

**USP - ICMC - SSC – Pós-Grad. CCMC
SSC 5888 (RMA) - 1o. Semestre 2010**

**Disciplina de
Robôs Móveis Autônomos
SSC-5888**

Prof. Fernando Santos Osório

LRM – Laboratório de Robótica Móvel do ICMC-USP

INCT-SEC – Instituto Nacional de C&T em

Sistemas Embarcados Críticos

Email: fosorio [at] { icmc. usp. br , gmail. com }

Web: <http://www.icmc.usp.br/~fosorio/>



Aula 02 – Arquiteturas de Controle

Agenda: Parte I

Arquiteturas de Controle de Robôs Autônomos

- 1. Controle e Autonomia**
- 2. Arquitetura de Controle Reativa**
Modelo Sensorial-Motor
Controle em Tarefas Reativas
- 3. Arquitetura de Controle Deliberativa**
Mapas e Planejamento de Trajetórias
- 4. Exemplos de Aplicações**
- 5. Vantagens e Desvantagens**

1. Introdução RMA

Robôs Móveis Autônomos

Controle Autônomo

Introdução aos Sistemas Robóticos



SENSORES
ATUADORES
CONTROLE INTELIGENTE

Ciclo:

- Percepção
- [Planejamento / Decisão]
- Ação

Tarefas:

- Localização (Pose)
- Mapas (criar, usar)
- Planejar Rotas
- Desviar de Obstáculos

3

Maio 2010

Aula 02 – Arquiteturas de Controle

Robôs Móveis:

Agentes Autônomos dotados de SENSORES e ATUADORES

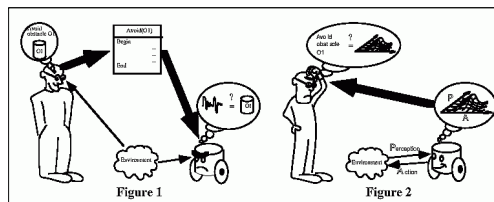
Controle e Autonomia



Como Agir?

Como Interpretar
as Percepções?

Como Tomar Decisões?



4

Maio 2010

Controle e Autonomia

Arquiteturas de Controle Robótico

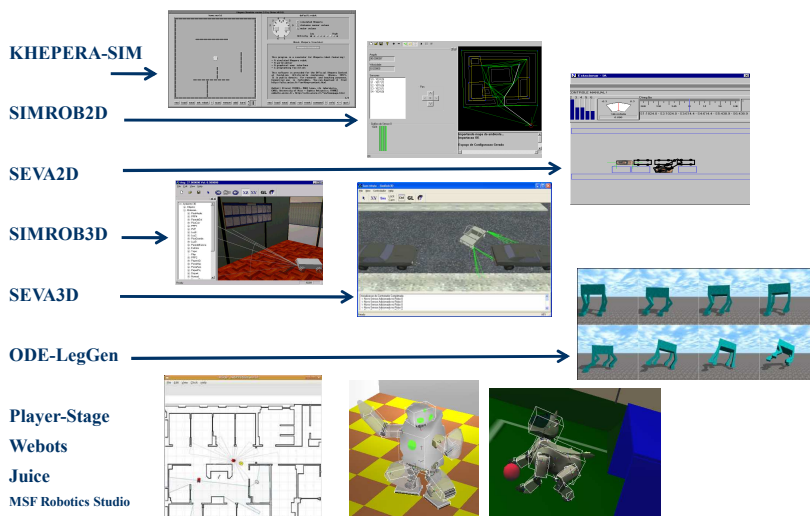
- Arquitetura Reativa
- Arquitetura Deliberativa
- Arquitetura Hierárquica
- Arquitetura Híbrida

Controle => Tomada de Decisão para Realizar Ações

Arquitetura => Modelo de Tomada de Decisão

Autonomia => Independência e Robustez na Tomada de Decisões

Simuladores: Modelos Sensores / Atuadores e Controle



Arquitetura de Controle Reativo

Controle Reativo

Controle: Percepção [=> Decisão] => Ação

Reativo: Percepção => Ação

- Reage diretamente aos estímulos externos;
- Esquema sensorio-motor;

Comportamentos e Tarefas típicas: *Reactive Behaviour*

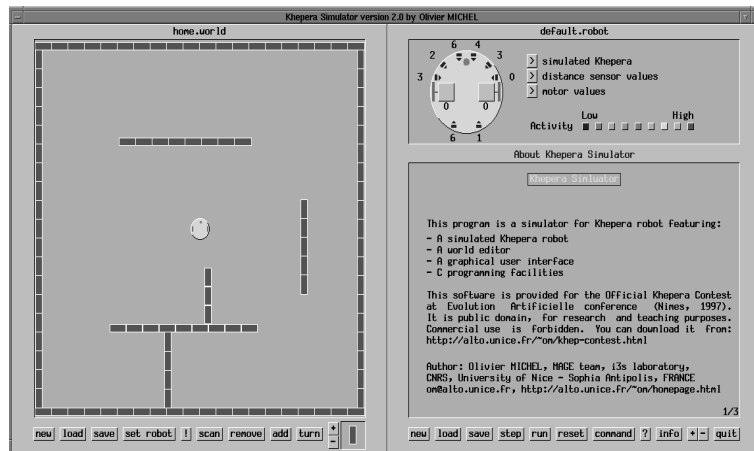
- Vagar pelo ambiente, evitando colisões e obstáculos;
- Acompanhar uma parede ou corredor;
- Comportamento direcionado pela luz;
- Ir em direção a uma determinada orientação

Composição de Comportamentos: Direção x Obstáculo

Arquitetura de Controle: Reativo

Simulador Khepera-SIM: Controle Reativo

Simulador do Khepera / SIM 2.0 Unix / Olivier MITCHELL / INRIA Sophia Antipolis

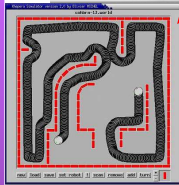


Sensores: 8 IR / Atuadores: 2 motores com cinemática diferencial

1997/98

Arquitetura de Controle: Reativo

Simulador do Khepera



Robô Khepera



9
 Maio 2010

> 2 DC brushed servo motors with incremental encoders
 > 8 infrared proximity and ambient light sensors (SFH900)

Arquitetura de Controle: Reativo

CONTROLE: Arquiteturas REATIVAS

• Reativo: Integração Sensorial-Motora

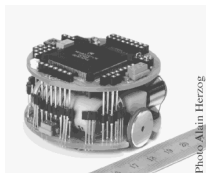
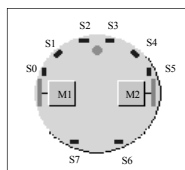
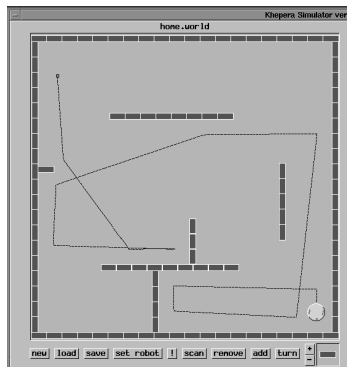


Photo: Allan Herzog



Controle Reativo

IF S1 < Limite and
 S2 < Limite and
 S3 < Limite and
 S4 < Limite
 THEN Action (Go_Forward)

IF S1 < Limite and
 S2 < Limite and
 S3 > Limite and
 S4 > Limite
 THEN Action(Turn_Left)

IF S2 > Limite and
 S3 > Limite and
 S2 > S3 and
 S1 > S4
 THEN Action(Turn_Right)

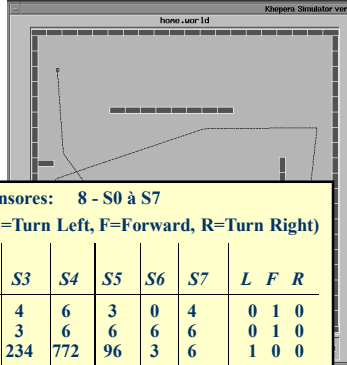
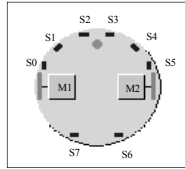
Evitar colisões e obstáculos
 Sensorial-Motor: Sentir => Agir

10
 Maio 2010

Arquitetura de Controle: Reativo

CONTROLE: Arquiteturas REATIVAS

• Reativo: Integração Sensorial-Motora



Sensores: 8 - S0 à S7
 Comandos: 3 ações (L=Turn Left, F=Forward, R=Turn Right)

