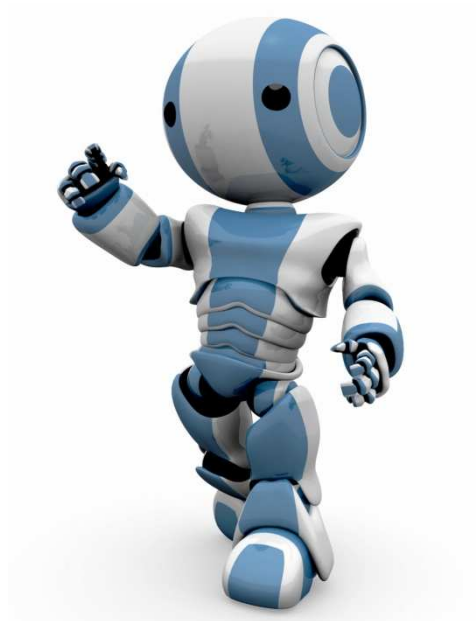


# **SISTEMAS ROBÓTICOS AUTÔNOMOS**

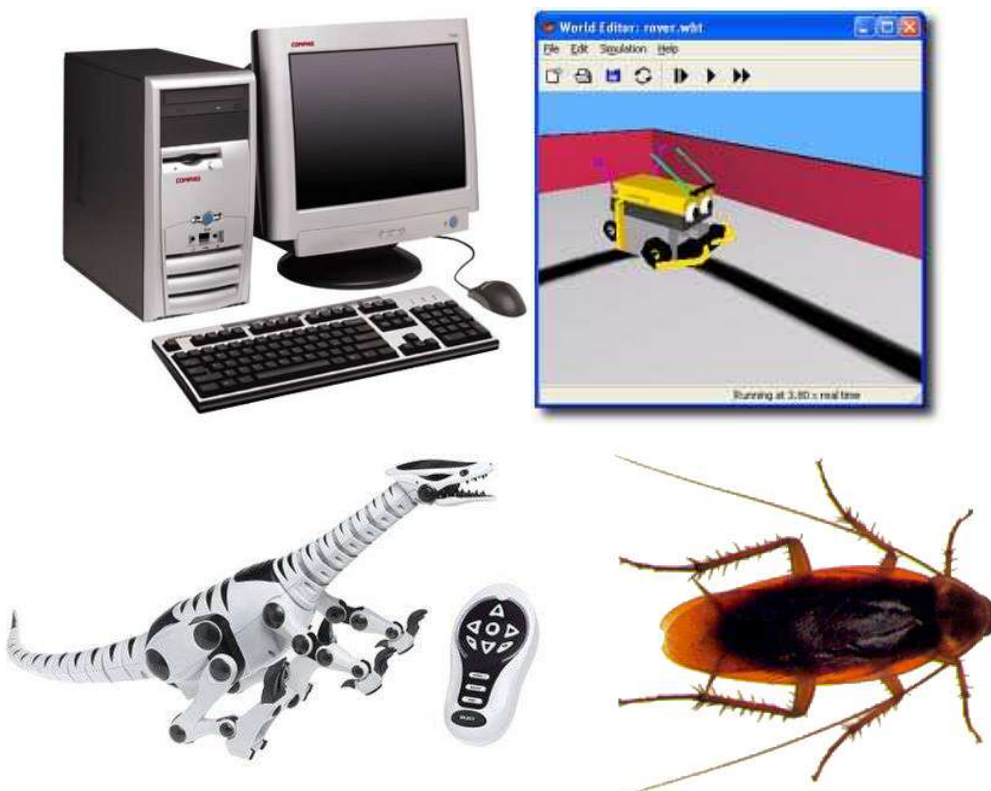
Pablo Javier Alsina



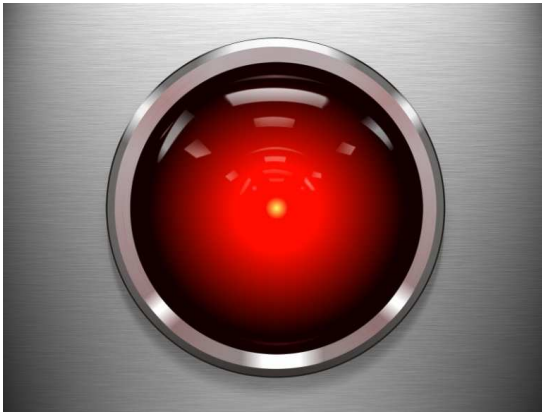
## O que é um robô autônomo?



