

DESENVOLVIMENTO DE UM SISTEMA MULTI-ROBÓTICO COM CONTROLE INTELIGENTE APLICADO NA IDENTIFICAÇÃO E COMBATE DE INCÊNDIOS EM ÁREAS FLORESTAIS

**PESSIN, Gustavo¹; OSÓRIO, Fernando S.²; MUSSE, Soraia R.³;
NONNEMMACHER, Vinícius⁴; FERREIRA, Sandro S.⁵.**

¹ Mestrando em Computação Aplicada na Universidade do Vale do Rio dos Sinos, pessin@gmail.com

² Prof. Dr. da Universidade do Vale do Rio dos Sinos, fosorio@unisinos.br

³ Profa. Dra. da Pontifícia Universidade Católica - RS, soraia.musse@pucrs.br

⁴ Graduando em Jogos e Entretenimento Digital na Unisinos, vnonnenmacher@gmail.com

⁵ Graduando em Jogos e Entretenimento Digital na Unisinos, sandro.s.ferreira@gmail.com

DESENVOLVIMENTO DE UM SISTEMA MULTI-ROBÓTICO COM CONTROLE INTELIGENTE APLICADO NA IDENTIFICAÇÃO E COMBATE DE INCÊNDIOS EM ÁREAS FLORESTAIS

RESUMO

O objetivo deste artigo é detalhar o projeto e o desenvolvimento de um sistema multi-agente que opera em um ambiente virtual de simulação realística¹. Neste sistema, uma equipe heterogênea de agentes autônomos trabalha cooperativamente a fim de realizar com sucesso a identificação e o combate de incêndios em áreas florestais, sem intervenção humana. Os robôs de combate são fisicamente simulados e as informações sensoriais de cada robô (*e.g.* GPS, bússola, sonar) servem de entrada para uma Rede Neural Artificial que controla os atuadores do veículo para assim realizar navegação com desvio de obstáculos de um ponto aleatório do terreno até um ponto de atuação no combate ao incêndio. 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 no uso dos robôs móveis. O terreno simula diversos aspectos naturais, como tipos de vegetação, topografia, e respeita questões como taxa de propagação do fogo baseado na intensidade e orientação do vento, vegetação e topografia. Os resultados das simulações demonstram que a Rede Neural Artificial controla satisfatoriamente os robôs móveis 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.

PALAVRAS-CHAVE: Simulação; Robótica Móvel; Rede Neural Artificial.

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 (GO *et al*, 2004). Dotar robôs autônomos de capacidade de 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 (DUDEK; JENKIN, 2000).

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 (OSAGIE, 2006).

Um grande sonho de nossa sociedade é a aplicação de sistemas robóticos substituindo atividades que coloquem em risco a vida humana, em atividades onde a atuação de humanos é deficitária ou onde a atuação humana de certa forma é ineficiente. A tarefa de monitoração e combate de incêndios em áreas florestais é um caso onde a aplicação de um sistema multi-robótico poderia diminuir consideravelmente os prejuízos humanos, materiais e ambientais. Com relação a incêndios florestais, anualmente registram-se cerca de 45.000 incêndios nas florestas da Europa. Entre 1989 e 1993, só na zona mediterrânea 2,6 milhões de hectares florestais foram destruídos pelo fogo, o equivalente ao desaparecimento do mapa de um território com a dimensão da Bélgica

em cinco anos (CE, 2006). Os incêndios florestais causam diversos tipos de danos humanos, materiais e ambientais. A extensão territorial do Brasil e a diversidade de sua cobertura vegetal, bem como o número expressivo de ocorrências de incêndios florestais verificados no país, são fatores que enfatizam a necessidade de um sistema cada vez mais aprimorado e consistente de detecção e combate de incêndios florestais (BATISTA, 2004). Diversas iniciativas a fim de incrementar a capacidade de reação de órgãos públicos e civis no sentido de evitar desastres tem sido uma das preocupações junto a órgãos como a Secretaria Nacional de Defesa Civil, levando a criação de novos CEPEDs (Centro de Estudos para a Prevenção de Emergências e Desastres). Importantes iniciativas, como a RBV – Rede Brasileira de Visualização, financiada pela FINEP, também tem sido incentivadas, onde a competência de Segurança e Defesa (Civil e Militar) da Rede vem sendo foco de desenvolvimentos junto a nossa Universidade e na qual este projeto se integra.

Neste trabalho, um grupo de agentes autônomos trabalha cooperativamente a fim de realizar com sucesso a identificação e o combate de incêndios em áreas florestais, sem intervenção humana. Simulados diferentes tipos de agentes, com níveis hierárquicos diferentes, assim, cada etapa do processo conta com agentes específicos que possuem um conjunto de ações determinadas. Os agentes de combate são robôs móveis fisicamente simulados e as informações sensoriais de cada robô (*e.g.* GPS, bússola, sonar) servem de entrada para uma Rede Neural Artificial que controla os atuadores do veículo para assim realizar navegação com desvio de obstáculos de um ponto aleatório do terreno até um ponto de atuação no combate ao incêndio. Os sensores simulados retornam valores estocásticos (com ruído), tornando assim o modelo mais realista.

Um problema fundamental da robótica é a navegação, ir de um lugar para outro depende de três aspectos fundamentais: localização, orientação e controle motor. Para conhecer tanto sua localização como sua orientação, um robô móvel deve possuir sensores próprios (*e.g.* GPS, bússola). Para o controle motor, deve possuir um número adequado de motores (um carro, por exemplo, usualmente possui um motor angular, para giro das rodas e um motor linear, para tração). Normalmente sensores e atuadores são sujeitos a erros e interferências, assim o controle das ações de um robô deve sempre levar em conta a imprecisão dos sensores e motores envolvidos. Um sistema robusto deve permitir que, mesmo com sensores e atuadores imprecisos, o agente cumpra o seu objetivo. Uma técnica de Aprendizado de Máquina indicada para este controle é a de Redes Neurais Artificiais, dada sua capacidade de aprendizado a partir de exemplos e a respectiva generalização e adaptação das saídas. É uma técnica muito utilizada no controle de navegação de sistemas reativos (OSÓRIO, 1998; MARCHI, 2001).

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 OSG (OSG, 2007) junto com a biblioteca de programação Demeter (DEMETER, 2007) permitem combinar o mapa de elevação juntamente com uma determinada distribuição de vegetação, criando assim um terreno bastante realístico (OSÓRIO *et al.*, 2006). Além disso, o uso da biblioteca de programação ODE (ODE, 2007) permite implementar atributos físicos (*e.g.* atrito, fricção, gravidade, colisão) tornando o sistema ainda mais realístico.

Em nossos estudos e pesquisas, pelo que pudemos constatar, não existe um ambiente de simulação aberto que combine todas as características requisitadas para este trabalho, que são: (i) Realismo físico de modelagem robótica; (ii) Realismo físico de interação de agentes em terrenos irregulares; (iii) Facilidades de comunicação entre os agentes; (iv) Capacidade de aplicação de métodos de controle (*e.g.* Redes Neurais Artificiais); (v)

Capacidade de aplicação de métodos de evolução de estratégias (e.g. Algoritmos Genéticos); (vi) Simulação de processos naturais (como a propagação do fogo).

Nossas principais metas neste projeto são: (i) Recolher informações sobre dados florestais, tipos de vegetação, topografia, e comportamento de incêndios para criar o ambiente virtual mais realista possível; (ii) Simular incêndios em florestas, reproduzindo de forma bastante realista o ambiente e a propagação dos focos de incêndio; (iii) Pesquisar ferramentas e técnicas de combate à incêndios florestais utilizadas por bombeiros; (iv) Implementar a simulação de agentes móveis autônomos colaborativos capazes de formar uma brigada de combate a incêndios; (v) Estudar métodos de aprendizado de máquina e suas vantagens para o modelo; e (vi) Estudar a robustez das ações dos agentes pela leitura de dados de sensores sujeitos a erros.

Como resultado principal esperamos projetar melhores agentes autônomos de combate a incêndios, com sensores e métodos de comunicação eficientes para a execução satisfatória da tarefa proposta neste artigo, bem como propor métodos de deslocamento e regras de formação de equipes. O sucesso desta tarefa envolve o uso de uma variedade de tecnologias de diferentes campos, assim, a construção deste sistema é um bom estudo para avaliar eficiência de arquitetura multi-agente, estratégia cooperativa, fusão de sensores e modelagem robótica. A tarefa em si poderá ser generalizada para outras atividades práticas como acidentes nucleares ou desastres ambientais. Com técnicas de aprendizado de máquina esperamos obter agentes que suportem melhor um ambiente dinâmico, fazendo assim o sistema mais flexível e autônomo possível.