	S0	S1	S2	S3	S4	S5	S6	S7	L	F	R
4	1	3	4	6	3	0	4	0	1	0	
1	3	6	3	6	6	6	6	6	0	1	0
6	3	104	234	772	96	3	6	1	0	0	
2	6	104	229	724	107	3	4	1	0	0	
563	1023	57	6	6	0	3	1	0	0	1	
544	1023	1	3	4	0	5	1	0	0	1	

Controle Reativo

IF S1 < Limite and
 S2 < Limite and
 S3 < Limite and
 S4 < Limite
 THEN Action (Go_Forward)

IF S1 < Limite and
 S2 < Limite and
 S3 > Limite and
 S4 > Limite
 THEN Action(Turn_Left)

IF S2 > Limite and
 S3 > Limite and
 S2 > S3 and
 S1 > S4
 THEN Action(Turn_Right)

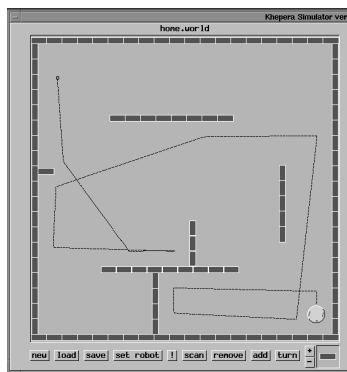
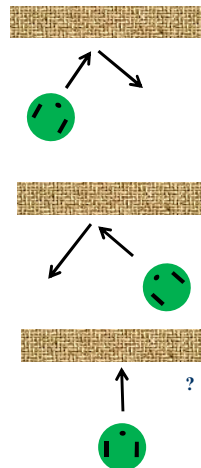
11

Maio 2010

Arquitetura de Controle: Reativo

CONTROLE: Arquiteturas REATIVAS

• Reativo: Integração Sensorial-Motora



Evitar colisões e obstáculos
 Sensorial-Motor: Sentir => Agir

Controle Reativo

IF S1 < Limite and
 S2 < Limite and
 S3 < Limite and
 S4 < Limite
 THEN Action (Go_Forward)

IF S1 < Limite and
 S2 < Limite and
 S3 > Limite and
 S4 > Limite
 THEN Action(Turn_Left)

IF S2 > Limite and
 S3 > Limite and
 S2 > S3 and
 S1 > S4
 THEN Action(Turn_Right)

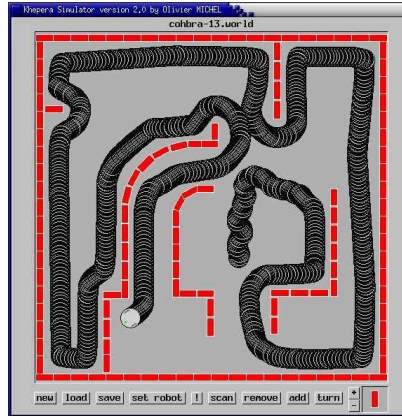
12

Maio 2010

Arquitetura de Controle: Reativo

CONTROLE: Arquiteturas REATIVAS

• Reativo: Integração Sensorial-Motora



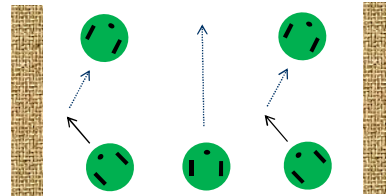
Seguir o contorno de uma parede
Sensorial-Motor: Sentir => Agir

Controle Reativo

Se Longe_Demais(Dist_Parede)
Então Gira_Direção(Lado_da_Parede);

Se Próximo_Demais(Dist_Parede)
Então Gira_Direção(Lado_Oposto_da_Parede);

Se Distância_Adequada(Dist_Parede)
Então Move_Para_Frente();

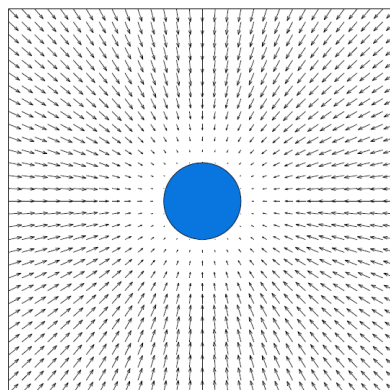


Arquitetura de Controle: Reativo

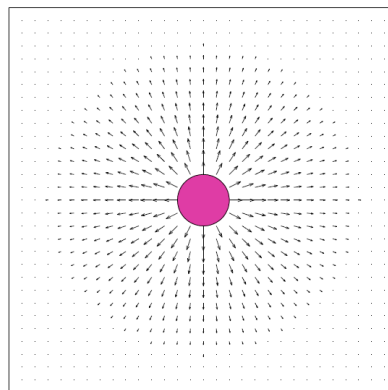
CONTROLE: Arquiteturas REATIVAS*

• Reativo: Integração Sensorial-Motora • Técnica: Campos Potenciais

Evitar colisões e obstáculos
Seguir em direção a um alvo
Sensorial-Motor: Sentir => Agir
* Reativo "não puro" baseado em
informações externas



Força de Atração



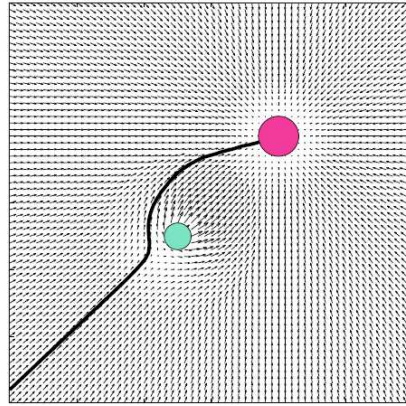
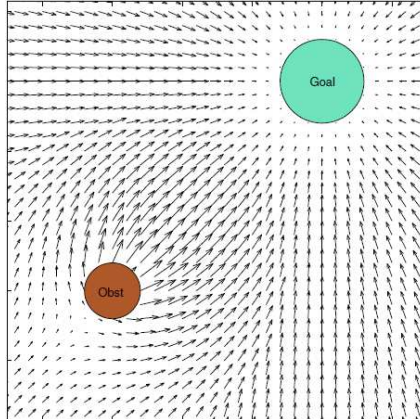
Força de Repulsão

Arquitetura de Controle: Reativo

CONTROLE: Arquiteturas **REATIVAS***

- Reativo: Integração Sensorial-Motora
- Técnica: Campos Potenciais

Evitar colisões e obstáculos
 Seguir em direção a um alvo
 Sensorial-Motor: Sentir => Agir
 * Reativo “não puro” baseado em informações externas



Forças Combinadas: Força de Atração + Força de Repulsão

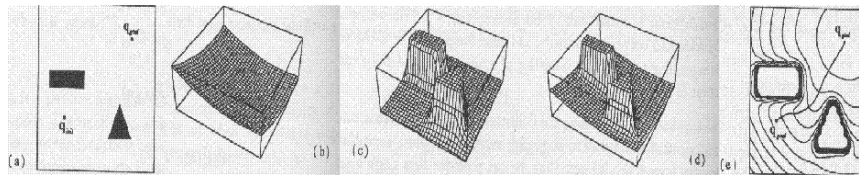
Potential Fields Tutorial
 Michael A. Goodrich

Arquitetura de Controle: Reativo

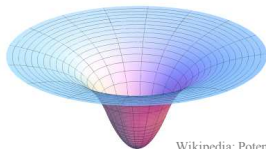
CONTROLE: Arquiteturas **REATIVAS***

- Reativo: Integração Sensorial-Motora
- Técnica: Campos Potenciais

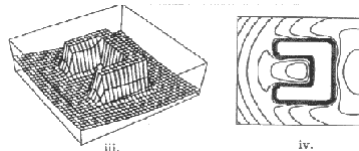
Evitar colisões e obstáculos
 Seguir em direção a um alvo
 Sensorial-Motor: Sentir => Agir
 * Reativo “não puro” baseado em informações externas



http://www.scf.usc.edu/~peiyingo/gra_planning.html



Wikipedia: Potential Filed



iii.

iv.

Houston, we have a problem!

