

Introdução à Inteligência Artificial

Especialização IAA

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

1

Conteúdo

1. Introdução
2. Resolução de Problemas
 - a. Busca
 - b. Jogos
 - c. CSP
3. Raciocínio
 - a. Agentes Lógicos
 - b. Lógica
 - c. Inferência
 - d. Planejamento
 - e. Representação do Conhecimento
4. Aprendizado
 - a. Aprendizado de Máquina
 - b. Redes Neurais Artificiais
 - c. SVM
 - d. Árvores
 - e. KNN
5. Comunicação, Percepção e Ação
 - a. Percepção
 - b. Robótica

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

2

2

1

1 Introdução

O que é?
História da IA
Estado da Arte
Assuntos da IA

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

3

3

O que é IA?



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

4

4

2

O que é IA?

- 4 abordagens do que é Inteligência Artificial



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

5

5

O que é IA?

- Definições segundo suas abordagens

Pensar como os Humanos

"O novo e interessante esforço para fazer os computadores pensarem (...) máquinas com mentes, no sentido total e literal." (Haugeland, 1985)

"[Automatização de] atividades que associamos ao pensamento humano, atividades como a tomada de decisões, a resolução de problemas, o aprendizado..." (Bellman, 1978)

Agir como os Humanos

"A arte de criar máquinas que executam funções que exigem inteligência quando executadas por pessoas." (Kurzweil, 1990)

"O estudo de como os computadores podem fazer tarefas que hoje são melhor desempenhadas pelas pessoas." (Rich e Knight, 1991)

Pensar Racionalmente

"O estudo das faculdades mentais pelo uso de modelos computacionais." (Charniak e McDermott, 1985)

"O estudo das computações que tornam possível perceber, raciocinar e agir." (Winston, 1992)

Agir Racionalmente

"Inteligência Computacional é o estudo do projeto de agentes inteligentes." (Poole *et al*, 1998)

"AI... está relacionada a um desempenho inteligente de artefatos." (Nilsson, 1998)

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

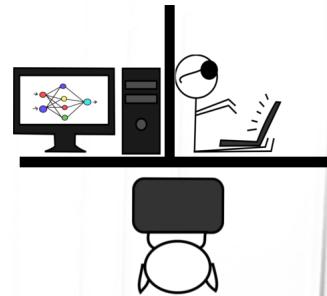
6

6

3

Agir como os Humanos

- Não implica na implementação do raciocínio
 - O que importa é se o resultado é parecido com um humano
 - O resultado dado pelo computador é um resultado humano?
- Turing (1950): Teste de Turing
- Nível satisfatório de inteligência?
 - Um humano entra em uma conversa, com outro humano e outra máquina
 - Conversa em língua natural
 - Todos os participantes estão separados
 - Se não for capaz de distinguir com segurança a máquina do humano, a máquina passa no teste
 - Não verifica respostas corretas, mas quão perto estão das respostas de humanos



Agir como os Humanos

- Objetivo não é definir o que é pensamento
- Nem implementar algum processo cognitivo
- Computador precisa imitar o comportamento humano
- “O jogo da imitação” (*The Imitation Game*)
- TURING, A. Computing Machinery and Intelligence. *Mind*, 59, 433-460. 1950.
Disponível em: <https://academic.oup.com/mind/article/LIX/236/433/986238>. Acesso em 21 de ago. de 2020.

Agir como os Humanos.

- Para passar no teste o computador precisa:
 - Processamento de linguagem natural
 - Representação do conhecimento
 - Raciocínio
 - Aprendizagem
 - Visão e Robótica

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

9

9

Pensar como os Humanos

- Para saber se um programa pensa, deve-se saber como os seres humanos pensam
- Entrar na mente dos humanos
 - **Introspecção:** captar nossos pensamentos conforme formos pensando
 - **Experimentos psicológicos:** observando uma pessoa em ação, em várias situações
- Quando for determinado como o humano pensa, pode-se criar um programa de computador
- O fato de um algoritmo dar resultados corretos não significa que simule o comportamento humano

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

10

10

Pensar como os Humanos.

- Abordagem voltada à **ciência cognitiva**
 - Estudo dos processos de funcionamento da mente humana
 - De forma empírica
- Dependem muito da experiência de cada um
- Muitíssimo difícil de se implementar, se for viável

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

11

11

Pensar Racionalmente

- Aristóteles (384 – 322 a.C) tenta codificar o "pensamento correto"
- Processos de raciocínio irrefutáveis
- Silogismos
 - Um argumento composto por 3 proposições
 - Duas afirmativas, uma conclusiva
- Este estudo deu início à lógica

Sócrates é um homem
Todos os homens são mortais
Então, Sócrates é mortal

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

12

12

Pensar Racionalmente.

- Hoje, dado premissas corretas, programas conseguem resolver problemas, codificados em lógica
 - Ex. Provedores Automáticos de Teoremas
- Entretanto, se não houver solução, podem entrar em um laço infinito
- Problemas
 - Não é fácil enunciar problemas informais em lógica
 - Principalmente com conhecimento incompleto ou incerto
 - Poder computacional: até mesmo problemas simples podem facilmente esgotar recursos computacionais

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

13

13

Agindo Racionalmente

- Um **agente** é algo que age:
 - Opera sob controle autônomo
 - Percebe seu ambiente
 - Adapta-se a mudanças
 - É capaz de criar e perseguir metas
- Um **agente racional** é aquele que age para alcançar o melhor resultado ou, quando há incerteza, o melhor resultado esperado

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

14

14

Agindo Racionalmente.

- A abordagem de agente racional tem algumas características
 - Uma inferência lógica pode ser um dos mecanismos para alcançar racionalidade
 - Existem modos de agir racionalmente que podem não envolver inferência (ex, atos reflexos)
 - Habilidades para passar no teste de Turing também permitem que o agente haja racionalmente
 - Representação do conhecimento e raciocínio também permite que agente tomem boas decisões
- Assim, ela é mais abrangente que outras abordagens

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

15

15

História da IA

- Período Inicial - 1943-1955:
 - 1943: Warren McCulloch e Walter Pitts, modelo de neurônios artificiais
 - 1949: Donald Hebb, aprendizado das redes
 - 1950: Marvin Minsky e Dean Edmonds, SNARC, primeiro computador de rede neural (simulava 40 neurônios)
 - 1950: Alan Turing, "Computing Machinery and Intelligence", Teste de Turing

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

16

16

História da IA

- Nascimento - 1956:
 - John McCarthy, Marvin Minsky, Claude Shannon, Nathaniel Rochester: Seminário de 2 meses para tentar mapear o aprendizado e construir uma máquina para simulá-lo
 - Allen Newell e Herbert Simon já tinham um programa de raciocínio (Logic Theorist)
 - O programa foi capaz de demonstrar a maioria dos teoremas do Cap. 2 do "Principia Mathematica" de Russel e Whitehead
 - Apresentou todos os atores da área
 - A IA foi dominada por estes nomes, seus alunos e colegas nos anos seguintes

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

17

17

História da IA

- Grandes expectativas – 1952 - 1969:
 - Computadores limitados
 - Computadores eram vistos como resolvedores de operações aritméticas, e nada mais
 - Classe intelectual indicava que a máquina não resolveria determinado problema
 - Pesquisadores de IA iam demonstrando problemas atrás de problemas
 - 1952: Arthur Samuel, programas para jogar damas e aprendiam a jogar melhor
 - 1957: Surge o GPS (General Problem Solver)
 - Imitar protocolos humanos e resolução de problemas
 - Primeiro programa a incorporar a abordagem "pensar de forma humana"
 - 1958: Surge o LISP – John McCarthy
 - 1959: Herbert Galernter, Geometry Theorem Prover
 - 1962: Berni Widrow, redes adalines
 - 1962: Frank Rosenblatt, perceptron (e o teorema da convergência do perceptron, algoritmo de aprendizagem)

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

18

18

História da IA

- 1966 - 1973:
 - Previsões para 10 anos:
 - Computador se tornar mestre do xadrez
 - Computador resolver um teorema matemático significativo
 - Só foram realizadas em 40 anos
 - Muitas dificuldades:
 - Primeiros programas só faziam manipulações sintáticas simples, faltava conhecimento sobre o assunto
 - Impossibilidade de tratar muitos problemas que a IA tentava resolver
 - Antes da Teoria da Complexidade, achava-se que tratar problemas maiores era só aumentar o hardware
 - Incapacidade de conviver com a explosão combinatorial
 - Governo Britânico corta apoio à pesquisa em IA em todas as universidades (exceto duas)

História da IA

- Sistemas Baseados em Conhecimento - 1969 - 1979:
 - 1969: DENDRAL, inferir estruturas moleculares
 - Era eficiente porque adicionava conhecimento
 - Início de 1970: MYCIN, para diagnosticar infecções sanguíneas. Usava 450 regras, ia tão bem quanto alguns especialistas e muito melhor que médicos em início de carreira
 - Conhecimento era obtido através de conversas com especialistas
 - Surgem os Sistemas Especialistas
 - Mais esforços em Compreensão de Linguagem Natural

História da IA

- Indústria – 1980 – até agora:
 - 1982 - McDermott – DEC - Primeiro Sistema Especialista bem sucedido – R1, ajudava a configurar pedidos de novos sistemas de computadores. Em 1986 ajudou a economizar 40 milhões de dólares por ano
 - 1988 – DEC tinha 40 sistemas especialistas entregues, com outros sendo produzidos
 - Du Pont tinha 100 sistemas especialistas em operação e 500 em desenvolvimento, economia de aproximadamente 10 milhões de dólares por ano
 - 1981 – Japão anuncia o projeto "Fifth Generation", plano para em 10 anos montar computadores inteligentes usando Prolog. EUA monta consórcio com o mesmo objetivo
 - Indústria de IA expandiu de alguns milhões de dólares em 1980 para bilhões de dólares em 1988

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

21

21

História da IA

- Indústria – 1980 – até agora:
 - Havia muita expectativa com os projetos subsidiados
 - Promessas grandiosas
 - Não conseguia produzir resultados imediatos
 - Setor foi enfraquecendo
 - Surge o **INVERNO DA IA**
 - Na década seguinte muitos pesquisadores trocam para áreas relacionadas com metas mais modestas
 - Aprendizado de máquina
 - Robótica
 - Visão

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

22

22

História da IA

- Retorno das Redes Neurais – 1986 – até agora:
 - Meados de 1980: grupos reinventam o algoritmo de aprendizado por **retropropagação**
 - Descoberto em 1969 por Bruson e ho
 - Modelos Conexionistas
 - Eram encarados como concorrentes dos Modelos Simbólicos (Newell e Simon) e Abordagem Logicista (McCarthy)

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

23

23

História da IA

- IA como ciência– 1987 – até agora:
 - Antigamente, IA era um tanto isolada da estatística e computação simbólica
 - Entendia-se que a IA ia contra as limitações destas áreas
 - Hoje IA encorpora estas áreas
 - IA adota o método científico
 - Para serem aceitas, **hipóteses** devem ser submetidas a rigorosos **experimentos empíricos** e **resultados** devem ser **analisados estatisticamente**.
 - Exemplo: Pesquisas com RNAs
 - Inicialmente eram sobre definir a abrangência do que poderia ser feito, bem como sobre a maneira como diferem das técnicas tradicionais.
 - Hoje sabe-se que podem ser comparadas a técnicas correspondentes da estatística, reconhecimento de padrões e aprendizado de máquina
 - Pode-se, inclusive, para cada experimento, determinar qual técnica é a melhor
 - Surge a **mineração de dados**
 - Surgem também as **redes bayesianas**, agregando o estudo de probabilidades para raciocinar sobre conhecimento incerto

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

24

24

História da IA

- Agentes Inteligentes – 1995 – até agora:
 - Agentes completos
 - Percebe seu ambiente
 - Age sobre o ambiente
 - Físicos ou de software
 - Internet é um ambiente propício para agentes
 - Exemplo: bots
 - Mecanismos de pesquisa, Sistemas de recomendação, agregadores de conteúdos
 - Para agentes físicos (como carros autônomos):
 - Progresso veio de uma mistura de abordagens: melhores sensores, geolocalização, mapeamento, planejamento, etc
 - Alguns pensadores, descontentes com a evolução da IA, querem que retorne à essência
 - Colocar menos ênfase em criar versões melhores de aplicações: jogador de xadrez melhor, dirigir carro, etc
 - Querem: "máquinas que pensam, que aprendem, que criam" (Simon)

História da IA.

- Disponibilidade de Conjuntos Grandes de Dados – 2001 – até agora:
 - Antigamente a ênfase maior era o **algoritmo**
 - Trabalho recente indica que para muitos problemas deve-se preocupar com os **dados**
 - Deve-se à crescente disponibilização de informações
 - Ex: preenchimento de buracos em fotos (Hay e Efros, 2007)
 - Retirou a foto de um ex-amigo em uma foto e quer preencher com algo que combine com o fundo
 - Análise de outras fotos
 - Desempenho pobre com 10 mil fotos
 - Excelente com 2 milhões de fotos

Estado da Arte

- **Veículos Autônomos:**

- Sensores: Radar (Radio Detection And Ranging), Sonar (Sound Navigation And Ranging) e Lidar (Light Detection And Ranging), detecção do que está a sua frente e ao redor
- Câmeras Stereo: detecção de objetos, distâncias
- GPS, Acelerômetro, Odômetro, etc
- Níveis de Automação
 - Nível 0 : Nenhuma automação – motorista faz tudo
 - Nível 1 : Assistência do motorista – uma ou mais funções específicas de controle, ex: cruise control, manter na pista, freios automáticos, etc
 - Nível 2 : Automação parcial – ajuda com aceleração e direção, mas o motorista precisa estar a postos para retomar o controle a qualquer momento
 - Nível 3 : Automação condicional – veículo controla o ambiente inteiro em determinadas situações (como estradas), nestes casos o motorista fica livre
 - Nível 4 : Automação alta – capaz de dirigir, frear, acelerar e monitorar o que ocorre ao seu redor, determinar quando trocar de faixa e virar, em um grande número de situações
 - Nível 5 : Automação total – autonomia total em qualquer situação, não é necessário nem um volante

Estado da Arte

- **Veículos Autônomos:**

- *Tesla Autopilot, Volvo XC60-* nível 2 (automação parcial): só funcionam sem intervenção na estrada e, mesmo assim, o motorista deve ter uma mão no volante
- *Audi A8 2018* – nível 3 (automação condicional): libera o motorista em engarrafamentos, estacionamentos, etc (não será vendido com 100% do SW, até que as leis necessárias forem criadas)
- *Renault Symbioz* – nível 4 (automação alta): (carro conceito) depois de iniciado o motorista fica liberado, o carro muda de faixa e até acessa o pedágio. Pode até mudar para uma faixa segura se ocorrer um acidente inesperado

Estado da Arte

- **Veículos Autônomos:**

- O nível 5 ainda não foi atendido:
 - Geração de 5TB a 20TB de dados por dia por veículo, para processamento instantâneo
 - 5G : Comunicação V2V e V2I
 - Infraestrutura: Países mais capacitados – Singapura, Holanda, Noruega, Estados Unidos e Finlândia. Brasil é um dos piores países (Estudo da KPMG/2020).

- **Necessário hardware especializado**

- NVIDIA DRIVE PX Pegasus: hardware com IA para atingir o nível 5
 - 2017/2018
 - 320 trilhões de operações por segundo
 - Objetivo seria criar taxis completamente autônomos
- NVIDIA DRIVE Atlan: chip único
 - Será lançado em 2023, com foco no mercado de 2025
 - 1000 trilhões de operações por segundo

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

29

29

Estado da Arte

- **Assistentes**

- Alexa, Google Home, Siri
- Conversa em inglês
- Tocar música, responder uma pergunta, clima, contar piada, ligar o rádio, dar notícias, programar alarme
- Reconhecimento de voz
- Integração com hardware (casa inteligente)

- **Planejamento Autônomo e Escalonamento**

- 2000: Remote Agent – primeiro planejador dentro de uma nave espacial
- MAPGEN, MEXAR2
- Terra-Marte : atraso de 13 minutos nas comunicações
- Mars 2020 Rover: deverá ser mais autônomo em seus experimentos

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

30

30

15

Estado da Arte

- **Jogos:**

- Jogo de Damas, (Samuel, 1959), função de avaliação que era aprendida a cada jogo. Jogava contra ele mesmo. Em 1962 derrotou Robert Nealy, campeão mundial
- Deep Blue (1997) derrotou Garry Kasparov no Xadrez.
- AlphaGo (Google, 2016), derrotou Lee Se-dol, que detinha 18 títulos internacionais no jogo
- AlphaZero (Google, 2017), aprendeu xadrez em 4hrs, derrotou o Stockfish, programa que detinha o título mundial de xadrez, em 100 partidas. Só foi programado com as regras do jogo, toda a estratégia foi montada pela máquina
- Jogos físicos (2018): Robô Curly venceu atletas coreanos de *curling*
- DeepMind (Google, 2019) venceu 99,8% das partidas de StarCraft II, com pessoas reais
- Cubo Mágico, Pôquer, Perguntas e Respostas,
- Controle de NPCs: regras, estados, etc

Estado da Arte

- **Combate ao Spam**

- Sistemas que aprendem o que é e o que não é spam, dependendo da pessoa
- Aprendizado

- **Planejamento Logístico**

- DART (Dynamic Analysis and Replanning Tool):
 - Forças armadas Estadunidenses para resolver problemas logísticos de transporte de materiais e pessoas
 - Gera planos para diminuir o custo e tempo na implantação das decisões logísticas
 - RDA - Resource Description and Access system
 - DRPI - Knowledge-Based Planning and Scheduling Initiative, sucessor do DART
- Veículos autônomos, entregas inteligentes
- Armazéns eficientes

Estado da Arte

- **Robótica**

- Aspiradores robóticos
- Drones semi-autônomos: desvio de obstáculos, reconhecimento de trilha, piloto automático

- **Tradução Automática**

- Sistemas melhoraram muito durante os anos
- Aplicação de técnicas de aprendizado
- iFlytek (China, 2018) – plataforma nacional de tradução com base em IA

- **Música e Arte:**

- Projeto Magenta (Google)
- Taryn Southern (youtube, 2018) – lançou álbum pop usando IA (software chamado Amper)

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

33

33

Estado da Arte.

- **Redes Sociais:**

- Recomendação Youtube
- Playlist Spotify
- Efeitos no Snapchat
- Reconhecimento de rosto no Facebook
- Feed de notícias
- Canadá, 2018 – monitorar redes sociais para detectar comportamentos suicidas

- **Bots**

- MIT, 2018 – Sistema usando Redes neurais para detectar depressão em textos e áudios
- Chatbot – bots de conversa:
 - baseados em regras : árvore de navegação ou árvore de diálogo
 - baseados em IA: Processamento de linguagem natural, aprendizado de máquina
 - Bancos, financeiras, lojas, setor público em alguns países, etc

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

34

34

Assuntos Abordados Sobre IA

- Resolução de Problemas
 - Busca
 - Jogos
 - CSP
- Raciocínio
 - Agentes Lógicos
 - Lógica
 - Inferência
 - Planejamento
 - Representação do Conhecimento
- Aprendizado
 - Redes Neurais Artificiais
 - SVM
 - Aprendizado Estatístico
- Comunicação, Percepção e Ação
 - Processamento de Linguagem Natural
 - Percepção
 - Robótica

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

35

35

Resolução de Problemas

- Agentes baseados em objetivos
- Definição do que é problema e solução
- Algoritmos de busca de propósito geral
- Problemas do mundo real
 - Caixeiro viajante
 - Navegação de robôs
- Busca
 - Busca em árvores
 - Busca sem informação: busca em largura, busca de custo uniforme, busca em profundidade, etc
 - Busca informada: heurística, busca gulosa, algoritmo A*, etc
 - Funções heurísticas

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

36

36

18

Resolução de Problemas

- Buscas não-clássicas
 - Subida de encosta
 - Têmpera simulada
 - Feixe local
 - Algoritmos genéticos
 - Etc
- Jogos
 - Algoritmo MINIMAX
 - Poda Alfa-Beta
- CSP: Problemas de Satisfação de Restrições
 - Coloração de Mapa
 - Agendamento de Tarefas

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

37

37

Raciocínio

- Agentes Lógicos
 - Representações internas de Conhecimento
 - Agentes baseados em Conhecimento
 - Mundo de Wumpus
 - Lógica
- Lógica de Primeira ordem
 - Inferência
- Planejamento
 - PDDL
 - STRIPS
- Representação do Conhecimento

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

38

38

19

Aprendizado

- Aprendizado a partir de Exemplos
- Árvores de decisão
- Regressão e Classificação
- Redes Neurais Artificiais
- Vizinho mais próximo
- Máquinas de Vetores de Suporte
- Agrupamento

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

39

39

Comunicação, Percepção e Ação..

- Processamento de Linguagem Natural
- Tradução Automática
- Reconhecimento de Voz
- Processamento de Imagens
- Reconhecimento de Objetos
- Robótica
- Sensores, Efetuadores
- Percepção Robótica

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

40

40

20

2 Resolução de Problemas

Problema
Busca
Jogos
CSP

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

41

41

Problema

- Assumimos uma abordagem baseada em Agentes de Resolução de Problemas, baseados em objetivos
- Agente:** elemento que percebe o ambiente (sensores) e age sobre ele (atuadores), a fim de resolver um problema
- Programa do Agente:** implementa a função do agente, isto é, recebem leituras dos sensores e devolvem uma ação para os atuadores
- Solução para um problema:** uma sequência fixa de ações

Estado inicial -> (*efetuar ações*) -> **Estado objetivo**

 - Estado Atual:** situação em que o agente se encontra agora
 - Estado Inicial:** situação de partida do agente
 - Estado Objutivo:** situação desejada

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

42

42

21

Problema

- Problema possui 5 componentes
 - **Estados:** Como um estado é definido, quais os elementos
 - **Estado Inicial:** Posição inicial do agente e estado inicial
 - **Ações:** Ações disponíveis
 - **Modelo de Transição:** Dado um Estados e uma Ação, descreve do que cada ação faz (transição) e qual o estado resultante
 - **Teste de objetivo:** como saber se o objetivo foi atingido
 - **Custo do Caminho:** Função de custo de caminho percorrido até o objetivo
- Problemas Simplificados
- Problemas do Mundo Real

Problema

Estado inicial -> (efetuar ações) -> Estado objetivo

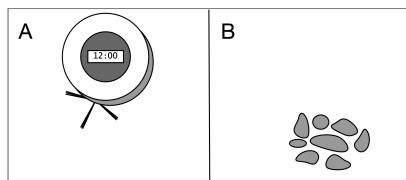
- **Busca:** processo de procura pela sequência de ações que fazem o agente alcançar seu objetivo
- Tem-se várias formas de busca, definidas por algoritmos diferentes e estratégias diferentes
- **Espaço de estados:** Estado inicial, ações possíveis em cada estado e modelo de transição. Conjunto de todos os estados acessíveis a partir do estado inicial, aplicando-se todas as ações possíveis
 - Grafo dirigido: cada vértice é um estado e cada aresta a aplicação de uma ação

Problema.

