

Uma Arquitetura de um sistema virtual de controle inteligente para robótica móvel

RESUMO

Este trabalho apresenta uma arquitetura para a construção de um sistema virtual de controle inteligente aplicado à robótica móvel. Este modelo baseia-se na arquitetura em três camadas definido para robótica, considerando questões relevantes de engenharia de software, a fim de obter um sistema robusto e flexível. A arquitetura foi elaborada para atender um conjunto de procedimentos essenciais à interação entre um móvel e seu habitat. Estes procedimentos levantados são discutidos, ressaltando os principais problemas e possíveis soluções. Levou-se em consideração aspectos na interseção entre robótica móvel autônoma e semi-autônoma; robôs reais e virtuais; ambiente gráfico em um browser ou desktop.

1. INTRODUÇÃO

A robótica móvel é uma área que vem crescendo a cada dia. Existem diversos aspectos tecnológicos que compõem as pesquisas nesta área, porém os dois conceitos mais relevantes em um robô móvel são: os aspectos de “inteligência” e eletrônica. Não serão abordados os aspectos eletrônicos de um robô, porém muitos destes componentes serão simulados computacionalmente. Será apresentada uma breve descrição de como se desenvolve a inteligência referente à percepção do ambiente e tomada de decisão quanto ao deslocamento.

Os aspectos de inteligência se diferenciam um pouco para um robô autônomo em relação a um semi-autônomo. Porém, há uma interseção considerável entre as duas linhas de pesquisa. É nesse aspecto que é direcionada a arquitetura proposta neste texto.

Este trabalho visa apresentar um modelo de um simulador de um robô móvel, autônomo ou semi-autônomo, que se desloca em certo ambiente, de uma posição inicial a uma posição final, previamente estabelecida, através de um caminho eficiente.

Esta arquitetura pode ser utilizada na elaboração de um robô real, embora não seja o alvo deste texto. Porém, para tal, é necessário acrescentar ao projeto os aspectos referentes a hardware não abordados. Um projeto interessante é simular um ambiente virtual a partir do deslocamento de um robô em um ambiente real, como será mais comentado a seguir.

Existem diversos modelos de arquitetura conhecidos na literatura. O que será proposto é uma variação do modelo de três camadas para robótica móvel.

2. ARQUITETURA DE ROBÔ

Existem diferentes definições para a arquitetura de um robô presente na literatura. Este trabalho considera a definição abordada por Arkin [Ark98] a qual considera que uma arquitetura de um robô está mais relacionada com os aspectos de softwares do que os de hardwares. Apesar das diferenças conceituais todos os modelos de arquiteturas para robôs móveis descrevem um mecanismo para construção de um sistema de controle inteligente de um robô, apresentando, principalmente, quais os módulos presentes e como estes interagem.

De uma forma geral os módulos de uma arquitetura preocupam-se com aspectos como percepção, planejamento e atuação/execução. A percepção refere-se à compreensão do ambiente e dos elementos nele contido; o planejamento à inteligência do robô; e a atuação, ou execução, ao modo como o robô procede no ambiente, ou seja, movimentos, captura de informações, etc.

Existem diversos modelos de arquitetura para robótica entre os quais ressaltamos os modelos de três camadas como: SSS [Con92], Atlantis [Gat91], 3T [Bon91]. Todas as camadas se dividem semelhantemente da seguinte forma:

- *camada reativa: orienta os sensores e toma decisões de baixo nível, como visão e movimento*
- *camada deliberativa: é responsável pela inteligência do robô, aspectos mais globais, que não são alterados a cada iteração*
- *camada de execução: intermedia essas as duas outras camadas*

As diferenças entre arquiteturas no modelo em três camadas distinguem-se no funcionamento e na comunicação entre camadas (e módulos).

3. ARQUITETURA DO SIMULADOR

Este artigo apresenta uma arquitetura que segue o modelo de três camadas da robótica. Nela as camadas reativas, deliberativas e de execução, juntamente com o modelo do ambiente implementado como habitat do robô, dividem-se primeiramente

segundo o padrão Model-View-Controller [GaHeJoVi95].

