

Desenvolvimento de um Ambiente para Simulação de Robótica Móvel em Áreas Florestais

Gustavo Pessin¹, Fernando Osório¹, Soraia Musse², Vinícius Nonnemacher³,
Sandro Souza Ferreira³

¹PIPCA – Universidade do Vale do Rio dos Sinos (UNISINOS)
Av. Unisinos 950 – São Leopoldo – RS – Brasil

²Faculdade de Informática – PUC-RS
Av. Ipiranga, 6681 – Porto Alegre – RS – Brasil

³GT JEDi – Universidade do Vale do Rio dos Sinos (UNISINOS)
Av. Unisinos 950 – São Leopoldo – RS – Brasil

fosorio@unisinos.br, soraia.musse@puccrs.br, {pessin, vnonnemacher,
sandro.s.ferreira}@gmail.com

***Resumo.** O objetivo deste artigo é detalhar o projeto e o desenvolvimento de um ambiente de simulação que representa uma área florestal e que tem a finalidade de suportar a operação de um sistema multi-robótico fisicamente simulado¹. Este sistema tem por meta ser aplicado no processo de automatização da identificação e combate de incêndios em áreas florestais. 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. Os primeiros resultados demonstraram que este ambiente de simulação pode vir a ter um papel muito importante no planejamento e execução de operações reais de robótica móvel em áreas florestais.*

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 visão, 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 e Jenkin 2000]. Existem diversas áreas onde a habilidade de um único agente autônomo 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-

¹ Código-fonte, vídeos e outras informações deste trabalho disponíveis em <http://pessin.googlepages.com/robombeiros>

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. Para que seja possível desenvolver uma simulação realística onde um grupo de robôs autônomos trabalhe cooperativamente a fim de realizar com sucesso a identificação e o combate de incêndios em áreas florestais, sem intervenção humana necessitamos de um ambiente que combine diversas funcionalidades. 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 3D não planos (terreno irregular, como um ambiente natural); (iii) Facilidades de comunicação entre os agentes; (iv) Capacidade de aplicação de diversos métodos de controle (*e.g.* Regras, Redes Neurais Artificiais); (v) Capacidade de aplicação de métodos de evolução de estratégias (*e.g.* Algoritmos Genéticos); e (vi) Simulação de processos naturais (como a propagação do fogo). Assim, optamos por desenvolver nosso próprio ambiente. Ambientes com terrenos 3D gerados com a biblioteca de programação OSG junto com a biblioteca de programação *Demeter* permitem combinar o mapa de elevação juntamente com uma determinada distribuição de vegetação, criando assim um terreno bastante realista [Osório et al. 2006]. Além disso, o uso da biblioteca de programação ODE permite implementar atributos físicos (*e.g.* atrito, fricção, gravidade, colisão) tornando o sistema ainda mais realístico.

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. Quanto a prejuízos humanos, por exemplo, na Austrália, em 1983 um incêndio que atingiu cerca de 400.000ha matou 75 pessoas. No Brasil, um incêndio no Paraná, em 1973 provocou 110 mortes [LIF 2006]. 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].

As principais metas relacionadas a este 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 de incêndios; (iii) Pesquisar ferramentas e técnicas reais de combate à incêndios florestais; (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.

Neste artigo apresentamos na Seção 2 uma pequena descrição de simulação e modelagem, com as ferramentas pesquisadas e as bibliotecas de simulação selecionadas para o desenvolvimento do ambiente. Na Seção 3 descrevemos sucintamente características importantes de robótica móvel que tenham relação com as necessidades deste trabalho. Na Seção 4 detalhamos o desenvolvimento do ambiente e a simulação do sistema multi-robótico. Finalizamos apresentando a conclusão 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 realista [Pfeifer e 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 este 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.

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).