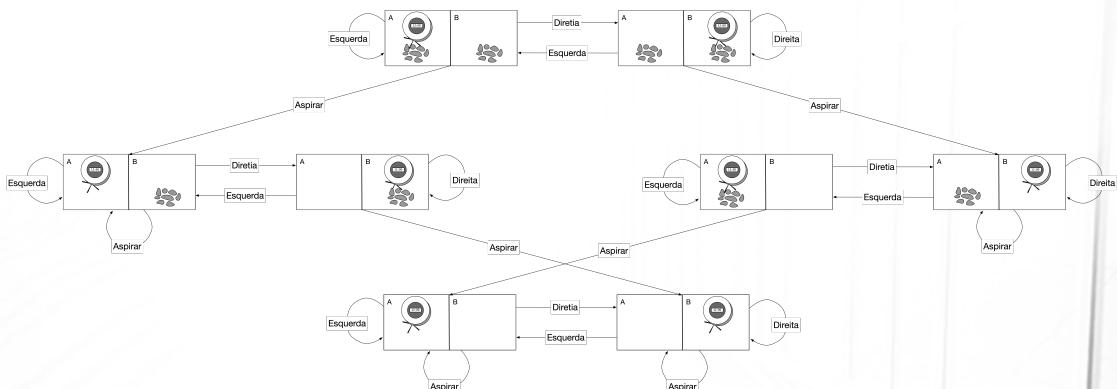
- **Busca por uma solução:** caminhar no espaço de estados, a partir do estado inicial, até encontrar o estado objetivo
- **Solução:** é o caminho encontrado, sequência de ações aplicadas
- **Solução ótima:** dentre todas as soluções possíveis, é a melhor. Depende da medida de qualidade da solução
 - Exemplo: Rotas entre duas cidades: melhor solução é a que tem menor distância nas estradas
 - Exemplo: Jogo de Xadrez: melhor solução é que leva à vitória

Mundo do Aspirador de Pó

- **Estados:** Posição do Aspirador e Situação da Sujeira
- **Estado Inicial:** Ex: duas salas, ambas sujas, aspirador na primeira sala
- **Ações:** Esperda, Direita e Aspirar
- **Modelo de Transição:** As ações tem efeitos esperados, exceto: *Direita* na sala mais à direita, *Esquerda* na sala mais à esquerda e *Aspirar* uma sala limpa, não possuem efeito
- **Teste de Objetivo:** Verifica se todas as salas estão limpas
- **Custo do Caminho:** Cada passo custa 1, o custo do caminho é o número de passos



Mundo do Aspirador de Pó – Espaço de Estados.



Fonte: (RUSSEL; NORVIG, 2013)

Prof. Dr. Razer A N R Montaño

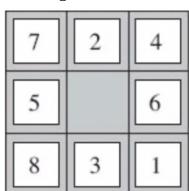
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

47

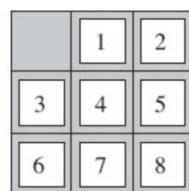
47

Quebra-cabeça de 8 peças

- Dada uma configuração embaralhada, movimentar o quadrado vazio até conseguir o padrão desejado
 - Estados:** Posição de cada uma das 8 peças e do quadrado vazio, em nove quadrados
 - Estado Inicial:** Qualquer estado pode ser estado inicial
 - Ações:** Ações do quadrado vazio: *Esperda, Direita, Para Cima, Para Baixo*
 - Modelo de Transição:** Efeito da aplicação de uma ação em algum estado
 - Teste de Objetivo:** Verifica se o estado corresponde ao que se deseja
 - Custo do Caminho:** Cada passo custa 1, o custo do caminho é o número de passos



Estado inicial



Estado objetivo

Fonte: (RUSSEL; NORVIG, 2013)

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

48

48

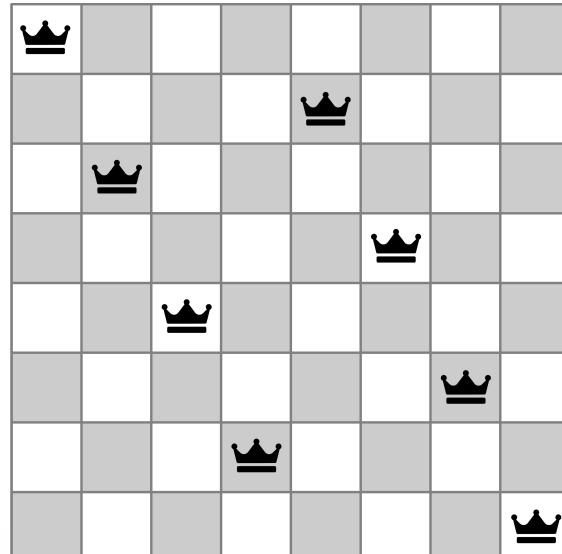
Quebra-cabeça de 8 peças.

- Com 8 peças (3x3): possui $9! / 2 = 181.440$ estados acessíveis
- Com 15 peças (4x4): possui 1,3 trilhão de estados
- Com 24 peças (5x5): 10^{25} estados

Problema das 8 rainhas

- Posicionar em um tabuleiro de xadrez 8 rainhas sem que uma ataque a outra
 - **Estados:** Qualquer tabuleiro com 0~8 rainhas é um estado
 - **Estado Inicial:** Nenhuma rainha no tabuleiro
 - **Ações:** Colocar uma rainha em qualquer casa vazia
 - **Modelo de Transição:** Dá o tabuleiro com a rainha recém adicionada
 - **Teste de Objetivo:** 8 rainhas estão no tabuleiro e nenhuma é atacada
 - **Custo do Caminho:** Não aplicável

Problema das 8 rainhas – Um estado



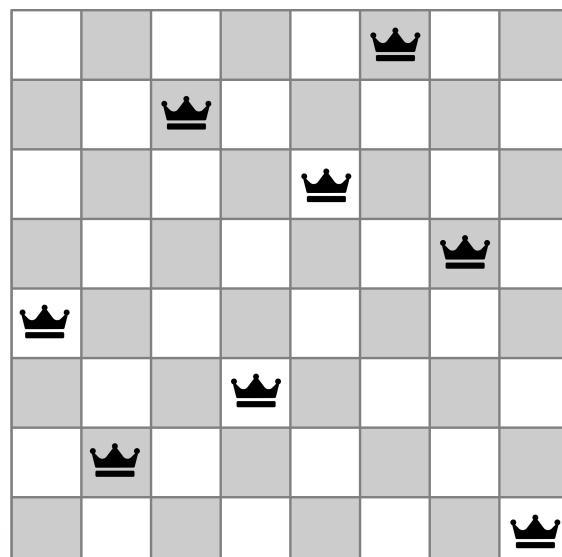
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

51

51

Problema das 8 rainhas – Uma Solução



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

52

52

Problema das 8 rainhas

- Total de soluções: 92 distintas, 12 soluções únicas (simetria)
- Com 8 rainhas
 - Coloca a 1^a rainha (64 possibilidades)
 - Depois a 2^a (63 possibilidades)
 - Depois a 3^a (62 possibilidades)
 - E assim sucessivamente até a 8^a
 - Deve-se testar $64 * 63 * 62 * \dots * 57 = 1,8 * 10^{14}$ possíveis sequências
- Espaço de estados
 - 64 elementos tomados 8 a 8
 - $C_8^{64} = \frac{64!}{(8!(64-8)!)} = 4.426.165.368$ possíveis tabuleiros
- Há uma forma de restringir as ações?

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

53

53

Problema das 8 rainhas.

- Sim! Diminuir o espaço de estados
 1. Somente permitir a inserção de rainhas colunas diferentes
 - Espaço de Estados: $8^8 = 16.777.216$ possíveis posições
 2. Somente permitir a inserção de rainhas em colunas e linhas diferentes
 - Espaço de Estados: $8! \Rightarrow 40.320$ possíveis posições
- Para um algoritmo que insere rainhas em casas que não são atacadas
 - Redução : de $1,8 * 10^{14}$ para 2.057 sequências a serem analisadas

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

54

54

27

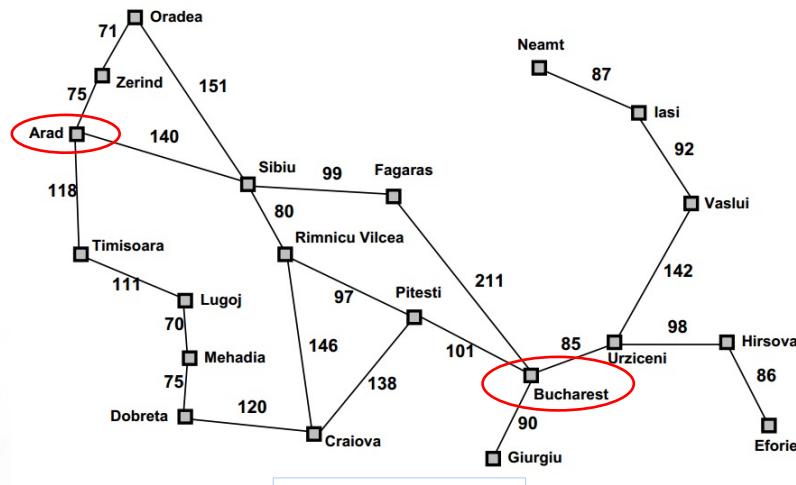
Problema das 8 rainhas.

- Com 100 rainhas
 - Aprox. 10^{400} estados
 - Solução melhorada cerca de 10^{52} estados
- Claramente é um problema **intratável**
 - Tem solução mas demora muito – ou consome muito recurso – para que consiga terminar

Busca por Soluções

- Uma **solução** é uma sequência de ações possíveis
- **Espaço de estados** é um grafo contendo todos os estados possíveis e todas as aplicações de ações possíveis
- Para encontrar a solução:
 - Percorrer o espaço de estados por meio de uma **árvore de busca**
- **Árvore de Busca**
 - Árvore onde a **raiz** é o estado inicial
 - **Ramos** são as ações aplicadas
 - **Nós** são estados no espaço de estados
- Exemplo
 - Encontrar uma rota entre **Arad** e **Bucareste** na Romênia

Busca por Soluções



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

57

57

Busca por Soluções

- Algoritmo Básico
 1. Testar se o nó raiz é o nó objetivo
 2. Se for, retorna
 3. Se não for Expande este nó
 4. Os nós expandidos são armazenados ("borda")
 5. Escolhe um nó da borda para analisar
 6. Testa se é nó objetivo
 7. Se não for, expande este nó, retira-o da borda e adiciona seus filhos na borda
 8. Repete até o objetivo ser encontrado

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

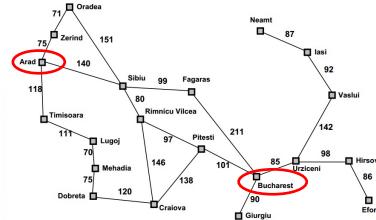
58

58

29

Busca por Soluções

Arad



Estado Inicial

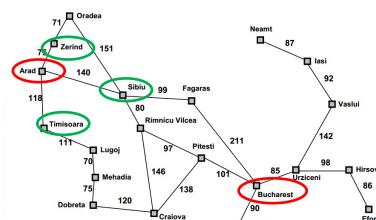
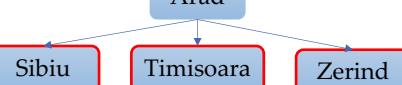
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

59

59

Busca por Soluções



Borda: Sibiu, Timisoara, Zerind

Analisa e Expande Arad

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

60

60

Busca por Soluções

```

graph TD
    Arad[Arad] --> Sibiu[Sibiu]
    Arad --> Timisoara[Timisoara]
    Arad --> Zerind[Zerind]
    Sibiu --> AradR[Arad]
    Sibiu --> Fagaras[Fagaras]
    Sibiu --> Oradea[Oradea]
    Sibiu --> RimnicuVilcea[Rimnicu Vilcea]
  
```

Borda: Arad, Fagaras, Oradea, Rimnicu Vilcea, Timisoara, Zerind

Analisa e Expande Sibiu

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

61

61

Busca por Soluções

```

graph TD
    Arad[Arad] --> Sibiu[Sibiu]
    Arad --> Timisoara[Timisoara]
    Arad --> Zerind[Zerind]
    Sibiu --> AradR[Arad]
    Sibiu --> Fagaras[Fagaras]
    Sibiu --> Oradea[Oradea]
    Sibiu --> RimnicuVilcea[Rimnicu Vilcea]
    RimnicuVilcea --> SibiuR[Sibiu]
    RimnicuVilcea --> Craiova[Craiova]
    RimnicuVilcea --> Pitesti[Pitesti]
  
```

Borda: Arad, Fagaras, Oradea, Sibiu, Craiova, Pitesti, Timisoara, Zerind

Analisa e Expande Rimnicu Vilcea

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

62

62

Busca por Soluções

Borda: Arad, Fagaras, Oradea, Sibiu, Craiova, Rimnicu Vilcea, Bucharest, Timisoara, Zerind

Analisa e Expande Pitesti

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

63

Busca por Soluções

Borda: Arad, Fagaras, Oradea, Sibiu, Craiova, Rimnicu Vilcea, Bucharest, Timisoara, Zerind

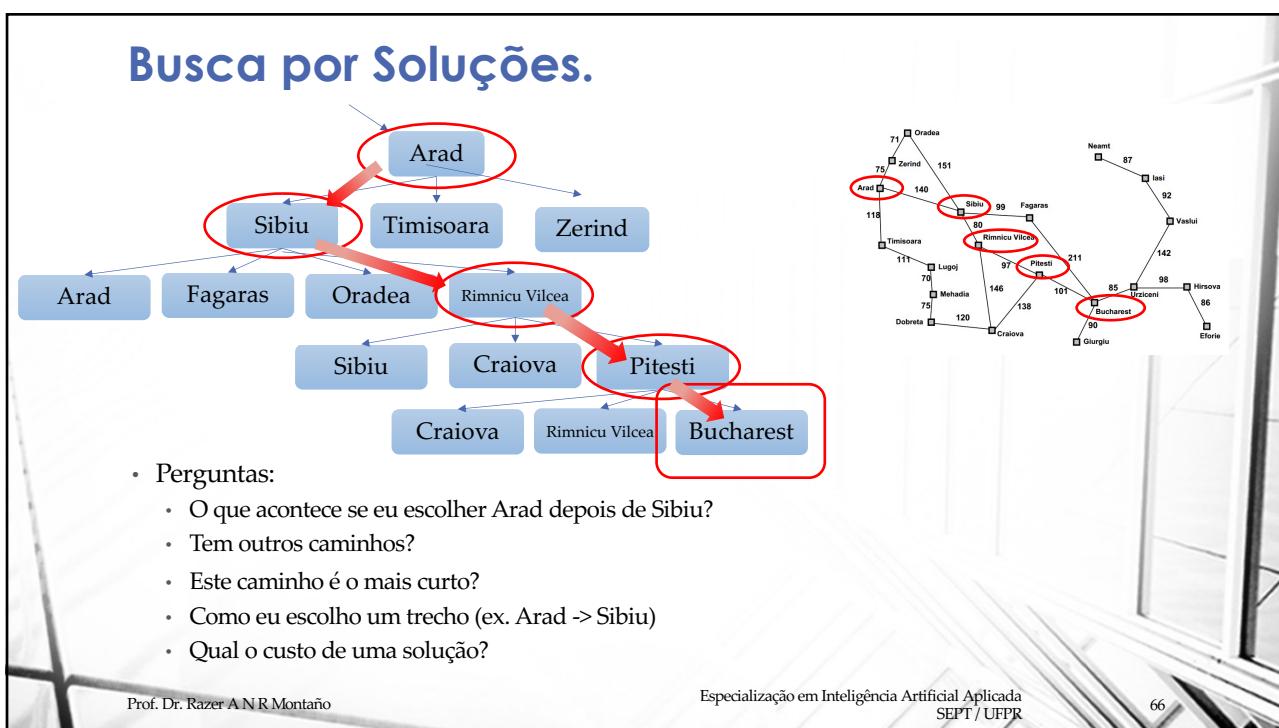
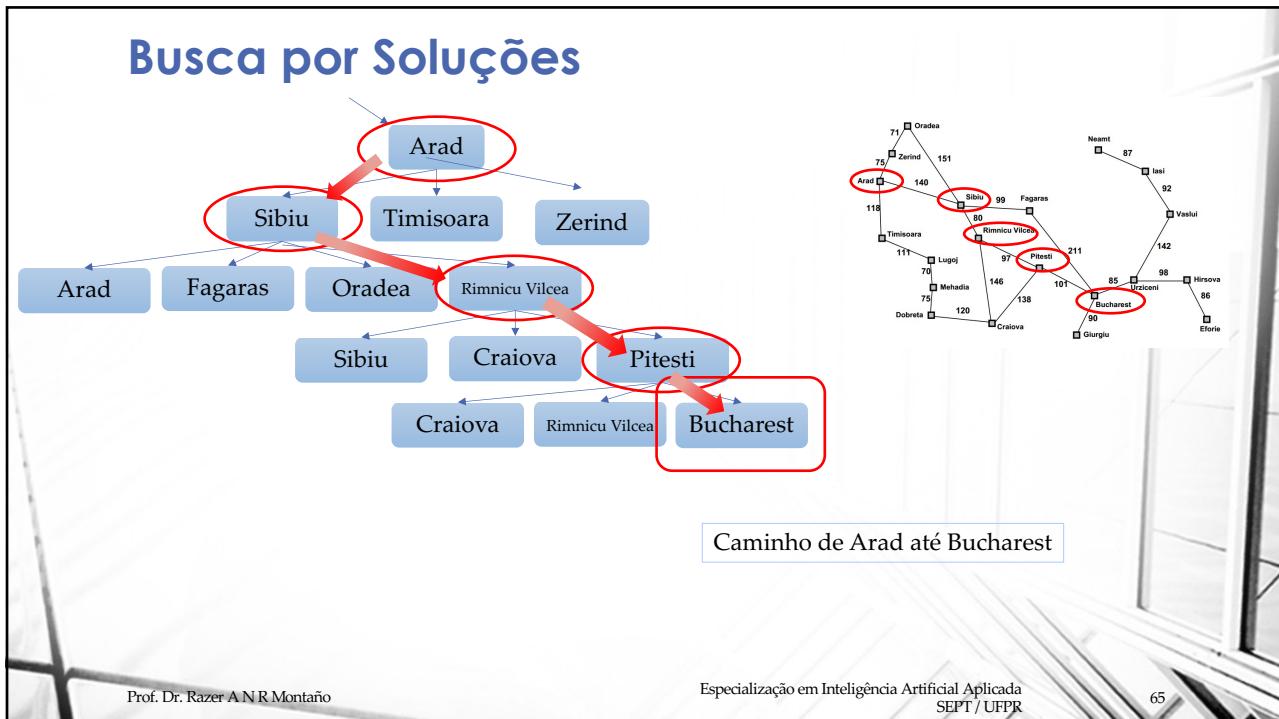
É o objetivo – Algoritmo Para!

Analisa Bucharest

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

64



Características.

- **Completude:** O algoritmo garante encontrar uma solução quando existir?
- **Otimização:** Encontra uma solução ótima?
- **Complexidade de Tempo:** Quanto tempo leva para encontrar uma solução?
- **Complexidade de Espaço:** Quanta memória é necessária para encontrar a solução?

Estratégias de Busca

- **Busca Cega:**
 - Sem informação adicional, além da colocada no problema
 - Tudo o que pode ser feito é gerar sucessores, distinguindo entre objetivo e não
 - Exemplos:
 - Busca em Largura
 - Busca de Custo Uniforme
 - Busca em Profundidade
 - Busca Bidirecional
- **Busca Heurística:**
 - Utiliza conhecimento do problema, além da informada no problema
 - O nó selecionado para expansão é escolhido com base em uma função de avaliação
 - Exemplos:
 - Busca Gulosa
 - A*

Estratégias de Busca.

- **Busca Local:**

- Bom quando o custo da solução não interessa
- Analisa um ou mais estados atuais
- Exemplos:
 - Subida de Encosta
 - Têmpera Simulada
 - Algoritmos Genéticos

- **Busca Competitiva (Jogos):**

- Mais de um agente interagindo
- Exemplos:
 - Algoritmo MINIMAX

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT/UFPR

69

69

Busca Cega

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT/UFPR

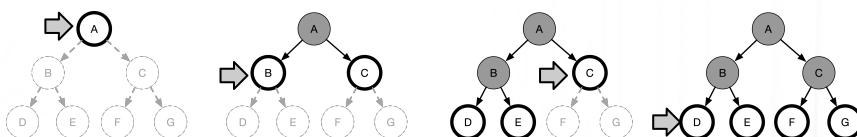
70

70

35

Busca Cega: Busca em Largura

- Busca em largura (*Breadth-first search*)
 - O nó raiz é expandido
 - Depois todos os sucessores do nó raiz são expandidos
 - Depois os sucessores destes nós



- Algoritmo usa uma FILA (FIFO) para manter os nós **borda**
 - **ENQUEUE**: enfileira um nó
 - **DEQUEUE**: desenfileira um nó
- Também Marca os nós que já foram testados se são objetivo

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

71

71

Busca Cega: Busca em Largura

```

ALGORITMO BuscaEmLargura
  Marque a raiz (estado inicial)
  ENQUEUE (FIFO, raiz)
  ENQUANTO ! EMPTY(FIFO) FAÇA
    v = DEQUEUE (FIFO)
    Adjacentes = ADJACENTES (v)
    PARA CADA w em Adjacentes FAÇA
      SE w não está marcado ENTÃO
        SE w é Estado Objetivo ENTÃO
          RETORNE Solução
        FIM SE
        Marque w
        ENQUEUE (FIFO, w)
      FIM SE
    FIM PARA
  FIM ENQUANTO
  
```

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

72

72

Busca Cega: Busca em Largura.