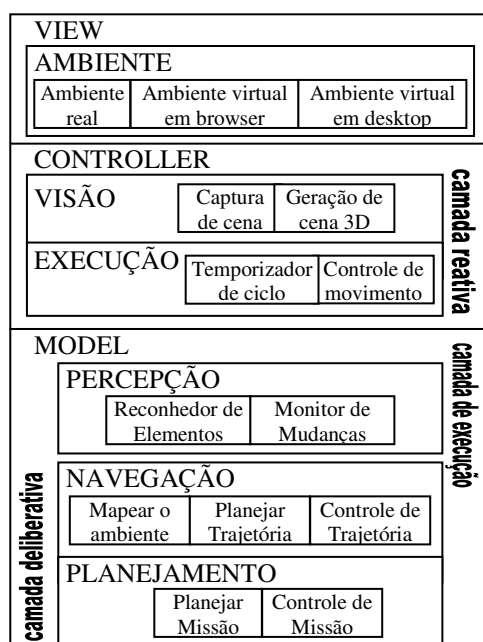


Figura 01: Arquitetura do simulador

A *View* apresenta a seção Ambiente. Esta seção apresenta o(s) módulo(s) que implementa(m) o(s) ambiente(s) onde o robô estará agindo. A *View* pode ser um ambiente real, para um robô real; pode ser um ambiente elaborado para um browser, utilizando, por exemplo, a linguagem VRML; como pode ser um ambiente para um simulador desktop, elaborado, por exemplo, com o OGRE 3D, OpenSceneGraph ou OpenGL; ou um ambiente que funcione em algum outro dispositivo eletrônico que de suporte a tal processamento.

O *Controller* contém as seções de Visão e Execução. Estas seções, e seus respectivos módulos, estão intimamente ligadas à implementação do ambiente, visto que suas entradas e/ou saídas são elementos específicos deste ambiente. Logo os módulos de controle podem não ser reutilizáveis em uma mudança de ambiente. Estas duas seções compõem a camada reativa do robô, pois se referem ao modo de como o robô interage com o ambiente.

O *Model* contém as seções de percepção do ambiente, com o reconhecedor de elementos da cena e o monitor de mudanças. Esta seção compõe a camada de execução, pois propicia ao robô uma compreensão do ambiente para suas tomadas de decisão de inteligência. A inteligência do robô, que é implementada na camada deliberativa, divide-se em duas atividades a princípio: Navegação e Planejamento. A seção de Navegação refere-se ao conjunto de

módulos responsáveis pelo Mapeamento do Ambiente, Planejamento e Controle de Trajetória. Esta é uma atividade de inteligência destacada por ser uma atividade amplamente utilizada em diferentes sistemas em robótica. A Seção de Planejamento está ligada com a missão do robô, não sendo possível assim estender a especificação.

4. PROCEDIMENTOS DO SISTEMA

A arquitetura foi elaborada baseada nos procedimentos de visão, navegação e planejamento de missão. Embora este último não seja amplamente abordado, a arquitetura provê a capacidade de compreensão do ambiente possibilitando a execução da maioria das tarefas adotadas por um robô. A construção e a possibilidade de comunicação de diversos tipos de ambientes, assim como a possibilidade de um sistema social também foram considerados na construção desta arquitetura. Estes aspectos serão abordados nas seções subseqüentes.

4.1. PROCEDIMENTO DE VISÃO

O Sistema de Visão é um dos mais complexos e completos do ser humano, pois fornece um conjunto de informações necessárias à interação do homem com o ambiente. Inicia-se com a captação dos estímulos luminosos do ambiente formando uma imagem, que juntamente aos outros estímulos captados por demais sensores do corpo (som, temperatura, pressão, umidade, cheiro, etc) e as informações contidas na memória, compõe uma cena compreendida pelo cérebro.

O procedimento de visão, na arquitetura apresentada, inicia-se com a captura da cena. Esta pode ser feita por um ou mais sensores, que podem ser de diferentes tipos. O sensoramento provê ao robô as informações necessárias para a construção de uma representação do ambiente onde está inserido. Isto permite uma descrição dinâmica da cena captada, com seus elementos (obstáculos e marcos úteis para sua orientação). Sistemas com uma variedade de sensores tendem a obter resultados mais precisos. A fusão de dados de sensores, ou como são mais conhecidos, fusão de sensores, é o processo de combinação de dados de múltiplos sensores para estimar ou prever estados dos elementos da cena. Esta combinação propicia vantagens como:

- *redundância*: observações de sensores independentes geram uma redundância de dados que se combinam em uma estimativa melhorada das medidas.