Neste artigo apresentamos na Seção 2 uma pequena conceituação teórica de agentes e sistemas multi-agentes. Na Seção 3 detalhamos características importantes de robótica móvel, conceitos de simulação e modelagem, as ferramentas pesquisadas e as bibliotecas de simulação selecionadas para o desenvolvimento do trabalho. A Seção 4 apresenta conceitos de Aprendizado de Máquina e a técnica escolhida para o controle de navegação dos robôs móveis, que é a de Redes Neurais Artificiais. Na Seção 5 descrevemos técnicas e operações reais de identificação e combate a incêndios florestais, informações importantes para que seja feita uma adequada simulação do ambiente e dos incêndios. Na Seção 6 descrevemos o ambiente desenvolvido, a operação multi-agente de identificação e combate e o desenvolvimento e a aplicação da Rede Neural Artificial no controle dos robôs móveis. Finalizamos apresentando a conclusão do trabalho realizado.

2. AGENTES INTELIGENTES

Uma das primeiras e mais simples definições de agentes inteligentes é de que são sistemas computacionais que habitam um dado ambiente, sentem e agem autonomamente nesse ambiente (MAES, 1996). Segundo (PFEIFER; SCHEIER, 1994), um agente de software pode ser visto como um sistema dinâmico, onde a percepção e a ação constituem processos simultâneos e inseparáveis. Em termos gerais, ambiente é onde um agente ou um conjunto de agentes está inserido, pode ser físico (como ambientes onde estão inseridos robôs), de software ou de realidade virtual (onde se faz simulação do ambiente físico) (GARCIA; SICHMAN, 2003). A Figura 1 mostra o esquema típico de um agente.

De acordo com (PFEIFER; SCHEIER, 1994) o projeto de um agente de software envolve a definição de três componentes que são rigorosamente interconectados e mutuamente interdependentes: (i) Definição do ambiente para atuação; (ii) Estabelecimento de comportamentos e tarefas a serem realizadas; (iii) Determinação da morfologia do agente. A concretização do projeto de um agente robótico deve priorizar quatro aspectos principais (PFEIFER; IIDA; BONGARD, 2005): (i) Autonomia: Deve

ser capaz de operar com o mínimo possível de intervenção, supervisão e instrução humana; (ii) Auto-suficiência: Deve ser capaz de operar por tempo prolongado, ou, pelo menos, suficiente para realizar a tarefa e buscar recarga de energia; (iii) Localização: Deve ser capaz de adquirir informação sobre o ambiente somente através de seus próprios sensores; (iv) Corporificarão: Deve ser desenvolvido como um sistema físico ou computacional, porém a corporificarão não se dá necessariamente pela materialização, como a apresentada em animais ou robôs físicos, mas por uma relação dinâmica com o ambiente, ou seja, a corporificarão pode ser realizada em ambientes de simulação computacionais, desde que estes ambientes sejam realísticos do ponto de vista físico (PFEIFER; IIDA; BONGARD, 2005).

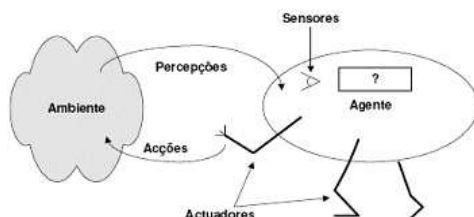


Figura 1 – Esquema típico de um agente

Fonte: (REIS, 2003)

2.1 SISTEMAS MULTI-AGENTES

Sistemas multi-agentes são sistemas constituídos de múltiplos agentes que interagem ou trabalham em conjunto de forma a realizar um determinado conjunto de tarefas ou objetivos. Um sistema multi-agente pode ser visto como uma rede, fracamente acoplada, de solucionadores de problemas que trabalham em conjunto para resolver problemas que vão além da sua capacidade individual. Estes solucionadores de problemas são essencialmente autônomos e, muitas vezes, heterogêneos em sua natureza (SILVEIRA, 2006). Um sistema multi-agente deve ter alguma forma de controle. A utilização de ambientes dinâmicos, robôs com mal-funcionamento, múltiplas regras e restrições em comportamentos individuais adicionam complexidade ao problema de controle (OSAGIE, 2006). Nos trabalhos de (STONE; VELOSO, 2000; MONDADA *et al*, 2005; OSAGIE, 2006) sistemas multi-agentes são definidos em duas arquiteturas de controle: sistemas distribuídos ou sistemas centralizados. Em um sistema multi-agente centralizado, uma única unidade detém todo o poder de tomada de decisão para controle e definição de objetivos dos outros agentes do sistema. É comum que neste tipo de sistema exista sobrecarga da unidade de controle. A abordagem centralizada ainda tem relação com o problema de falha em pontos únicos do sistema, que podem inviabilizar toda a operação, porém, traz a vantagem de poder produzir planos de ação globais ótimos que podem ser teoricamente comprovados como corretos (OSAGIE, 2006). Em um sistema multi-agente distribuído, cada robô é responsável pela coordenação ao de suas próprias ações. Em um sistema completamente distribuído, a busca da solução do problema surge da interação entre os agentes. Estas soluções são tipicamente difíceis ou mesmo impossíveis de terem sua eficiência comprovada analiticamente (OSAGIE, 2006).

Cooperação pode ser entendida como a capacidade que os agentes tem de trabalhar em conjunto de forma a concluírem tarefas de interesse comum (REIS, 2003). Nwana (NWANA, 1996) acredita que a cooperação entre agentes é fundamental, sendo a razão principal para a existência de um ambiente multi-agente. Para que exista cooperação, deve existir alguma forma de interação entre os agentes, ou com humanos, através de alguma forma de comunicação (WOOLDRIDGE; JENNINGS, 1995). A cooperação

não depende obrigatoriamente de comunicação entre todos os agentes, visto que, como no mundo real, um coordenador (unidade de controle) pode delegar tarefas (enviar mensagens) para um grupo de pessoas (agentes) que não tenham comunicação direta entre si. Neste caso, as pessoas podem trabalhar em prol do mesmo objetivo sem ter conhecimento do que as outras pessoas estão fazendo.

A morfologia de um robô deve, ao incluir a capacidade de comunicação, possuir um módulo de controle de comunicações, assim, deve possuir componentes de percepção (recepção de mensagens) e de ação (envio de mensagens). O módulo de comunicação está diretamente ligado ao módulo de controle do agente (REIS, 2003). No nível referente aos mecanismos de comunicação, é possível identificar quatro diferentes modelos de trocas de mensagens entre agentes (WOOLDRIDGE, 2000): (i) Troca direta de mensagens: envolve o estabelecimento de uma ligação direta entre os agentes, através de um conjunto de protocolos que permite o estabelecimento de comunicação ponto a ponto. Este tipo de comunicação implica que cada um dos agentes envolvidos deve ter conhecimento da existência dos seus parceiros e da forma de endereçar mensagens a eles (SILVEIRA, 2006); (ii) Sistemas federados: neste modelo são introduzidos agentes facilitadores (GENESERETH; KETCHPEL, 1994) que são agentes especiais que provem serviços de comunicação. Uma estrutura hierárquica de agentes é estabelecida e toda troca de mensagens se dá através destes agentes facilitadores; (iii) Comunicação por difusão: utilizado quando for necessária alguma comunicação a todos os agentes do sistema, ou quando o remetente da mensagem não sabe quem é ou qual o endereço do seu destinatário (SILVEIRA, 2006); (iv) Quadro-negro: no modelo de comunicação através de quadro-negro (*blackboard*) a comunicação se dá através de escrita e leitura em uma estrutura de dados comum (LEE *et al.*, 1980; HAYES-ROTH, 1995). Neste tipo de modelo não existe qualquer tipo de comunicação diretamente entre agentes, sendo esta efetuada sempre através da estrutura de dados comum (REIS, 2003), como mostra a Figura 2.