## O que não é um robô autônomo?



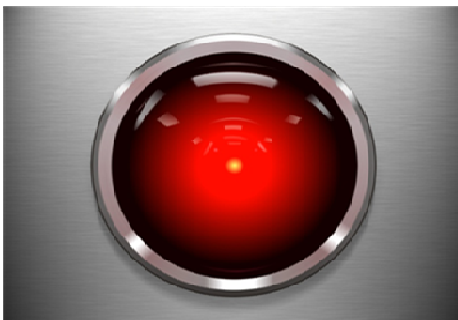
## O que não é um robô autônomo?



## O que é um robô autônomo?

- Um robô autônomo é uma máquina programável de propósito geral, que existe no mundo físico, percebe o mundo através de sensores, processa a informação sensorial de acordo com um modelo do mundo e atua no mundo através de movimentos.
- Um robô autônomo estabelece uma conexão “inteligente” entre percepção e ação.
- Quanto menos um operador humano interfere nas suas ações, mais autônomo é o robô.

## O que é um robô autônomo?



+



**O que é um robô autônomo?**

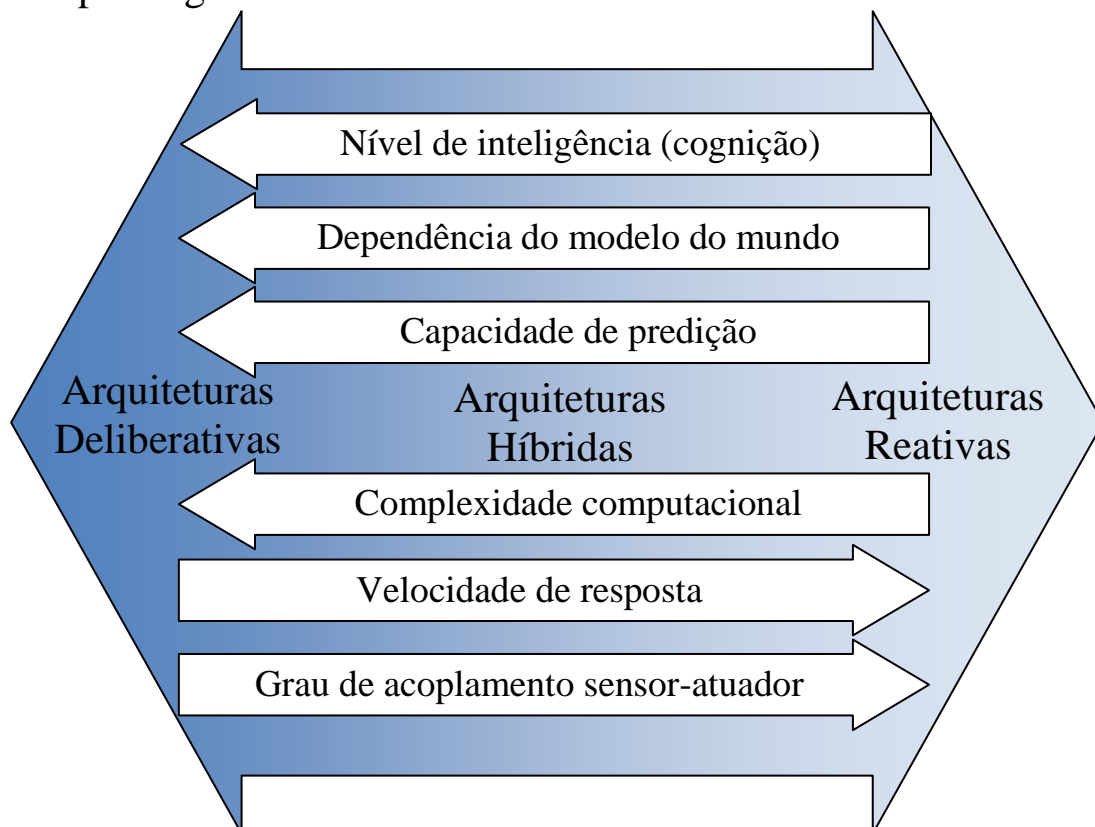


## **PARADIGMAS DE ARQUITETURAS DE CONTROLE**

De acordo com Arkin [Ronald C. Arkin. Behavior-Based Robotics (Intelligent Robotics and Autonomous Agents), volume I. MIT Press, 1998.], as arquiteturas de controle de robôs podem ser classificadas em três grandes paradigmas:

- Paradigma Deliberativo.
- Paradigma Reativo.
- Paradigma Híbrido Deliberativo/Reativo

De fato, as diversas arquiteturas de controle encontradas na literatura podem ser classificadas dentro de uma gama, que vai de arquiteturas totalmente deliberativas a arquiteturas puramente reativas. As arquiteturas híbridas buscam explorar as vantagens dos paradigmas reativos e deliberativos.