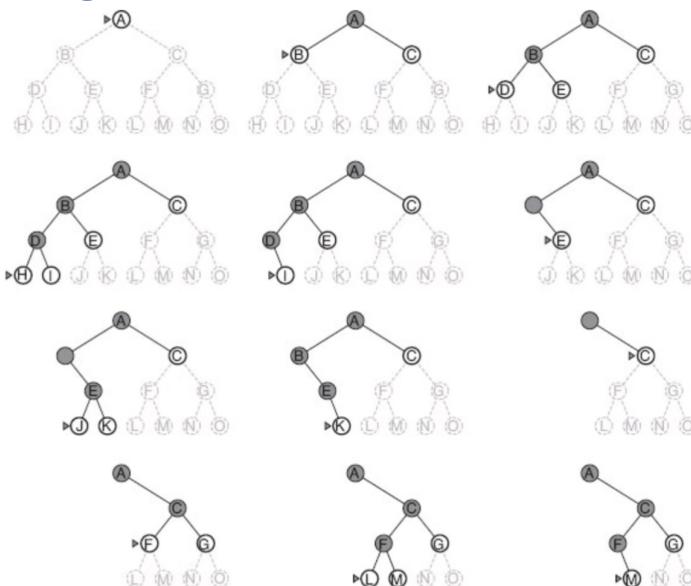
- Exemplo de Requisitos de Tempo e Memória
 - Fator de ramificação = 10
 - Um milhão de nós por segundo
 - 1.000 bytes por nó

Profundidade	Nós	Tempo	Memória
2	110	0,11 milissegundo	107 kilobytes
4	$11 \cdot 10^6$	11 milissegundos	10,6 megabytes
6	10^6	1,1 segundo	1 gigabyte
8	10^8	2 minutos	103 gigabytes
10	10^{10}	3 horas	10 terabytes
12	10^{12}	13 dias	1 petabyte
14	10^{14}	3,5 anos	99 petabytes
16	10^{16}	350 anos	10 exabytes

Busca Cega: Busca em Profundidade

- Busca em Profundidade (*Depth-first search*)
- Sempre expande o nó mais profundo
 - Vai até não ter mais sucessores
- Algoritmo usa uma PILHA (LIFO) para manter os nós **borda**
 - **PUSH:** empilha um nó
 - **POP:** desempilha um nó
- Ou implementa-se como uma Recursão
- Também Marca os nós que já foram testados se são objetivo

Busca Cega: Busca em Profundidade



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

75

75

Busca Cega: Busca em Profundidade

```

ALGORITMO BuscaEmProfundidade(v)
    Filhos = FILHOS(v)
    PARA CADA w em Filhos FAÇA
        SE w é Estado Objetivo ENTÃO
            RETORNE Solução
        SENÃO
            BuscaEmProfundidade(w)
            SE encontrou Solução ENTÃO
                RETORNE Solução
            FIM SE
        FIM SE
    FIM PARA

```

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

76

76

38

Busca Cega: Busca em Profundidade

- Em um espaço de estados infinito
 - Facilmente o algoritmo entra em **loop infinito**
- Uma abordagem é limitar a altura da busca
 - **Busca em Profundidade Limitada**
 - A partir de uma profundidade, considera que um nó não tem sucessores
- Resolve o problema de loop infinito
 - Mas gera incompletude
 - Ex.: Se escolhe-se n como altura máxima, o que acontece se a solução está em n+1?
- Para tentar resolver este problema
 - **Busca de Aprofundamento Iterativo**
 - Faz Busca em Profundidade Limitada, mas aumenta o limite gradualmente
 - Primeiro 0, depois 1, depois 2, etc

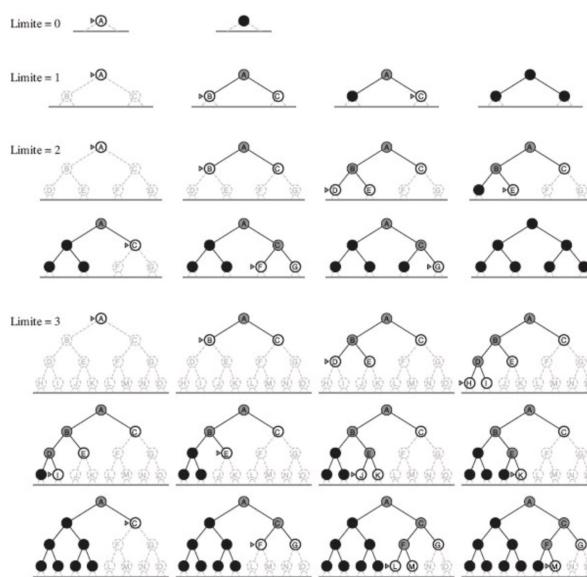
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

77

77

Busca Cega: Busca em Profundidade.



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

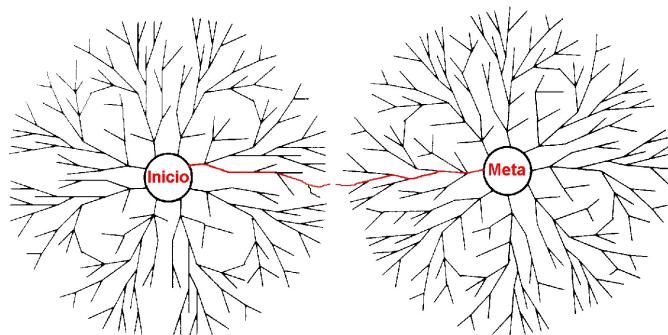
78

78

39

Busca Cega: Busca Bidirecional..

- Efetuar duas buscas ao mesmo tempo
 - Uma do Estado Inicial até o Estado Objetivo
 - Outra do Estado Objetivo até o Estado Inicial
- A ideia é que as duas buscas se encontrem em algum ponto



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

79

79

Busca Heurística

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

80

80

40

Busca Heurística

- Busca Heurística / Busca Informada
 - Utiliza informação do problema, além da definição do problema
 - Pode encontrar soluções de forma mais eficiente
- Dados os algoritmos anteriores
 - Escolhe-se um nó para expansão com base em uma **função de avaliação**
 - É uma estimativa de custo
 - Escolhe-se o nó com menor estimativa de custo
 - A maioria dos algoritmos usa uma função heurística ($h(n)$) como componente da função de avaliação
 - $h(n)$ = custo estimado do caminho de menor custo, do nó n para o estado objetivo
 - Simplificando:
 - Funções não negativas
 - $h(n) = 0$ quando n é estado objetivo

Busca Heurística.

- Busca Gulosa
- Busca A*

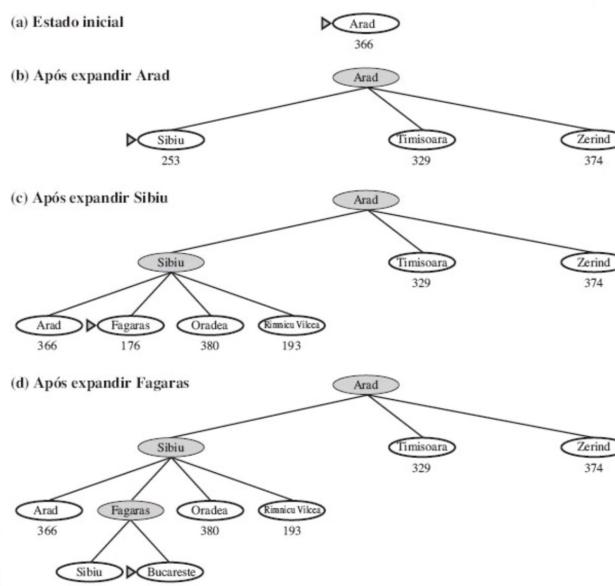
Busca Gulosa

- Tenta expandir o nó mais próximo do objetivo
 - Ideia: Isso pode conduzir a uma solução mais rapidamente
- Heurística usada:
 - Distância em linha reta até o objetivo
 - Essa informação precisa ser conhecida

Arad	366	Mehadia	241
Bucareste	0	Neamt	234
Craiova	160	Oradea	380
Drobeta	242	Pitesti	100
Eforie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hirsova	151	Urziceni	80
Iasi	226	Vaslui	199
Lugoj	244	Zerind	374

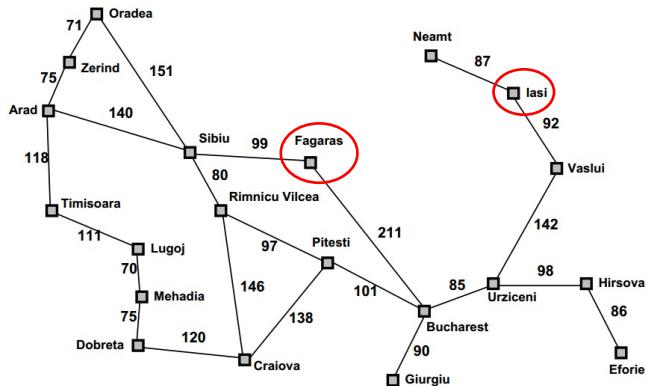
Figura 3.22 Valores de $hDLR$ — distâncias em linha reta para Bucareste.

Busca Gulosa



Busca Gulosa.

- Essa busca não é completa. Ex.: Iasi -> Fagaras
 - Neamt -> Fagaras
 - Vaslui -> Fagaras
 - O algoritmo sugere ir para Neamt
 - Depois de Neamt, expande Iasi
 - Laço infinito
 - **Beco sem saída!!!!**



Prof. Dr. Razer A N R Montaño

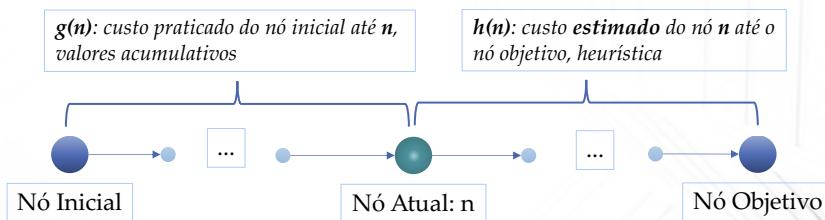
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

85

85

Algoritmo A*

- Sua função de avaliação é:
$$f(n) = g(n) + h(n)$$
 - Onde:
 - $g(n)$ = Custo do caminho do nó inicial até n
 - $h(n)$ = Custo estimado do caminho de menor custo de n até o objetivo
 - $f(n)$ = Custo estimado da solução de menor custo através de n



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

86

86

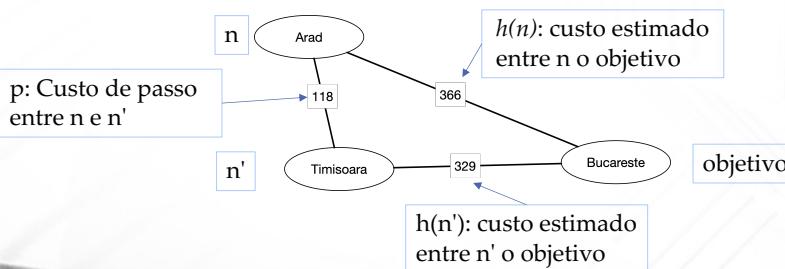
Algoritmo A*

- Estratégia: Tentar em primeiro lugar o nó com menor $f(n)$
 - A borda é implementada como uma Fila de Prioridades
 - Mantém uma ordenação conforme $f(n)$
 - Facilita a implementação e busca do próximo nó

Algoritmo A*

- Desde que $h(n)$ satisfaça certas condições, é completa e ótima
 - Deve ser **Admissível**: nunca superestima o custo de atingir o objetivo
 - Deve ser **Consistente ou Monotônica**: dado n e seus sucessores n' , o custo estimado para alcançar o objetivo a partir de n sempre for menor ou igual ao custo de passo para seus sucessores mais o custo estimado para alcançar o objetivo a partir de um sucessor

$$h(n) \leq p + h(n')$$



Algoritmo A*

```

Nó ← Estado inicial do problema
Custo-Caminho ← 0
Borda ← CRIAR-FILA-PRIORIDADES()
INSERIR-PRIORIDADES(Borda, Nó)
Explorado ← Ø
REPITA
    SE VAZIA(Borda) ENTÃO
        RETORNAR Falha
    FIM SE
    Nó ← REMOVER-PRIORIDADES(Borda)
    SE Nó é Objetivo ENTÃO
        RETORNAR Solução
    FIM SE
    Explorado ← Explorado ∪ {Nó}
    AçõesPossíveis ← Ações possíveis a partir de Nó
    PARA CADA Ação e AçõesPossíveis FAÇA
        Filho ← NÓ-FILHO(Nó, Ação)
        SE Filho não está na Borda E
            Filho ∉ Explorado ENTÃO
                INSERIR-PRIORIDADES(Borda, Filho)
        SENÃO
            SE Filho Está na Borda E
                g(n)+h(n) na Borda é maior ENTÃO
                    Substitui o nó na Borda por Filho
            FIM SE
        FIM SE
    FIM PARA
FIM REPITA

```

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

89

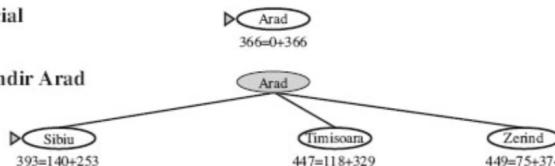
89

Algoritmo A*

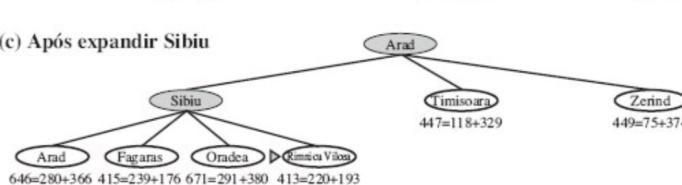
(a) Estado inicial



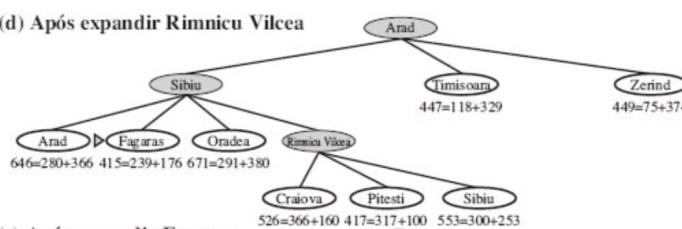
(b) Após expandir Arad



(c) Após expandir Sibiu



(d) Após expandir Rimnicu Vilcea



Prof. Dr. Razer A N R Montaño

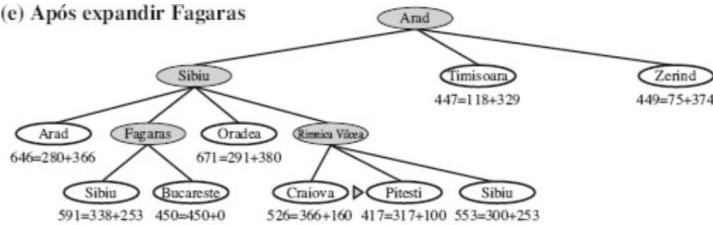
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

90

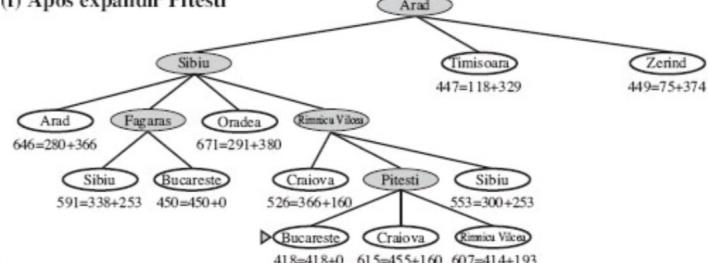
90

Algoritmo A*.

(e) Após expandir Fagaras



(f) Após expandir Pitesti



Prof. Dr. Razer A N R Montaño

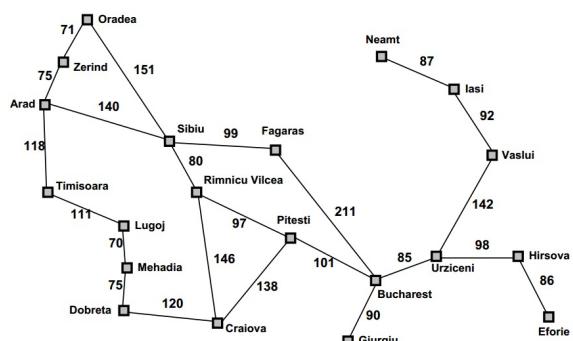
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

91

91

Exercício..

- Fazer a busca A* de ORADEA -> BUCHAREST



Arad	366	Mehadia	241
Bucarest	0	Neamt	234
Craiova	160	Oradea	380
Drobeta	242	Pitesti	100
Eforie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hirsova	151	Urziceni	80
Iasi	226	Vaslui	199
Lugoj	244	Zerind	374

Figura 3.22 Valores de $hDLR$ — distâncias em linha reta para Bucarest.

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

92

92

Busca Local

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

93

93

Busca Local

- Executa busca local no espaço de estados
 - Avaliando e modificando um ou mais estados atuais
 - Não explorar sistematicamente os caminhos a partir de um estado inicial
 - Têmpera Simulada
 - Algoritmos Genéticos
- Problemas onde o caminho até a solução é irrelevante
 - Ex.: 8 rainhas – a ordem de colocação de rainhas independe, só a configuração final é importante
- Não há necessidade de armazenar os caminhos
 - Analisam um único estado atual
 - Em geral se movem na vizinhança deste estado

Prof. Dr. Razer A N R Montaño

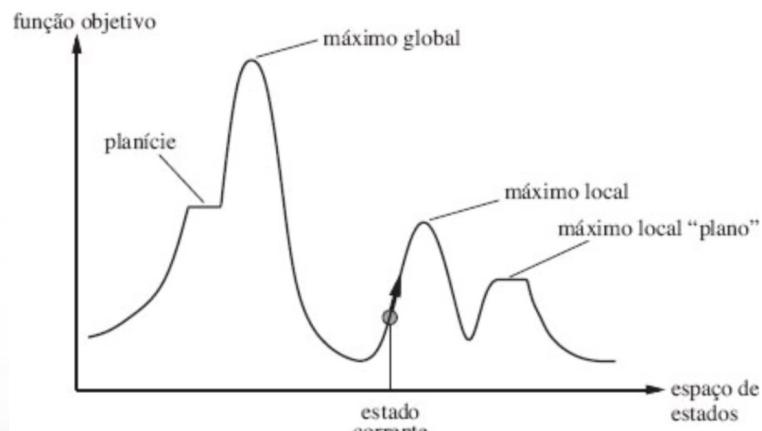
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

94

94

Busca Local

- Topologia do Espaço de Estados
 - Objetivo é encontrar o melhor estado de acordo com uma **função objetivo**
 - Neste caso é encontrar o **máximo global**



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

95

95

Busca Local.

- Subida de Encosta
- Têmpera Simulada
- Algoritmos Genéticos

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

96

96

Busca Local: Subida de Encosta

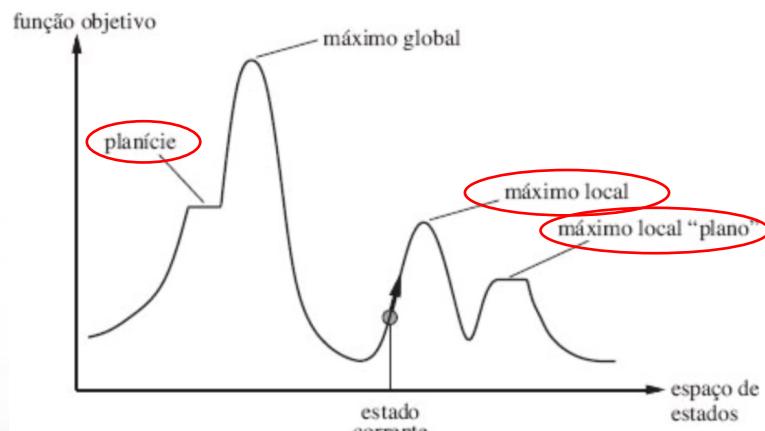
- Subida de Encosta (ou *Hill-Climbing*) ou Busca Gulosa Local
- Um laço repetitivo que se move para valores mais altos
 - Encosta acima
- O algoritmo termina quando encontra um **pico**
 - Em que nenhum vizinho tem valor mais alto
 - Só analisa os vizinhos imediatos do estado corrente
 - Não precisa manter a árvore de busca
 - Guarda só o estado atual e tenta melhorá-lo

Busca Local: Subida de Encosta

```
ALGORITMO SubidaDeEncosta()
    Atual = Estado Inicial
    REPITA
        Vizinho = sucessor de "Atual" com melhor avaliação
        SE AVAL(Vizinho) <= AVAL(Atual) ENTÃO
            RETORNE Estado do nó corrente
        SENÃO
            Atual = Vizinho
        FIM SE
    FIM REPITA
```

Busca Local: Subida de Encosta

- Pode levar o algoritmo a paralisar em:
 - Máximos locais : onde nenhum valor na vizinhança é maior, mas não é o objetivo
 - Platôs : onde os valores na vizinhança são iguais



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

99

99

Busca Local: Subida de Encosta.

- Como resolver os problemas?
 - Movimentos Laterais:** Acreditando que o platô possa ser uma planície (com valores superiores mais à frente), permite-se um movimento para um estado com a mesma avaliação. Deve-se limitar a quantidade de movimentos laterais
 - Reinício Aleatório:** Quando não há solução, reinicia com um estado inicial aleatório
- Problemas reais e difíceis, em geral, têm um número exponencial de máximos locais em que ficam presos
 - Apesar disso, um máximo local razoavelmente bom pode ser encontrado facilmente com reinícios aleatórios

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

100

100

Busca Local: Têmpera Simulada

- Problema de completude x eficiência
- Subida de Encosta: nunca anda para um estado com valor de avaliação menor
 - Não é completo, pois pode ficar preso em um máximo local
- Escolher um sucessor ao acaso, é completo mas extremamente ineficiente
- Ao juntar as duas estratégias tem-se a Têmpera Simulada (ou *Simulated Annealing*)
 - Subida de encosta com um percurso aleatório
 - Chacoalhada nos estados sucessores
 - Estados com avaliação pior podem ser escolhidos com uma certa probabilidade
 - Esta probabilidade diminui com o tempo
 - Permite que o algoritmo, além de subir encostas, também desça encostas e ande por platôs
- Analogia com cozimento de metais
 - Aquece-se a uma temperatura alta, depois diminui a temperatura T gradativamente
 - Se T diminui devagar o suficiente, a qualidade do metal é melhor

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

101

101

Busca Local: Têmpera Simulada.

- Escolhe um vizinho aleatório
- Se melhorar a situação, é sempre aceito
- Caso contrário, aceita o vizinho com alguma probabilidade menor que 1
 - Esta probabilidade decresce conforme a temperatura T decresce
 - Movimentos ruins são mais frequentes no início, quanto T é alto
 - Portanto T deve diminuir com lentidão

Prof. Dr. Razer A N R Montaño

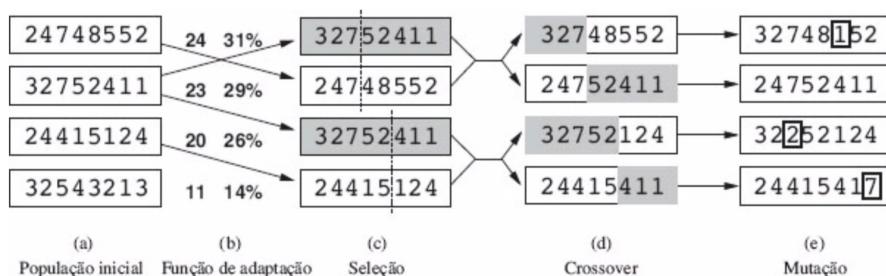
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

102

102

Busca Local: Algoritmos Genéticos

- Inicia-se com k estados aleatórios: população
- Os estados sucessores são gerados a partir da combinação de dois estados pais
- Cada estado (ou indivíduo) é representado por uma cadeia de valores (conforme modelagem do problema)
 - Ex.: para as 8-rainhas, cada valor representa em que linha está a rainha , para cada uma das colunas



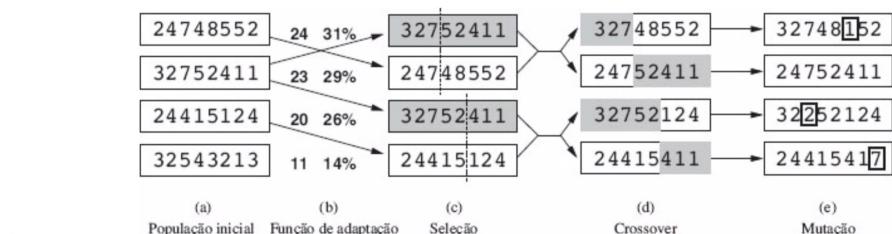
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

103

103

Busca Local: Algoritmos Genéticos



- (a) mostra a população com 4 indivíduos
- (b) aplica-se a função de avaliação (em AG – função de adaptação)
 - Neste caso, 28 seria o valor maior (nr de pares de rainhas não atacantes)
 - Um valor maior, representa um estado mais próximo da solução
 - Também é indicada a probabilidade de um indivíduo ser escolhido para reprodução, proporcional ao valor da função de adaptação

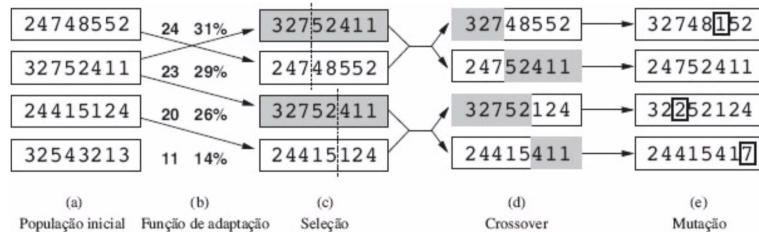
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

104

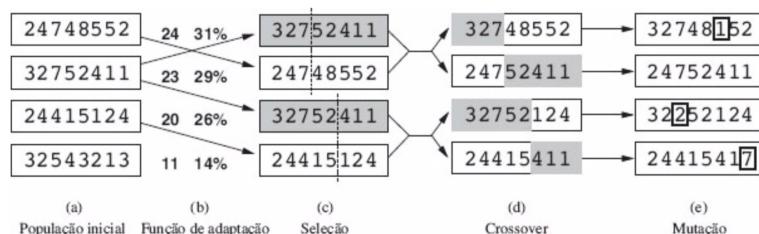
104

Busca Local: Algoritmos Genéticos

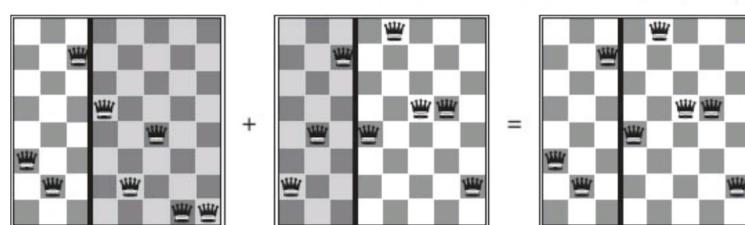


- (c) dois pares de indivíduos são escolhidos ao acaso, de acordo com as probabilidades
 - Um indivíduo foi escolhido duas vezes
 - Um não foi escolhido
 - Para cada par escolhido, é escolhido um ponto de cruzamento ao acaso
- (d) os indivíduos resultantes do crossover são apresentados

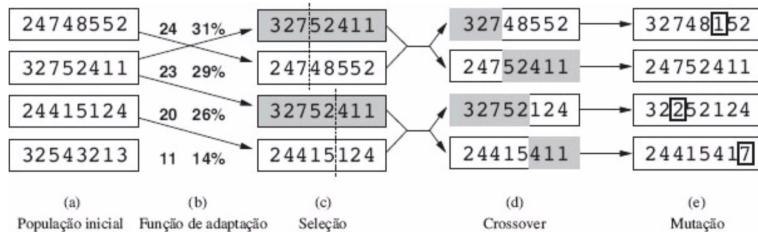
Busca Local: Algoritmos Genéticos



- (d) os indivíduos resultantes do crossover são apresentados



Busca Local: Algoritmos Genéticos..