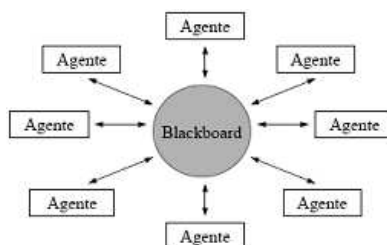


Figura 2 – Exemplo de estrutura de um sistema baseado em Quadro-Negro

Fonte: (FONSECA, 2000)

3. ROBÓTICA MÓVEL

Um robô móvel é um dispositivo mecânico montado sobre uma base não fixa, que age sob o controle de um sistema computacional, equipado com sensores e atuadores que o permitem interagir com o ambiente (MARCHI, 2001; BEKEY, 2005). A interação com o ambiente se dá através de ciclos de percepção-ação que consistem em três passos fundamentais: (i) Obtenção de informação através de sensores; (ii) Processamento das informações para seleção de ação; e, (iii) Execução da ação através do acionamento dos atuadores. Esse conjunto de operações, em uma análise superficial, pode parecer simples, porém o controle robusto de sistemas robóticos tem complicações físicas e mecânicas, eletrônicas e computacionais que tornam a criação de um conjunto de regras uma tarefa árdua e sujeita a erros. Sensores são os mecanismos de percepção de um robô e realizam medições físicas (*e.g.* contato, distância, orientação, temperatura),

provêm sinais ou dados crus que precisam ser interpretados pelo “cérebro” do robô. A interpretação destes sinais deve ser a única maneira de um robô autônomo entender o ambiente que o cerca para poder realizar as mudanças de ação necessárias (JPL/NASA 2007). Atuadores são os mecanismos de ação de um robô (*e.g.* motores, pistões, braços manipuladores), controlados por circuitos eletrônicos que recebem valores de ação, valores estes que devem ser calculados pelo “cérebro” do robô e devem estar de acordo com especificações de fabricantes (JPL/NASA 2007).

3.1 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 realista (PFEIFER; SCHEIER, 1999). Usualmente, experimentos em robótica móvel utilizando um robô real exigem enorme despendimento 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 (GO *et al*, 2004). É 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 (GO *et al*, 2004). No caso deste trabalho, é fundamental que existam restrições físicas no modelo e que exista a possibilidade de trabalho em um terreno irregular, provido de obstáculos. Para o desenvolvimento deste trabalho foram pesquisadas algumas ferramentas de simulação, porém, nenhuma mostrou possuir o conjunto completo das características requisitadas. Uma pequena descrição de cada uma das ferramentas pesquisadas é fornecida a seguir.

3.1.1 FERRAMENTAS PESQUISADAS

O *Microsoft Robotics Studio* (msdn.microsoft.com/robotics) tem como objetivo prover um ambiente para desenvolvimento de simulação robótica. Permite a simulação em terrenos irregulares e é livre apenas para uso não comercial. O desenvolvimento dos sensores, dos controles inteligentes e da comunicação entre os robôs depende de programação (*e.g.* C#, VB.NET). O *Webots* (www.cyberbotics.com) é um simulador de robôs móveis que tem como base a biblioteca de simulação física ODE, inclui modelos prontos de alguns robôs comerciais e modela sensores com a capacidade de detecção de obstáculos, visão, e manipuladores simples. O usuário pode programar cada robô utilizando C/C++. É um produto comercial. Outros ambientes de simulação estudados foram o *Khepera Simulator* (diwww.epfl.ch/lami/team/michel/khep-sim), o *Mission Simulation Facility* (ase.arc.nasa.gov/msf), o *JUICE* (www.natew.com/juice) e o *Simulator BOB* (www.tu-harburg.de/ti6/mitarbeiter/pst/Sim/Simulator.html).

3.1.2 BIBLIOTECAS

Como optamos por desenvolver nosso próprio ambiente, todas as bibliotecas de programação selecionadas para o desenvolvimento do ambiente e dos agentes robóticos são software livre, multi-plataforma e em linguagem C/C++. Uma pequena descrição de cada uma das bibliotecas utilizadas é fornecida a seguir: (i) **Open Dynamics Engine** (ODE, 2007): é uma biblioteca desenvolvida para a simulação física de corpos rígidos articulados (SMITH, 2006). Uma estrutura articulada é criada quando corpos rígidos de vários tipos são conectados por algum tipo de articulação, como, por exemplo, um

veículo terrestre que tem a conexão de rodas em um chassi. A ODE foi projetada para ser utilizada de modo interativo em simulações de tempo real e é especialmente indicada para a simulação de objetos móveis em ambientes dinâmicos. A ODE não tem como objetivos realizar simulação de outras dinâmicas além da dinâmica de corpos rígidos (*e.g.* partículas, roupas, ondas, fluidos, corpos flexíveis, fraturas). A ODE possui contatos rígidos, assim, quando dois corpos colidem não há penetração. O sistema de detecção de colisão é nativo e suporta diversas primitivas (*e.g.* esfera, caixa, cilindro, plano, raio).

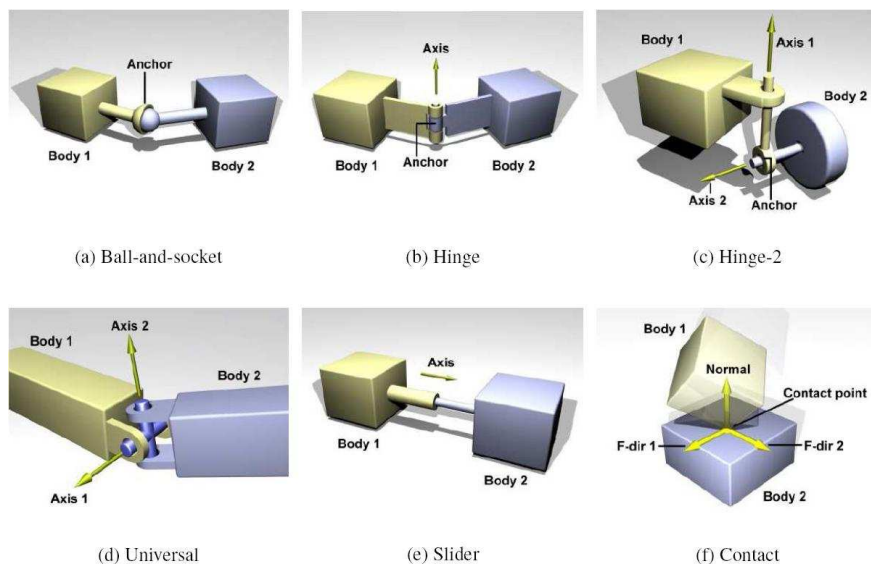


Figura 3 – Articulações disponíveis na ODE

Fonte: (SMITH, 2006)

A utilização da ODE no nosso ambiente é fundamental por fornecer restrições físicas, principalmente na definição da morfologia dos robôs. Do ponto de vista físico, um robô pode ser considerado simplesmente como um conjunto de corpos rígidos conectados (SMITH, 2006). Cada um destes corpos pode interagir com os demais, assim como um motor faz com que um veículo se movimente. Além disso, a atuação da gravidade e da inércia devem estar presentes. Na ODE, estas propriedades são tratadas em dois conceitos, os corpos rígidos e as articulações. Com relação aos corpos rígidos, suas propriedades, do ponto de vista da simulação, que mudam com o passar do tempo, são: (i) vetor de posição do ponto de referência do corpo; (ii) velocidade linear do ponto de referência do corpo; (iii) orientação do corpo; (iv) velocidade angular; (v) posição do centro de massa; e (vi) matriz de inércia. As articulações (Figura 3) são relacionamentos entre corpos de modo que possam existir posições e orientações que sejam relativas a ambos os corpos. Este relacionamento realiza restrições/obrigações de movimento; (ii) **Open Scene Graph** (OSG, 2007): é uma biblioteca para desenvolvimento de aplicações gráficas 3D de alta performance. Baseada no conceito de grafos de cena, provê ao desenvolvedor um ambiente orientado a objeto sobre a *OpenGL* (www.opengl.org), liberando este da necessidade de implementação e otimização de chamadas gráficas de baixo nível. Um grafo de cena permite a representação de objetos em uma cena com uma estrutura que permite a criação de grupos que podem compartilhar propriedades, assim, podemos definir uma propriedade comum em um nível hierárquico mais alto e todos os objetos inferiores receberão esta propriedade (BARROS; GONZAGA; RAPOSO, 2007); (iii) **DrawStuff** (ODE, 2007): como a ODE é completamente independente de visualizador, iniciar a criação dos corpos e das simulações pode ser

uma tarefa bastante árdua caso não tenhamos uma forma simples e fácil de visualizar os objetos. Por este motivo, a biblioteca DrawStuff é disponibilizada em conjunto com a ODE. Basicamente, o DrawStuff é um ambiente de visualização de objetos 3D que tem o propósito de permitir a demonstração visual da ODE sendo uma biblioteca bastante simples e rápida para utilização; (iv) **Demeter** (DEMETER, 2007): é uma biblioteca desenvolvida para renderizar terrenos 3D, desenvolvida para ter rápida performance e boa qualidade visual, pode renderizar grandes terrenos em tempo-real sem necessidade de hardware especial, depende da *Simple DirectMedia Layer* (www.libsdl.org) para realizar o tratamento das texturas do terreno e da *Geospatial Data Abstraction Library* (www.gdal.org) para carregar arquivos de elevação.