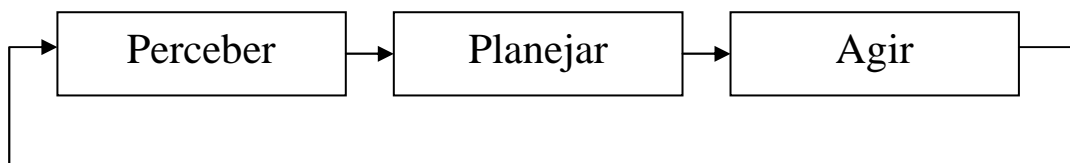


<b>Arquiteturas Deliberativas</b>	<b>Arquiteturas Reativas</b>
Dependem de uma representação simbólica interna do mundo	Independem de uma representação do mundo. O mundo é o seu próprio modelo
Processos de decisão puramente simbólicos	Processos de decisão puramente reflexos
Tempos de resposta lentos	Resposta em tempo real
Custo computacional alto (planejar é custoso)	Baixo custo computacional

- **Primitivas Robóticas:**

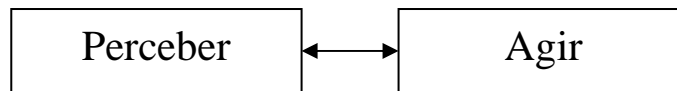
<b>Primitiva</b>	<b>Entrada</b>	<b>Saída</b>
<b>Percepção</b>	Dados sensoriais	Informação percebida
<b>Planejamento</b>	Informação percebida	Diretivas
<b>Ação</b>	Informação percebida ou Diretivas	Comandos para atuadores

- **Paradigma Deliberativo:**



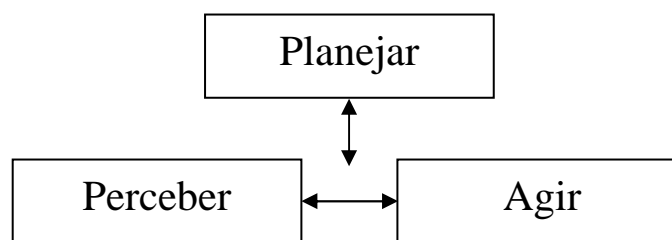
<b>Primitiva</b>	<b>Entrada</b>	<b>Saída</b>
<b>Percepção</b>	Dados sensoriais	Informação percebida
<b>Planejamento</b>	Informação percebida	Diretivas
<b>Ação</b>	Informação percebida Diretivas	Comandos para atuadores

- **Paradigma Reativo:**



Primitiva	Entrada	Saída
Percepção	Dados sensoriais	Informação percebida
Planejamento	Informação percebida	Diretivas
Ação	Informação percebida Diretivas	Comandos para atuadores