Forças Combinadas: Força de Atração + Força de Repulsão

Arquitetura de Controle: Reativo

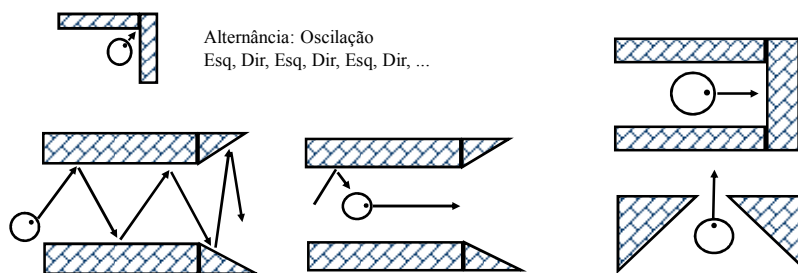
CONTROLE: Arquiteturas REATIVAS*

- Reativo: Integração Sensorial-Motora
- Técnicas:
 - Desvio reativo de Obstáculos
 - Campos Potenciais

Evitar colisões e obstáculos
Seguir em direção a um alvo
Sensorial-Motor: Sentir => Agir
* Reativo "não puro" baseado em
informações externas **LOCAIS**

• Problemas...

Houston, we have a problem!
>> Mínimos Locais
>> Passado, Memória e Contexto



17

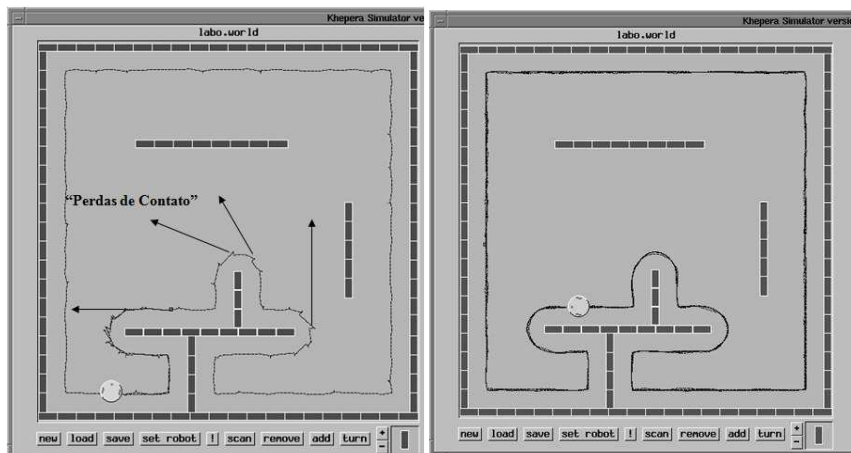
Maio 2010

Arquitetura de Controle: Reativo

CONTROLE: Arquiteturas REATIVAS*

- Reativo: Integração Sensorial-Motora
- Problemas...

Houston, we have a problem!
>> Mínimos Locais
>> Passado, Memória e Contexto



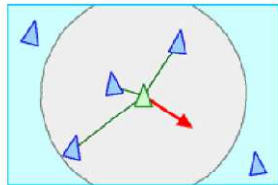
18

Maio 2010

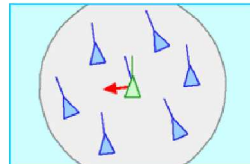
Arquitetura de Controle: Reativo

CONTROLE: Arquiteturas REATIVAS

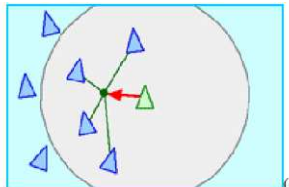
- Reativo: Múltiplos Robôs => Boids / Steering [Craig Reynolds]



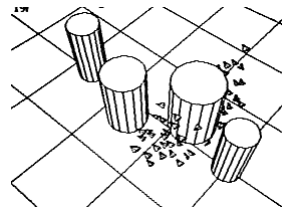
(a) **Separation:** Direcionar o movimento dos agentes para evitar um aglomeramento local dos membros do grupo.



(b) **Alignment:** Direcionar o movimento de modo a seguir o alinhamento médio (*heading*) dos membros do grupo.



(c) **Cohesion:** Direcionar o movimento de modo a ir em direção a posição central média dos membros do grupo



19

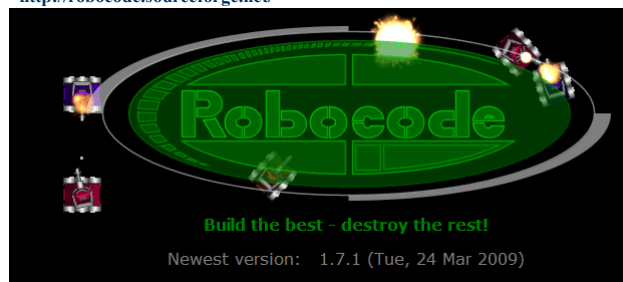
Maio 2010

Arquitetura de Controle: Reativo

CONTROLE: Arquiteturas REATIVAS*

- Reativo: *Em Direção ao Deliberativo...* **ROBO CODE**

<http://robocode.sourceforge.net/>



```
/**
 * Fire when we see a robot
 */
public void onScannedRobot(ScannedRobotEvent e)
{
    fire(1);
}
```

```
/**
 * We were hit! Turn perpendicular to the bullet,
 * so our seesaw might avoid a future shot.
 */
public void onHitByBullet(HitByBulletEvent e)
{
    turnLeft(90 - e.getBearing());
}
```

20

Maio 2010

Arquitetura de Controle Deliberativo

Controle Deliberativo

Controle: Percepção => [Conhecimento e Decisão] => Ação

Deliberativo: Percepção => Planeja, Delibera => Ação

- Possui *conhecimento* sobre a situação do robô e do ambiente;
- Usualmente baseado no uso de mapas e planejamento de trajetórias.

Tarefas típicas: *Behaviour : Task Planning, Action Sequence*

- Execução de scripts de ações planejadas previamente;
- Executar uma seqüência de ações previamente determinada;
- Seguir trajetórias especificadas com uso de mapas;
- Execução de Tarefas de Alto Nível;

21

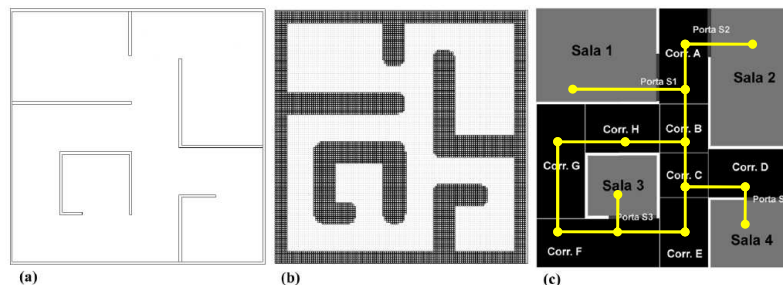
Maio 2010

Arquitetura de Controle Deliberativo

Controle Deliberativo

MAPAS DO AMBIENTE: Métrico / Topológico

- Mapa Métrico do tipo GRADE (Grid, Mapa de Ocupação)
- Mapa Métrico do tipo GEOMÉTRICO (Geometria do Ambiente)



22

Maio 2010

Farlei Heinen, 2002

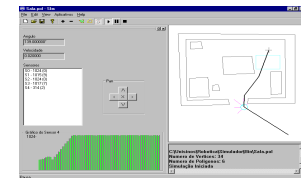
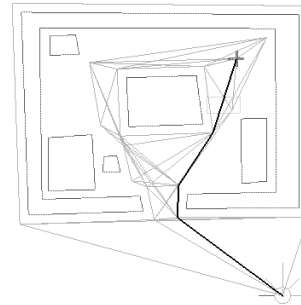
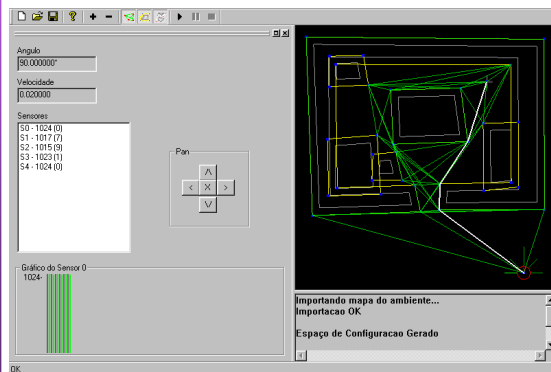
Geométrico (a); Baseado em Grade (b); Topológico/semântico (c);

SIMROB2D

Referência:

Farlei Heinen (Orientador: Fernando Osório)
Robótica Autônoma: A integração entre planificação e comportamento reativo, 2000.