4. 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 (REZENDE, 2003). Essas técnicas devem exibir um comportamento inteligente e realizar tarefas complexas com um nível de competência equivalente ou superior ao de um especialista humano (NIKOLOPOULOS, 1997). As técnicas de ML mais importantes citadas pelas bibliografias são (REZENDE, 2003): (i) aprendizado por analogia (Sistemas Baseados em Casos); (ii) aprendizado por indução (Árvores de Decisão, *Inductive Logic Programming*); (iii) aprendizado por evolução (Algoritmos Genéticos); (iv) aprendizado conexionista (Redes Neurais Artificiais, Redes de Função de Base Radial, *Support Vector Machines*); e, (v) aprendizado por reforço. Para a construção do sistema proposto neste trabalho utilizamos uma RNA, assim, descrevemos sucintamente suas características a seguir.

4.1 REDES NEURAIS ARTIFICIAIS

Redes Neurais Artificiais (RNA) são sistemas paralelos distribuídos, compostos por unidades de processamento simples que calculam determinadas funções matemáticas, normalmente não-lineares (BRAGA; CARVALHO; LUDERMIR, 2000). Seus atributos básicos podem ser divididos em arquitetura e neurodinâmica. A arquitetura determina a estrutura da rede, ou seja, o número de neurônios e sua interconectividade e a neurodinâmica, por sua vez, define as propriedades funcionais da rede, ou seja, como ela aprende, recupera, associa e compara novas informações com o conhecimento já armazenado (KARTALOPOULOS, 1996). Matematicamente, RNAs são aproximadores universais, que realizam mapeamentos em espaços de funções multi-variáveis (HORNIK; STINCHCOMBE; WHITE, 1989). A capacidade de aprender e generalizar das RNAs é um dos seus maiores atrativos. O processamento da informação em uma RNA é feito por meio de estruturas neurais artificiais (REZENDE, 2003), sendo que esta estrutura, bem como o próprio neurônio artificial são uma analogia biológica ao funcionamento do cérebro. O neurônio artificial proposto por (MCCULLOCH; PITTS, 1943) (Figura 4) pode ser descrito como um modelo com n terminais de entrada $x_1, x_2, x_3, \dots, x_n$ que, para cada entrada, possui um peso w_i correspondente. A soma das entradas x_i ponderadas pelos pesos correspondentes w_i produzem a chamada *saída linear* u . A saída y do neurônio é obtida pela aplicação de uma função $f(.)$ à saída linear u . A função $f(.)$ é chamada função de ativação e pode assumir diferentes formas (e.g. linear, semi-linear ou *sigmoid*). Roseblatt (ROSEMBLATT, 1958) propôs o modelo *Perceptron*, introduzindo o conceito de aprendizado em RNA. O modelo *Perceptron* era composto por uma estrutura de rede tendo como unidades básicas nodos MCP e de uma

regra de aprendizado. O aprendizado no modelo *Perceptron* sempre converge caso o problema em questão seja linearmente separável (ROSEBLATT, 1962).

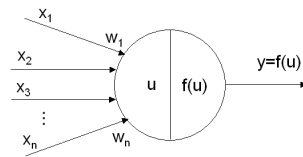


Figura 4 – Neurônio de McCulloch e Pitts (MCP)
 Fonte: (REZENDE, 2003)

A solução de aprendizado para redes com *Perceptrons* combinados em diversas camadas, formando a chamada Rede Neural do Tipo *Multi Layer Perceptron* (MLP) seria proposta por (RUMELHART; MCCLELLAND, 1986) com o algoritmo de aprendizado *Backpropagation*. O algoritmo *Backpropagation* é um algoritmo supervisionado, sendo que a base de treinamento é um conjunto de dados que deve apresentar, para cada entrada, a saída prevista do sistema. Este tipo de aprendizado ocorre em várias épocas, sendo que cada época representa a apresentação do conjunto inteiro de dados à RNA para o ajuste dos pesos. O *Backpropagation* é baseado na Regra Delta Generalizada (WIDROW; HOFF, 1960) e os ajustes dos pesos são realizados pelo método da descida do gradiente. Este método não garante que o aprendizado não tenha ficado preso em mínimos locais (Figura 5(a)), assim, 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. Quando uma RNA é treinada, os pesos sinápticos vão se ajustando para que sejam dadas respostas satisfatórias a base de treinamento apresentada. Se o aprendizado for feito por um número muito grande de épocas pode ocorrer o chamado *overfitting*, onde a RNA começa a decorar a saída, perdendo sua capacidade de generalização. A solução deste problema usualmente envolve o uso de uma base de validação usada em paralelo a base de treinamento, como mostra a Figura 5(b).

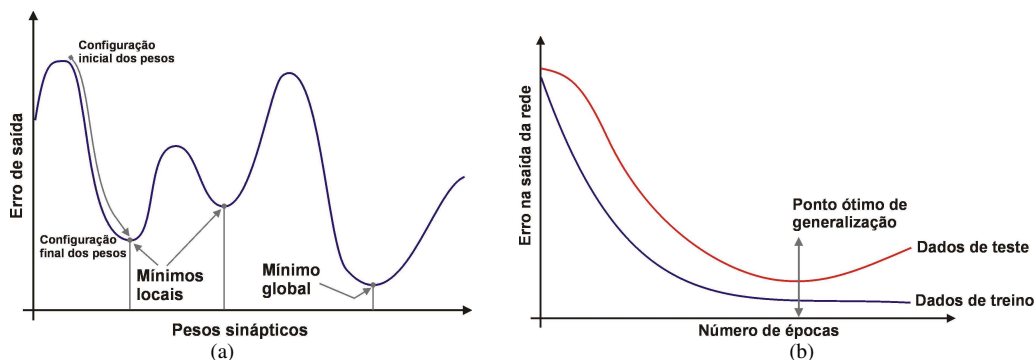


Figura 5 – (a) Descida do gradiente de uma superfície de erro e (b) Curvas de erro em aprendizado e validação

4.1.1 SIMULAÇÃO DE REDES NEURAIS ARTIFICIAIS

O *Stuttgart Neural Network Simulator* (SNNS, 2007) é um simulador de Redes Neurais Artificiais criado no *Institute for Parallel and Distributed High Performance Systems* (IPVR) da Universidade de Stuttgart, que proporciona um ambiente para a criação, o treinamento e a manutenção de Redes Neurais Artificiais. O SNNS possui um grande número de algoritmos de aprendizado (e.g. *Backpropagation*, *Quickprop*, *RProp*). O kernel do sistema é desenvolvido em C e sua utilização pode ser completamente feita

através de linha de comando, porém também possui uma interface desenvolvida em JAVA (JavaNNS). Esta interface gráfica possibilita acompanhar a evolução de taxas de erro e aprendizado da RNA através de gráficos. Um aplicativo do pacote SNNS, o SNNS2C, permite a conversão de uma RNA em código em C, que pode ser inserido em uma outra aplicação.

5. INCÊNDIOS EM AMBIENTES NATURAIS

Fogo, de um modo geral, é o termo aplicado ao fenômeno físico resultante da rápida combinação entre o oxigênio e uma substância combustível com produção de calor, luz e chamas. A reação da combustão completa da madeira, que poderia ser generalizada para todo o material combustível da floresta, envolve três elementos básicos: combustível, oxigênio e calor. Em qualquer incêndio florestal é necessário haver combustível para queimar, oxigênio para manter as chamas e calor para iniciar e continuar o processo de queima. Essa inter-relação entre os três elementos básicos da combustão é conhecida como triângulo do fogo. A ausência, ou redução abaixo de certos níveis, de qualquer um dos componentes do triângulo do fogo inviabiliza o processo da combustão (LIF, 2006).

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 estudo sobre as técnicas reais de operação. Este estudo teve como base os trabalhos de (ANTUNES, 2000; BATISTA, 2004; CPTEC/INPE, 2006; LIF 2006; REMMEL; PERERA, 2001). Para a implementação da propagação do fogo obtivemos de (KOPROSKI, 2005) medições reais de velocidades. O estudo dos modelos de florestas e resíduos florestais é de grande importância para o aprimoramento dos modelos de simulação a serem implementados em ambientes virtuais (PESSIN *et al*, 2007).