- **Paradigma Híbrido:**



Primitiva	Entrada	Saída
Percepção	Dados sensoriais	Informação percebida
Planejamento	Informação percebida	Diretivas
Ação	Informação percebida ou Diretivas	Comandos para atuadores

Primitiva	Entrada	Saída
Planejamento	Informação percebida	Diretivas
Percepção-Ação (Comportamentos)	Dados sensoriais	Comandos para atuadores

## **NAVEGAÇÃO**

**Níveis hierárquicos típicos de um Sistema de Navegação de Robô.**

PERCEPÇÃO
DELIBERAÇÃO E TOMADA DE DECISÃO
PLANEJAMENTO DE CAMINHO
ADEQUAÇÃO DE CAMINHO E GERAÇÃO DE TRAJETÓRIA
SISTEMA DE CONTROLE
ATUAÇÃO

- **Navegação em um ambiente conhecido:**

- Dispõe-se previamente de uma representação do ambiente (por exemplo, um mapa).
- Localização: onde estou?
- Busca: aonde vou? O problema de cobertura.
- Planejamento de caminhos, geração de trajetória: como eu chego lá?
- Execução de plano: seguir o caminho planejado. Como eu vou daqui a acolá?
- Movimentação: como vou daqui para ali?

- **Navegação em um ambiente desconhecido:**

- Não se dispõe previamente de uma representação do ambiente.
- Localização: onde estou?
- Mapeamento: o que há por aqui? Como representar?
- Exploração: como construir um mapa garantindo cobertura.
- Localização e Mapeamento Simultâneos (SLAM): o que veio primeiro, o ovo ou a galinha?



## **MAPEAMENTO**

- **Mapa**: representação do ambiente no qual o robô atua.
- **Mapa Métrico**: representa as dimensões físicas dos objetos presentes no ambiente
- **Mapa Topológico**: representa as relações de conectividade das regiões navegáveis do ambiente.
- **Mapa Híbrido Métrico/Topológico**: incorpora informações métricas e de conectividade do ambiente.
- **Mapeamento por Grade de Ocupação**: representação do ambiente como uma grade de configurações igualmente espaçadas, às quais se associa uma probabilidade de ocupação por obstáculo.
- O mapa é gerado a partir de dados de medição ruidosos e incertos proveniente dos sensores embarcados.
- Assume-se que a localização do robô é conhecida a cada instante.
- À medida que o robô se movimenta, novas informações sensoriais são incorporadas à grade, melhorando a sua precisão.

## PERCEPÇÃO

- Medições obtidas a partir de Sensores embarcados permitem obter informações do ambiente e do próprio estado do robô, para fins de mapeamento, localização, navegação e manipulação.
- Medições sensoriais incorporam incertezas e ruídos inerentes.
- **Sensores Proprioceptivos:** capturam informações relativas ao próprio robô.
- **Sensores Externoceptivos:** capturam informações relativas ao ambiente.

## LOCALIZAÇÃO

- A Localização do robô no ambiente pode ser feita a partir de medições sensoriais.
- **Localização Relativa (Dead Reckoning):**
  - Assume-se uma localização inicial conhecida.
  - Mede-se deslocamentos incrementais (*encoders*, unidades de medida inercial).
  - Os deslocamentos incrementais são integrados no tempo, determinando a localização em relação à localização inicial.
  - Erros de medição se acumulam.
- **Localização Absoluta:**
  - A cada instante mede-se a localização em relação a um referencial global.
  - Erros de medição não se acumulam.