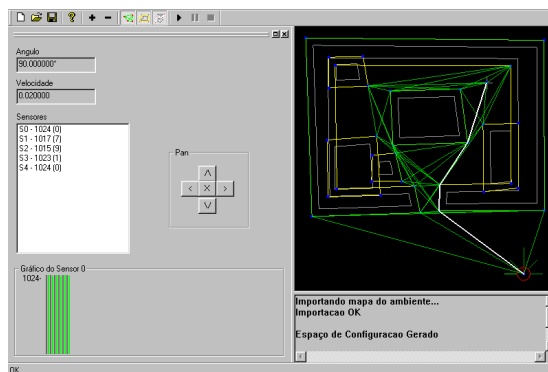
Sensores: 5 IR / Atuadores: 2 motores (diferencial)



23

Maio 2010

SIMROB2D



Farlei Heinen, 2000

Planejando a Trajetória do Robo com o uso do Mapa

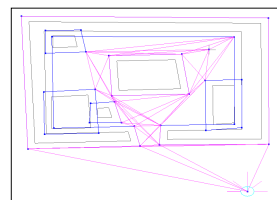
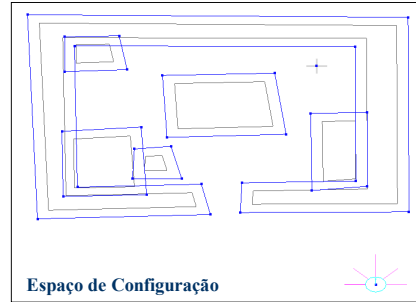
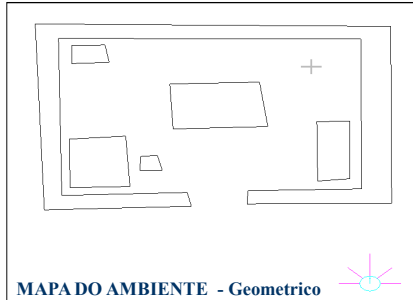
1. Determinar a área onde o robô pode navegar: Espaço de Configuração
2. Gerar o conjunto de rotas possíveis: Grafo de Visibilidade
3. Obter a melhor trajetória dentro das rotas possíveis: Algoritmo de Dijkstra

24

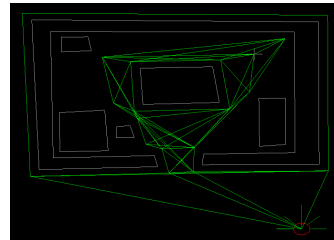
Maio 2010

Arquitetura de Controle: **Deliberativo**

SIMROB2D



Grafo de Visibilidade

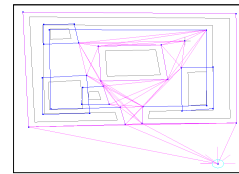
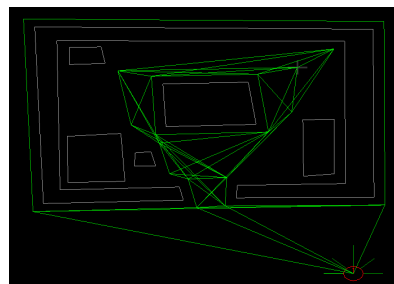


25

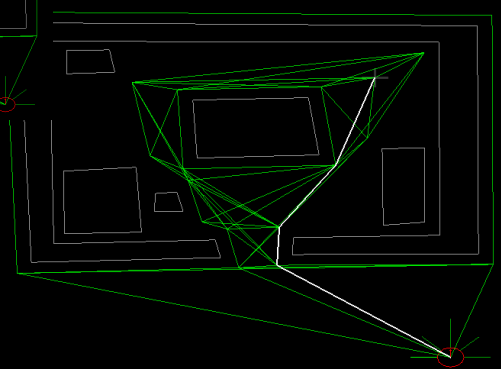
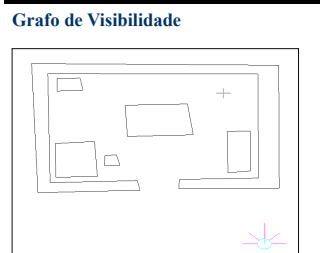
Maio 2010

Arquitetura de Controle: **Deliberativo**

SIMROB2D



Caminho Ótimo no Grafo: Dijkstra



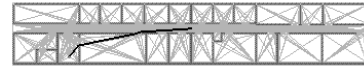
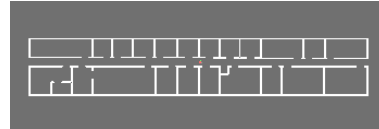
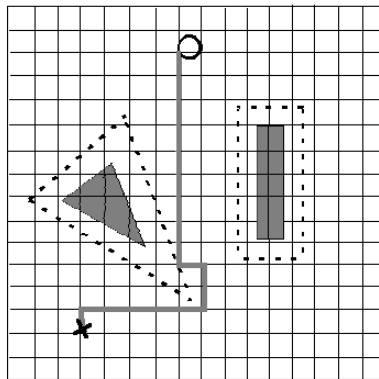
26

Maio 2010

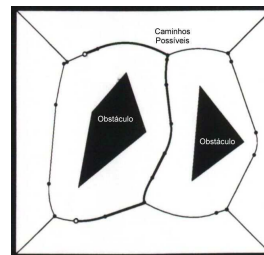
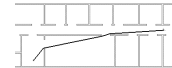
Arquitetura de Controle: **Deliberativo**

PLANEJAMENTO DE TRAJETÓRIAS Baseado em Mapas – Múltiplos Algoritmos

Navegação baseada em
Mapa de Ocupação:
Planejamento A*



Navegação baseada em
Mapa Geométrico:
Grafo+Dijkstra



Navegação baseada em
Mapa Geométrico:
Diagramas de Voronoi

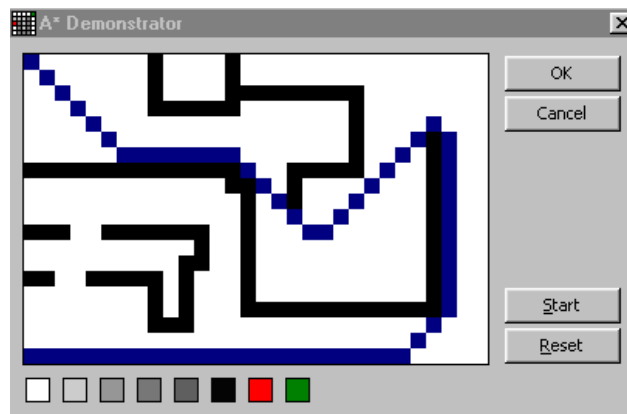
27
Maio 2010

Figura 4.3 Navegação baseada em *Grid*

Farlei Heinen, 2002

Controle Robótico Inteligente: **Deliberativo**

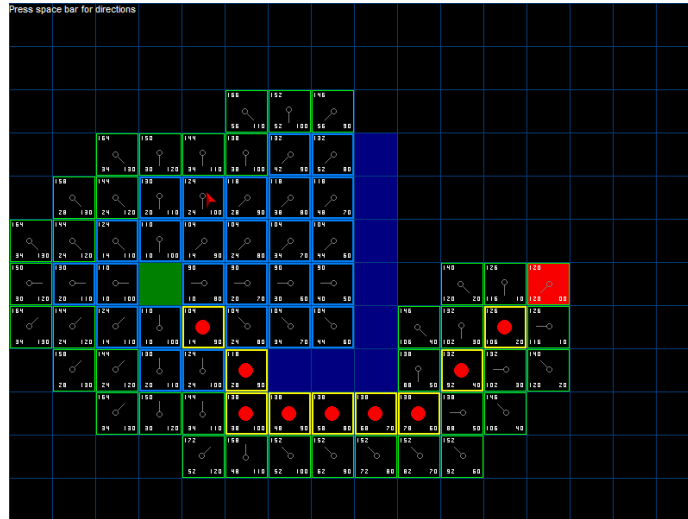
A* Search: [Tutorial *SBGames07*] Busca Heurística de Caminhos



28
Maio 2010

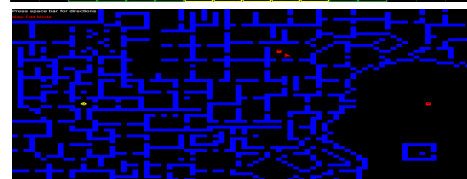
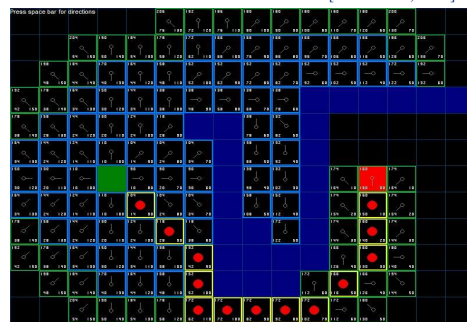
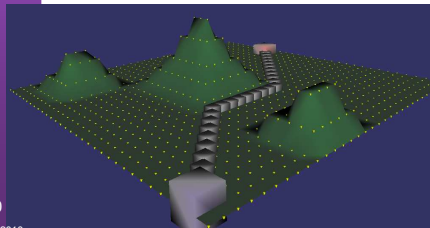
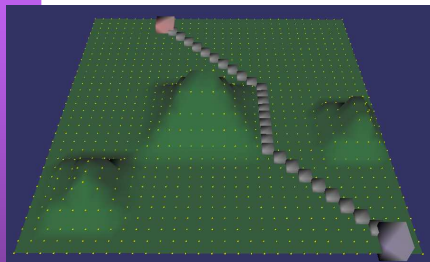
[Matthews 2003]