Os combustíveis florestais, que são produto da cobertura vegetal e sua dinâmica, são importantes parâmetros nos processos de ignição e propagação dos incêndios. O conhecimento das características básicas dos combustíveis, tais como: tipo, quantidade, continuidade e arranjo, são muito úteis na previsão do comportamento dos incêndios (CASTRO; TUDELA; SEBASTIA, 2001). Quanto aos tipos de combustíveis florestais, sugere-se os seguintes modelos (MAB, 2006): Modelos 1, 2 e 3 são do tipo herbáceo, sendo pasto fino, contínuo, e contínuo espesso respectivamente, os incêndios propagam-se com grande velocidade; Modelos 4, 5, 6 e 7 são do tipo arbustivo, indo de árvores jovens até mato denso; Modelos 8, 9 e 10 são do tipo manta morta; Modelos 11 e 12 são do tipo resíduos lenhosos. Cada um destes modelos (de 1 a 12) possui velocidade de propagação de incêndio específica.

O estudo das técnicas reais de operação permite que possamos planejar melhor o formato da operação e as estratégias a serem implementadas no sistema. O conjunto das técnicas listadas a seguir é de consenso geral para aplicações em monitoramento e combate a incêndios. A operação de combate ou supressão de um incêndio envolve seis etapas distintas (LIF, 2006): (i) Detecção; (ii) Comunicação; (iii) Mobilização; (iv) Deslocamento; (v) Planejamento; (vi) Combate. Existem quatro métodos de combate ao fogo nos incêndios florestais (LIF, 2006): (i) Método direto: usado quando a intensidade do fogo permite uma aproximação suficiente da brigada à linha de fogo, são usadas as seguintes técnicas e materiais: água (bombas costais, baldes ou moto-bombas); terra (pás); ou batidas (abafadores); (ii) Método paralelo ou intermediário: usado quando não é possível o método direto e a intensidade do fogo não é muito grande, consiste em limpar, com ferramentas manuais, uma estreita faixa, próxima ao fogo, para deter o seu avanço e possibilitar o ataque direto; (iii) Método indireto: usado em incêndios de intensidade muito grande, consiste em abrir aceiros com equipamento pesado (*e.g.*

trator, motoniveladeira); (iv) Método aéreo: usado nos incêndios de copa, de grande intensidade e área e em locais de difícil acesso às brigadas de incêndio, são usados aviões e helicópteros, especialmente construídos para o combate ao incêndio.

A rapidez e a eficiência na detecção e monitoramento dos incêndios florestais são fundamentais para a viabilização do controle do fogo, redução dos custos nas operações de combate e atenuação dos danos. Para países de grande extensão territorial, como o Brasil, o monitoramento dos incêndios florestais através de imagens de satélites é o meio mais eficiente e de baixo custo quando comparado com os demais meios de detecção (BATISTA, 2004). O lançamento em 1972 do primeiro satélite *Landsat* possibilitou detectar alterações nas áreas florestais do espaço. Desde então, as imagens termais e de infravermelho têm sido usadas na detecção de incêndios e estudos de mapeamento, permitindo que áreas queimadas e não queimadas sejam detectadas através do contraste entre os gradientes térmicos (REMMEL; PERERA, 2001).

6. IMPLEMENTAÇÃO DO PROTÓTIPO

O protótipo do sistema foi implementado em C++, desenvolvemos inicialmente uma versão 2D com a biblioteca SDL. A atual versão tridimensional (Figura 6) 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.



Figura 6 – Ambiente de simulação desenvolvido com OSG+ODE+Demeter

A implementação do protótipo (Figura 6) iniciou com a criação de um mapa que simula a integração das informações de vegetação, topografia e comportamento de fogo detalhados na Seção 5. A criação dos mapas teve como base cartas topográficas (Figura 7(a)) e o mapa de modelos de combustíveis florestais (Figura 7(b)) (MAB, 2006).

Para a simulação da vegetação e correta propagação do incêndio, 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. O fogo pode ser iniciado em uma posição aleatória ou parametrizado em uma posição inicial fixa. A velocidade de propagação respeita dados do modelo retirados de (KOPROSKI, 2005), considerando intensidade do vento e sua orientação. Quanto ao vento, tanto a sua

intensidade como a sua orientação podem ser gerados aleatoriamente ou configurados a partir de dados parametrizados pelo usuário. O tempo de permanência do fogo em uma área é relacionado diretamente ao tipo da vegetação presente e se comporta baseado nos valores de tipo de vegetação, inclinação do terreno, intensidade e orientação do vento. Desta forma a propagação do fogo busca simular de modo bastante realístico a forma como o fogo se propagaria em um ambiente real. A simulação da comunicação é feita utilizando um sistema baseado em Quadro-Negro. O sistema de comunicação simula a troca de mensagens que vão do agente monitor para o agente líder, do agente líder aos agentes de combate, e dos agentes de combate ao agente líder. Existem casos de comunicação um para um e de um para todos. A fila usada como função de Quadro-Negro armazena as seguintes informações: indicador de remetente, indicador de destinatário, e tipo da mensagem (*e.g.* aviso de incêndio, aviso de fim de incêndio, negociação de times, posição do incêndio). No ambiente desenvolvido, cada árvore existe como um modelo OSG e como um cilindro ODE, assim, o veículo da Figura 6(a) está colidindo com um cilindro (não visualizado). O ambiente permite que, através de parâmetros, possamos habilitar ou não a exibição dos objetos físicos ao invés de apenas a sua representação gráfica, neste exemplo, uma árvore.

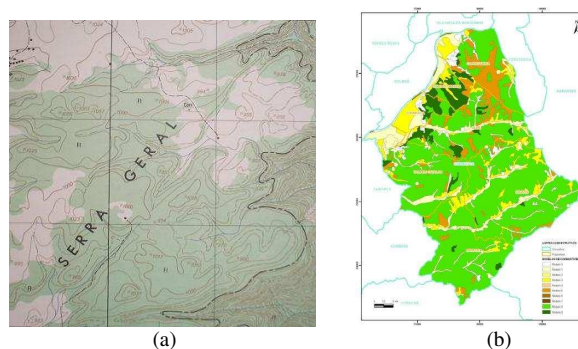


Figura 7 – (a) Fragmento de carta topográfica representando um quadrante de 6km x 6km e (b) Mapa de modelos de combustíveis

Fonte: (MAB, 2006)

6.1 OPERAÇÃO DE IDENTIFICAÇÃO E COMBATE DE INCÊNDIO

O ambiente conta com um grupo de n robôs de comportamento reativo (controlados por uma RNA explicada na próxima Seção) e com um agente deliberativo (mecanismo de planejamento). Em se considerando arquiteturas de sistemas multi-agente, o ambiente desenvolvido possui controle centralizado de planejamento e controle distribuído de ações (local em cada robô móvel). 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, envia uma mensagem para o agente líder. Esta mensagem contém a posição (x,y) do incêndio (simulando uma posição UTM) e a densidade da vegetação na área. O agente líder é o agente responsável pela definição das posições de atuação dos robôs bombeiros no combate ao incêndio. Após receber o *aviso de incêndio* do agente monitor, o agente líder envia para todos os agentes de combate uma mensagem informando *início de incêndio na posição (x,y)* e recebe a distância estimada d (Equação 1) de cada agente de combate ao incêndio.

$$d(P,Q) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (1)$$

Após o agente líder receber as respostas dos agentes de combate, envia mensagens de solicitação de formação para atuação da seguinte forma: se não existe vento, solicita os

8 agentes mais próximos ao incêndio em formação circular equidistante com um raio predeterminado; se existe vento, solicita os 4 agentes mais próximos ao incêndio em formação semi-circular (ferradura) no sentido contrário ao do vento, com raio definido de acordo com a intensidade do vento. Esta formação é baseada em regras e pode ser visualizada nas Figuras 8 e 9. A regra pré-programada que define a posição final de cada robô na composição da formação da equipe (quando circular equidistante) pode ser vista nas Equações 2 e 3.

$$x_f = x_a + r \times \cos\left(i \times \frac{360}{q} \times \frac{\pi}{180}\right) \quad (2)$$

$$y_f = y_a + r \times \text{sen}\left(i \times \frac{360}{q} \times \frac{\pi}{180}\right) \quad (3)$$

Considere x_f e y_f como as coordenadas da posição final do agente, x_a e y_a como as coordenadas da posição central do incêndio, r como o raio de atuação, q a quantidade total de agentes e i o índice do agente. As posições são negociadas e confirmadas com a comunicação entre os agentes da equipe. Os agentes do time possuem quatro sensores de temperatura que servem como alerta, quando a temperatura de um deles excede o máximo especificado, o agente se desloca no sentido do sensor com a menor temperatura e solicita ao agente líder a atualização da formação do time. O comportamento dos agentes de combate é reativo, deslocando-se em direção a posição de seu objetivo específico desviando de obstáculos. O método de combate de incêndio simulado é o método indireto. Os agentes de combate simulados são motoniveladeiras 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 as Figuras 8 e 9.