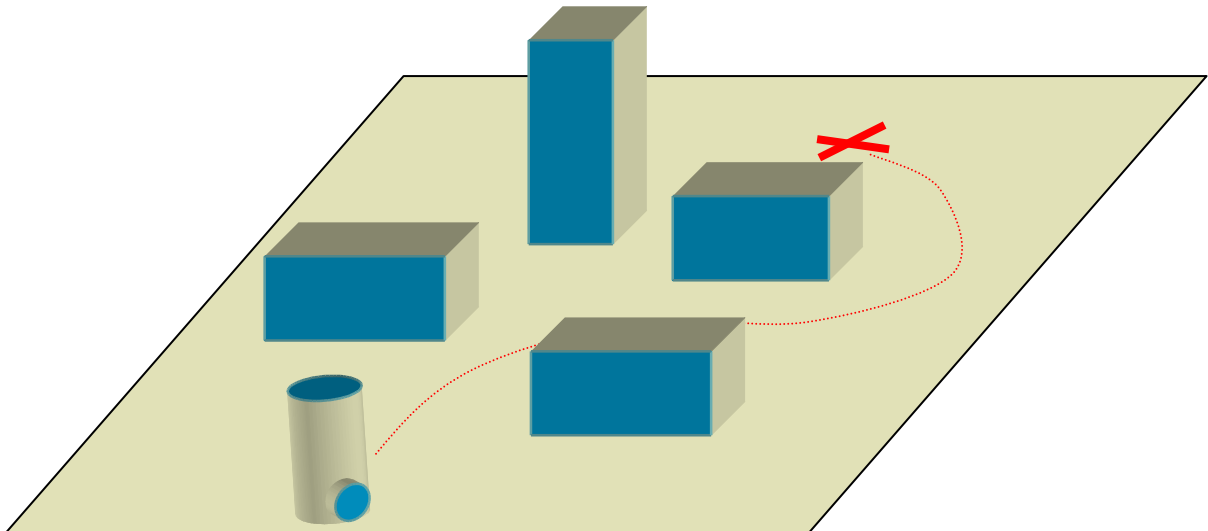
- **Localização baseada em Balizas Ativas:**
  - A localização absoluta de um conjunto de balizas é conhecida.
  - Mede-se a distância ou o azimuth das balizas no referencial do robô.
  - A localização do robô é obtida por trilateração ou triangulação.
  - Exemplo: GPS, Radio faróis.
- **Localização baseada em Marcos:**
  - A localização absoluta de um conjunto de Marcos é conhecida.
  - As características dos Marcos são conhecidas.
  - **Marcos artificiais:** marcos são projetados para otimizar o processo de localização.
  - **Marcos Naturais:** escolhidos no ambiente de modo a serem fáceis de detectar.
  - Geralmente, um único marco detectado nas vizinhanças do robô pode ser utilizado para determinar a localização absoluta.
- **Localização baseada em Mapas:**
  - O mapa é o marco.
  - Sensores de alcance são usados para determinar a localização dos objetos na vizinhança do robô.
  - Um mapa local é construído.
  - Busca-se a correspondência do mapa local com o mapa global disponível.

## **Localização e Mapeamento Simultâneos (SLAM)**

- O que veio primeiro, o ovo ou a galinha?
- Em ambiente desconhecido, o robô deve construir o mapa e se localizar em relação a ele, simultaneamente.
- Problema: se a localização é imprecisa, o mapa é impreciso.
- Solução: SLAM (*Self Localization and Map*) – Se suficientes novas informações sensoriais forem incorporadas quando o robô navega, superando as perdas de informação devido ao aumento das incertezas, tanto a localização do robô como o mapa vão se tornando mais precisos.

## PLANEJAMENTO DE CAMINHOS

- **O Problema do Carregador de Piano:**
- Como levar um piano no interior de um edifício, através de corredores povoados de obstáculos, até a sua localização final dentro do prédio?
  - Piano = corpo rígido móvel.
  - Obstáculos = corpos rígidos fixos.
  - Localização = posição e orientação = configuração.
- O Problema é formulado em **Espaço de Trabalho**.