A* - Planejamento de Trajetórias [Tutorial SBGames 2007]



A* Search: [Tutorial SBGames07] Busca Heurística de Caminhos

[Lester 2004, 2007]

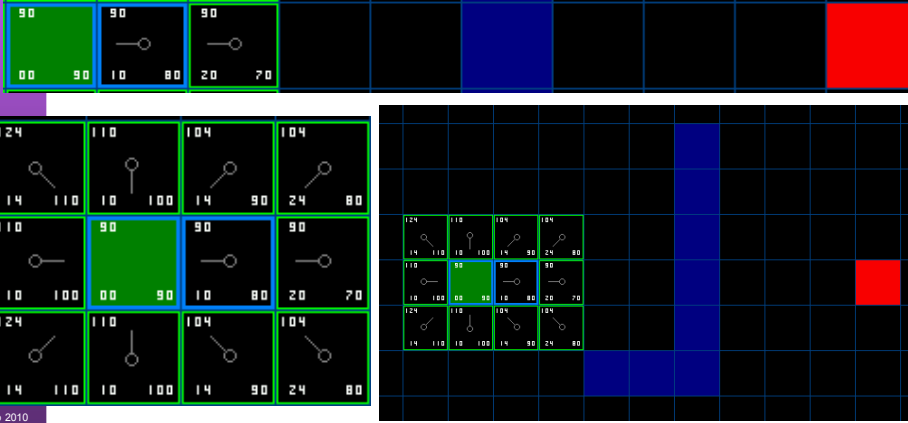
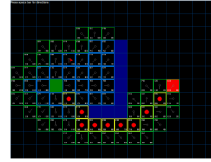


A* Search: [Tutorial SBGames07]

Busca Heurística de Caminhos

Custo de um Caminho

$$F(x) = G(x) + H(x) \quad \dots \quad G = \text{Percorrido}, H = \text{Estimado}$$



Maio 2010

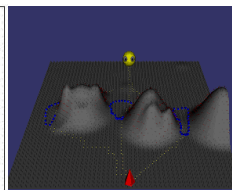
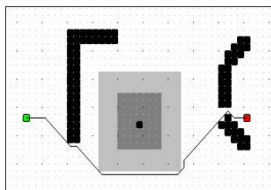
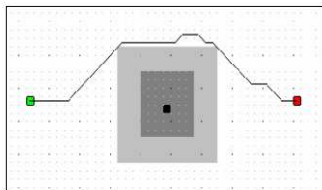
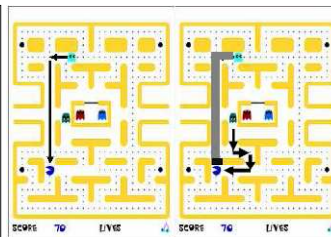
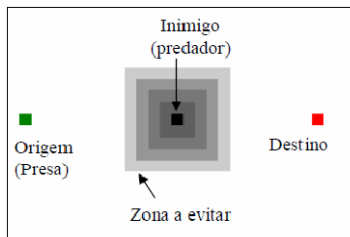
A* Search: [Tutorial SBGames07]

Busca Heurística de Caminhos

MAPA: Custo do Terreno a ser Atravessado

Custo de um Caminho

$$F(x) = G(x) + H(x) \quad \dots \quad G = \text{Percorrido}, H = \text{Estimado}$$



32

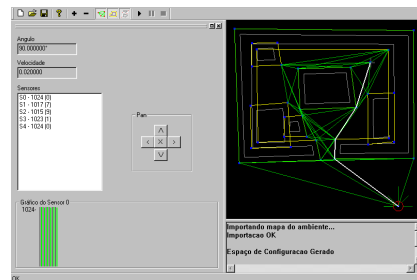
Maio 2010

Arquitetura de Controle: **Deliberativo**

CONTROLE: Arquiteturas **Deliberativo**

- **Deliberativo: Planejamento**
- **Técnicas:**
 - Grafo de Visibilidade
 - Voronoi
 - A* ...

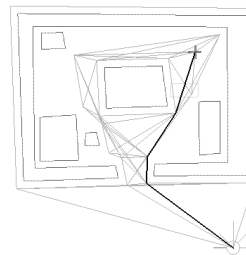
• Problemas...



Seguir em direção a um alvo
Planejamento de Trajetória com Mapa
* **Deliberativo "puro"** baseado em
informações externas **GLOBAIS**

Houston, we have a problem!

- >> Mapas com informações imprecisas
- >> Obstáculos Dinâmicos (não mapeados)
- >> Erros de localização do robô



Localização => *Tem que ser precisa!*
Mapa do Ambiente => *Tem que ser exato!*

Arquitetura de Controle

CONTROLE: Arquiteturas **Reativas**

CONTROLE: Arquiteturas **Deliberativas**



**Ambas possuem
problemas e limitações!**

Solução?

**Buscar aproveitar o que de melhor tem cada
uma das duas abordagens...**

SISTEMAS HÍBRIDOS!

Referências citadas nesta aula...

Craig Reynolds – Boids, OpenSteer, Flocks (múltiplos agentes autônomos)
<http://www.red3d.com/cwr/boids/>

Robocode – Batalha de Robôs Simulados
<http://robocode.sourceforge.net/>

Simulador do Khepera
<http://diwww.epfl.ch/lami/team/michel/khep-sim/>

Simulador com Planejamento de Trajetória (SimRob 2D)
<http://ncc.unisinos.br/robotica/robotica.html>

Referências do Algoritmo A* (A Star)

A* Demo1: http://www.policvalmanac.org/games/aStarTutorial_port.htm

A* Demo2: <http://www.gamedev.net/REFERENCE/ARTICLES/ARTICLE2003.ASP>

A* Múltiplo: <http://www.inf.unisinos.br/~sbgames/anais/tutoriais/Tutorial3.pdf>

Agenda: Parte II

Arquiteturas de Controle de Robôs Autônomos

1. Controle Reativo e Deliberativo [Puro]
2. Controle Hierárquico
Arquitetura Subsumption (Brooks), SMPA
3. Controle Híbrido e Localização
Arquitetura COHBRA
Arquitetura SLAM
4. Exemplos de Sistemas com Arquitetura Híbrida

Arquitetura de Controle Reativo Puro

Controle Reativo Puro

Controle: Percepção [=> ~~Decisão~~ =>] => Ação

Reativo: Percepção => Ação

- Reage diretamente aos estímulos externos;
- Esquema sensorio-motor;

Comportamentos e Tarefas típicas: *Reactive Behaviour*

- Vagar pelo ambiente, evitando colisões e obstáculos;
- Acompanhar uma parede ou corredor;
- Comportamento direcionado pela luz;
- Ir em direção a uma determinada orientação

Composição de Comportamentos: Direção x Obstáculo



37

Março 2010

Arquitetura de Controle Deliberativo

Controle Deliberativo

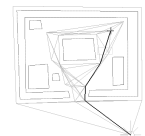
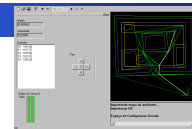
Controle: Percepção => [Conhecimento e Decisão] => Ação

Deliberativo: Percepção => Planeja, Delibera => Ação

- Possui *conhecimento* sobre a situação do robô e do ambiente;
- Usualmente baseado no uso de mapas e planejamento de trajetórias.