- (e) os indivíduos resultantes podem sofrer mutação aleatória, independente e com uma pequena probabilidade

Busca Competitiva

Jogos - Busca Competitiva

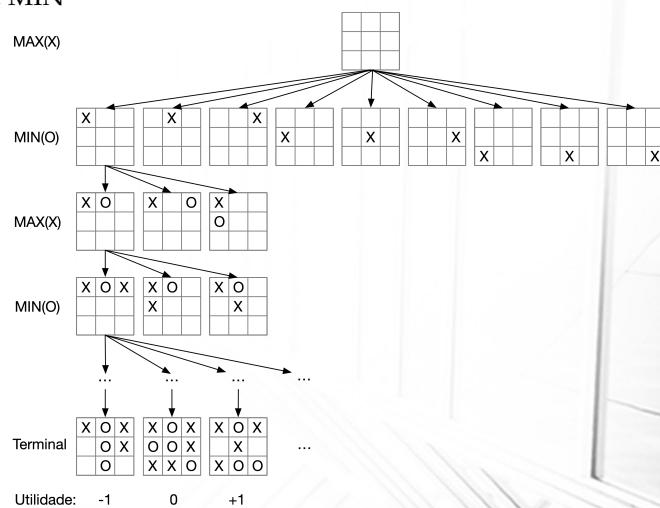
- Teoria dos Jogos
 - Ramo da Economia
 - Qualquer ambiente multiagentes é um Jogo
 - Desde que o impacto de um agente sobre os outros é significativo
 - Não importa se são cooperativos ou competitivos
- Em IA existem vários tipos de jogos, os mais comuns são:
 - Determinísticos
 - Soma-zero (o ganho de um jogador necessariamente é a perda do outro)
 - Possuem informações perfeitas
 - Ex.: Xadrez, Damas
 - A posição (favorável ou desfavorável) de um jogador num determinado instante (estado) do jogo pode ser medida por uma função de utilidade.
 - Os valores de utilidade dos agentes no fim do jogo são iguais e opostos (simétricos): +1 (ganha), ou -1 (perde).
 - Planejar com antecedência em um mundo onde os outros fazem planos contra nós

Jogos - Busca Competitiva

- Definição formal:
 - S_0 : **estado inicial**, como o jogo está no início;
 - JOGADORES(s): retorna o jogador que deve se mover em um estado s;
 - AÇÕES(s): retorna os válidos no estado s;
 - RESULTADO(s, a): o **modelo de transição**, define o resultado da aplicação de uma jogada a no estado s;
 - TESTE DE TÉRMINO(s): é o teste de objetivo, ou **teste de término**, que retorna verdadeiro quando o jogo termina. Os estados em que o jogo termina são chamados **estados terminais**;
 - UTILIDADE(s, p): **função utilidade**, ou função objetivo ou função compensação, que define um valor numérico para um jogo que termina no estado terminal s por um jogador p. Por exemplo, no xadrez tem-se vitória, derrota ou empata (+1, 0 ou $\frac{1}{2}$).

Jogos - Busca Competitiva

- Considera-se 2 jogadores: MAX e MIN
 - MAX faz o primeiro movimento $\text{MAX}(X)$
 - Depois se revezam
- Para o jogo da velha:
 - Menos de $9! = 362.880$ nós
- Para o Xadrez:
 - Aprox. 10^{40} nós



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

111

111

Jogos - Busca Competitiva

- Utilidade:** valor objetivo, valor numérico para um jogo que termina no nó por um jogador. Ex.: -1, 0, +1 (perde, empata, ganha)
- Cada nó possui um **valor minimax**
- Se é nó terminal (folha):
 - É sua utilidade
- Se não:
 - É a utilidade PARA MAX de se encontrar neste nó

Prof. Dr. Razer A N R Montaño

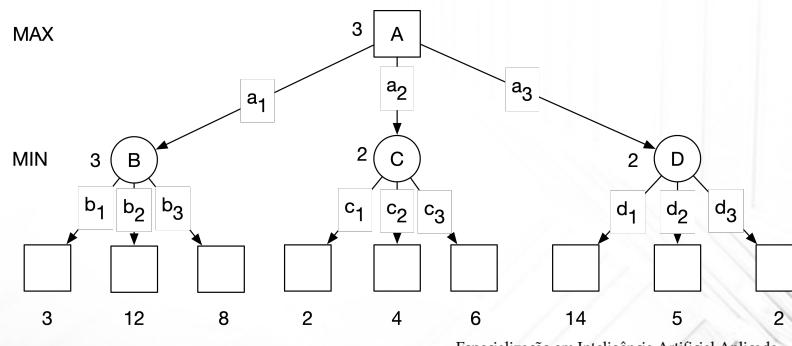
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

112

112

Jogos – Busca Competitiva

- MAX sempre prefere se mover para um estado com valor MÁXIMO
- MIN sempre prefere se mover para um estado com valor MÍNIMO
- Isso indica que tanto MAX como MIN fazem sua **melhor jogada**
 - MAX pressupõe que MIN jogue de forma ótima
 - Se ele não jogar, MAX jogará melhor ainda



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

113

113

Jogos - Busca Competitiva

```

ALGORITMO minimax(nó, profundidade)
  SE nó é um nó terminal OU profundidade = 0 ENTÃO
    RETORNE o valor da heurística do nó
  SENÃO SE o nó representa a jogada de algum adversário ENTÃO
    α ← +∞
    PARA CADA filho DE nó
      α ← min(α, minimax(filho, profundidade-1))
    FIM PARA
    RETORNE α
  SENÃO
    α ← -∞
    PARA CADA filho DE nó
      α ← max(α, minimax (filho, profundidade-1))
    FIM PARA
    RETORNE α
  FIM SE
FIM ROTINA
  
```

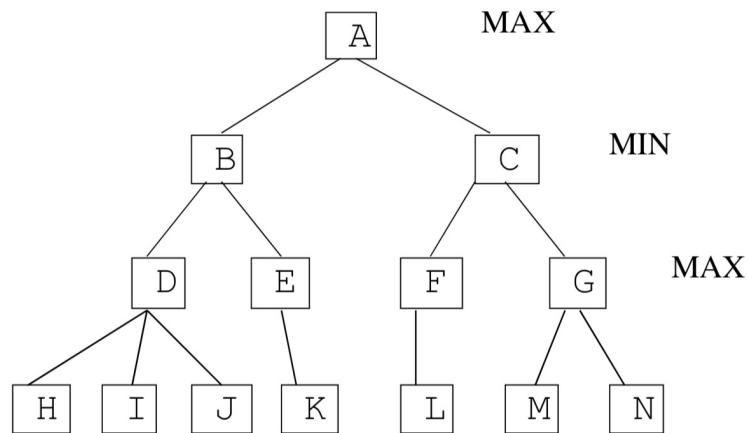
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

114

114

Jogos – Busca Competitiva



Fonte: Prof. Paulo Martins Engel/UFRGS

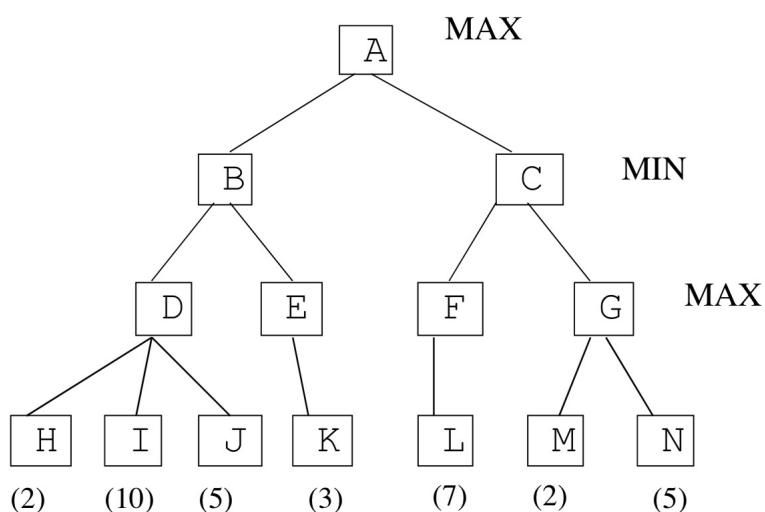
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

115

115

Jogos – Busca Competitiva



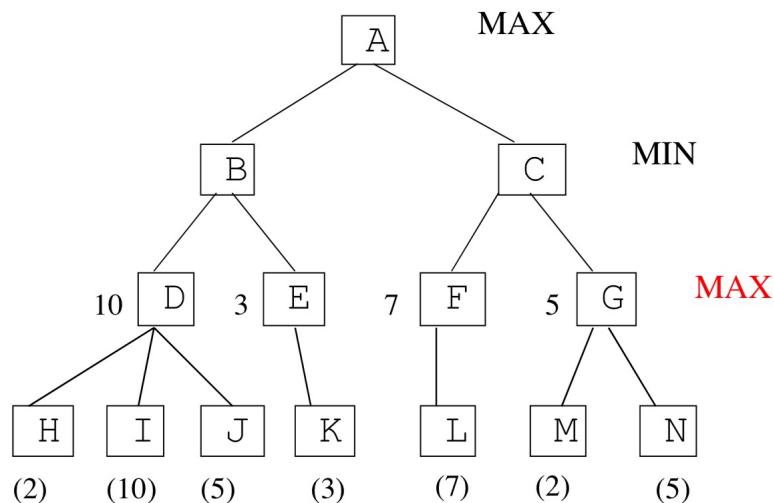
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

116

116

Jogos – Busca Competitiva



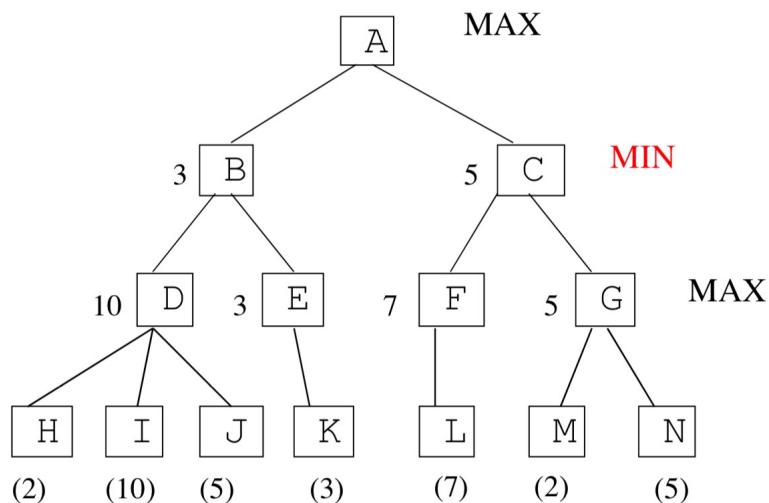
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

117

117

Jogos – Busca Competitiva



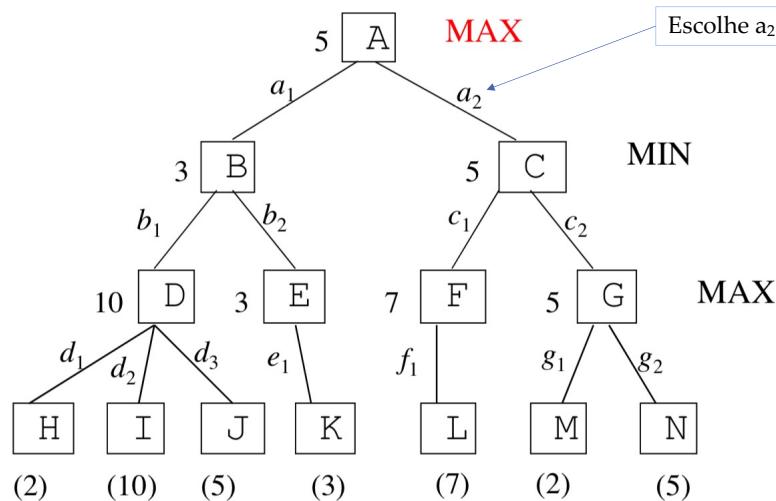
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

118

118

Jogos – Busca Competitiva



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

119

119

Jogos – Busca Competitiva

- Grande problema do MINIMAX:
 - Busca em profundidade na árvore de jogo
 - Se tem profundidade m e b movimentos válidos a cada jogada
 - Complexidade $\mathcal{O}(b^m)$
- Tempo de execução impraticável
- Deve-se usar técnicas para melhorar isso
 - Exemplo: Poda Alfa-Beta

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

120

120

Jogos – Busca Competitiva..

- Algoritmo mantém dois valores
 - Alfa: valor mais alto encontrado para MAX
 - Beta: valor mais baixo encontrado para MIN
 - São atualizados ao longo da execução
- Assim:
 - Valores menores que Alfa : nunca serão escolhidos por MAX
 - Valores maiores que Beta : nunca serão escolhidos por MIN
- Complexidade
 - Ordenação das jogadas influencia a complexidade
 - Estratégias levam a analisar uma árvore 2x mais profunda (aprox.)

CSP – Problemas de Satisfação de Restrições

CSP

- Problemas de Satisfação de Restrições (*Constraint Satisfaction Problems*)
- Conjunto de objetos que devem satisfazer uma série de restrições
 - Coloração de Grafos
 - 8-rainhas
 - Agendamento de Horários de professores
 - Transportes
 - SUDOKU
 - Escalonamento: tarefas dependentes que compartilham de recursos
- São temas de IA e de Pesquisa Operacional
- Alta complexidade
 - Aborda-se com busca heurística e busca combinatória
 - Tenta-se eliminar de uma só vez várias combinações que violam as restrições

CSP

- Um CSP é definido formalmente como uma tripla $\langle X, D, C \rangle$ onde:
 - X : conjunto de variáveis $\{X_1, X_2, \dots, X_n\}$
 - D : conjunto de domínios, um para cada variável $\{D_1, D_2, \dots, D_n\}$
 - C : conjunto de restrições que especificam as combinações possíveis de valores
- Cada Domínio D_i é composto por um conjunto de valores possíveis $\{v_1, v_2, \dots, v_k\}$ para a variável X_i
- Cada Restrição C_j é um par $\langle \text{escopo}, \text{rel} \rangle$ onde escopo é uma tupla de variáveis que participam da restrição e rel é uma relação definindo os valores que essas variáveis podem assumir
- Ex.: Se X_1 e X_2 têm domínio $\{A, B\}$, então uma possível restrição que indica que os valores devem ser diferentes pode ser:
 - $\langle (X_1, X_2), [(A, B), (B, A)] \rangle$, ou $\langle (X_1, X_2), X_1 \neq X_2 \rangle$

CSP

- Para resolver um CSP deve-se definir um espaço de estados e uma noção de solução
 - Cada estado é definido por uma **atribuição** de algumas ou todas as variáveis
 - Uma atribuição que não viola qualquer restrição é uma **atribuição consistente**
 - Uma atribuição para todas as variáveis é uma **atribuição completa** (**atribuição parcial**, caso contrário)
 - Uma **solução** para um CSP é uma atribuição **consistente e completa**

Prof. Dr. Razer A N R Montaño

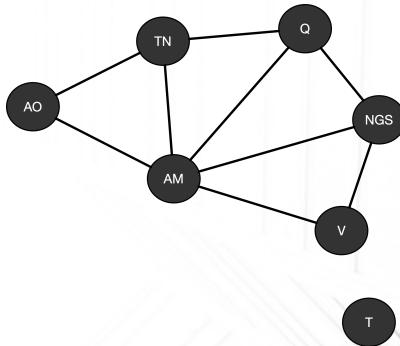
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

125

125

CSP

- Exemplo: Coloração de Mapa
 - Colorir o mapa da Austrália (seus territórios) em **Vermelho**, **Verde** e **Azul**, de modo que regiões vizinhas não tenham a mesma cor



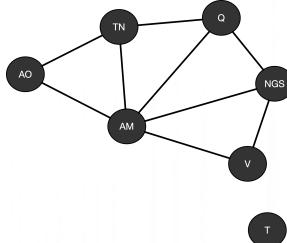
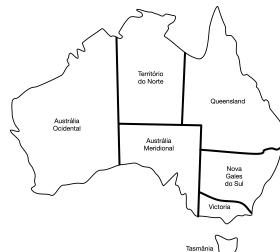
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

126

126

CSP



- Variáveis
 - $X = \{ AO, TN, Q, NGS, V, AM, T \}$
- Domínio de todas as variáveis é o mesmo
 - $D_i = \{ \text{vermelho, verde, azul} \}$
- Restrições, são aplicadas a regiões fronteiriças
 - $C = \{ AM \neq AO, AM \neq TN, AM \neq Q, AM \neq NGS, AM \neq V, AO \neq TN, TN \neq Q, Q \neq NGS, NGS \neq V \}$

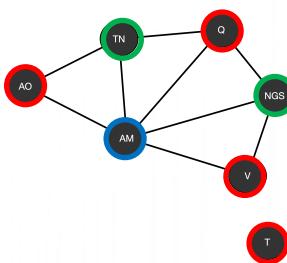
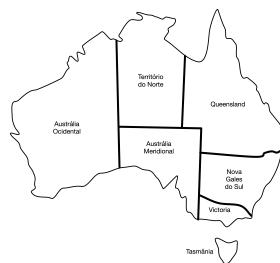
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

127

127

CSP



- Várias possíveis soluções, ex:
 - { AO = vermelho, TN = verde, Q = vermelho, NGS = verde, V = vermelho, AM = azul, T = vermelho }

Prof. Dr. Razer A N R Montaño

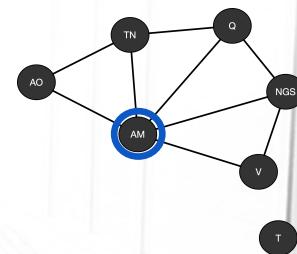
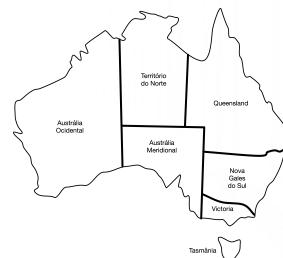
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

128

128

CSP

- O processo simples para resolução de um CSP
 - Busca no espaço de estados
 - Efetuar inferências – propagação de restrições
- As inferências diminuem os valores que uma variável pode receber, em vista do que já foi atribuído
- Ex.:
 - Ao atribuir: **AM = Azul**
 - Já é sabido que AO, TN, Q, NGS e V não podem ser Azul
 - Não precisa buscar estados em que esses valores são Azul



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

129

129

CSP..

```

ALGORITMO PesquisaComRetrocesso(csp)
    RETORNE RetrocessoRecursivo({}, csp)

ALGORITMO RetrocessoRecursivo(atribuição, csp)
    SE atribuição é completa ENTÃO
        RETORNE atribuição
    FIM SE
    var = Selecionar-Variável-Não-Atribuída(csp)
    PARA CADA valor em Valores-de-Ordem-No-Dominio(var, atribuição, csp) FAÇA
        SE valor é consistente com atribuição ENTÃO
            Adicionar {var = valor} em atribuição
            inferência = Inferência(csp, atribuição, var)
            SE inferência != Falha ENTÃO
                Adicionar inferência em atribuição
                resultado = RetrocessoRecursivo(atribuição, csp)
                SE resultado != Falha ENTÃO
                    RETORNE resultado
                FIM SE
            FIM SE
        FIM SE
        Remover { var = valor } de atribuição
    FIM PARA
  
```

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

130

130

3 Raciocínio

Agentes Lógicos
Lógica
Inferência
Planejamento
Representação do Conhecimento

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

131

131

Agentes Lógicos.

- Seres humanos sabem das coisas, e o que sabem os ajudam a fazer coisas
 - **Raciocínio** sobre **Representações** internas do conhecimento
- Em IA são **Agentes Baseados em Conhecimento**
- Os agentes já apresentados são limitados e inflexíveis
 - Ex.: Para o quebra-cabeça de 8 peças o conhecimento está embutido em uma função implementada
 - Consegue prever resultado de ações, mas não se duas peças podem ocupar o mesmo espaço
- Em CSP faz-se a atribuição de valores a determinadas variáveis
 - Um passo em direção à independência de domínio
 - O agente resolve "qualquer" problema

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

132

132

Conceitos Básicos.