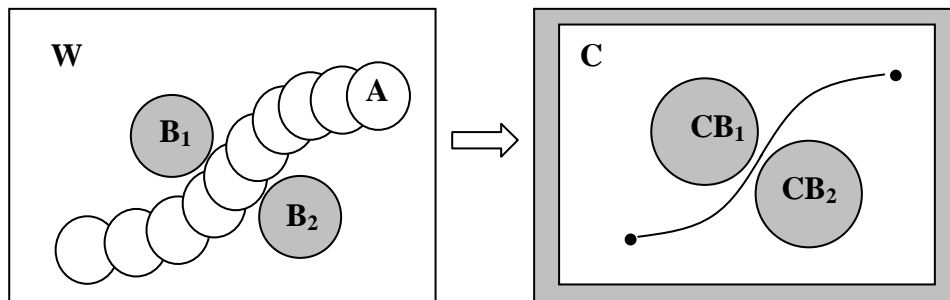
## ESPAÇO DE TRABALHO

- **Robô A:** corpo rígido que pode movimentar-se dentro de um Espaço de Trabalho.
- **Espaço de Trabalho W:** é o espaço físico no qual o robô se movimenta.
- **Obstáculo no Espaço de Trabalho B:** região conexa de **W** na qual é impossível posicionar qualquer ponto do Robô.

- Problema: Planejamento em Espaço de Trabalho requer teste de colisão dos infinitos pontos que compõem o robô com os infinitos pontos que compõem os obstáculos.

### **Planejamento de Caminhos - Solução:**

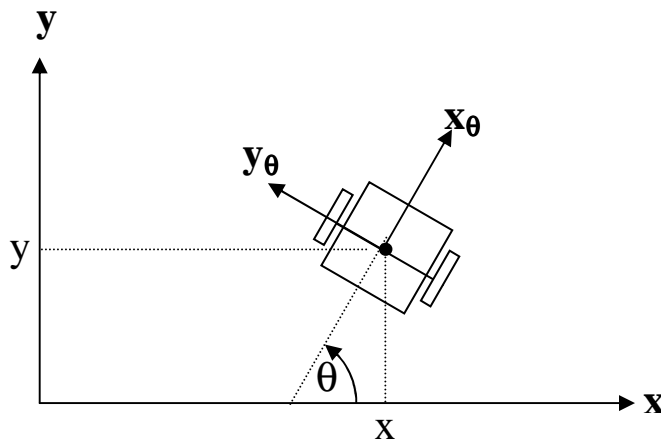
- Movimento de um robô A no Espaço de Trabalho **W** povoado de obstáculos  $B_i$ 's.  $\Rightarrow$
- Movimento de um ponto no Espaço de Configuração **C** povoado de C-obstáculos  $CB_i$ 's.



## ESPAÇO DE CONFIGURAÇÃO

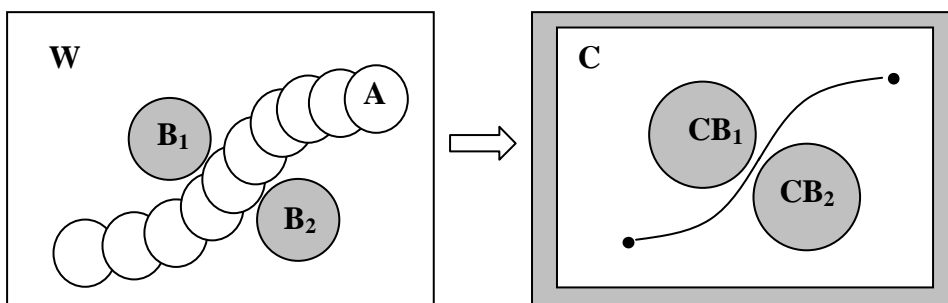
- Configuração  $q$ :

- Especificação da Posição e Orientação do robô.
- Exemplo:  $q = [x \ y \ \theta]^T$ .



- Espaço de Configuração  $C$ : é o conjunto de todas as possíveis configurações do robô.
- Espaço de Configuração Livre  $C_L$ : é o conjunto de todas as possíveis configurações em que o robô não colide com os obstáculos  $B_i$ 's.
- Obstáculo em Espaço de Configuração  $CB$ :
  - Um obstáculo  $B$  no espaço de trabalho  $W$  pode ser representado de forma equivalente por um C-obstáculo  $CB$  no espaço de configuração  $C$ .
  - C-Obstáculo é o conjunto de todas as configurações em que o robô se superpõe parcial ou totalmente ao obstáculo.

⇒ Solução para o problema de planejamento: Mapear  $W \Rightarrow C$



- **Métodos de Planejamento:**
  
- **Métodos Combinacionais (1980's):** baseados na construção de estruturas no espaço de configuração **C** que capturam completamente as informações para efetuar o planejamento.
  