Tarefas típicas: *Behaviour : Task Planning, Action Sequence*

- Execução de scripts de ações planejadas previamente;
- Executar uma seqüência de ações previamente determinada;
- Seguir trajetórias especificadas com uso de mapas;
- Execução de Tarefas de Alto Nível;



38

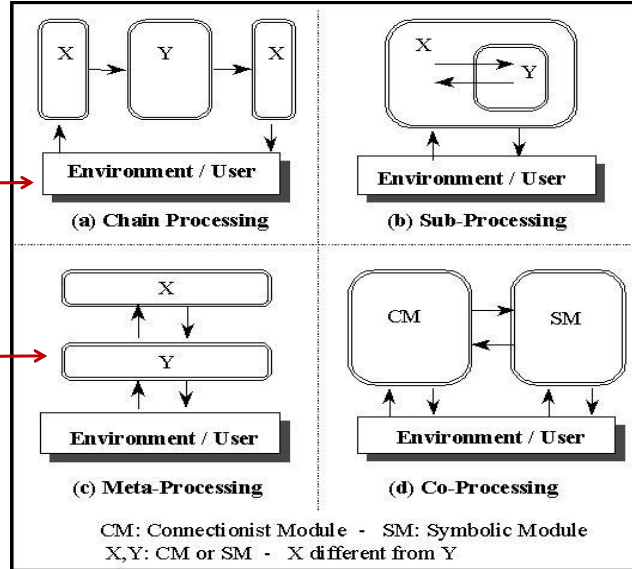
Março 2010

Arquiteturas de Controle

Hybrid Systems

Camadas Horizontal

Camadas Vertical

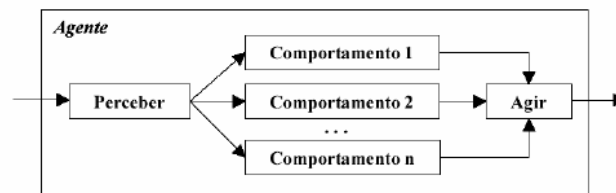


41

Maio 2010

Arquiteturas de Controle Híbridas

Controle Hierárquico



Hierarquia de Comportamentos:

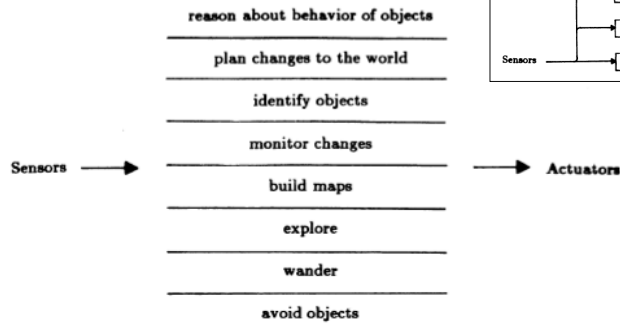
- Evitar/Desviar de obstáculos
- Seguir em uma determinada direção
- Seguir uma determinada rota

42

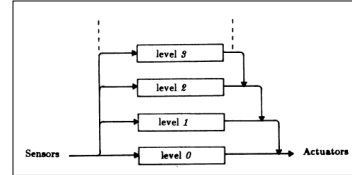
Abril 2010

Arquiteturas de Controle Híbridas

Controle Hierárquico



Brooks - Arquitetura Subsumption



43
 Abril 2010

Figure 2. A decomposition of a mobile robot control system based on task achieving behaviors.

Figures From:
 Brooks, R. A.
 MIT A.I. Memo 864
 Sept. 1985

Arquiteturas de Controle Híbridas

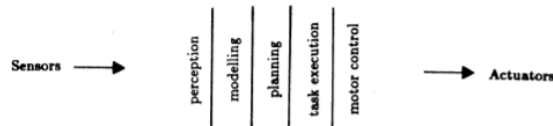
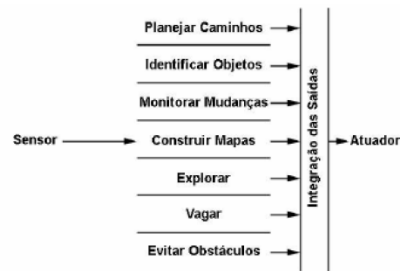
Controle Hierárquico

Arquiteturas Hierárquicas

Brooks-Subsumption

SMPA

“Sense” - “Model” - “Plan” - “Act”



44
 Abril 2010

Figure 1. A traditional decomposition of a mobile robot control system into functional modules.

Arquiteturas de Controle Híbridas

Controle Hierárquico

Arquiteturas Hierárquicas

Brooks-Subsumption

SMPA

“Sense” - “Model” - “Plan” - “Act”

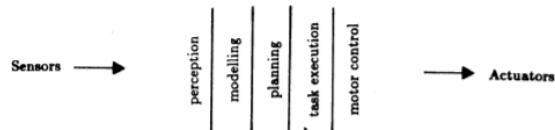
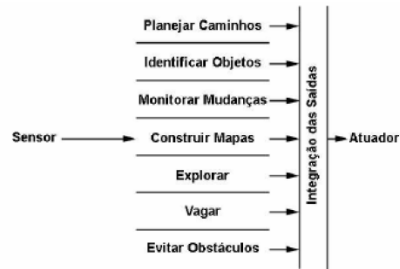
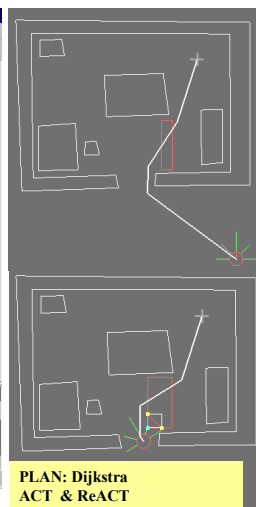


Figure 1. A traditional decomposition of a mobile robot control system into functional modules.

Arquiteturas de Controle Híbridas

Controle Hierárquico/Híbrido: SMPA

Controle Híbrido



Controle Hierárquico/Híbrido: SMPA

Sense - **MODEL** - Plan - Act

Environment Map:
 Occupancy Grid

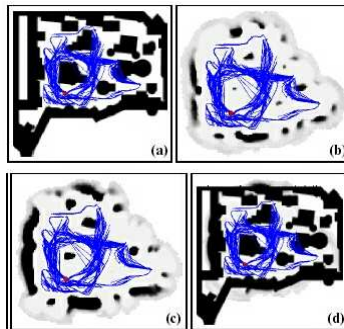


Fig. 9. Integrating multiple maps: (a) CAD map of the museum ($21 \times 20m^2$) modeling only the static obstacles, (b) laser map, (c) sonar map, and (d) the integrated map used for path planning.

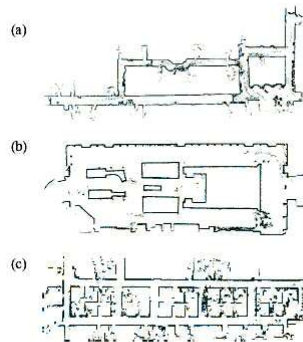
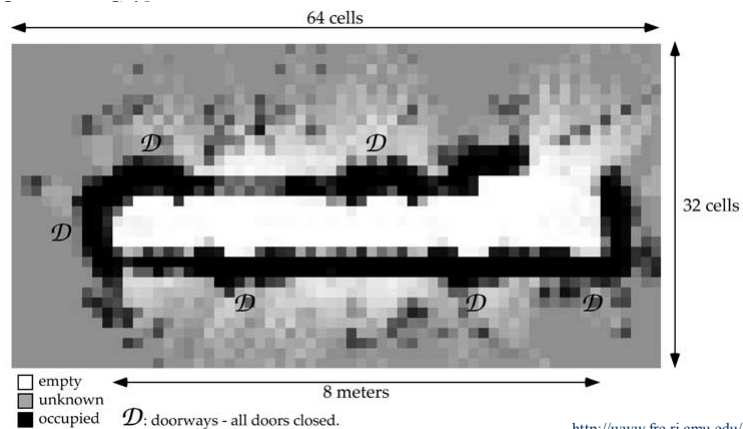


Figure 5: Maps generated in other large-scale environments of sizes (a) 75m, (b) 45m, and (c) 50m. In some of these runs, the cumulative odometric error exceeds 30 meters and 90 degrees.