- **Base de Conhecimento (KB):**
 - É o conhecimento de um agente
 - Formado por **sentenças**
- **Sentença:**
 - Representa alguma informação sobre o mundo
 - Não é escrita em PT, mas em uma linguagem específica: **linguagem de representação do conhecimento**
- **Axioma:**
 - Quando há uma sentença que é verdadeira, ié, não é derivada
- **Inferência:**
 - Uma forma de adicionar sentenças, derivando das que já estão na KB
- **Conhecimento Inicial:**
 - Uma KB de início sobre o mundo

Prof. Dr. Razer A N R Montaño

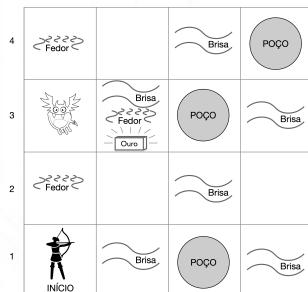
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

133

133

Mundo de Wumpus

- Objetivo: Guerreiro pegar o Ouro e sair em segurança
- Uma caverna com salas conectadas
 - Vários Poços
 - Um Tesouro
 - Em algum lugar está o Terrível Wumpus (não se move, *tks God*)
- O Guerreiro só tem uma flecha, que mata o Wumpus
- Guerreiro morre:
 - Se entrar no quadrado do Wumpus
 - Se cair em um poço
- Sensores:
 - Ao redor do Wumpus : Fedor
 - Ao redor do poço: Brisa
 - No ouro: Brilho
 - Quando caminha em uma parede: Impacto
 - Quando Wumpus morre: Grito em toda a caverna



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

134

134

Mundo de Wumpus

- Guerreiro começa em [1, 1]
- **Desconhece o conteúdo das salas**
- Guerreiro se move:
 - Para Frente
 - Girar Esquerda 90º
 - Girar Direita 90º
- Guerreiro tem ações:
 - Agarrar: pega o ouro que está no mesmo quadrado
 - Atirar: atira uma flecha para frente
 - Escalar: sair da caverna, apenas no quadrado [1, 1]
- Sempre que o guerreiro entra em um quadrado é dado um vetor de sensores:
 - [FEDOR, BRISA, BRILHO, IMPACTO, GRITO]
 - [Fedor, Brisa, Nada, Nada, Nada]

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
[1,1]	[2,1]	[3,1]	[4,1]
A			
OK			

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

135

135

Mundo de Wumpus

- Primeira conclusão do agente
 - Não tem nada no sensor
 - Salas [1,2] e [2,1] são seguras

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
[1,1]	[2,1]	[3,1]	[4,1]
A			
OK			

Marca salas seguras

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
[1,1]	[2,1]	[3,1]	[4,1]
OK			
A			
OK			

[nada, nada, nada, nada, nada]

[nada, nada, nada, nada, nada]

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

136

136

Mundo de Wumpus

- Caminha para a Sala [2, 1]
 - Percebe BRISA no sensor
 - Salas [2,2] ou [3,1] possuem um poço (P?)
 - Sala [1,1] é segura, pois estava nela

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK	P?		
[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

[nada, Brisa, nada, nada, nada]

Mundo de Wumpus

- Não pode caminhar para uma sala insegura
- Única sala segura nesse cenário é [1,2]
 - Caminha até lá

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK	P?		
[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

[nada, Brisa, nada, nada, nada]

Mundo de Wumpus

- Na sala [1,2]: Recebe FEDOR no sensor
- Vamos analisar as salas
 - Por causa do FEDOR: Wumpus está em [1,3] ou [2,2]
 - Wumpus NÃO está em [1,1], é sabidamente segura
 - Wumpus NÃO está em [2,2]: Senão o agente teria percebido FEDOR quando visitou [2,1]
 - Com certeza Wumpus está em [1,3] (W!!!)

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
W!!!			
[1,2]	[2,2]	[3,2]	[4,2]
A OK		OK	
[1,1]	[2,1]	[3,1]	[4,1]
V OK	B OK		P!!!

[Fedor, nada, nada, nada, nada]

Mundo de Wumpus

- Outra conclusão
 - A falta de Brisa em [1,2] indica que não há poço em [2,2]
 - Portanto a sala [2,2] é segura
 - Portanto a sala [3,1] tem um poço (P!!!)

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
W!!!			
[1,2]	[2,2]	[3,2]	[4,2]
A OK		OK	
[1,1]	[2,1]	[3,1]	[4,1]
V OK	B OK		P!!!

[Fedor, nada, nada, nada, nada]

Mundo de Wumpus

- No próximo passo vai para:
 - [2,2] sala segura
- Duas opções:
 - [2,3]
 - [3,2]

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
W!!!			
[1,2]	[2,2]	[3,2]	[4,2]
A OK	OK		
[1,1]	[2,1]	[3,1]	[4,1]
V OK	B OK		P!!!

Prof. Dr. Razer A N R Montaño

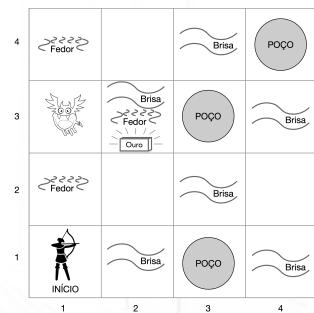
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

141

141

Mundo de Wumpus.

- Suponha que o agente chegar em [2,3]
 - Percebe Ouro, Brisa e Fedor
 - Agarra o Ouro
 - Volta pelo caminho feito
- Lembrando que ele NÃO tem conhecimento do mundo, vai inferindo conforme obtém os sensores



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

142

142

Agentes Baseados em Conhecimento

- Problema com Algoritmos de Busca???
- Conhecimento sobre estados, transições, etc está NO CÓDIGO
- É implementado para resolver AQUELE PROBLEMA ESPECÍFICO
- Não consegue deduzir outras informações

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

143

143

Agentes Baseados em Conhecimento

- Precisa-se de outro tipo de agente:
- Agentes baseados em CONHECIMENTO
 - Mantém representações internas do conhecimento
 - Operam essas representações como forma de raciocínio
 - Usam processos lógicos
 - Podem tirar conclusões
 - Não estão limitados a um tipo específico de problema

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

144

144

Agentes Baseados em Conhecimento.

- Como representar esse conhecimento???
- Usar um formalismo matemático com determinadas propriedades e métodos
 - Valores verdade
 - Inferência
 - etc
- Exemplo: Lógica Proposicional

Lógica

- Ao lado tem-se um raciocínio lógico
 - As duas primeiras linhas são **premissas**.
 - A última linha é a **conclusão**.
 - As afirmações são **sentenças**.
- Lógica teve origens na filosofia e matemática dos gregos antigos
 - Vários princípios lógicos foram difundidos por Platão (discípulo de Sócrates)
 - O primeiro estudo sistemático foi feito por Aristóteles, consolidado por seus alunos em 322 a.C. => *Organon* (trabalho fundamental para a lógica dedutiva tradicional)
 - Os **silogismos** de Aristóteles são o que chamamos de **regras de inferência**
- Inferência Lógica como processo mecânico, aplicado a uma Linguagem Formal
 - Primeira ideia: Wilhelm Leibniz (1646-1716)
 - Primeiro sistema completo, modelado sobre álgebra: George Boole (1847)
- Lógica Proposicional Moderna: Gottlob Frege (1879)

*Todos os homens são mortais.
Sócrates é homem.
Portanto, Sócrates é mortal.*

Lógica

- Demostrador de Stanhope (1753-1816):
 - 3º Conde de Stanhope
 - Primeiro dispositivo mecânico para executar inferências
 - Tratava silogismos
 - <https://history-computer.com/ModernComputer/thinkers/Stanhope.html>
- Logic Theorist (1957)
 - Newell, Shaw e Simon
 - Primeiro programa de computador para inferência lógica



Lógica - Conceitos

- **Sentenças:** afirmações acerca do problema
- **Sintaxe:** linguagem na qual as sentenças são expressas
 - Ex: $x + 1 = 10$
- **Semântica:** significado das sentenças, definindo a verdade em cada mundo possível
 - Ex: $x + 1 = 10$ é verdade num mundo em que $x = 9$, mas é falsa num mundo onde $x = 2$
- **Inferência:** processo para se chegar a uma conclusão
 - Ex: descobrir que Sócrates é mortal, somente sabendo que ele é um homem, e que homens são mortais
- **Atribuição:** são valores atrelados às variáveis
 - Ex: $x + 1 = 10$, é verdade quando a **atribuição** $x = 9$ é feita
- Em lógica clássica, toda sentença deve ser OU **Verdadeira** OU **Falsa**, não havendo um terceiro estado

Lógica - Conceitos

- **Modelo** : é um mundo estruturado com relação à verdade de sentenças
 - Ex.: No mundo possível onde $x = 9$ (M), a sentença $x + 1 = 10$ (α) é verdadeira. Portanto M é um modelo de α
 - Ex.: No mundo possível onde $x = 1$ (M'), a sentença $x + 1 = 10$ (α) é falsa. Portanto M' não é um modelo de α
 - Assim, M é um modelo de α , se α é verdade em M
 - Também diz-se que M **satisfaz** α
- Já que pode-se expressar sentenças, logicamente, deve-se ter a noção de **consequência lógica**
 - Significa que uma sentença *decorre logicamente* de outra
 - $\alpha \models \beta$
 - Indica que α tem como *consequência lógica* β
 - Formalmente: $\alpha \models \beta$ SE, E SOMENTE SE, todo modelo onde α é verdade, β também é
 - Ex.: $x = 0 \models x \cdot y = 0$

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

149

149

Lógica - Aplicação

- Deve-se aplicar estes conceitos a um problema real: Ex. Mundo de Wumpus
- A **Base de Conhecimento** (BC) são as sentenças de afirmações sobre o mundo
- Ex.
 - Em [1,1] - Agente detectou NADA
 - Em [2,1] – Agente detectou BRISA
 - BC = percepções + regras do mundo

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK	P?		
[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

[nada, Brisa, nada, nada, nada]

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

150

150

Lógica - Aplicação

- Assim, pode-se fazer perguntas, ou avaliar inferências
- Tem poço em volta da salas por onde passou [1,1] e [2,1] ???
 - 3 adjacentes: [1,2], [2,2], [3,1]
 - Pode ou não pode conter o poço = 2^3 modelos possíveis
- $BC = [1,1]$ não tem nada; [2,1] tem brisa

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK		P?	
[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

[nada, Brisa, nada, nada, nada]

Prof. Dr. Razer A N R Montaño

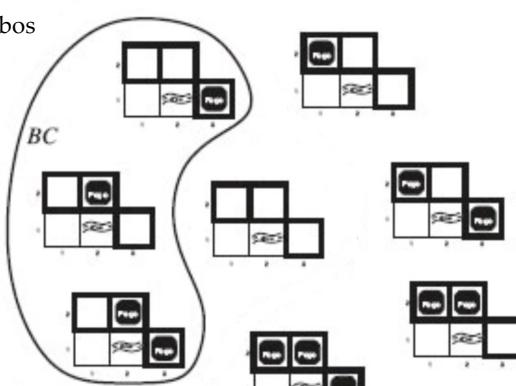
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

151

151

Lógica - Aplicação

- Tem-se somente 3 modelos válidos
 - Poço em [2,2]
 - Poço em [3,1]
 - Poço em ambos



[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK		P?	
[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

[nada, Brisa, nada, nada, nada]

Prof. Dr. Razer A N R Montaño

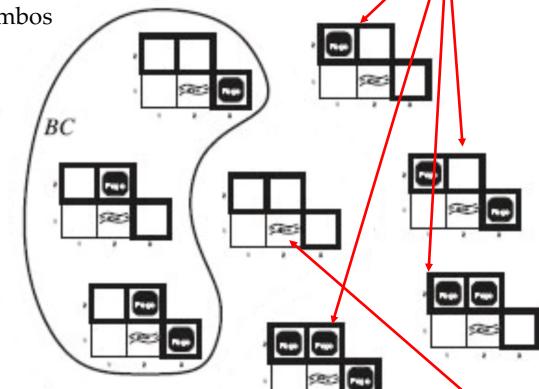
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

152

152

Lógica - Aplicação

- Tem-se somente 3 modelos válidos
 - Poço em [2,2]
 - Poço em [3,1]
 - Poço em ambos



[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK	P?		
[1,1]	[2,1]	[3,1]	[4,1]
v OK	A B OK	P?	

[nada, Brisa, nada, nada, nada]

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

153

153

Lógica - Aplicação

- Com este conhecimento
 - $BC = [1,1]$ não tem nada; $[2,1]$ tem brisa
 - Tem-se somente 3 modelos válidos
 - Poço em [2,2]
 - Poço em [3,1]
 - Poço em ambos
- Considera-se tirar 2 conclusões
 - α_1 = "Não existe poço em [1,2]"
 - α_2 = "Não existe poço em [2,2]"

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK	P?		
[1,1]	[2,1]	[3,1]	[4,1]
v OK	A B OK	P?	

[nada, Brisa, nada, nada, nada]

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

154

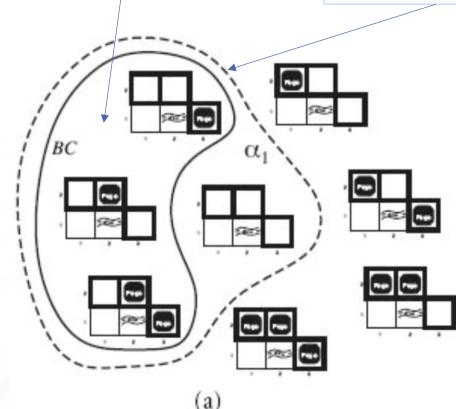
154

Lógica – Conclusão α_1

- Modelos Possíveis

$BC = [1,1]$ nada; $[2,1]$ brisa

$\alpha_1 = [1,2]$ sem poço



Prof. Dr. Razer A N R Montaño

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
[1,1]	[2,1]	[3,1]	[4,1]
v OK		P?	
	A OK		P?

[nada, Brisa, nada, nada, nada]

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

155

155

Lógica – Conclusão α_1

- Para a primeira
 - $\alpha_1 = \text{"Não existe poço em } [1,2]\text{"}$
- Observa-se que:
 - Em todo modelo que BC é verdadeiro
 - α_1 também é
- Logo, pode-se escrever
 - $BC \models \alpha_1$
- Assim, pode concluir que não há poço em $[1,2]$

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

156

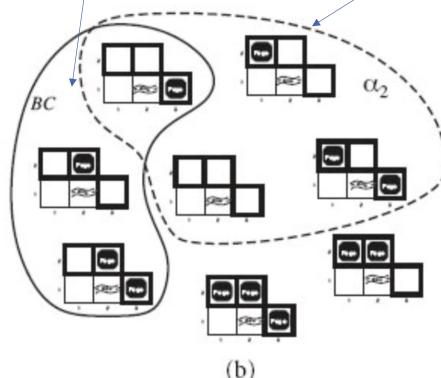
156

Lógica – Conclusão α_2

- Modelos Possíveis

$BC = [1,1]$ nada; $[2,1]$ brisa

$\alpha_2 = [2,2]$ sem poço



(b)

Prof. Dr. Razer A N R Montaño

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
[1,1]	OK	P?	
V OK	[2,1]	A B OK	P?
		[3,1]	[4,1]

[nada, Brisa, nada, nada, nada]

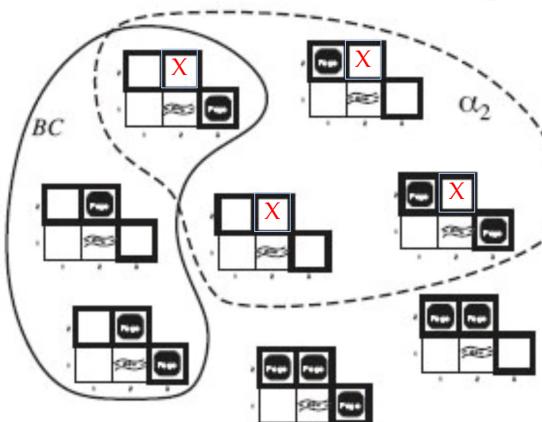
Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

157

157

Lógica – Conclusão α_2

- Para a segunda
 - $\alpha_2 = \text{"Não existe poço em } [2,2]$ "
- Observa-se que:
 - Em alguns modelos que BC é verdadeiro
 - α_2 é falsa
- Logo, pode-se escrever
 - $BC \not\models \alpha_2$
- Assim, **não pode concluir** que não há poço em $[2,2]$



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

158

158

Lógica

- Este método é conhecido como **Verificação de Modelos**
 - Enumera todos os modelos
 - Verifica em quais modelos BC é verdadeira
 - Verifica em quais modelos a conclusão que se quer é verdadeira
 - Verifica se em todos os modelos em que BC é verdadeiro a conclusão também é

Lógica.

- Um **algoritmo de inferência** é um algoritmo que usa o conhecimento (BC) para tirar conclusões
- Obviamente, enumerar todos os modelos pode não ser viável
- Pode-se **derivar** o conhecimento, mantendo a verdade
- Derivar pode ser gerar (a partir de BC) novas afirmações
 - Usam-se regras de inferência que mantém a verdade
 - Usam-se transformações conhecidas que mantém a verdade
- Pode-se escrever
 - $BC \vdash \alpha_1$
 - Que significa: α_1 é **derivável** de BC

Lógica Proposicional

- Lógica muito simples
 - Sintaxe
 - Semântica
 - Maneira de verificar a verdade de sentenças
 - Consequência lógica
- **Sintaxe**
 - Símbolos Atômicos - indivisíveis: P, Q, R, W_{1,3}, Norte
 - Podem ser Verdadeiro ou Falso (V ou F)
 - Os símbolos representam algo, Ex.: W_{1,3} indica se o Wumpus está em [1,3]
 - Composição de símbolos por meio de conectivos

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

161

161

Lógica Proposicional

- Sentenças complexas podem ser escritas com parênteses e conectivos lógicos
- Conectivos:
 - \neg Negação: $\neg P$
 - \wedge E / conjunção: $P \wedge Q$
 - \vee OU / disjunção: $P \vee Q$
 - \rightarrow Condicional / implica / Se P, Então Q: $P \rightarrow Q$. P é chamado de premissa ou antecedente, Q é chamado de conclusão ou consequente
 - \leftrightarrow Bicondicional / Se e somente Se: $P \leftrightarrow Q$

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

162

162

Lógica Proposicional

- **Semântica:** define as regras para determinação do valor verdade de uma sentença
- Sentenças Atômicas (somente um símbolo) é fácil:
 - V é verdadeiro em todo modelo, F é falso em todo modelo
 - Os símbolos (Ex. P) deve ser especificado diretamente no modelo
- Para Sentenças Complexas
 - $\neg P$: é V se, e somente se, P for F
 - $P \wedge Q$: É V se, e somente se, P e Q forem V
 - $P \vee Q$: É V se P for V, Q for V ou ambos forem V
 - $P \rightarrow Q$: É V exceto se P for V e Q for F
 - $P \leftrightarrow Q$: É V se P e Q forem ambos V, ou se P e Q forem ambos F

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

163

163

Lógica Proposicional

- A partir desta semântica consegue-se as **tabelas verdade** dos conectivos

P	Q	$\neg P$	$P \vee Q$	$P \wedge Q$	$P \rightarrow Q$	$P \leftrightarrow Q$
V	V	F	V	V	V	V
V	F	F	V	F	F	F
F	V	V	V	F	V	F
F	F	V	F	F	V	V

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

164

164

Lógica Proposicional

- A tabela verdade de expressões mais complexas decorre dos conectivos
 - Todas as possibilidades de valores verdade entre todos os símbolos
- Tamanho depende da quantidade de símbolos atômicos:
 - 1 símbolo : 2 linhas
 - 2 símbolos : 4 linhas
 - 3 símbolos : 8 linhas
 - 4 símbolos : 16 linhas
 - n símbolos: 2^n linhas
- Exemplo: $(P \vee Q) \rightarrow \neg R$
 - 3 símbolos: 8 linhas

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

165

165

Lógica Proposicional

- Exemplo: $(P \vee Q) \rightarrow \neg R$

P	Q	R	$P \vee Q$	$(P \vee Q) \rightarrow \neg R$	$\neg R$
V	V	V	V	F	F
V	V	F	V	V	V
V	F	V	V	F	F
V	F	F	V	V	V
F	V	V	V	F	F
F	V	F	V	V	V
F	F	V	F	V	F
F	F	F	F	V	V

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

166

166

Exercício.

- Fazer a tabela verdade de:
 - $(P \vee \neg P)$
 - $(P \wedge \neg P)$

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

167

167

BC Simples no Mundo de Wumpus

- Para o Mundo de Wumpus, sejam os seguintes símbolos:
 - $P_{x,y}$: V se existe Poço em $[x,y]$
 - $W_{x,y}$: V se o Wumbus está em $[x,y]$
 - $B_{x,y}$: V se tem Brisa em $[x,y]$
 - $S_{x,y}$: V se tem Fedor em $[x,y]$

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK		P?	
[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

168

168

BC Simples no Mundo de Wumpus

- Então escrevem-se as regras e percepções até o momento
- Não há Poço em [1,1]
 - $R_1 : \neg P_{1,1}$
- Tem Brisa se tem poço em um quadrado vizinho
 - $R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
 - $R_3 : B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
- Percepções
 - $R_4 : \neg B_{1,1}$
 - $R_5 : B_{2,1}$

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK	P?		

[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

BC Simples no Mundo de Wumpus

- Portanto, a BC é:
 - $R_1 : \neg P_{1,1}$
 - $R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
 - $R_3 : B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
 - $R_4 : \neg B_{1,1}$
 - $R_5 : B_{2,1}$
- Quer-se concluir
 - $\neg P_{1,2}$
 - $\neg P_{2,2}$
- Isto é:
 - $\neg P_{1,2}$ é consequência lógica de BC?
 - $\neg P_{2,2}$ é consequência lógica de BC?

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK	P?		

[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

Verificando Modelos

- No processo de verificação de modelos, faz-se a tabela verdade de todos os símbolos, inclusive as afirmações R_i : 7 variáveis = $2^7 = 128$ linhas
 - Toma-se os modelos onde todos os R_i são V
 - Verifica-se se as conclusões também são V
- Neste caso percebe-se que
 - Em somente 3 modelos BC é V
 - Nestes 3 modelos, $\neg P_{1,2}$ é V ou $P_{1,2}$ é F
 - Nestes 3 modelos, $P_{2,2}$ é V em somente 2 modelos
- Assim conclui-se que:
 - $\neg P_{1,2}$: É V, isto é, não tem poço em [1,2]
 - $\neg P_{2,2}$: É indeterminado, não há como saber se tem ou não poço em [2,2]

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK	P?		
[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

171

171

Verificando Modelos.

$B_{1,1}$	$B_{2,1}$	$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$P_{2,2}$	$P_{2,3}$	R_1	R_2	R_3	R_4	R_5
V	V	V	V	V	V	V					
V	V	V	V	V	V	F					
V	V	V	V	V	F	V					
V	V	V	V	V	F	F					
...											