Um dos requisitos básicos do nosso ambiente é que todas as bibliotecas de programação sejam software livre e em linguagem C/C++. As bibliotecas utilizadas são: (i) *Open Dynamics Engine* (www.ode.org) é uma biblioteca desenvolvida para a simulação física de corpos rígidos articulados [Smith 2006], utilizada para o desenvolvimento dos robôs e do mundo colisivo. Uma estrutura articulada é criada quando corpos rígidos são conectados por algum tipo de articulação, como, por

exemplo, um veículo terrestre que tem a conexão de rodas em um chassi. A ODE não tem como objetivo realizar simulações de outras dinâmicas além da dinâmica de corpos rígidos (e.g. ondas, fluídos, corpos flexíveis, fraturas). O sistema de detecção de colisão é nativo e suporta diversas primitivas (e.g. esfera, caixa, cilindro, plano, raio). A utilização da ODE no nosso ambiente é fundamental por fornecer restrições físicas, principalmente na definição da morfologia dos robôs; (ii) **Open Scene Graph** (www.openscenegraph.org) 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; (iii) **DrawStuff**: 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** (www.tbgssoftware.com) 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.

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 [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, mecânicas, eletrônicas (como falta de precisão de sensores) e computacionais que tornam a criação de um conjunto de regras uma tarefa árdua e sujeita a erros.

4. Implementação do Protótipo

O ambiente simulado conta com um grupo de n robôs de comportamento reativo (reação sensorial-motora) e com um agente deliberativo (mecanismo de planejamento), assim, outra característica deste ambiente é permitir a definição de tipos de comportamentos diferentes para cada tipo de agente. 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 aproximada (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 vetorial d de cada agente de combate ao incêndio. 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 na Figura 3. 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 1 e 2.

$$x_f = x_a + r \times \cos\left(i \times \frac{360}{q} \times \frac{\pi}{180}\right) \quad (1) \quad y_f = y_a + r \times \sin\left(i \times \frac{360}{q} \times \frac{\pi}{180}\right) \quad (2)$$

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 e de áreas densas de vegetação. O método de combate de incêndio simulado é o método indireto. 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 Figura 3. O desenvolvimento dos objetivos mais específicos deste trabalho são detalhados a seguir.

- **Simulação de Processos Naturais:** 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]. 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]. A criação dos mapas teve como base cartas topográficas e o mapa de modelo de combustíveis florestais do Ministério da Agricultura do Brasil. 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. 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.

- **Facilidades de Comunicação entre os Agentes:** A comunicação permite que os agentes em um ambiente multi-agente troquem informações que servem de base para coordenar suas ações e realizar cooperação. Desenvolvemos o ambiente com possibilidade de comunicação baseado em Quadro-Negro (*blackboard*). 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, *timestamp*, e tipo da mensagem (*e.g.* aviso de incêndio, aviso de fim de incêndio, negociação de times, posição do incêndio).

- **Capacidade de aplicação de diversos métodos de controle:** O ambiente desenvolvido permite que cada agente tenha seu próprio método de controle, por exemplo, os robôs móveis foram programados de duas formas, inicialmente foi aplicado um conjunto de regras de controle motor e posteriormente foi aplicada uma Rede Neural Artificial que permite aos robôs móveis realizar navegação autônoma entre dois pontos evitando colisão com obstáculos.

- **Capacidade de aplicação de métodos de evolução de estratégias:** Desenvolvemos para esse trabalho uma unidade monitora (agente deliberativo), que deve evoluir estratégias de formação de atuação dos robôs móveis (agentes reativos). A evolução de estratégias está sendo implementada com a biblioteca de programação de Algoritmos Genéticos GALib (lancet.mit.edu/ga) e o ambiente permitirá, durante a evolução, que a saída gráfica seja desabilitada, possibilitando a aceleração das simulações até atingir o valor esperado da função objetivo, baseada na minimização da área de vegetação queimada. Atualmente, a unidade monitora tem sua operação baseada em regras.

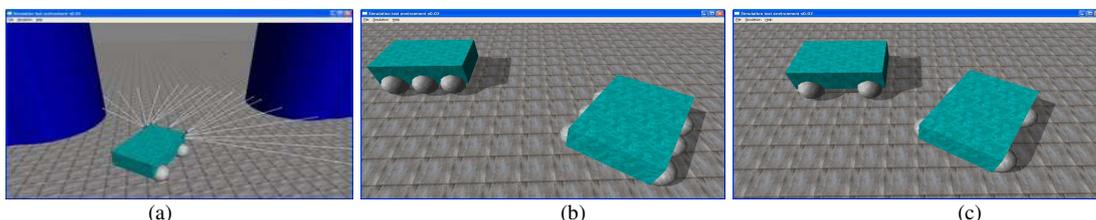


Figura 1. Simulações de: (a) Sensores de distância, (b) Cinemática de esteira e (c) Cinemática de automóvel.