- **Métodos baseados em Amostragem (1990's):** usam algoritmos de detecção de colisão para explorar o espaço de configuração **C** e buscar incrementalmente uma solução, ao invés de caracterizar completamente a estrutura do espaço livre.
  
- **Mapa de Rotas:**
  - Extração da conectividade do Espaço de Configuração Livre na forma de uma rede de curvas (Mapa de Rotas).
  - Construção de um grafo de conectividade do Mapa de Rotas.
  - Busca de um caminho no grafo de conectividade.
  
- **Decomposição em Células Convexas:**
  - Decomposição do Espaço de Configuração Livre em células convexas.
    - Decomposição Exata: a união das células é exatamente igual ao Espaço de Configuração Livre.
    - Decomposição Aproximada: a união das células é uma aproximação conservadora do Espaço de Configuração Livre.
  - Construção de um grafo de conectividade de acordo com as relações de adjacência entre as células.
  - Busca de um canal no grafo de conectividade.
  - Extração de um caminho a partir do canal.



- **Campo de Potencial:**

- Robô considerado como uma partícula imersa em um campo de potencial artificial.
- Obstáculos = potencial repulsivo; Alvo = potencial atrativo.
- Planejamento de caminho realizado incrementalmente, seguindo a direção de força artificial induzida na direção do negativo do gradiente da função de potencial.

- **Rapidly-exploring Random Trees (RRTs):**

- Sondagem e exploração agressiva do espaço de configuração, expandindo a busca incrementalmente a partir da configuração inicial.
- O território explorado é demarcado por uma árvore com raiz na configuração inicial.
- A cada iteração, a árvore é expandida adicionando novas configurações escolhidas aleatoriamente no espaço de configuração e tentando conectá-las ao ponto mais próximo da árvore por uma aresta contida no espaço livre. Continua-se até achar a configuração final.

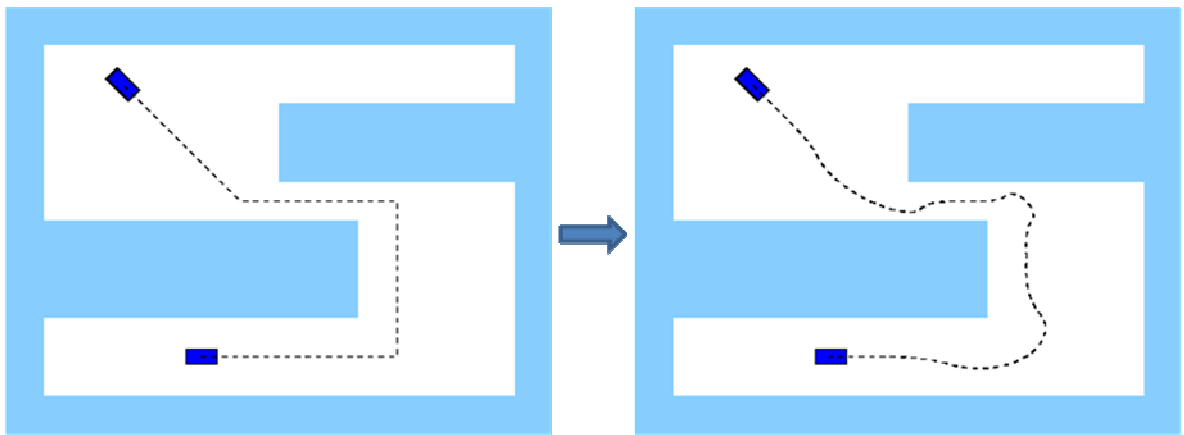
- **Mapa de Rotas Probabilístico:**

- Seleciona-se aleatoriamente um conjunto grande de configurações aleatórias no espaço livre e são consideradas vértices do mapa de rotas.
- Arestas do mapa de rotas são construídas tentando conectar cada vértice a um conjunto de vizinhos próximos.
- Se é possível construir um mapa de rotas que preserva acessibilidade e conectividade do espaço livre, pode ser utilizado para busca de pares ( $q_{ini}$ ,  $q_{fin}$ ) múltiplos.