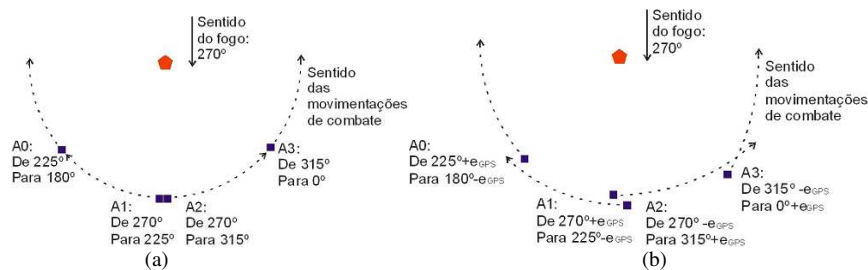


Figura 8 – Exemplo de trajetórias de combate quando o sentido do vento é de 270° e existem 4 agentes de combate: (a) Não considera erro no GPS e (b) Considerando erro no GPS, apresentando limites propositalmente redundantes

Quanto ao controle de posicionamento, um sensor do tipo GPS é simulado em cada robô. Em experimentos que realizamos com um GPS *Garmin Etrex* (www.garmin.com) obtivemos um erro médio de 18,6 metros. Considerando que cada agente possui seu próprio GPS o tratamento deste erro é crucial na criação dos aceiros. Tratamos esta informação de duas maneiras, a primeira faz com que o erro médio deste sensor durante o deslocamento seja usado somado a distância de criação do aceiro e também é somado ao final da área de criação, como mostra a Figura 8(b).

A Figura 8(a) apresenta as posições e trajetórias com “posicionamento perfeito”, o que não é possível de se obter em uma situação real. Devido ao erro de posicionamento do GPS não é possível se estabelecer uma rota que se encaixe perfeitamente como a apresentada nesta Figura. Deste modo a preparação do aceiro (semi-círculo indicado na figura) não será executada de forma correta. A Figura 8(b) apresenta as correções

adicionadas ao algoritmo de modo a obter uma trajetória que permita criar um aceiro incluindo o erro do GPS (e_{GPS}) no modelo de deslocamento dos agentes. As Figuras 9(a) e 9(b) demonstram os resultados obtidos em uma das simulações realizadas com o protótipo 2D.

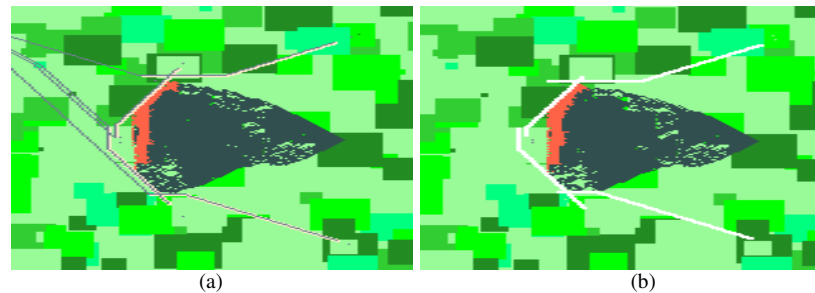


Figura 9 – Simulação com o protótipo 2D considerando o erro de GPS: (a) Trajetórias dos robôs-bombeiros e (b) Criação do aceiro. Ambas apresentam limites propositalmente redundantes

6.2 MORFOLOGIA DOS ROBÔS MÓVEIS

Os robôs móveis foram desenvolvidos com a biblioteca de simulação de corpos rígidos artigos ODE. A Figura 10(b) apresenta o veículo desenvolvido. Dada a existência de restrições físicas, a única maneira de controlar o veículo das Figuras 10(b) e 10(c) 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 e termômetro descritos na Seção anterior, cada robô possui também uma bússola (Figura 10(a)), responsável pela obtenção da orientação do veículo e, em conjunto com o GPS, é responsável pela obtenção do azimute (orientação em que deve se deslocar para atingir o alvo).

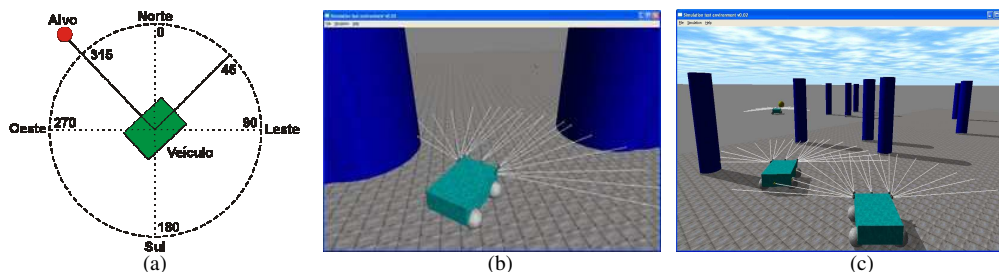


Figura 10 – Sensores desenvolvidos: (a) A bússola é responsável pela orientação, e, junto com o GPS, permite obter o azimute para o alvo, (b) e (c) Sensores de distância

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). Nas Figuras 10(b) e 10(c), cada três linhas brancas representam um sensor.

6.3 REDE NEURAL ARTIFICIAL

Quando um robô recebe a coordenada da posição do incêndio do agente líder, ele deve se deslocar de sua posição atual (aleatória) até a posição de atuação. Para realizar esta navegação, desviando de obstáculos, o agente usa uma Rede Neural Artificial (RNA) do tipo *Backpropagation*. As entradas da RNA são obtidas pelos sensores presentes no

robô e as saídas são as forças a serem aplicadas nos atuadores. A RNA possui 9 entradas: orientação do veículo e orientação para o alvo (obtidos através de bússola e GPS simulados) mais as proximidades de obstáculos medidas por 7 sensores do tipo sonar. As saídas são: força a aplicar no motor linear (torque, de 0.0 a 3.0) e força a aplicar no motor angular (giro da barra da direção, de -0.8 a 0.8).

A base de dados para treinamento da RNA foi obtida a partir de um sistema de regras simples, rodamos 15 simulações com diferentes pontos de início e destino para obter dados específicos de navegação e 12 simulações para obter dados de desvios de obstáculos, obtendo um total de 7.720 registros. Esta base de dados foi dividida em 70% para treino e 30% para teste, e, usando o SNNS foram geradas seis Redes Neurais Artificiais com 4, 9, 18, 24, 30 e 36 neurônios na camada oculta. Foi realizado o treino de cada RNA e análise de 500, 1.000, 1.500 e 2.000 ciclos. A Tabela 1 apresenta os dados de treino e teste das Redes Neurais desenvolvidas, os valores apresentados são o Erro Médio Absoluto (*Mean Absolute Error* - MAE) (Equação 4).