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

172

172

Prova de Teoremas

- Ao invés de enumerar os modelos (n variáveis = 2^n modelos)
 - Aplicação de regras de inferência na BC
 - Aplicação de transformações na BC
- A ideia é derivar uma conclusão sem enumerar todos os modelos
- Conceitos Adicionais:
 - **Equivalência Lógica:** duas sentenças são equivalentes se são verdadeiras no mesmo conjunto de modelos
 - **Validade:** uma sentença é válida se é verdadeira em todos os modelos (tautologias)
 - Ex.: $P \vee \neg P$
 - Para quaisquer sentenças A, B , $A \models B$ se, e somente se, $A \rightarrow B$ é válida
 - **Satisfatibilidade:** uma sentença é satisfatível se for verdadeira em algum modelo
 - Para quaisquer sentenças A, B , $A \models B$ se, e somente se, $A \wedge \neg B$ não é satisfatível

Prova de Teoremas: Equivalências

Elemento neutro de \wedge : $\alpha \wedge V \equiv \alpha$;

Elemento neutro de \vee : $\alpha \vee F \equiv \alpha$;

Elemento anulador de \wedge : $\alpha \wedge F \equiv F$;

Elemento anulador de \vee : $\alpha \vee V \equiv V$;

Comutatividade de \wedge : $\alpha \wedge \beta \equiv \beta \wedge \alpha$;

Comutatividade de \vee : $\alpha \vee \beta \equiv \beta \vee \alpha$;

Associatividade de \wedge : $(\alpha \wedge \beta) \wedge \gamma \equiv \alpha \wedge (\beta \wedge \gamma)$;

Associatividade de \vee : $(\alpha \vee \beta) \vee \gamma \equiv \alpha \vee (\beta \vee \gamma)$;

Dupla negação: $\neg(\neg \alpha) \equiv \alpha$;

Contrapositiva: $\alpha \rightarrow \beta \equiv \neg \beta \rightarrow \neg \alpha$;

Condicional: $\alpha \rightarrow \beta \equiv \neg \alpha \vee \beta$;

Bicondicional: $\alpha \leftrightarrow \beta \equiv (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$;

De Morgan: $\neg(\alpha \wedge \beta) \equiv \neg \alpha \vee \neg \beta$;

De Morgan: $\neg(\alpha \vee \beta) \equiv \neg \alpha \wedge \neg \beta$;

Distributiva de \wedge sobre \vee : $\alpha \wedge (\beta \vee \gamma) \equiv (\alpha \wedge \beta) \vee (\alpha \wedge \gamma)$;

Distributiva de \vee sobre \wedge : $\alpha \vee (\beta \wedge \gamma) \equiv (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$

Prova de Teoremas: Equivalências

- Também conhecidas como Leis
- Exemplo de aplicação da **lei do condicional**

$$(p \wedge q) \rightarrow (q \rightarrow (r \vee s)) \equiv \neg(p \wedge q) \vee (q \rightarrow (r \vee s))$$

Inferência e Prova

- **Regras de Inferência:** uma regra onde dadas premissas, uma conclusão pode ser tomada de forma imediata
- Uma **Prova** é um encadeamento de conclusões, aplicações de regras de inferência ou equivalências notáveis, que levam a um objetivo desejado
- Exemplo de regra:
 - *Modus Ponens*

$$\frac{A \rightarrow B, A}{B}$$

- Exemplo:
 - A: SÓCRATES É HOMEM
 - B: SÓCRATES É MORTAL
 - A → B : Se SÓCRATES É HOMEM, então SÓCRATES É MORTAL

Regras de Inferência

Nome	Sigla	Regra
Adição	AD	$\alpha \vdash \alpha \vee \beta$
Eliminação da Conjunção	EC	$\alpha \wedge \beta \vdash \alpha$
Conjunção	CONJ	$\alpha, \beta \vdash \alpha \wedge \beta$
Absorção	ABS	$\alpha \rightarrow \beta \vdash \alpha \rightarrow (\alpha \wedge \beta)$
Modus Ponens	MP	$\alpha \rightarrow \beta, \alpha \vdash \beta$
Modus Tollens	MT	$\alpha \rightarrow \beta, \neg \beta \vdash \neg \alpha$
Silogismo Disjuntivo	SD	$\alpha \vee \beta, \neg \alpha \vdash \neg \beta \quad \text{e} \quad \alpha \vee \beta, \neg \beta \vdash \neg \alpha$
Silogismo Hipotético	SH	$\alpha \rightarrow \beta, \beta \rightarrow \gamma \vdash \alpha \rightarrow \gamma$
Dilema Construtivo	DC	$\alpha \rightarrow \beta, \gamma \rightarrow \delta, \alpha \vee \gamma \vdash \beta \vee \delta$
Dilema Destruutivo	DD	$\alpha \rightarrow \beta, \gamma \rightarrow \delta, \neg \beta \vee \neg \delta \vdash \neg \alpha \vee \neg \gamma$

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

177

177

Inferência e Prova

- Exemplo de regra:
 - *Eliminação da Conjunção (E)*

$$\frac{A \wedge B}{A}$$

- Exemplo:
 - A: Wumpus está Vivo
 - B: Wumpus está Adiante
 - Portanto, se A e B são verdadeiros, conclui-se que Wumpus está Vivo

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

178

178

Inferência e Prova

- Dado BC
 - $R_1 : \neg P_{1,1}$
 - $R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
 - $R_3 : B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
 - $R_4 : \neg B_{1,1}$
 - $R_5 : B_{2,1}$
- Quer-se provar que não tem poço em [1,2] :
 - $\neg P_{1,2}$

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK	P?		
[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

179

179

Inferência e Prova

- $R_1 : \neg P_{1,1}$
- $R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
- $R_3 : B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
- $R_4 : \neg B_{1,1}$
- $R_5 : B_{2,1}$
- Aplica-se Eliminação da bicondicional em R_2
 - $R_6 : (B_{1,1} \rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1})$

Eliminação da Bicondicional

$$\frac{P \leftrightarrow Q}{(P \rightarrow Q) \wedge (Q \rightarrow P)}$$

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

180

180

Inferência e Prova

- $R_1 : \neg P_{1,1}$
- $R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
- $R_3 : B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
- $R_4 : \neg B_{1,1}$
- $R_5 : B_{2,1}$
- $R_6 : (B_{1,1} \rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1})$
- Aplica-se Eliminação do E em R_6
 - $R_7 : (P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1}$

Eliminação do E

$$\frac{P \wedge Q}{P}$$

Inferência e Prova

- $R_1 : \neg P_{1,1}$
- $R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
- $R_3 : B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
- $R_4 : \neg B_{1,1}$
- $R_5 : B_{2,1}$
- $R_6 : (B_{1,1} \rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1})$
- $R_7 : (P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1}$
- Aplica-se Contrapositiva em R_7
 - $R_8 : \neg B_{1,1} \rightarrow \neg(P_{1,2} \vee P_{2,1})$

Contrapositiva

$$\frac{P \rightarrow Q}{\neg Q \rightarrow \neg P}$$

Inferência e Prova

- $R_1 : \neg P_{1,1}$
- $R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
- $R_3 : B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
- $R_4 : \neg B_{1,1}$
- $R_5 : B_{2,1}$
- $R_6 : (B_{1,1} \rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1})$
- $R_7 : (P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1}$
- $R_8 : \neg B_{1,1} \rightarrow \neg (P_{1,2} \vee P_{2,1})$
- Aplica-se *Modus Ponens* em R_4 e R_8
 - $R_9 : \neg (P_{1,2} \vee P_{2,1})$

Modus Ponens

$$\frac{P \rightarrow Q, P}{Q}$$

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

183

183

Inferência e Prova

- $R_1 : \neg P_{1,1}$
- $R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
- $R_3 : B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
- $R_4 : \neg B_{1,1}$
- $R_5 : B_{2,1}$
- $R_6 : (B_{1,1} \rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1})$
- $R_7 : (P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1}$
- $R_8 : \neg B_{1,1} \rightarrow \neg (P_{1,2} \vee P_{2,1})$
- $R_9 : \neg (P_{1,2} \vee P_{2,1})$
- Aplica-se *De Morgan* em R_9
 - $R_{10} : \neg P_{1,2} \wedge \neg P_{2,1}$

De Morgan

$$\frac{\neg(P \vee Q)}{\neg P \wedge \neg Q}$$

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

184

184

Inferência e Prova

- $R_1 : \neg P_{1,1}$
- $R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
- $R_3 : B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
- $R_4 : \neg B_{1,1}$
- $R_5 : B_{2,1}$
- $R_6 : (B_{1,1} \rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1})$
- $R_7 : (P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1}$
- $R_8 : \neg B_{1,1} \rightarrow \neg(P_{1,2} \vee P_{2,1})$
- $R_9 : \neg(P_{1,2} \vee P_{2,1})$
- $R_{10} : \neg P_{1,2} \wedge \neg P_{2,1}$

- Aplica-se *Eliminação do E* em R_{10}

Eliminação do E

$$\frac{P \wedge Q}{P}$$

OBJETIVO ENCONTRADO

BC Resultado

- $R_1 : \neg P_{1,1}$
- $R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
- $R_3 : B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
- $R_4 : \neg B_{1,1}$
- $R_5 : B_{2,1}$
- $R_6 : (B_{1,1} \rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1})$
- $R_7 : (P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1}$
- $R_8 : \neg B_{1,1} \rightarrow \neg(P_{1,2} \vee P_{2,1})$
- $R_9 : \neg(P_{1,2} \vee P_{2,1})$
- $R_{10} : \neg P_{1,2} \wedge \neg P_{2,1}$
- $R_{11} : \neg P_{1,2}$

Próximos objetivos

- Com o BC anterior, assumindo que o guerreiro vai até a casa [1,2]
- Precisa-se adicionar o sensor
 - $R_{12}: S_{1,2}$
 - $R_{13}: \neg S_{2,1}$
- Precisa-se adicionar as regras sobre o Wumpus
 - $R_{14}: S_{1,2} \leftrightarrow (\bar{W}_{1,3} \vee \bar{W}_{2,2})$
 - $R_{15}: S_{2,2} \leftrightarrow (\bar{W}_{1,2} \vee \bar{W}_{2,3} \vee \bar{W}_{2,1} \vee \bar{W}_{3,2})$
 - $R_{16}: S_{2,1} \leftrightarrow (\bar{W}_{1,1} \vee \bar{W}_{2,2} \vee \bar{W}_{3,1})$

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK	P?		
[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

Próximos objetivos

- Deseja-se descobrir
 - $\neg \bar{W}_{1,3}$
 - $\neg \bar{W}_{2,2}$
 - $\neg \bar{P}_{1,3}$
 - $\neg \bar{P}_{2,2}$
- Repete-se o processo para cada objetivo

[1,4]	[2,4]	[3,4]	[4,4]
[1,3]	[2,3]	[3,3]	[4,3]
[1,2]	[2,2]	[3,2]	[4,2]
OK	P?		
[1,1]	[2,1]	[3,1]	[4,1]
V OK	A B OK	P?	

Algoritmos, Como Implementar?

- Algoritmos de Busca
 - Alternativa à enumeração de modelos
 - Representar sentenças com alguma estrutura de dados
 - Representar as regras
- **Estado Inicial:** BC de início
- **Ações:** Todas as regras de inferência, que são aplicadas se a parte superior corresponder a sentenças
- **Resultado:** O resultado de uma ação é acrescentar a parte inferior da regra de inferência aplicada
- **Objetivo:** Obter a sentença que se quer provar

Problemas???

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

189

189

Aplicação

- Aplicação
 - Algoritmos e otimizações podem ser aplicados a vários problemas
 - Isso porque muitos problemas podem ser transformados para SAT ou Prova de Teoremas
 - Ex: Cardápio da Sra Montagny (<https://br.spoj.com/problems/CARDAPIO/>)
 - Sra Montagny quer realizar uma festa e verificar se é possível montar um cardápio que satisfaça ao menos um desejo de cada convidado da festa.
 - Cada convidado irá fazer 2 pedidos, contendo sua preferência de comida na festa, podendo ser uma das três hipóteses:
 - Duas comidas que ele gostaria de ver no cardápio;
 - Uma comida que gostaria e uma que não gostaria de ver no cardápio;
 - Duas que não gostaria de ver no cardápio.

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

190

190

Além.

- Algoritmo muito usado
 - Resolução
 - Aplicado somente a bases na forma normal conjuntiva (conjunção de disjunções)
- Cláusulas de Horn
- Lógica de Primeira Ordem
 - $\forall x \exists y \text{ Pessoa}(x) \rightarrow (\text{Pessoa}(y) \wedge \text{AlmaGemea}(y, x))$
- Lógicas não-clássicas
 - Tempo
 - Intenção
 - Crença
 - Recursos

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

191

191

Planejamento Clássico

- Elaboração de um plano de ações para atingir determinados objetivos
- Agente de Resolução de Problemas
 - Só trata estados atômicos
 - Depende de boas heurísticas
 - Conhecimento implementado no código
- Agente Baseado em Inferência
 - Muitos estados e muitas ações
 - Explosão combinatória

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

192

192

Planejamento Clássico

- Usar representações mais adequadas
 - Exemplo: PDDL
- **PDDL** (*Planning Domain Description Language*)
 - Derivada do **STRIPS** (*Stanford Research Institute Problem Solver*)
 - Representação simples de estados, ações, pré-condições, efeitos de ações
- **Fluente**: uma condição que pode mudar ao longo do tempo
 - Exemplo: **Em(Caminhão₁, CWB)** – indica que Caminhão₁ está em Curitiba
- Cada estado:
 - Conjunção de fluentes
 - Exemplo: **Em(Caminhão₁, SãoPaulo) \wedge Em(Caminhão₂, Curitiba)**
 - Nos estados não se permitem fluentes não instanciados: Ex: **Em(X, Y)**
- Usa a **Suposição do Mundo Fechado**
 - O que não está afirmado é falso

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

193

193

Planejamento Clássico

- **Ações**
 - Esquemas que descrevem o que muda com a aplicação de uma ação
 - Assume-se que a maioria das coisas permanece inalterada
 - O que permanece inalterado não é mencionado
 - Descreve as pré-condições para a aplicação de uma ação
- Exemplo de ação

AÇÃO (Voar(p, de, para)),

PRECOND: $\text{Em}(p, \text{de}) \wedge \text{Avião}(p) \wedge \text{Aeroporto}(\text{de}) \wedge \text{Aeroporto}(\text{para})$

EFEITO: $\neg\text{Em}(p, \text{de}) \wedge \text{Em}(p, \text{para})$

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada
SEPT / UFPR

194

194

97

Planejamento Clássico

- Os esquemas, no tratamento dos problemas, são instanciados (valores atribuídos)
 - Exemplo, Avião ABC10 deve voar de CWB para GRU:

AÇÃO (**Voar(ABC10, CWB, GRU)**),

PRECOND: **Em(ABC10, CWB) \wedge Avião(ABC10) \wedge Aeroporto(CWB) \wedge Aeroporto(GRU)**

EFEITO: **\neg Em(ABC10, CWB) \wedge Em(ABC10, GRU)**

- Uma ação pode ser executada no estado S se sua pré-condição for consequência lógica de S
 - Pré-condição: conjunção de literais (ou afirmações simples)
 - Consequência lógica pode ser vista como: todo literal positivo está em S e todo literal negativo não está em S

Planejamento Clássico

- **Resultado da aplicação** de uma ação no estado s
 - Um estado s'
 - Formado pelos fluentes de s
 - Remove-se os fluentes negativos dos efeitos: lista de exclusão
 - Adiciona-se os fluentes positivos dos efeitos: lista de adição
- Exemplo:

AÇÃO (**Voar(ABC10, CWB, GRU)**),

PRECOND: **Em(ABC10, CWB) \wedge Avião(ABC10) \wedge Aeroporto(CWB) \wedge Aeroporto(GRU)**

EFEITO: **\neg Em(ABC10, CWB) \wedge Em(ABC10, GRU)**

- Lista de Exclusão: **\neg Em(ABC10, CWB)**
- Lista de Adição: **Em(ABC10, GRU)**

Planejamento Clássico

- Estado inicial
 - Conjunto de fluentes instanciados
- Objetivo
 - Conjunto de fluentes, que podem conter variáveis
 - Exemplo: $\text{Em}(p, \text{CGH}) \wedge \text{Avião}(p)$
- Problema é resolvido:
 - É encontrada uma sequência de ações que gera um estado que satisfaz o objetivo
 - Exemplos:
 - $\text{Em}(\text{XY1}, \text{CGH}) \wedge \text{Avião}(\text{XY1})$
 - $\text{Em}(\text{TT5}, \text{CGH}) \wedge \text{Avião}(\text{TT5}) \wedge \text{Avião}(\text{J7}) \wedge \text{Em}(\text{J7}, \text{CWB})$

Planejamento Clássico

- Exemplo de Problema: Mundo dos Blocos
- ```

INÍCIO (Sobre(A, Mesa) \wedge Sobre (B, Mesa) \wedge Sobre (C, A) \wedge
 Bloco(A) \wedge Bloco(B) \wedge Bloco(C))

OBJETIVO (Sobre(A, B) \wedge Sobre(B, C))

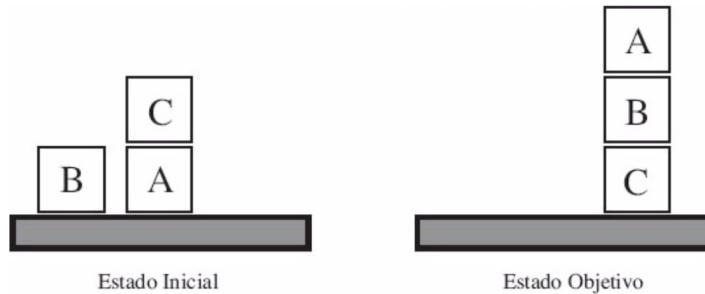
AÇÃO (Mover(b, x, y)),
PRECOND: Sobre(b, x) \wedge Livre(b) \wedge Livre(y) \wedge Bloco(b) \wedge
 Bloco(y) \wedge (b ≠ x) \wedge (b ≠ y) \wedge (x ≠ y),
EFEITO: Sobre(b, y) \wedge Livre(x) \wedge ¬Sobre(b, x) \wedge ¬Livre(y)

AÇÃO (MoverParaMesa(b, x)),
PRECOND: Sobre(b, x) \wedge Livre(b) \wedge Bloco(b) \wedge (b ≠ x),
EFEITO: Sobre(b, Mesa) \wedge Livre(x) \wedge ¬Sobre(b, x)

```

## Planejamento Clássico

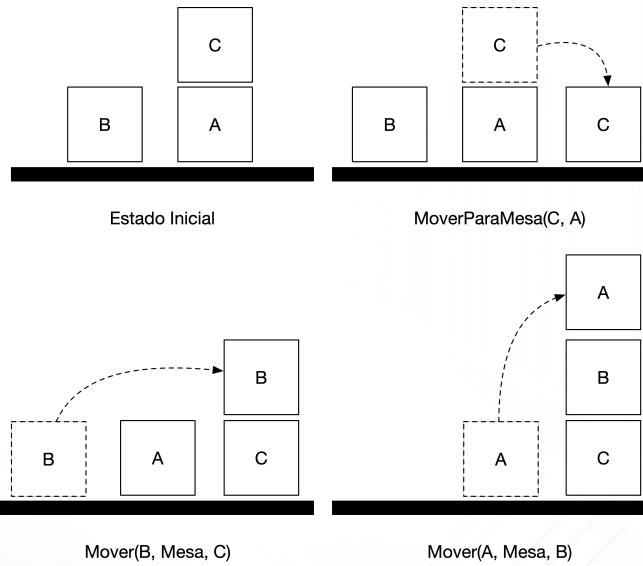
**INÍCIO** ( Sobre(A, Mesa)  $\wedge$  Sobre (B, Mesa)  $\wedge$  Sobre (C, A)  $\wedge$   
 Bloco(A)  $\wedge$  Bloco(B)  $\wedge$  Bloco(C) )  
**OBJETIVO** ( Sobre(A, B)  $\wedge$  Sobre (B, C) )



## Planejamento Clássico

- Sequência de ações que resolve o problema
- **MoverParaMesa (C, A)**
- **Mover (B, Mesa, C)**
- **Mover (A, Mesa, B)**

## Planejamento Clássico



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

201

201

## Planejamento Clássico.

- Para sair do Estado Inicial e chegar no Estado Objetivo
  - Quais ações e em que ordem?
- Várias estratégias
  - Busca para frente
  - Busca para trás
  - Heurísticas: com um algoritmo de busca A\*
  - Grafos de Planejamento
  - Tradução para SAT
  - Etc
- Planejamento não-clássico
  - Tempo
  - Recursos
  - Escalonamento

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

202

202

## Planejamento Hierárquico

- Espaço de estados MUITO grande
- Separar os estados em níveis diferentes de abstração
  - Decomposição hierárquica
- Exemplo: Viagem para as Maldivas
  - Ir ao aeroporto de Curitiba;
  - Pegar o Voo XY6 para Maldivas;
  - Aproveitar as férias durante sete semanas;
  - Ir ao aeroporto de Maldivas;
  - Pegar o Voo AB5 para Curitiba;
  - Ir para casa.