## CONTROLE DE TRAJETÓRIA

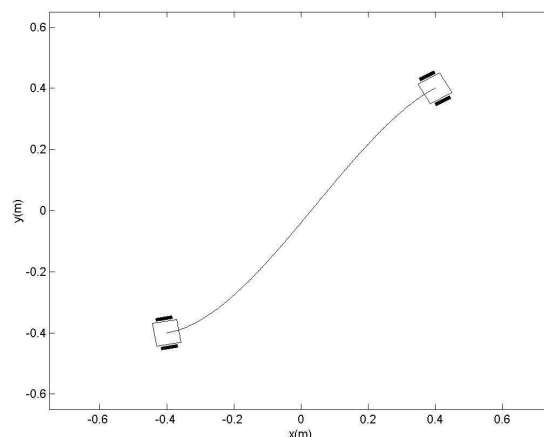
- Adequação de um Caminho:

- Transformação do caminho planejado (curva geométrica), através de pequenas deformações, em um novo caminho (nova curva) que satisfaz as restrições cinemáticas do robô (exemplo: raio de giro mínimo, restrições não holonômicas, etc.).



- Geração de Trajetória:

- Associação de restrições temporais (exemplo: velocidades máximas, tempo de percurso, etc.) e restrições dinâmicas (exemplo: forças de atrito, acelerações máximas, etc.) ao caminho gerado.

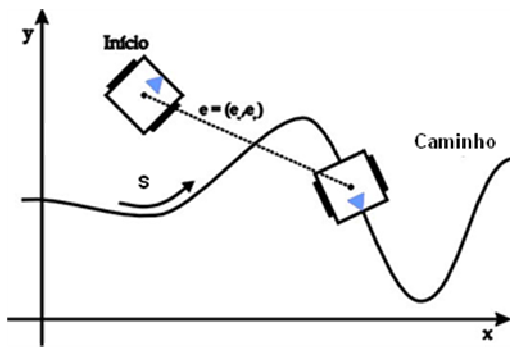


- **Execução de Trajetória:**

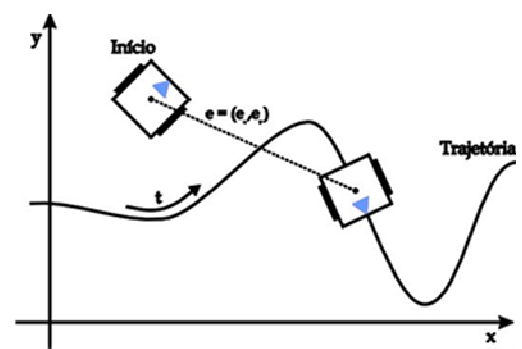
- Aplicação de leis de controle cinemático e dinâmico para que o robô siga a trajetória planejada.
- Leis de controle podem requerer o conhecimento do modelo cinemático e dinâmico do robô.
- Restrições cinemáticas e dinâmicas devem ser levadas em conta (restrições não holonômicas, raio de giro mínimo, etc.).

- **Controladores:**

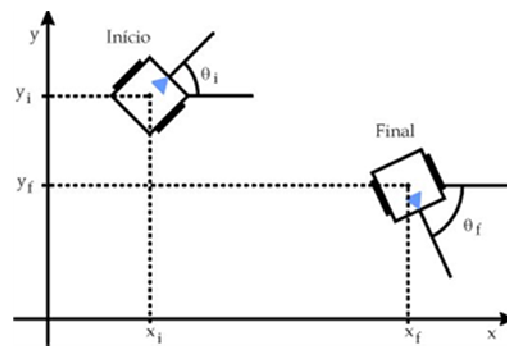
- Seguidores de Caminho – projetados para seguir um caminho.
- Seguidores de Trajetória – projetados para seguir uma trajetória contínua.
- Controladores Estabilizadores - Projetados para atingir uma configuração final.



Seguidor de Caminho.



Seguidor de Trajetória.



Controlador Estabilizante.