$$|E_{MAE}| = \frac{1}{N} \sum_{i=1}^N |x_i - \bar{x}_i| \quad (4)$$

Tabela 1 – Erro médio Absoluto das RNAs

		Exp. 1				Exp. 2				Exp. 3			
		Treino		Teste		Treino		Teste		Treino		Teste	
RNA	Ciclos	Giro	Veloc.	Giro	Veloc.	Giro	Veloc.	Giro	Veloc.	Giro	Veloc.	Giro	Veloc.
9x4x1	500	0,228	0,479	0,223	0,478	0,220	0,450	0,222	0,445	0,230	0,472	0,227	0,472
9x4x1	1000	0,170	0,478	0,174	0,477	0,158	0,401	0,161	0,404	0,153	0,437	0,157	0,438
9x4x1	1500	0,136	0,424	0,139	0,424	0,150	0,388	0,153	0,390	0,142	0,411	0,147	0,412
9x4x1	2000	0,138	0,412	0,143	0,412	0,149	0,386	0,152	0,389	0,143	0,408	0,149	0,409
9x9x1	500	0,235	0,465	0,229	0,462	0,135	0,454	0,139	0,453	0,141	0,426	0,146	0,428
9x9x1	1000	0,144	0,413	0,145	0,413	0,131	0,406	0,135	0,408	0,128	0,376	0,134	0,379
9x9x1	1500	0,131	0,402	0,136	0,401	0,130	0,395	0,135	0,398	0,125	0,361	0,131	0,366
9x9x1	2000	0,128	0,397	0,134	0,396	0,129	0,391	0,133	0,394	0,125	0,349	0,132	0,354
9x18x1	500	0,231	0,468	0,223	0,463	0,290	0,476	0,275	0,473	0,183	0,475	0,186	0,471
9x18x1	1000	0,133	0,403	0,138	0,405	0,131	0,408	0,136	0,409	0,137	0,416	0,141	0,417
9x18x1	1500	0,134	0,396	0,139	0,399	0,125	0,344	0,130	0,346	0,134	0,392	0,139	0,396
9x18x1	2000	0,137	0,391	0,142	0,393	0,112	0,311	0,117	0,316	0,135	0,382	0,141	0,387
9x24x1	500	0,190	0,412	0,191	0,410	0,209	0,474	0,203	0,473	0,172	0,446	0,171	0,444
9x24x1	1000	0,142	0,314	0,135	0,318	0,177	0,392	0,176	0,392	0,130	0,340	0,133	0,341
9x24x1	1500	0,127	0,285	0,128	0,292	0,140	0,220	0,140	0,229	0,132	0,289	0,136	0,294
9x24x1	2000	0,096	0,261	0,100	0,269	0,130	0,205	0,132	0,215	0,112	0,183	0,116	0,194
9x30x1	500	0,145	0,383	0,150	0,384	0,159	0,382	0,157	0,383	0,188	0,309	0,187	0,317
9x30x1	1000	0,135	0,355	0,139	0,358	0,142	0,340	0,145	0,346	0,127	0,294	0,131	0,301
9x30x1	1500	0,124	0,338	0,128	0,341	0,130	0,317	0,135	0,325	0,122	0,288	0,126	0,296
9x30x1	2000	0,116	0,319	0,121	0,323	0,126	0,304	0,132	0,311	0,115	0,283	0,119	0,291
9x36x1	500	0,142	0,452	0,144	0,445	0,229	0,471	0,221	0,464	0,246	0,470	0,232	0,462
9x36x1	1000	0,139	0,400	0,142	0,401	0,136	0,402	0,138	0,401	0,172	0,452	0,147	0,446
9x36x1	1500	0,137	0,374	0,140	0,374	0,123	0,373	0,126	0,374	0,141	0,397	0,143	0,396
9x36x1	2000	0,133	0,360	0,136	0,359	0,114	0,342	0,118	0,346	0,137	0,384	0,141	0,385

A RNA que apresentou o menor MAE tanto no giro como na velocidade foi a com 24 neurônios na camada oculta, com 2.000 ciclos de treino da terceira semente aleatória. Esta RNA foi convertida em um programa C e utilizada nos robôs.

6.4 APLICAÇÃO DA REDE NEURAL ARTIFICIAL

Após a implementação da RNA no controle de navegação com desvio de obstáculos nos robôs móveis, realizamos uma série de simulações com o ambiente contendo 3 robôs, com pontos de início e destino aleatórios e com uma quantidade parametrizada de cilindros. Os resultados podem ser visto nas Tabelas 2 e 3. Para tornar as simulações mais rápidas, trabalhamos em um terreno pequeno, representado $80m \times 100m$, cada veículo tem dimensões de $2,5m \times 3m$ e cada cilindro tem diâmetro de $1m$. A Figura 11 apresenta imagens de seqüências de uma simulação onde podemos ver o resultado correto de navegação com desvio e o posicionamento final dos robôs.

Tabela 2 – Resultado das simulações com 10 cilindros

Simulação	Quantidade de cilindros	Quantidade de chegadas satisfatórias	Quantidade de colisões	Quantidade de trajetos errados
1	10	3	0	0
2	10	3	0	0
3	10	2	0	1
4	10	3	0	0
5	10	3	0	0
6	10	3	0	0

Tabela 3 – Resultado das simulações com 20 cilindros

Simulação	Quantidade de cilindros	Quantidade de chegadas satisfatórias	Quantidade de colisões	Quantidade de trajetos errados
7	20	2	0	1
8	20	3	0	0
9	20	2	0	1
10	20	3	0	0
11	20	1	1	1
12	20	2	1	0

Nas duas colisões observadas, apenas o sensor frontal identificava o cilindro, e o veículo seguia em frente com velocidade reduzida. Esse tipo de desvio existia na base de treino, que devia priorizar giro para direita, porém, por ter produzido alguns erros na simulação, encoraja a construção de uma base de treino maior.

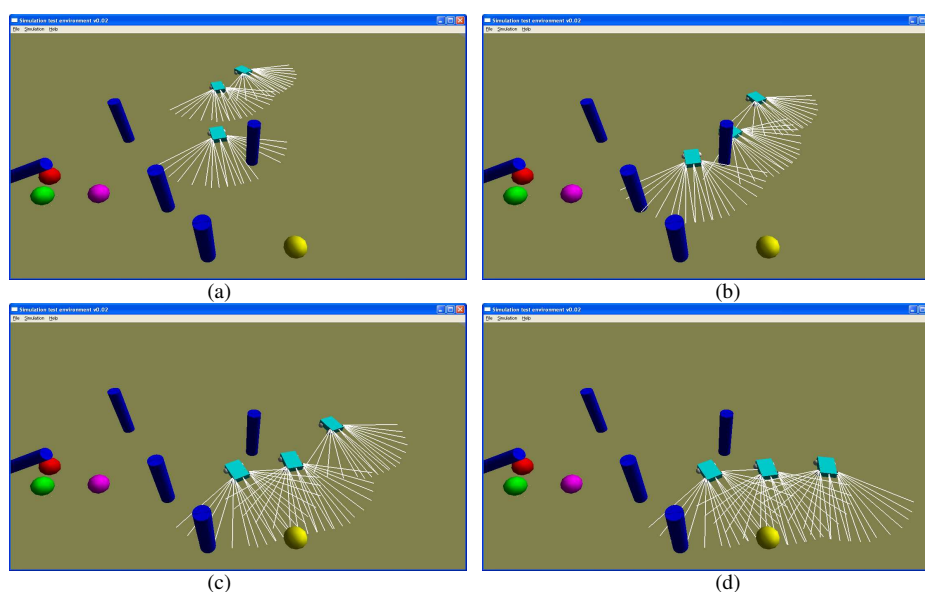


Figura 11 – Seqüências de uma simulação com navegação e desvio satisfatórios

7. CONCLUSÃO

O objetivo deste artigo foi detalhar o projeto e o desenvolvimento de um sistema multi-agente que opera em um ambiente virtual de simulação realística. Neste sistema, uma equipe heterogênea de agentes autônomos trabalha cooperativamente a fim de realizar com sucesso a identificação e o combate de incêndios em áreas florestais, sem intervenção humana. A abordagem implementada adota estratégia centralizada, coordenada pelo QG, mas com controle de ação distribuído, onde cada agente possui autonomia na execução das tarefas que lhe são passadas. Os robôs de combate são fisicamente simulados e as informações sensoriais de cada robô (*e.g.* GPS, bússola, sonar) servem de entrada para uma Rede Neural Artificial que controla os atuadores do veículo para assim realizar navegação com desvio de obstáculos de um ponto aleatório do terreno até um ponto de atuação no combate ao incêndio. O ambiente desenvolvido apresenta uma série de características fundamentais para a simulação realística da operação, como terrenos irregulares, processos naturais e restrições físicas na criação e no uso dos robôs móveis. A comunicação através do uso de *blackboard* se mostrou bastante prática e simples de implementar. O terreno simula diversos aspectos naturais, como tipos de vegetação, topografia, e respeita questões como taxa de propagação do fogo baseado na intensidade e orientação do vento, vegetação e topografia. Os resultados das simulações demonstram que a Rede Neural Artificial controla satisfatoriamente os robôs móveis 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.