Controle Hierárquico/Híbrido: SMPA

Sense - **MODEL** - Plan - Act

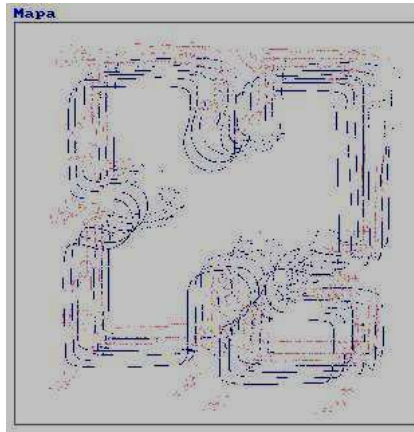
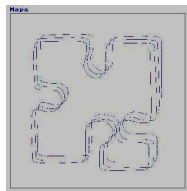
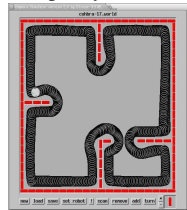
Environment Map:



Controle Hierárquico/Híbrido: SMPA

Sense – **MODEL** – Plan - Act

Environment Map:
 Occupancy Grid



Controle Hierárquico/Híbrido:

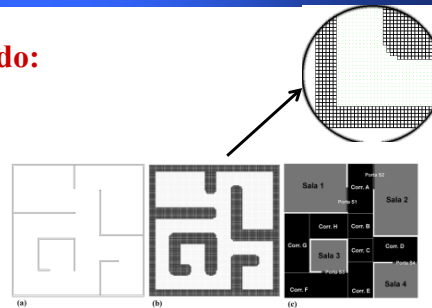
MAPAS:

- Construção - Exemplo: SMPA
- Utilização - Exemplo: A*

PROBLEMA:

➤ **LOCALIZAÇÃO!**

- Para poder construir um mapa precisamos controlar a **posição** e o **deslocamento** do robô;
- Para poder navegar usando um mapa precisamos saber **onde está** o robô.



Where am I? [Borenstein 1996]
<http://www-personal.umich.edu/~johannb/position.htm>

Controle Hierárquico/Híbrido:

> POSE

Posição (X, Y [Z])

Orientação Θ ou $\Theta_x, \Theta_y, \Theta_z$

POSE 2D = X, Y, Θ

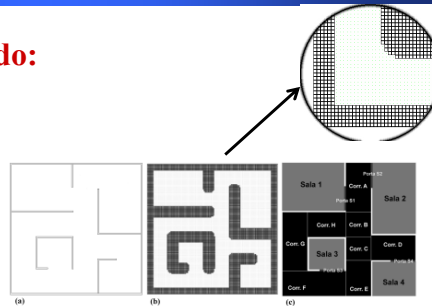
POSE 3D = X, Y, Z, $\Theta_x, \Theta_y, \Theta_z$

> LOCALIZAÇÃO!

- Para poder construir um mapa precisamos controlar a posição e o deslocamento do robô;
- Para poder navegar usando um mapa precisamos saber onde está o robô.

Where am I? [Borenstein 1996]

<http://www-personal.umich.edu/~johannb/position.htm>



Controle Hierárquico/Híbrido:

>> Estimativa da Posição <<

Posição Inicial + Deslocamentos => Posição Final

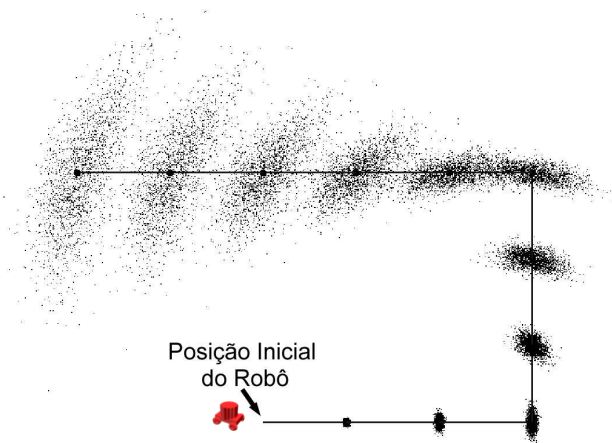
Integração dos deslocamentos no tempo: Odometria

PROBLEMAS:

- Como determinar a posição inicial?
- Como medir precisamente os deslocamentos:
 - * Avanço em linha reta
 - * Giro para os lados

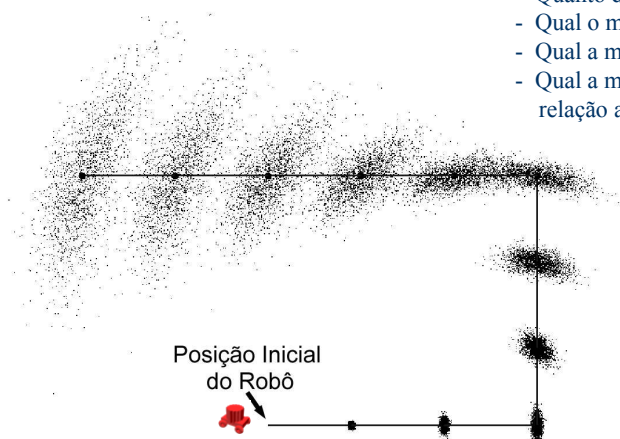
Controle Hierárquico/Híbrido:

Erro de Odometria...



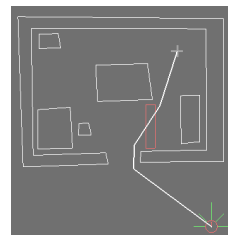
Localização do Robô:

Erro de Odometria...



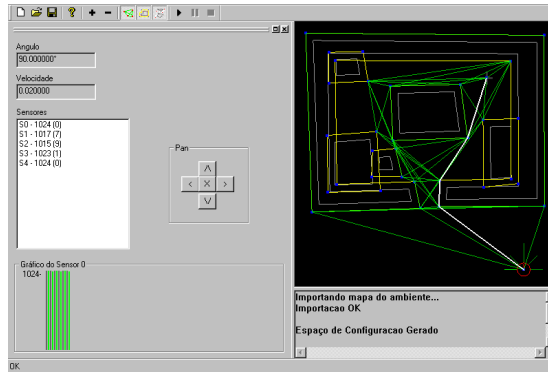
Questões:

- Onde estou?
- Em que direção devo ir?
- Quanto devo avançar?
- Qual o meu destino?
- Qual a minha posição no mapa?
- Qual a minha posição em relação aos obstáculos do mapa?



Arquiteturas de Controle Híbridas

PROBLEMA: Erro de Localização => IMPACTO no Algoritmo de Navegação e Detecção de Obstáculos



O que ocorre se a posição estimada estiver incorreta ?

Farlei Heinen, 2000

Planejando a Trajetória do Robo com o uso do Mapa

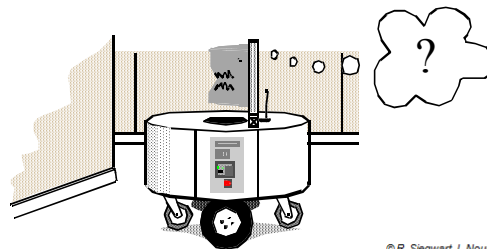
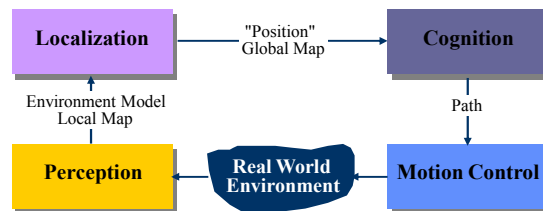
1. Determinar a área onde o robô pode navegar: Espaço de Configuração
2. Gerar o conjunto de rotas possíveis: Grafo de Visibilidade
3. Obter a melhor trajetória dentro das rotas possíveis: Algoritmo de Dijkstra

57

Maio 2010

Localização e Uso de Mapas

Controle de Robôs Móveis Autônomos



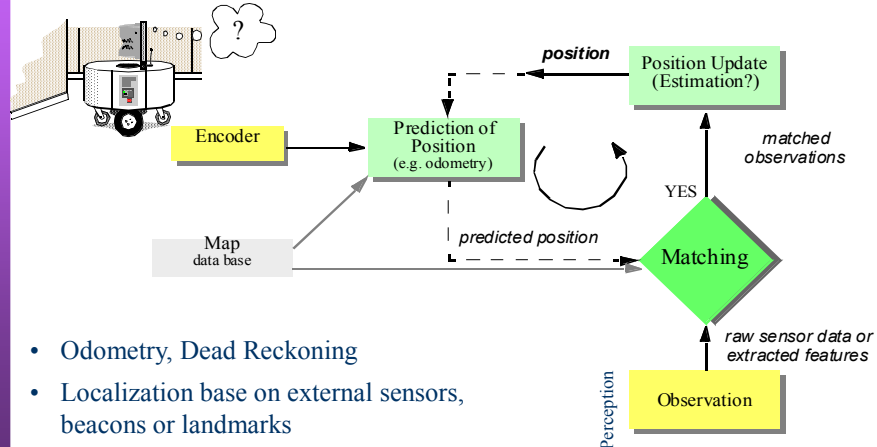
58

Maio 2010

© R. Siegwart, I. Nourbakhsh

Localização e Uso de Mapas

Controle de Robôs Móveis Autônomos



- Odometry, Dead Reckoning
- Localization based on external sensors, beacons or landmarks
- Probabilistic Map Based Localization