- *cooperatividade: a disposição e a movimentação de múltiplos sensores cooperam na inferência do dado observado.*
- *complementaridade: dados provenientes de sensores distintos podem ter informações distintas e complementares à obtenção da estimativa da entidade observada.*

Na robótica móvel, sistemas utilizando visão artificial são utilizados para garantir uma navegação segura ao robô móvel, através do reconhecimento e localização dos objetos. Assim, após a captura da cena pelos sensores o módulo Geração de cena 3D manda uma visão do ambiente para o módulo Reconhecedor de elementos. Nesta etapa a imagem terá um significado compreendido pelo móvel. Esta informação é passada para a inteligência do robô para saber como proceder na ação que estiver executando.

4.2. PROCEDIMENTO DE NAVEGAÇÃO

O principal aspecto na robótica móvel é a ênfase nas questões referentes à locomoção em ambientes complexos e que se modificam dinamicamente. Para cumprir seus objetivos um robô deve ser capaz de adquirir e utilizar informações extraídas do ambiente, estimar sua posição e direção e reconhecer os obstáculos. Essas habilidades tornam-no capaz de reagir, em tempo real, às diferentes ocorrências ao seu redor.

A camada reativa determina processos simples denominados de comportamentos. Entre esses pode-se citar: parar, reduzir velocidade, acelerar, desviar de obstáculos, virar, inverter direção, dirigir-se a meta. Através de uma sequência destes comportamentos é possível alcançar objetivos mais complexos como seguir a trajetória, manter a integridade física do móvel (desviando de obstáculos) e cumprir a missão do robô.

A forma como mapea-se o ambiente internamente no sistema é determinante na sua precisão e performance. As diferentes abordagens para controle de móveis autônomos interagem fortemente com a representação do ambiente. Uma proposta é mapeá-lo através de um grafo conexo, cujos nós referem-se a: entradas/saídas do ambiente, acessos aos cômodos, os cômodos, obstáculos fixos e esquinas. O peso das arestas será calculado de acordo com o custo de processamento no deslocamento entre a posição de um nó ao outro. Este custo verifica-se empiricamente quando o robô detectar uma passagem ainda desconhecida. Este procedimento é realizado no módulo Mapear o ambiente. O mapeamento pode ser um processo contínuo ou realizado inicialmente, em uma fase de treinamento.

Após um mapeamento inicial (e não necessariamente completo) do ambiente é possível um deslocamento planejado para cumprir um objetivo. Este deslocamento deve ser baseado em uma trajetória (caminho) obtido no grafo do ambiente previamente determinado.

O sistema deverá ser capaz de estimar sua posição local para localizar-se globalmente e se recuperar de possíveis erros de localização. Um correto mapeamento do ambiente junto a aplicação correta das leis da cinemática podem resolver tal problema. Uma proposta para a localização do robô no ambiente é a utilização do método Monte Carlo [FoBuDeTh99]. A verificação da posição do móvel e a verificação do cumprimento da trajetória é a função do módulo Controle de trajetória.

Um problema inerente ao controle da trajetória é determinar como o robô deve reagir à presença de um obstáculo. Existem diferentes tipos de obstáculos. Estes podem ser classificados como transponível (aquele que não interrompe a trajetória), intransponível (aquele que o exigirá recalculer a trajetória por outro caminho) e redutor (aquele que permite ao robô seguir pela trajetória, porém a uma velocidade mais lenta). Mesmo mediante aos obstáculos transponíveis e redutores pode ser conveniente recalculer o caminho devido ao aumento do custo do percurso. Uma proposta para o desvio de trajetória é o modelo baseado em Campos Potenciais proposto por [BoKo89].

4.3. PROCEDIMENTO DE PLANEJAMENTO

Aspectos físicos referentes ao ambiente, como pressão atmosférica, atrito do piso, temperatura, umidade do ar, entre outros, podem ser relevantes dependendo do que se pretende. Estas informações extraídas do ambiente podem influenciar diretamente no planejamento da missão, tal como no planejamento da navegação, devendo assim ser levadas em consideração pelos módulos da camada deliberativa.

Como cada sistema de controle inteligente em robótica móvel é criado com um propósito distinto, em um projeto que utilize esta arquitetura pode-se dividir a seção de Planejamento diferentemente do proposto. Isto não fere a arquitetura, pois esta seção está sendo apresentada em caráter genérico.

4.4. AMBIENTE

A independência de ambiente é uma característica que dá uma maior robustez a arquitetura. Uma possibilidade também abordada é a comunicação entre estes ambientes. Por exemplo, pode-se simular um

ambiente virtual, com as informações inferidas pelo Gerador de cena 3D e o reconhecedor de Elementos, obtidos a partir do sensoriamento de um robô real. Um projeto como este pode ser aplicado na realização de alguma tarefa em um ambiente inóspito ou inacessível ao ser humano.