- **Realismo Físico de Modelagem Robótica:** Iniciamos o desenvolvimento focando o realismo de simulação física utilizando as bibliotecas de programação ODE e *DrawStuff*. Com estas bibliotecas podemos definir a morfologia do veículo móvel, bem como sensores e atuadores (Figura 1). Na Figura 2 podemos ver um grupo de 3 veículos interagindo com o ambiente.

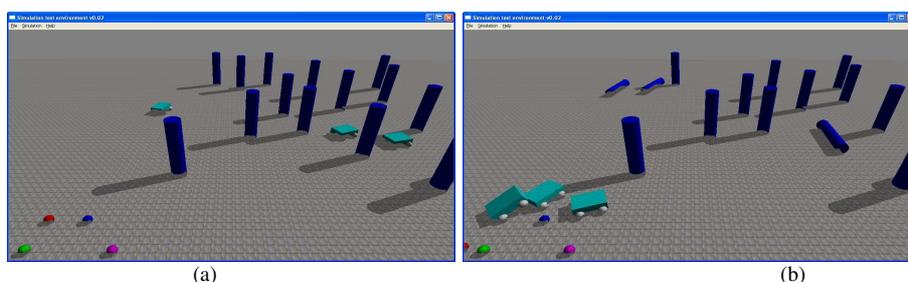


Figura 2. Grupo de 3 veículos (baseado em regras) indo até um objetivo: (a) Início e (b) Chegada ao objetivo sem controle de colisão, cilindros derrubados e dois veículos “acidentados”.

Um dos sensores simulados é o sensor de posicionamento GPS. Em levantamento de campo que realizamos com um GPS *Garmin Etrex* (www.garmin.com) obtivemos um erro médio de 18,6 metros. Considerando que cada robô possui seu próprio GPS, o tratamento deste erro é crucial na criação dos aceiros [Pessin et al. 2007]. 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 3(b).

A Figura 3(a) apresenta o esquema de movimentação com “posicionamento perfeito”, o que não é possível de se obter em uma situação real. Dada a existência de restrições físicas, a única maneira de controlar o veículo da Figura 1(c) é com a aplicação de forças em seus dois motores simulados, um motor angular (para o giro do volante) e um motor linear (para o torque). O veículo da Figura 1(b) foi simulado com cada lateral possuindo três rodas e um motor linear, assim, apresenta dinâmica de veículos com esteira e possui controle de giro e direcionamento baseado na aplicação de forças independentes em cada motor linear.

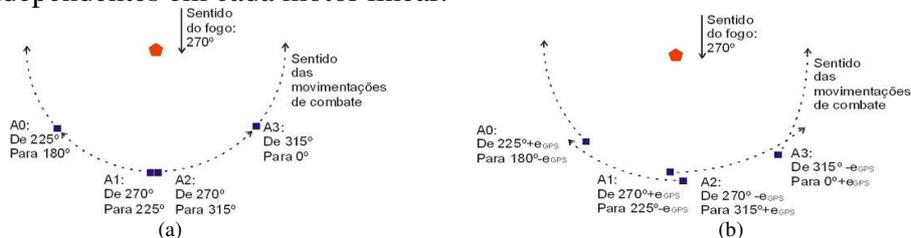


Figura 3. Exemplo de formação de combate quando o sentido do vento é de 270° e existem 4 agentes de combate: (a) não considera erro no GPS e (b) considerando erro no GPS, apresentando limites propositalmente redundantes

- **Realismo Físico de Interação de agentes em terrenos 3D irregulares:** A base do realismo de interação é o uso integrado das bibliotecas de programação ODE, OSG e *Demeter*. Este conjunto permite que os robôs simulados fisicamente respeitem questões como gravidade, inércia e atrito.