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

203

203

## Planejamento Hierárquico.

- Ação
  - Ir ao aeroporto de Curitiba;
- Envolve
  - Dirigir até o estacionamento do aeroporto de Curitiba;
  - Estacionar;
  - Pegar o *transfer*;
  - Entrar no terminal.
- Cada ação pode ainda ser decomposta
- Faz-se planejamento nos diversos níveis

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

204

204

## Representação do Conhecimento

- Como representar os fatos do mundo
- Como representar a base de conhecimento
- Domínios complexos, do mundo real
  - Compras na Internet
  - Dirigir um Carro
  - Etc
- Conceitos: Eventos, Tempo, Objetos Físicos, Crenças, etc
- Obviamente é impossível representar TUDO
  - Conceitos superiores
  - São "especializados" depois
- Ontologia:
  - Descrição de conceitos e seus relacionamentos.
  - Uma especificação formal e explícita de uma conceitualização compartilhada
  - Permitir Compartilhamento e Reutilização do conhecimento

Prof. Dr. Razer A N R Montaño

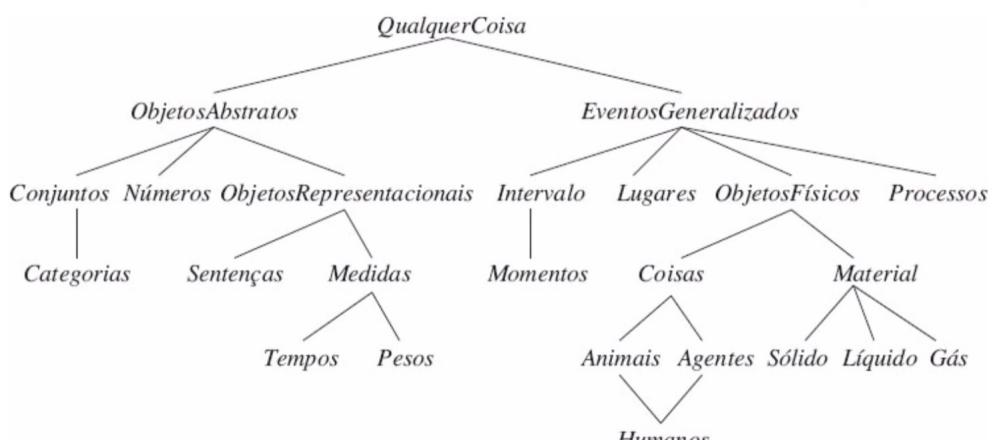
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

205

205

## Representação do Conhecimento

- Ontologia Superior



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

206

206

## Representação do Conhecimento

- Organização dos objetos em **categorias**
  - Grande parte do raciocínio acontece em nível de categoria
  - Conceito de **subcategoria**: Bolas e BolasDeBasquete
  - Conceito de **herança**: Fruta é um Alimento
  - As subclasses organizam as categorias em uma **taxonomia**

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

207

207

## Representação do Conhecimento

- Representa-se com alguma linguagem: Ex, **Lógica de Primeira Ordem**
  - Um objeto é um elemento de uma categoria  
 $BB9 \in BolasDeBasquete$
  - Uma categoria é uma subclasse de outra categoria  
 $BolasDeBasquete \subset Bolas$
  - Todos os elementos de uma categoria têm algumas propriedades  
 $(x \in BolasDeBasquete) \Rightarrow Esférica(x)$
  - Elementos de uma categoria podem ser reconhecidos por algumas propriedades  
 $Laranha(x) \wedge Redonda(x) \wedge Diâmetro(x)=23,75cm \wedge x \in Bolas \wedge x \in BolasDeBasquete$
  - Uma categoria é um conjunto que tem algumas propriedades  
 $Cães \in EspéciesDomesticadas$
  - Exceções são tratadas de outras formas (Ex. bola furada não é esférica)

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

208

208

104

## Representação do Conhecimento

- Composição
  - Representar que um objeto faz parte de outro
- Relação ParteDe(x, y)
 

*ParteDe(Paraná, Brasil)*

*ParteDe(Brasil, AméricaDoSul)*
- Relação é Reflexiva e Transitiva
 

$(\text{ParteDe}(x, y) \wedge \text{ParteDe}(y, z)) \Rightarrow \text{ParteDe}(x, z)$

*ParteDe(x, x)*

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

209

209

## Representação do Conhecimento

- Objetos compostos são caracterizados por relações estruturais entre as partes
 

$Bípede(a) \Rightarrow \exists l_1, l_2, b, \text{Perna}(l_1) \wedge \text{Perna}(l_2) \wedge \text{Corpo}(b) \wedge$

$\text{ParteDe}(l_1, a) \wedge \text{ParteDe}(l_2, a) \wedge \text{ParteDe}(b, a) \wedge$

$\text{Presa}(l_1, b) \wedge \text{Presa}(l_2, b) \wedge$

$l_1 \neq l_2 \wedge [\forall l_3, (\text{Perna}(l_3) \wedge \text{ParteDe}(l_3, a)) \Rightarrow (l_3 = l_1 \vee l_3 = l_2)]$
- Perceber os mecanismos para excluir a possibilidade de uma terceira perna

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

210

210

## Representação do Conhecimento

- Tratamento de Eventos
- Somente indicar  
 $Em(Razer, Curitiba)$
- Não significa que é verdade, deve-se usar um predicado para indicar um tempo que é verdade no tempo  $t$   
 $T( Em(Razer, Curitiba), t )$
- Pode-se usar outro formalismo: **cálculo de eventos**

## Representação do Conhecimento

- No cálculo de eventos usam-se outros predicados para indicar os eventos
  - $T(f, t) : f$  é verdadeiro no tempo  $t$
  - $Acontece(e, i) :$  Evento  $e$  acontece no intervalo de tempo  $i$
  - $Inicia(e, f, t) :$  Evento  $e$  faz com que  $f$  passe a valer no instante  $t$
  - $Termina(e, f, t) :$  Evento  $e$  faz com que  $f$  deixe de valer no instante  $t$
  - $Cortado(f, i) : f$  deixa de valer em algum ponto durante o intervalo de tempo  $i$
  - $Restaurado(f, i) : f$  passa a valer em algum ponto durante o intervalo de tempo  $i$

## Representação do Conhecimento.

- Exemplo, no Mundo de Wumpus, só recebe uma flecha no início e perde a flecha ao atirar
 
$$\text{Inicia}(e, \text{TemFlecha}(a), t) \Leftrightarrow e = \text{Início}$$

$$\text{Termina}(e, \text{TemFlecha}(a), t) \Leftrightarrow e \in \text{AtoDeAtirar}(a)$$
- Deve-se estender o cálculos de eventos para tratar outros tipos de eventos
  - Simultâneos (duas crianças em uma gangorra)
  - Exógenos (vento soprando e alterando a localização de um objeto)
  - Contínuos (nível de água de uma banheira sempre crescendo)
  - etc

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

213

213

## Eventos Mentais

- Atitudes Proposicionais
  - Acreditar
  - Saber
  - Querer
  - Pretender
  - Informar
- Muitos problemas com lógica clássica: Usa-se **lógica modal**
  - Lógica Clássica: "P é verdadeiro"
  - Lógica Modal: "A sabe P"
- Representa-se com operadores modais
  - A sabe P :  $K_A P$

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

214

214

## Lógica Modal.

- Formalmente, tem-se a noção de **mundos possíveis**
  - Mundos onde as sentenças são verdadeiras ou falsas
  - Os mundos são acessíveis (relação de acessibilidade)
  - $\Box P$  : necessário  $P$  / sabe  $P$  :  $P$  é verdadeiro em todos os mundos acessíveis
  - $\Diamond P$  : possível  $P$  / acredita  $P$  :  $P$  é verdadeiro em pelo menos um mundo acessível
- Consegue-se criar sistemas formais para expressar:
  - Tempo
  - Obrigatório / Permitido
  - Acredita / Sabe
- Possui regras de inferência e sistemas de dedução, para tirar conclusões

## Outros Tipos de Raciocínio

- **Raciocínio Default** : senso comum
- Lógica é Monotônica
  - Adição de novas afirmações não altera as afirmações provadas
- Raciocínio humano Não é monotônico
- Dois formalismo:
  - Lógica Default
  - Circunscrição

## Outros Tipos de Raciocínio..

- Incerteza: raciocínio probabilístico
- Tempo
- Incerteza+Tempo
- Tomada de Decisões

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT/UFPR

217

217

## Aprendizado

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT/UFPR

218

218

109

## Aprendizado

- Um agente está aprendendo se melhorar seu desempenho em tarefas futuras
- Tomada de decisões a partir de experiências acumuladas
- **Indução:** uma forma de inferência lógica que permite obter conclusões genéricas sobre um conjunto particular de exemplos
- Assim, usam-se exemplos fornecidos para aprender
- **Aprendizado Indutivo:** raciocínio sobre exemplos fornecidos

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

219

219

## Aprendizado

- **Supervisionado:** é fornecido ao algoritmo de aprendizado (ou **indutor**) exemplos rotulados, ié, com resultados conhecidos (classificados ou com valores de resultado)
  - **Classificação:** Para rótulos discretos, categorias
  - **Regressão:** Para rótulos contínuos, valores reais
- **Não-Supervisionado:** os exemplos são fornecidos e o algoritmo tenta agrupá-los ou relacioná-los de alguma maneira. Após, é necessária uma análise para determinar o significado do agrupamento/relação
  - **Agrupamento/Cluster:** Tenta agrupar os exemplos em grupos "parecidos"
  - **Associação:** Tenta encontrar relacionamento entre os dados
- **Por Reforço:** sem informações prévias. Cada ação leva a uma recompensa ou punição, que direciona o algoritmo.

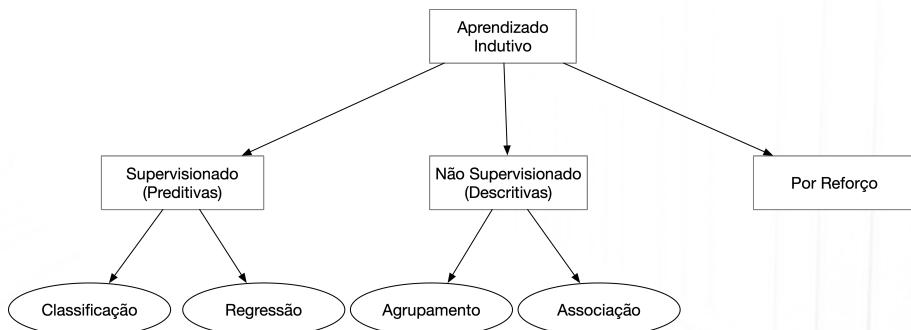
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

220

220

## Aprendizado



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

221

221

## Aprendizado

- Paradigmas
  - **Simbólico:** representações simbólicas dos conceitos. Ex Árvores de Decisão
  - **Estatístico:** modelos estatísticos para encontrar uma boa aproximação. Ex Bayesiano, SVM
  - **Baseado em Exemplos:** usam exemplos armazenados para compará-los e fazer a aproximação. Ex K-NN
  - **Conexionista:** redes de componentes conectados, fazendo uma analogia ao mecanismo biológico do sistema nervoso. Ex Redes Neurais
  - **Evolutivo:** derivado do modelo biológico de aprendizado, usando evolução de uma população de elementos. Ex Algoritmos Genéticos

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

222

222

## Exemplos de Problemas

- Reconhecimento de Placas de Carros
- Reconhecimento de Pessoas
- Reconhecimento de Letra Escrita
- Análise de Fidelidade de Clientes em Campanhas de Marketing
- Análise de Imagens de Satélite
- Análise de Compras para Recomendação de Produtos
- Predição de Volume de Madeira em uma Floresta Plantada
- Etc

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

223

223

## Processo

- **Descoberta de Conhecimento em Bases de Dados - KDD**
  - Bases de Dados muito grandes
  - Big Data
  - Mineração de Dados
- **Mineração de Dados:** processo de identificação de padrões válidos, novos, potencialmente úteis e comprehensíveis embutidos nos dados
- Etapas da MD:
  - **Identificação do Problema**
  - **Pré-processamento**
  - **Extração dos padrões:** onde os algoritmos são efetivamente aplicados
  - **Pós-processamento**
  - **Utilização do conhecimento**

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

224

224

## Aprendizado

- Para aprendizado supervisionado
- Dada uma base de dados observada e rotulada
- Separa-se (de alguma forma) em base de treinamento e base de teste (disjuntas)
  - **Base de Treinamento:** um algoritmo usa estes dados para treinar
  - **Base de Teste:** o algoritmo usa para avaliar se seu aprendizado é bom, a partir de métricas estatísticas
- Várias formas de separar:
  - **Holdout :** Porções fixas, exemplo 70% (treinamento), 30% (teste)
  - **Cross-validation :** Várias rodadas de aprendizado, a cada rodada parte da base é de teste e o resto treinamento

## Aprendizado

- Escolher o algoritmo não é tarefa simples
  - Todos são parametrizáveis
  - Pode ser necessária a execução de várias iterações
  - Pode ser necessária a execução de vários algoritmos diferentes
- Deve-se ter um método de avaliação dos algoritmos
  - Testes estatísticos
- Em aprendizado supervisionado
  - Tem-se os dados e seus rótulos
  - Usando uma base de testes, aplica-se uma métrica, por exemplo, Coeficiente de Correlação de Pearson

## Aprendizado.

- Coeficiente de Correlação de Pearson
  - Aplicado sobre dois conjuntos de dados
  - $= 1$  : correlação positiva perfeita
  - $= -1$  : correlação negativa perfeita, ié, se uma aumenta a outra diminui
  - $= 0$  : não dependem uma da outra
  - Assim, quanto mais perto de 1.0, mais dependentes são os dados
- Assim, dado os resultados do conjunto de teste (já conhecidos) e o resultado da aplicação do algoritmo treinado com o conjunto de teste
  - Aplica-se uma métrica, ex. Coeficiente de Correlação de Pearson
  - Se o valor for  $> 0.9$ , forte correlação, portanto o algoritmo treinado gerou resultados extremamente próximos do observado
- Outras métricas
  - RMSE : *root mean square error*
  - MAE: *mean absolute error*
  - etc

Prof. Dr. Razer A N R Montaño

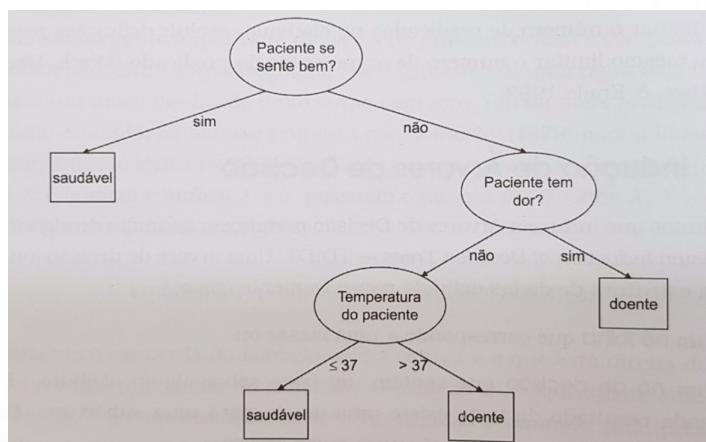
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

227

227

## Árvores de Decisão

- É uma árvore contendo dois tipos de nós
  - **Nó Folha:** corresponde a um resultando (ou uma classe ou um valor)
  - **Nó de Decisão:** que contém algum teste sobre algum atributo



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

228

228

## Árvores de Decisão

- Árvore é construída a partir dos dados de treinamento
- Uma vez treinada deve ser testada
- Os dados de teste são entrados
  - Os resultados são comparados: resultados observados vs resultados preditos
  - Assim, mede-se a qualidade da árvore treinada
- Se a qualidade for boa, pode ser usada em dados reais

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

229

229

## Construir uma Árvore de Decisão

---

### Algoritmo 1 Algoritmo de Indução de Árvores

---

```

if critério de parada(D) = Verdadeiro then
 return Um nó folha rotulado com a constante que minimiza a função perda
end if
Escolha o atributo que maximiza o critério de divisão em D
for all partição dos exemplos D_i baseado nos valores do atributo escolhido do
 Induz uma subárvore $Arvore = GeraArvore(D_i)$
end for
return Árvore contendo um nó de decisão baseado no atributo escolhido, e descendentes
 $Arvore_i$

```

---

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

230

230

## Construir uma Árvore de Decisão

- Escolha do atributo
  - Aleatório
  - Menos Valores: com menor quantidade de valores possíveis
  - Mais Valores: com maior quantidade de valores possíveis
  - Ganho Máximo: que maximiza o ganho de informação, dando um menor tamanho de sub-árvores
  - Índices (Gini, de Ganho, etc)

Prof. Dr. Razer A N R Montaño

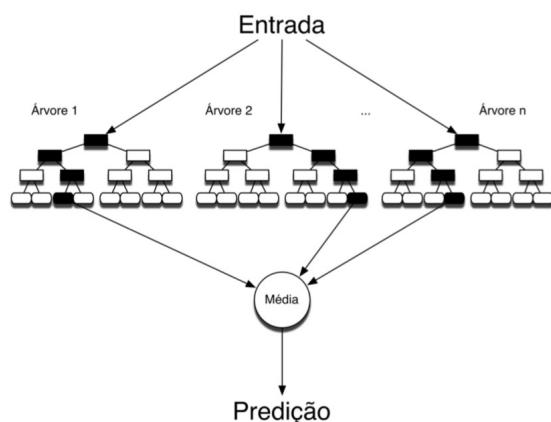
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

231

231

## Floresta.

- Já foi observado que várias árvores, treinadas de forma aleatória, podem dar resultados diferentes
- Se os resultados individuais forem combinados, o resultado final é melhor
- *Bagging*
- Floresta Aleatória (Random Forest)



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

232

232

## Redes Neurais

- Primeiros modelos em 1943 – McCulloch e Pitts
- Modelos matemáticos que se assemelham às estruturas neurais biológicas
- Processo de aprendizado interativo, melhorando seu desempenho a cada passo
  - Aprendizado é feito ajustando valores internos : pesos
- Uma rede formada por três camadas
  - Camada de Entrada
  - Camada Oculta
  - Camada de Saída
- Cada camada possui uma quantidade de nós ou neurônios
  - Cada neurônio é uma unidade de processamento

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

233

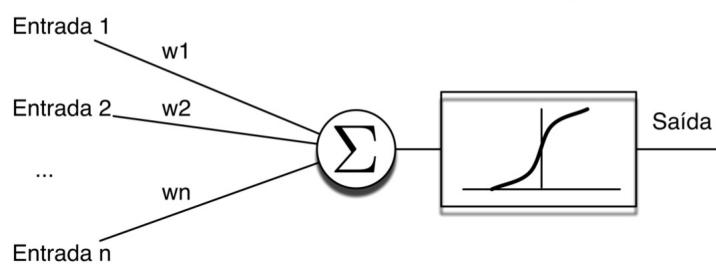
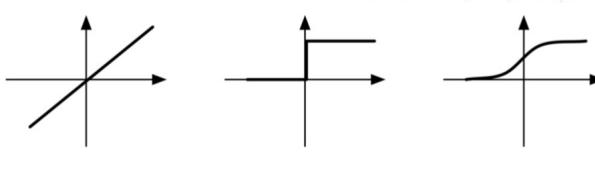
233

## Redes Neurais

- Um neurônio processa suas entradas e dá uma saída

$$u = \sum_{i=1}^m x_i \times w_i$$

$$f'(x) = f_a(u)$$

Função de Ativação  $f_a()$ 

Prof. Dr. Razer A N R Montaño

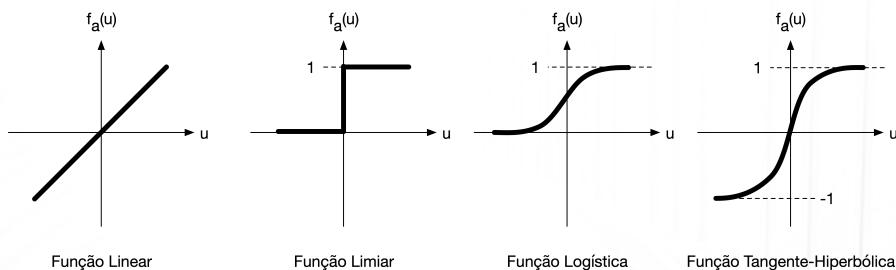
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

234

234

## Redes Neurais

- Funções de Ativação



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

235

235

## Redes Neurais

- Multicamadas
  - Vários neurônios interligados
  - Várias camadas
- Em geral
  - **Camada de Entrada:** Valores são apresentados à rede
  - **Camada de Saída:** Um ou mais neurônios, dão o resultado da rede
  - **Camadas Ocultas:** Várias camadas de neurônios

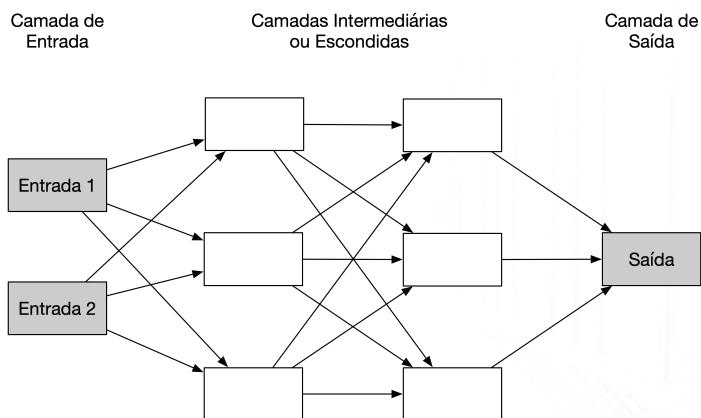
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

236

236

## Redes Neurais



Prof. Dr. Razer A N R Montaño

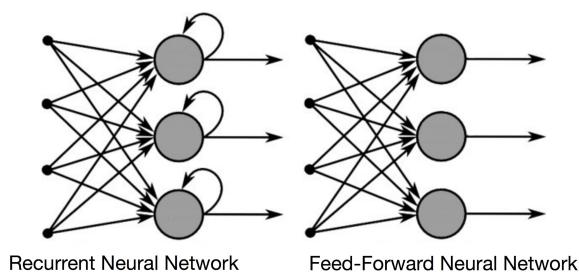
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

237

237

## Redes Neurais - Tipos

- Tipos de Redes Neurais, conforme suas conexões
  - *Feed-forward*: conexões seguem o fluxo para frente
  - **Recorrentes**: há retroalimentação



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

238

238

## Redes Neurais

- Treinamento é feito em duas fases
- Para Frente:
  - Objeto é apresentado à rede
  - Neurônios calculam seus valores, ponderações, e  $f(x)$  dá a saída
  - Até que todos os neurônios sejam calculados
  - Diferença entre o resultado computado e o esperado é o erro da rede
- Para Trás:
  - Erro cometido é usado para ajustar os pesos dos neurônios
  - Quanto cada neurônio influencia no erro total da rede.
  - Derivada parcial do erro total da rede com respeito a  $w_i$  ( $\partial E_{\text{total}} / \partial w_i$ )
  - Para cada peso  $w_i$ , calcula-se o novo valor aplicando-se a fórmula:
$$w_i' = w_i - \eta * \frac{\partial E_{\text{total}}}{\partial w_i}$$
- onde  $\eta$  é a taxa de aprendizado, um fator multiplicativo aplicado à rede.

