementação e a avaliação de eficiência de técnicas de Inteligência Artificial (IA) aplicadas em um sistema multi-agente que opera em um ambiente virtual de simulação realística. Neste sistema, uma equipe de agentes autônomos trabalha cooperativamente a fim de realizar com sucesso a identificação e o combate de incêndios em áreas florestais, sem intervenção humana. O ambiente suporta uma série de características fundamentais para a simulação realística da operação, como terrenos irregulares, processos naturais e restrições físicas na criação e uso de robôs móveis. A operação multi-agente depende essencialmente de duas etapas: planejamento e ação. No planejamento, usamos Algoritmos Genéticos (AG) para evoluir estratégias de posicionamento de atuação dos robôs bombeiros. Para a ação foram criados robôs de combate fisicamente simulados, sendo as informações sensoriais de cada robô (e.g. GPS, bússola, sonar) usadas na entrada de uma Rede Neural Artificial (RNA). Esta RNA que controla os atuadores do veículo permite navegação com desvio de obstáculos. O controle robusto foi validado com a demonstração de resultados aplicando ruídos nos sensores e nos atuadores dos robôs móveis. Os resultados das simulações demonstram que a RNA controla satisfatoriamente os robôs móveis, que o uso do AG configura a estratégia de combate ao incêndio de modo satisfatório e que o sistema multi-agente proposto pode vir a ter um papel

muito importante no planejamento e execução de operações reais de combate a incêndios florestais. Os trabalhos futuros vislumbrados durante o desenvolvimento desta pesquisa estão detalhados em [23].

Agradecimentos

Agradecemos ao CNPq, Capes, Unibic (Unisinos), FAPERGS e a FINEP (Projeto Rede Brasileira de Visualização) pelo apoio a este trabalho.

Referências

1. GO, J. et. al.: Accurate and flexible simulation for dynamic, vision-centric robots. International Joint Conference on Autonomous Agents (2004)
2. Dudek, G., Jenkin, M.: Computational Principles of Mobile Robotics. Cambridge, MIT Press (2000)
3. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators, Neural Networks, v. 2, p. 359–366 (1989)
4. Osagie, P.: Distributed Control for Networked Autonomous Vehicles. Dissertação de Mestrado, KTH-CSC, Royal Institute of Technology, Sweden (2006)
5. Rezende, S.O.: Sistemas inteligentes: fundamentos e aplicações. São Paulo (2003)
6. Nikolopoulos, C.: Expert Systems - Introduction to First and Second Generation and Hybrid Knowledge Based Systems. New York, USA: Marcel Dekker Inc. Press (1997)
7. Holland, J.: Adaptation in Natural and Artificial Systems, Michigan Press (1975)
8. Mitchell, M.: An Introduction to Genetic Algorithms, Cambridge, The MIT Press (1996)
9. Carvalho, A. C. F., Braga, A. P., Ludermir, T. B.: Sistemas inteligentes: Fundamentos e aplicações, Cap. Computação Evolutiva, p. 225-248 (2003)
10. Koproski, L.P.: O fogo e seus efeitos sobre a herpeto e a mastofauna terrestre no parque nacional de Ilha Grande. Dissertação de mestrado, UFPR (2005)
11. Pfeifer, R., Scheier, C.: Understanding Intelligence, Massachusetts, USA: The MIT Press (1999)
12. Osório, F. S.: INSS: Un Système Hybride Neuro-Symbolique pour l'Apprentissage Automatique Constructif, Tese de Doutorado, INPG, Grenoble, França (1998)
13. Osório, F. S., et al.: Increasing Reality in Virtual Reality Applications through Physical and Behavioural Simulation, Proc. of Virtual Concept Conference (2006)
14. Laboratório de Incêndios Florestais, www.floresta.ufpr.br/~firelab
15. Heinen, M. R., Osório, F. S.: AGs aplicados em roteamento de veículos, Revista Hifen (2006)
16. Stuttgart Neural Network Simulator (SNNS), www-ra.informatik.uni-tuebingen.de/SNNS
17. Open Dynamics Engine, www.ode.org
18. Open Scene Graph, www.openscenegraph.com
19. Demeter Terrain Engine, www.tbgssoftware.com
20. Pessin, G. et. Al: Simulação Virtual de Agentes Autônomos para a Identificação e Controle de Incêndios em Reservas Naturais. IX Symposium on Virtual and Augmented Reality (SVR), v. 1, p. 236-245 (2007)
21. Pessin, G., Osorio, F., Musse, S.: Utilizando Redes Neurais Artificiais no Controle Robusto de Navegação de Robôs Móveis. XII MT Digital, Cuiabá, MT (2008)
22. Pessin, G. et. Al: Simulação Virtual da Evolução de Estratégias e do Controle Inteligente em Sistemas Multi-Robóticos. X Symposium on Virtual and Augmented Reality (2008)
23. Pessin, G.: Evolução de Estratégias e Controle Inteligente em Sistemas Multi-Robóticos Robustos. Dissertação (Mestrado) — Universidade do Vale do Rio dos Sinos (UNISINOS), RS (2008)
24. Pessin, G. et. Al: Ambiente virtual para simulação de robôs moveis inteligentes aplicados no combate de incêndios florestais. Revista Disciplinary Scientia, Santa Maria, RS (2007)
25. Pessin, G. et. Al: Controle inteligente em um sistema multi-robótico aplicado na identificação e combate de incêndios em áreas florestais. In: Anais do IV Observatório. Chapecó, SC (2007)
26. Pessin, G. et. Al: Desenvolvimento de um ambiente para simulação de robótica móvel em áreas florestais. In: Anais do VI SIRC-RS, Santa Maria, RS (2007)
27. Pessin, G. et. Al: Evoluindo estratégias de posicionamento em um sistema multi-robótico aplicado ao combate de incêndios florestais. Revista Hifen (ISSN 0103-1155), Uruguaiana, RS (2007)
28. Pessin, G. et. Al: Utilizando agentes autônomos com aprendizado para a identificação e combate de incêndios em áreas florestais. In: Anais do VII SIPM. Passo Fundo, RS (2007)
29. Pessin, G. et. Al: Utilizando redes neurais artificiais no controle de robôs móveis aplicados ao combate de incêndios florestais. In: Anais do XVI SEMINCO, Blumenau, SC (2007)