59

Maio 2010

© R. Siegwart, I. Nourbakhsh

Arquiteturas de Controle Híbridas

PROPOSTA DE ARQUITETURA: COHBRA

Arquitetura Híbrida

- Módulo de Localização (MCL – Monte Carlo Localization)
- Representação do Ambiente através de Mapas (Ocupação/Grid, Geométrico e Topológico)
- Módulos com Comportamentos: Reativo + Deliberativo
Reativo: Desvio de Obstáculos (campos potenciais)
Deliberativo: Planejamento de Trajetórias (A*)
- Integração de Múltiplos Comportamentos:
Hierarquia Vertical

Arquitetura Híbrida por Co-Processamento: Localização x Navegação

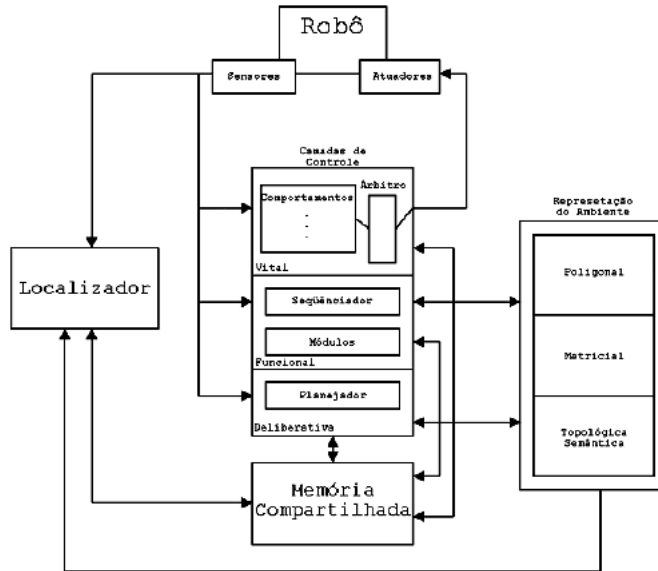
Arquitetura Híbrida em Camadas Hierárquicas: Módulos de Comportamentos

60

Março 2010

Arquiteturas de Controle Híbridas

Controle Híbrido Robusto - COHBRA / SimRob3D



Arquiteturas de Controle Híbridas

Controle Híbrido Robusto - COHBRA / SimRob3D

Mapa: Geométrico, Grade, Topológico
 Auto-Localização
 Planejamento de Trajetória
 Desvio de Obstáculos

Controle Híbrido
 Reativo / Deliberativo

SLAM

Controle Híbrido: Arquitetura SLAM

SLAM = Simultaneous localization and mapping

Refs:

SLAM

http://en.wikipedia.org/wiki/Simultaneous_localization_and_mapping

OpenSLAM

<http://openslam.org/>

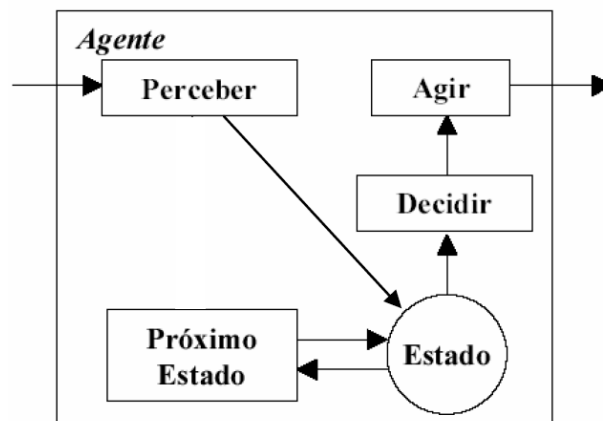
MATLAB SLAM

<http://eia.udg.es/~qsalvi/Slam.zip>

Book: Probabilistic Robotics - The MIT Press (2005)
by Sebastian Thrun, Wolfram Burgard, Dieter Fox

Arquiteturas Híbridas

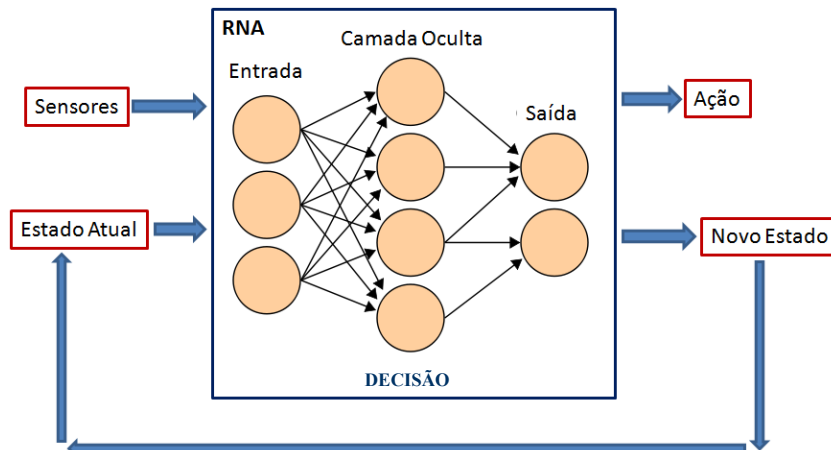
Controle Híbrido: Arquitetura com Estados Internos (Autômato)



Arquiteturas de Controle Híbridas

Arquiteturas Híbridas

Controle Híbrido: Arquitetura com Estados Internos (Autômato + Rede Neural)



65

Abril 2010

Arquiteturas de Controle Híbridas

SEVA 3D - Sistema de Estacionamento de Veículos

Virtual Environment: 3D Realistic Environment

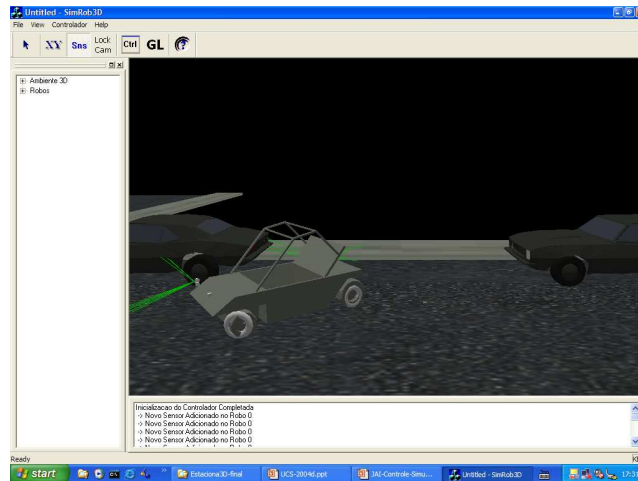


SEVA 3D
Simulation
Tool

66

Abril 2010

SEVA 3D – Sistema de Estacionamento de Veículos



*SEVA 3D
Simulation
Tool*

67

Abril 2010

Arquiteturas de AGENTES AUTÔNOMOS

AGENTES AUTÔNOMOS:

- Reativos
- Deliberativos
- Hierárquicos
- Híbridos

- Agentes Cognitivos
- BDI = Belief / Desire / Intention
- MAS Architectures (Multi-Agent)
- Agentes Adaptativos (Machine Learning)
- Agentes Evolutivos (Genetic Algorithms)

Conceitos de Agentes:

- Corpo (Agente é um ente físico)
- Percepção (Múltiplos “canais” de percepção)
- Cognição: Raciocínio, Aprendizagem, Adaptação,
- Ação (Múltiplas “formas” de interação com o mundo)
- Comunicação - Interação

68

Abril 2010



INFORMAÇÕES SOBRE A DISCIPLINA

USP - Universidade de São Paulo - São Carlos, SP
ICMC - Instituto de Ciências Matemáticas e de Computação
SSC - Depto. de Sistemas de Computação / Grupo SEER
LRM - Lab. de Robótica Móvel: <http://www.lrm.icmc.usp.br/>

Prof. Fernando Santos OSÓRIO
Web institucional: <Http://www.icmc.usp.br/ssc/>
Página pessoal: <Http://www.icmc.usp.br/~fosorio/>
E-mail: [fosorio \[at\] icmc. usp. br](mailto:fosorio@icmc.usp.br) ou [fosorio \[at\] gmail. com](mailto:fosorio@gmail.com)

Disciplina de Robôs Móveis Autônomos (PG-CCMC)
Web Disciplinas: <Http://www.icmc.usp.br/~fosorio/>
> Programa, Material de Aulas, Critérios de Avaliação,
> Material de Apoio, Trabalhos Práticos