Figura 4. Ambiente completo (a) Um veículo preso em uma árvore e (b) Vista do grupo em deslocamento até o incêndio.

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. A Figura 4 mostra o ambiente desenvolvido. O *Demeter*, junto com a SDL e o GDAL, são os responsáveis pela criação do terreno 3D. A OSG permite a leitura do terreno e a leitura dos modelos de árvores e do fogo. Cada árvore existe como um modelo OSG e como um cilindro ODE, assim, o veículo da Figura 4(a) está, na verdade, colidindo com um cilindro (não visualizado). O ambiente permite que

possamos habilitar ou não a exibição dos objetos físicos ao invés de seu modelo gráfico, neste exemplo, uma árvore.

5. Conclusão

O objetivo deste artigo foi detalhar o projeto e o desenvolvimento de um ambiente de simulação que representa uma área florestal e que tem a finalidade de suportar a operação de um sistema multi-robótico. Este sistema multi-robótico tem por meta ser aplicado no processo de automatização da identificação e combate de incêndios em áreas florestais. 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. Os robôs móveis planejam suas ações baseados no uso de dados de seus sensores (*e.g.* GPS, odômetro, sensor de temperatura, bússola). 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. A integração das bibliotecas de programação permitiu o desenvolvimento de um ambiente completamente livre, em C++, integrando as características requisitadas para a operação multi-robótica em áreas florestais. Os primeiros resultados demonstraram que este ambiente de simulação pode vir a ter um papel importante no planejamento e execução de operações reais de robótica móvel em áreas florestais, além de poder ser utilizado em outras aplicações, como jogos ou aplicações de realidade virtual.

Referências

- Antunes, M. A. H. (2000) “Uso de satélites para detecção de queimadas e para avaliação do risco de fogo”. *Ação Ambiental*, 12:24-27.
- Batista, A. C. (2004) “Detecção de incêndios florestais por satélite”, *Revista Floresta* 34, Mai/Ago, 237-241, Curitiba, Paraná.
- Bekey, G. A. (2005) “Autonomous Robots: From Biological Inspiration to Implementation and Control”. Cambridge, USA: The MIT Press, 2005. 577 p.
- CE - Comissão Européia (2006) “O que faz a Europa? Incêndios florestais”. <http://ec.europa.eu/research/leaflets/disasters/pt/forest.html>, setembro.
- CPTEC/INPE (2006) “Centro de previsão do tempo e estudos climáticos - Instituto nacional de pesquisas espaciais”, www.cptec.inpe.br/queimadas, outubro.
- Dudek, G., Jenkin, M. (2000) “Computational Principles of Mobile Robotics”, The MIT Press.
- Go, J., Browning, B., Veloso, M. “Accurate and flexible simulation for dynamic, vision-centric robots”. *Proceedings of International Joint Conference on Autonomous Agents*.
- Koproski, L.P. (2005) “O fogo e seus efeitos sobre a herpetofauna e a mastofauna terrestre no parque nacional de Ilha Grande”, *Dissertação de mestrado, UFPR*.
- LIF - Laboratório de Incêndios Florestais (2006) “Pesquisas e projetos em prevenção e combate de incêndios florestais”, UFPR, www.floresta.ufpr.br/~firelab, setembro.
- Osagie, P. (2006) “Distributed Control for Networked Autonomous Vehicles”. *Dissertação de Mestrado, KTH CSC, Royal Institute of Technology, Sweden*.
- Osório, F. S. et al. (2006) “Increasing Reality in Virtual Reality Applications through Physical and Behavioural Simulation” *Virtual Concept Conference*. Springer, v.2, p. 1-45.
- Pessin, G. et al. (2007) “Simulação Virtual de Agentes Autônomos para a Identificação e Controle de Incêndios em Reservas Naturais”, *IX SVR*, v.1, p. 236-245.
- Pfeifer, R., Scheier, C. (1999) “Understanding Intelligence”. Massachusetts, The MIT Press.
- Smith, R. (2006) “Open Dynamics Engine v0.5 User Guide”.