REFERÊNCIAS

- ANTUNES, M. A. H. **Uso de satélites para detecção de queimadas e para avaliação do risco de fogo**, Ação Ambiental, 12:24-27, 2000.
- BARROS, L. M., GONZAGA, L., RAPOSO, A. B. **Open Scene Graph: conceitos básicos e aplicações em realidade virtual**, Tutorial on IX Symposium on Virtual and Augmented Reality (SVR), 2007.
- BATISTA, A.C. **Deteção de incêndios florestais por satélite**, Revista Floresta, Curitiba, Paraná, n 34, Mai/Ago, 237-241, 2004.
- BEKEY, G. A. **Autonomous Robots: From Biological Inspiration to Implementation and Control**. Cambridge, Massachusetts, USA: The MIT Press, 2005.
- BRAGA, A.; CARVALHO, A.; LUDERMIR, T. **Redes Neurais Artificiais: Teoria e aplicações**. Rio de Janeiro, RJ, Brasil: LTC, 262 p, 2000.
- CASTRO, F. X.; TUDELA, A.; SEBASTIA, M. T. **Modeling moisture content in shrubs to predict fire risk in catalonia**. Agricultural and Forest Meteorology, Spain, v. 116, p. 49–59, 2001.
- CE - Comissão Européia. **O que faz a Europa? Incêndios florestais**. Disponível em: <<http://ec.europa.eu/research/leaflets/disasters/pt/forest.html>>. Acesso em set. 2006.
- CPTEC/INPE. **Centro de previsão do tempo e estudos climáticos - Instituto nacional de pesquisas espaciais**. Disponível em <<http://www.cptec.inpe.br/queimadas>>. Acesso em out. 2006.
- DEMETER, **Demeter Terrain Engine**. Disponível em: <<http://www.tbgssoftware.com/>>. Acesso em jun. 2007

DUDEK, G.; JENKIN, M. **Computational Principles of Mobile Robotics**, Cambridge, London, UK: The MIT Press, 280 p, 2000.

FONSECA, J. M. **Protocolos de negociacao com coligacoes em sistemas multi-agente**. Tese de Doutorado, Universidade Nova de Lisboa, Portugal, 2000.

GARCIA, A. C. B.; SICHMAN, J. S. **Sistemas inteligentes: Fundamentos e aplicações**. In: Barueri, SP, Brasil: Manole, Cap. Agentes e Sistemas Multi-Agentes, p. 269–306, 2003.

GENESERETH, M. R.; KETCHPEL, S. P. **Software agents**. Communications of the ACM, v. 37, n. 7, p. 49–53, 1994.

GO, J. *et al.* **Accurate and flexible simulation for dynamic, vision-centric robots**. International Joint Conference on Autonomous Agents, 2004.

HAYES-ROTH, B. **An architecture for adaptive intelligent systems**. Artificial Intelligence, v. 72, p. 329–365, 1995.

HEINEN, F. J. **Robótica Autônoma: Integração entre Planificação e Comportamento Reativo**, Ed. Unisinos, São Leopoldo, RS, Brasil, 84 p, 2000.

HORNIK, K.; STINCHCOMBE, M.; WHITE, H. **Multilayer feedforward networks are universal approximators**. Neural Networks, v. 2, p. 359–366, 1989.

JPL/NASA. **Jet Propulsion Laboratory / NASA**. Disponível em: <<http://www.robotics.jpl.nasa.gov>>. Acesso em maio 2007.

KARTALOPOULOS, S. V. **Understanding Neural Networks and Fuzzy Logic - Basic Concepts and Applications**. New York, USA: IEEE Press Understanding Science and Technology Series, 232p, 1996.

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.

LEE, E. *et al.* **The hearsay-ii speech-understanding system: Integrating knowledge to resolve uncertainty**. Computing Surveys, v. 12, n. 2, p. 213–253, 1980.

LIF - Laboratório de Incêndios Florestais. **Pesquisas e projetos em prevenção e combate de incêndios florestais**. Disponível em <<http://www.floresta.ufpr.br/~firelab>>. Acesso em set. 2006.

MAB – Ministério da Agricultura do Brasil, **Guia metodológico para elaboração de plano municipal/intermunicipal de defesa da floresta contra incêndios**, 2006

MCCULLOCH, W.; PITTS, W. **A logical calculus of the ideas immanent in nervous activity**. Bulletin of Mathematical Biophysics, v. 5, p. 115–133, 1943.

MAES, P. **Intelligent software: easing the burdens that computers put on people**. IEEE Expert, v. 11, p. 62–63, 1996.

MARCHI, J. **Navegação de robôs móveis autônomos: Estudo e implementação de abordagens**. Dissertação de Mestrado. Universidade Federal de Santa Catarina, 2001.

MONDADA, F. *et al.* **The cooperation of swarm-bots: Physical interactions in collective robotics**. IEEE Robotics and Automation Magazine, p. 21–28, 2005.

NIKOLOPOULOS, C. **Expert Systems - Introduction to First and Second Generation and Hybrid Knowledge Based Systems**. New York, USA: Marcel Dekker Inc. Press, 1997.

NWANA, H. S. **Software agents: An overview**. Knowledge Engineering Review, v. 1, n. 3, p. 1–40, 1996.

ODE, **Open Dynamics Engine**, Disponível em: <www.ode.org>. Acesso em jun. 2007.

OSAGIE, P. **Distributed Control for Networked Autonomous Vehicles**, Dissertação de Mestrado, KTH CSC, Royal Institute of Technology, Sweden, 2006.

OSG, **Open Scene Graph**, OSG Community. Disponível em: <<http://www.openscenegraph.com>>. Acesso em jun. 2007.

- OSÓRIO, F. S. **INSS: Un Système Hybride Neuro-Symbolique pour l'Apprentissage Automatique Constructif**. Tese de Doutorado. Institut National Polytechnique de Grenoble (INPG), Grenoble, França, 1998.
- OSÓRIO, F. S., *et al.* **Increasing Reality in Virtual Reality Applications through Physical and Behavioural Simulation**, Research in Interactive Design - Proceedings of the Virtual Concept Conference (Summer School). Springer Verlag, v.2, p. 1-45, 2006.
- 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.
- PFEIFER, R.; IIDA, F.; BONGARD, J. **New robotics: Design principles for intelligent systems**. Artificial Life, v. 11, p. 99–120, 2005.
- PFEIFER, R.; SCHEIER, C. **From perception to action: The right direction?** Proceedings of the From Perception to Action Conference, IEEE Press, p. 1–11, 1994.
- PFEIFER, R.; SCHEIER, C. **Understanding Intelligence**. Cambridge, Massachusetts, USA: The MIT Press, 697 p, 1999.
- REIS, L. P. **Coordenação em sistemas multi-agente: aplicações na gestão universitária e futebol robótico**. Tese de Doutorado, Univ. do Porto, Portugal, 2003.
- REMMEL, T. K., PERERA, A. H. **Fire mapping in a northern boreal forest: assessing AVHRR/NDVI methods of change detection**, Forest Ecology and Management 152:119-129, 2001.
- REZENDE, S.O. **Sistemas inteligentes: fundamentos e aplicações**, Ed. Manole, São Paulo, 2003.
- ROSEMBLATT, F. **The perceptron: A probabilistic model for information storage and organization in the brain**. Psychol, n. 65, p. 386–408, 1958.
- ROSEMBLATT, F. **Principles of Neurodynamics**. New York, USA: Spartan, 1962.
- RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. **Learning Internal Representations by Error Propagation**. Cambridge, MA, USA: The MIT Press, 1986.
- RUMELHART, D.; MCCLELLAND, J. **Parallel Distributed Processing**. Cambridge: MIT Press, 1986.
- SILVEIRA, R. A. **Introdução a Sistemas Multi-agente**, Universidade Federal de Santa Catarina (UFSC), 2006.
- SMITH, R. **Open Dynamics Engine v0.5 User Guide**. 2006
- SNNS. **Stuttgart Neural Network Simulator**. Disponível em: <<http://wwwra.informatik.uni-tuebingen.de/SNNS/>>. Acesso em maio 2007.
- STONE, P.; VELOSO, M. **Multiagent systems: A survey from a machine learning perspective**, Autonomous Robots, v. 8, n. 3, p. 345–383, 2000.
- WIDROW, B.; HOFF, M. E. **Adaptative switching circuits**. IRE Western Electric Show and Convention Record, p. 96–104, 1960.
- WOOLDRIDGE, M. **Reasoning about Rational Agents**. Cambridge, Massachusetts, USA: The MIT Press, 2000. 240 p.
- WOOLDRIDGE, M.; JENNINGS, N. **Intelligent agents: Theory and practice**. Knowledge Engineering Review, v. 10, n. 2, 1995.

AGRADECIMENTOS

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

ⁱ O código-fonte e os vídeos desenvolvidos neste trabalho podem ser obtidos em <http://pessin.googlepages.com/robombeiros>.