4.5. SISTEMA SOCIAL

Essa arquitetura pode ser utilizada em sistemas comportamentais com mais de um personagem animado. Basta que todos compartilhem o mesmo ambiente. Pode ser necessário à seção de Planejamento algum módulo, também compartilhado, para que todos os personagens tomem uma decisão com um propósito comum. Quando os personagens tiverem um objetivo comum será necessário mediá-los para alcançar a meta. Uma alternativa é utilizar o Padrão de Projeto *Mediator* [GaHeJoVI95] para tal controle dos personagens.

Muitos sistemas sociais implementam aspectos cognitivos nos personagens. Estes são ações não verbais de interação com os demais personagens e com o ambiente. Para tal pode se adicionar um módulo de cognição na seção Execução, ou dependendo da complexidade deste, criar uma seção de cognição na camada Reativa. Aspectos cognitivos estão intimamente ligados a percepção do ambiente e dos elementos nele contido.

Sistemas como estes costumam ser muito complexos, dependendo da inteligência de cada personagem. A implementação simplificada dos personagens, que seguiria a arquitetura em questão, pode ser uma alternativa para que o sistema não seja excessivamente complexo. Outra proposta é utilizar algum sistema de processamento distribuído, de modo que torne-se viável o processamento de todos os personagens.

5. CONCLUSÃO

Os principais procedimentos que nortearam a elaboração desta arquitetura foi o de visão, navegação e planejamento. Estes são amplamente abordados em sistemas de robótica móvel atualmente.

Existem diversos modelos de arquiteturas apresentados na literatura. Esta visa propor um sistema que agregue as boas características da Engenharia de Software. Foi elaborada com os aspectos considerados positivos em outras arquiteturas em três

camadas (citadas anteriormente) especificando melhor as seções e acrescentando módulos com clareza de propósito, facilitando a coesão e interação entre si, contemplando globalmente as principais questões referentes a elaboração de um sistema de controle inteligente de um robô. Além disso, uma correta implementação, seguindo as boas práticas em OO possibilita que o sistema seja, de fato, robusto e flexível. Consequentemente uma característica alcançada é que o modelo seja invariável em relação à visão.

Um modelo como este pode ser aplicado na elaboração de um simulador para testar a resistência e eficiência de um robô em um ambiente (desde que as características do robô e do ambiente sejam devidamente modeladas); em jogos, considerando o robô como um personagem; em sistemas educacionais, onde o usuário oriente a navegação do personagem (robô) por um ambiente que propicie um aprendizado; entre outras áreas da realidade virtual e da Computação gráfica.

6. REFERÊNCIAS BIBLIOGRÁFICAS

- [Ark98] ARKIN, R.C. **Behavior-based robotics**. Cambridge, Massachusetts: MIT Press, 1998.
- [BoKo89] Borenstein, J.; Koren, Y. **Real-time Obstacle Avoidance for Fast Mobile Robots**. In: IEEE Transactions on Systems, Man, and Cybernetics, Vol. 19, No. 5, pp. 1179-1187. 1989.
- [Bon91] BONASSO, R. P. **Integrating reaction plans and layered competences through synchronous control**. In: Proceedings of the Twelfth International Joint Conference on Artificial Intelligence. San Francisco: Morgan Kaufmann Publishers, 1991, p. 1225-1231.
- [CON92] CONNEL, J.H. **SSS: A hybrid architecture applied to robot navigation**. In: Proceedings of 1992 IEEE Int Conf. on Robotics and Automation. Nice, France: 1992. p. 2719-2724.
- [FoBuDeTh99] FOX, D.; BURGARD, W.; DELLAERT, F.; THRUN, S. **Monte Carlo localization: Efficient position estimation for mobile robots**. In: Proc. of the National Conference on Artificial Intelligence (AAAI). 1999
- [GaHeJoVI95] GAMMA, E.; HELM, R.; Johnson, R.; VLISSIDES, J.; **Padrões de Projetos. Soluções reutilizáveis de Softwares Orientado a Objetos**. Bookman, 1995.
- [Gat91] GAT, E. **reliable Goal-directed Reactive Control for Real-World Autonomus Mobile Robots**. Tese – Dept. of Computer Science, Virginia polytechnic Institute and State University, 1991.