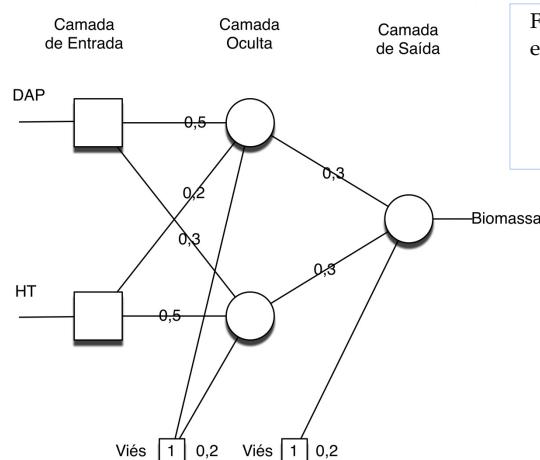
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

239

239

## Redes Neurais



Prof. Dr. Razer A N R Montaño

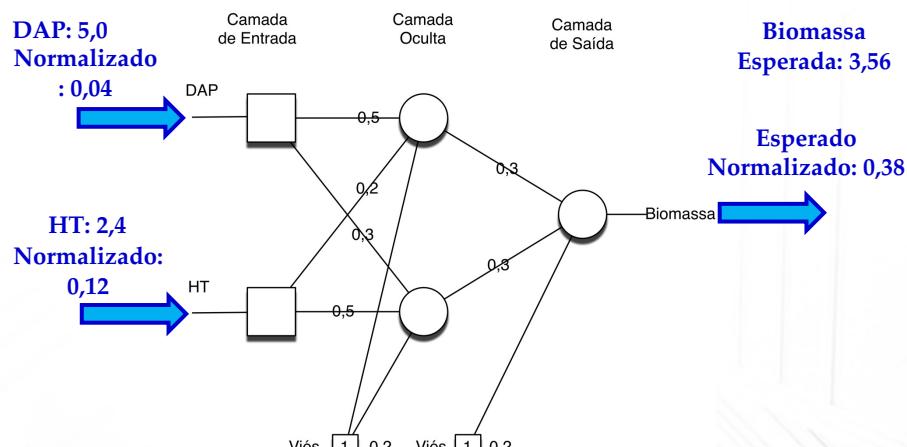
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

240

240

120

## Redes Neurais



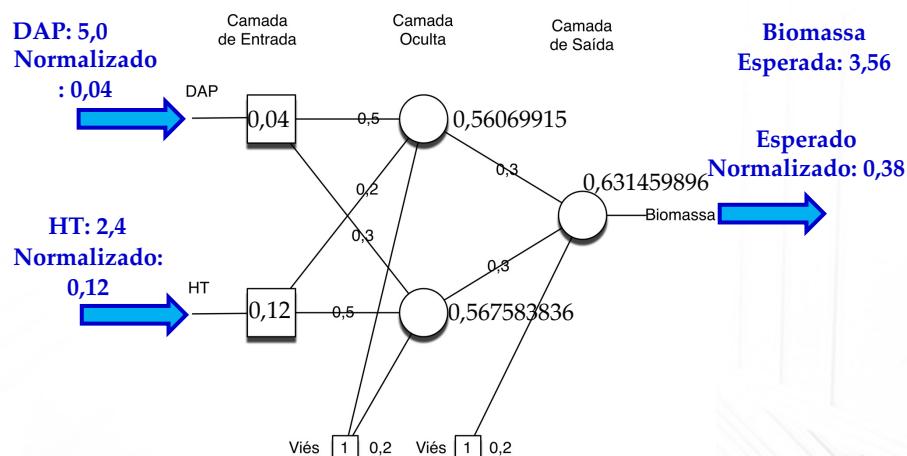
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

241

241

## Redes Neurais



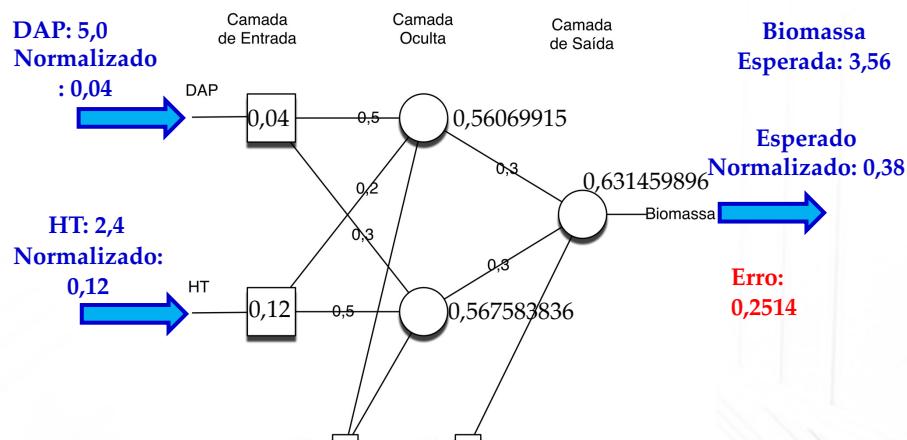
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

242

242

## Redes Neurais



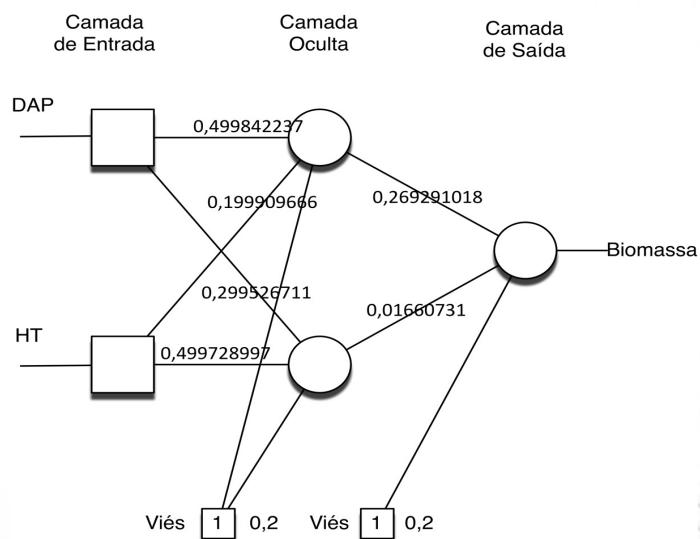
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

243

243

## Redes Neurais



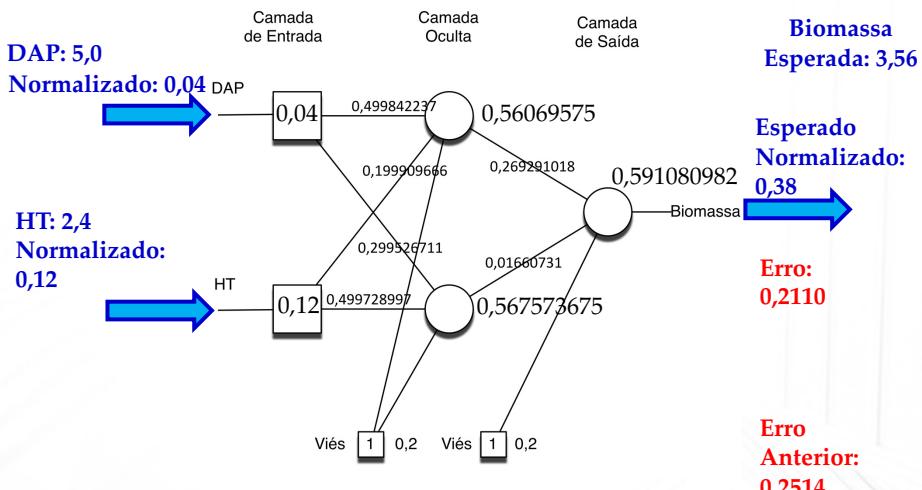
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

244

244

## Redes Neurais



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

245

245

## Redes Neurais.

- Altamente parametrizável:
  - Quantidade de camadas escondidas
  - Quantidade de neurônios nas camadas escondidas
  - Taxa de Aprendizado
  - Momentum
  - Quantidade de épocas (vezes que o conjunto de dados é apresentado à rede)
  - etc

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

246

246

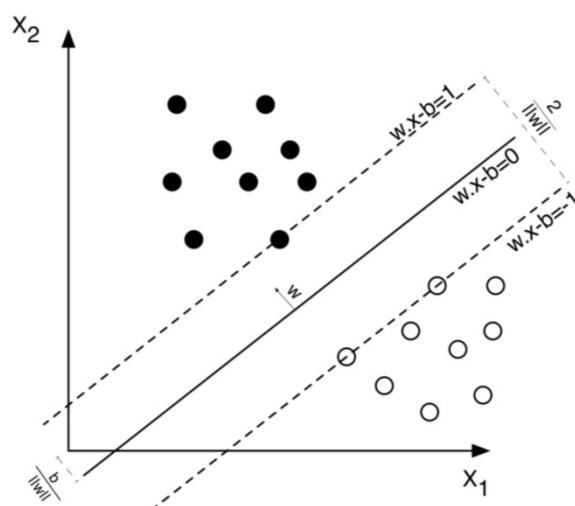
## SVM

- SVM : Máquinas de Vetores de Suporte
- Aprendizado estatístico, Vapnik 1971
- Para Classificação, encontrar um hiperplano que separa os objetos das classes

$$h(x) = w \times x + b$$

- Onde  $w \times x$  é o produto escalar entre os vetores  $w$  e  $x$

## SVM



## SVM

- Difícil encontrar aplicações com dados linearmente separáveis
  - Variável de folga  $\xi_i$
  - Permite alguns erros de classificação
- SVMs com Margens Suaves
- Constante C : peso à minimização dos erros em relação à minimização da complexidade do modelo.

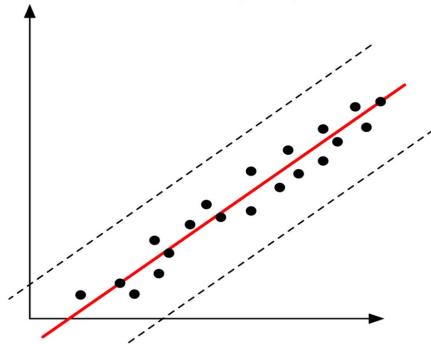
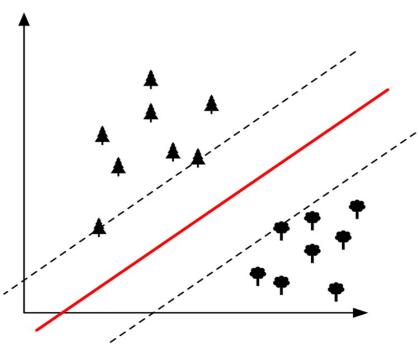
$$\text{Maximizar}_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j (x_i x_j)$$

- com as restrições:

$$\begin{cases} 0 \leq \alpha_i \leq C, \forall i = 1, \dots, n \\ \sum_{i=1}^n \alpha_i y_i = 0 \end{cases}$$

## SVM

- Problemas Lineares



## SVM

- Para problemas não-lineares deve-se aplicar funções *kernel*
  - Mapeiam os vetores de entrada para um espaço de mais alta dimensão
  - Neste espaço os dados são separáveis linearmente

Prof. Dr. Razer A N R Montaño

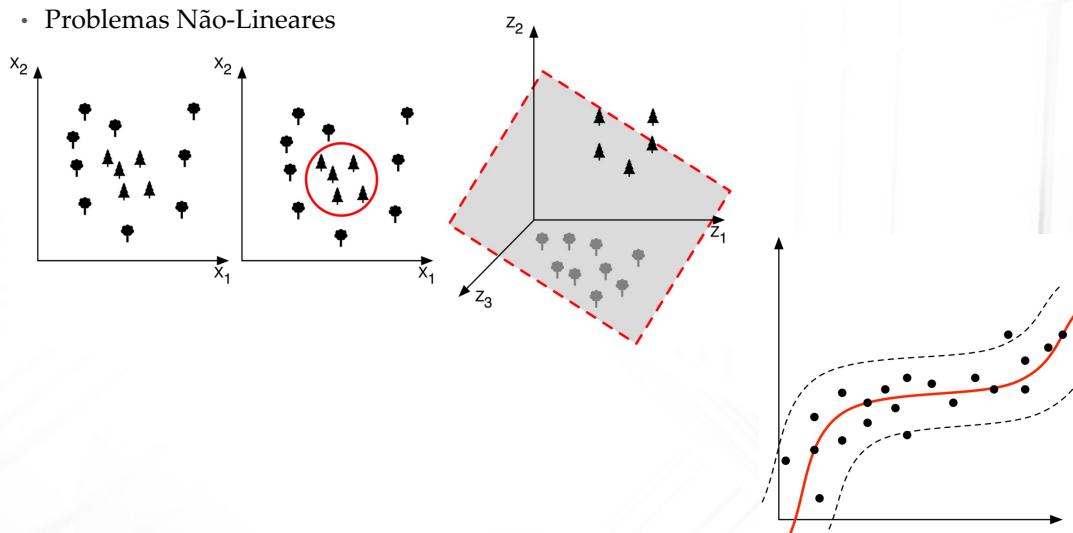
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

251

251

## SVM

- Problemas Não-Lineares



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

252

252

## SVM.

- Várias parametrizações
  - C: custo que penaliza erros de estimativa
  - Função de Kernel: várias disponíveis – Polinomial, Gaussiano, Sigmoidal, etc
  - Parâmetros da própria função kernel

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

253

253

## K-NN

- K-NN : k-Nearest neighbors
- Aprendizado por exemplos (instâncias)
- Verifica os elementos conhecidos mais próximos e se baseia neles
- Usa os K vizinhos mais próximos (ex, 3-NN)
- Usa-se uma composição (ex. média) dos valores dos vizinhos mais próximos

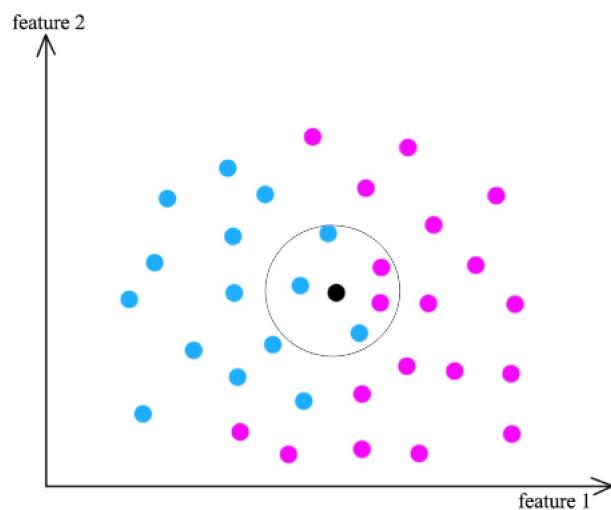
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

254

254

## K-NN



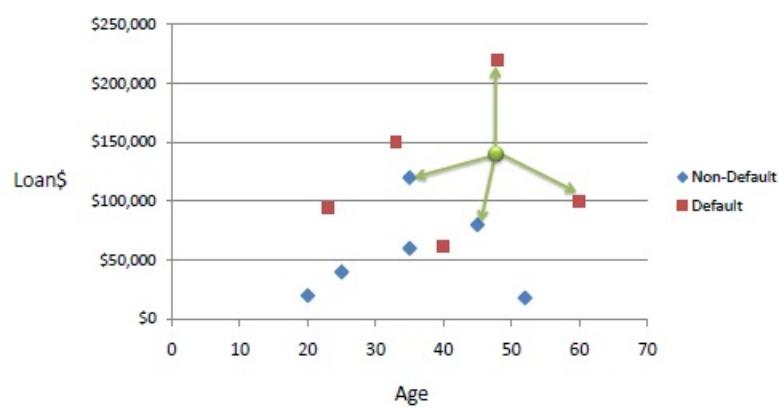
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

255

255

## K-NN



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

256

256

128

## Ambientes para Aprendizado de Máquina..

- WEKA
  - Universidade de Waikato/NZ
  - Ambiente concentrador de implementações de algoritmos de aprendizado de máquina
  - <http://www.cs.waikato.ac.nz/ml/weka/>
- R
  - Ambiente de análise estatística
  - <https://www.rstudio.com/>
- JCarbon
  - Software de estimativa de carbono usando técnicas de mineração de dados
  - Jaime Wojciechowski / BIOFIX / UFPR
  - <http://www.jcarbon.ufpr.br>

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT/UFPR

257

257

## Comunicação, Percepção e Ação

Percepção  
Robótica

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT/UFPR

258

258

129

## Percepção

- Fornece informações sobre o mundo, interpretando sinais de sensores
- Sensor: dispositivo que mede algum aspecto do ambiente
  - Visão
  - Audição
  - Toque
  - Rádio
  - Infravermelho
  - GPS
- Sensoriamento Ativo: quando um sinal é enviado e o reflexo do sinal é sentido
  - Radar
  - Ultrasom

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

259

259

## Visão

- Observações Visuais são extremamente ricas
  - Se comparados a outros sensores: Ex, Se bateu ou não em uma parede
- Problema: Que aspectos devem ser considerados para que um agente faça suas escolhas?
- Três grandes abordagens sobre o problema
  - Extração de Características: cálculos para obter partes das observações
  - Reconhecimento: distinção entre objetos
  - Reconstrução: construção de um modelo geométrico do mundo

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

260

260

130

## Visão – Processamento de Imagens

- Imagens Médicas
- Processamento de Imagens de Documentos
- Reconhecimento de Impressões Digitais
- OCR
- Imagens de Satélites
- Compressão

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

261

261

## Formação de Imagens

- Imagem distorce a geometria
  - Trilhos de trem que convergem
  - Mão na frente da Lua, que não é menor que sua mão
- Sensores de Imagem: reúnem luz espalhada pelos objetos em uma cena e criam uma **imagem** bidimensional
- Câmera: grade retangular de **pixels** fotossensíveis
  - Cada um de CMOS ou CCD
  - Cada fóton produz um efeito que depende da força e comprimento de onda
  - Saída do sensor é a soma de todos os efeitos (observados em uma janela de tempo)
  - Média ponderada da intensidade da luz que chega
- Uma imagem com foco
  - Deve-se assegurar que todos os fótons aproximadamente do mesmo local na cena cheguem aproximadamente no mesmo ponto no plano da imagem

Prof. Dr. Razer A N R Montaño

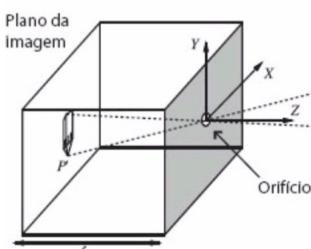
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

262

262

## Formação da Imagem – Câmara Escura

- Quanto mais longe o objeto, menor sua imagem
- Se o orifício for pequeno o suficiente a imagem fica focada
  - Quanto menor o furo, menos luz passa, imagem mais escura
- Melhora-se com um sistema de lentes



Prof. Dr. Razer A N R Montaño

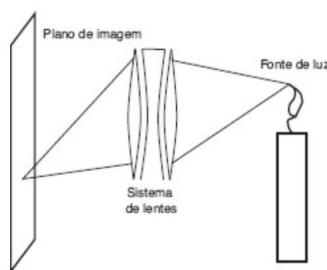
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

263

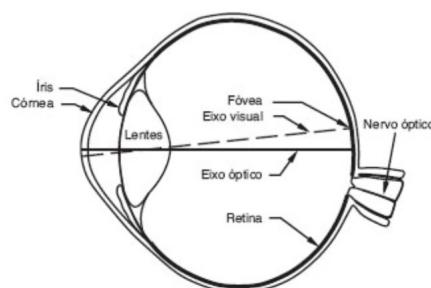
263

## Formação da Imagem – Sistema de Lentes

- Uma abertura grande é coberta por lentes
- Lentes focalizam objetos desde próximos até no plano da imagem
- Mas possuem profundidade de campo limitada
  - Só focam objetos dentro de uma faixa de profundidade: **plano focal**
  - Lentes precisam mudar de forma (olho) ou se mover (câmeras)



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

264

264

## Formação da Imagem – Luz e Sombras

- O brilho de um pixel é baseado no brilho da superfície na cena
- O brilho dá uma forte, porém ambígua, sugestão de forma e identidade
- 3 principais causas de variação de brilho
  - Intensidade geral da luz
  - Reflexão, pontos da cena podem refletir menos ou mais
  - Sombreamento, objetos na frente da fonte de luz são mais brilhantes
- Reflexão Difusa: dispersa luz de forma uniforme
- Reflexão Especular: espelho perfeito
- Especularidades: pequenos trechos em objetos com reflexão especular (ex. colheres)

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

265

265

## Formação da imagem – Luz e Sombras



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

266

266

133

## Formação da Imagem - Cor

- A luz chega ao olho com quantidades diferentes de energia em comprimentos de onda diferentes
- Olho humano detecta uma pequena parcela da função de densidade de energia espectral
  - Suficiente para saber quando uma fruta está madura, por exemplo
- Olho possui 3 tipos de células Cones, que respondem a diferentes comprimentos de onda (vermelho, verde, azul)
- **Princípio da Tricromacia:** qualquer densidade de energia espectral pode ser construída por uma mistura de 3 cores: vermelho, verde e azul.
- Por isso temos R/G/B

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

267

267

## Operações em Processamento de Imagens

- Operações iniciais
  - Detecção de Areias
  - Análise de Textura
  - Fluxo Ótico
- São operações locais, que não precisam de conhecimento sobre a imagem

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

268

268

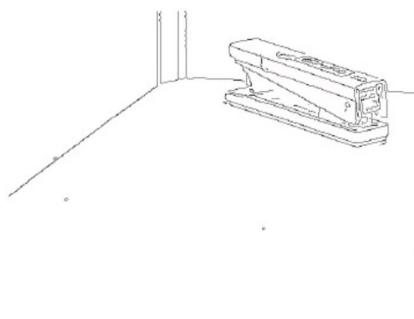
134

## Detecção de Areias

- Areias:
  - Linhas retas ou curvas
  - Mudança significativa no brilho da imagem
- Motivação: contornos de areias correspondem a contornos importantes na cena



(a)



(b)

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

269

269

## Detecção de Areias

- Em geral, primeiro passo é eliminar ruídos
  - Suavizar a imagem com Filtro Gaussiano
  - Cada pixel será uma média ponderada dos seus vizinhos
- Segundo passo, busca por gradientes
  - Gradiente aponta na direção da mudança mais rápida de intensidade
  - Descobrir a direção dos gradientes

$$\nabla f = \left[ \frac{\partial f}{\partial x}, 0 \right]$$



$$\nabla f = \left[ 0, \frac{\partial f}{\partial y} \right]$$

$$\nabla f = \left[ \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]$$

- Ligar os pixels que pertencem à mesma curva
  - Areias adjacentes com mesmas orientações

Prof. Dr. Razer A N R Montaño

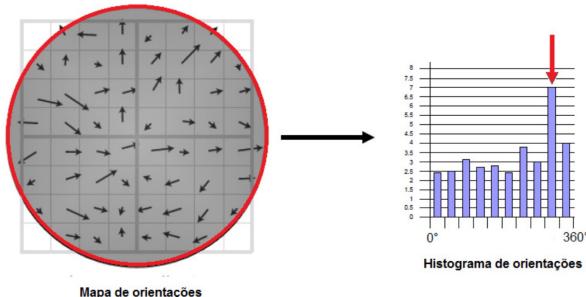
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

270

270

## Textura

- Textura:
  - Relativo a uma superfície (mais de um pixel)
  - Sensação visual de uma superfície
  - Padrão que se repete e pode ser percebida visualmente
- Calcular a orientação de cada pixel e montar um histograma de orientações



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

271

271

## Fluxo Ótico

- Sequência de imagens (ex vídeo)
- Fluxo ótico descreve direção e velocidade do movimento de atributos dentro da imagem



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

272

272

## Fluxo Ótico

- Deve-se encontrar pontos correspondentes num ponto no tempo e no próximo
  - Um Método Simples: pequenas áreas em torno dos pontos correspondentes possuem os mesmos padrões de intensidade
  - Usar alguma medida de similaridade, como **soma dos quadrados das diferenças**
- Permite
  - Distinguir objetos distantes: movimentação menor
  - Reconhecer ações
- Melhor com texturas

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

273

273

## Segmentação de Imagens

- Desmembrar a imagem em seções
- Regiões com pixels semelhantes
- Em uma imagem:
  - Pixels dentro de limites de objetos mudam muito pouco
  - Pixels fora dos limites entre objetos tendem a mudar muito

Prof. Dr. Razer A N R Montaño

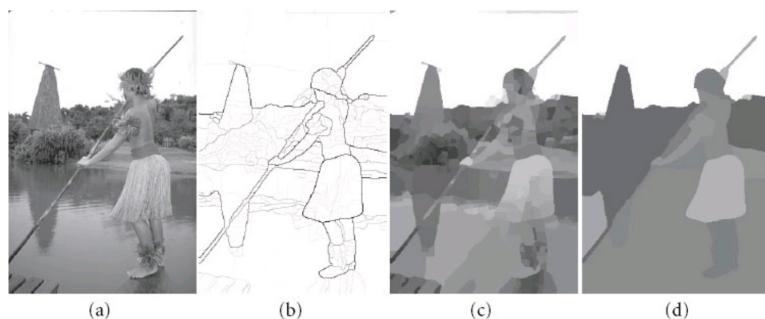
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

274

274

## Segmentação de Imagens

- A) Imagem Original
- B) Contornos
- C) Segmentação mais fina, mais regiões
- D) Segmentação mais grosseira, menos regiões



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

275

275

## Reconhecimento de Objetos por Aparência

- Aparência: o que o objeto tende a parecer
- Alguns objetos mudam pouco: ex. bolas de basquete
- Outros mudam muito: ex. casas, pessoas
- Uma abstração útil:
  - Objetos são feitos de padrões locais que tendem a se movimentar
  - Pode-se identificar com histogramas, que expõe a característica mas não indica onde está
- Usa-se um classificador (aprendizado de máquina)
- Iluminação é importante
  - Rostos olhando para câmera a baixa resolução e pouco iluminado são muito parecidos

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

276

276

138

## Reconhecimento de Objetos por Aparência

- Janela Deslizante
  - Para reconhecer um rosto, p. ex, passa-se uma janela redonda de tamanho fixo pela imagem, calculando alguns atributos
  - Os atributos devem ser robustos e sofrer pouca alteração com sombras, brilhos, etc
  - Uma estratégia é calcular orientações de gradientes
  - Outra estratégia é estimar e corrigir a iluminação
  - Para encontrar rostos de tamanhos menores, repete-se o processo com janelas menores
  - Com várias janelas, de tamanhos diferentes, tem-se muitos dados sobrepostos
    - Em geral o tamanho da janela escolhido não é exatamente o do rosto
  - Usa-se um classificador que possa encontrar os rostos em meio a tantos dados

Prof. Dr. Razer A N R Montaño

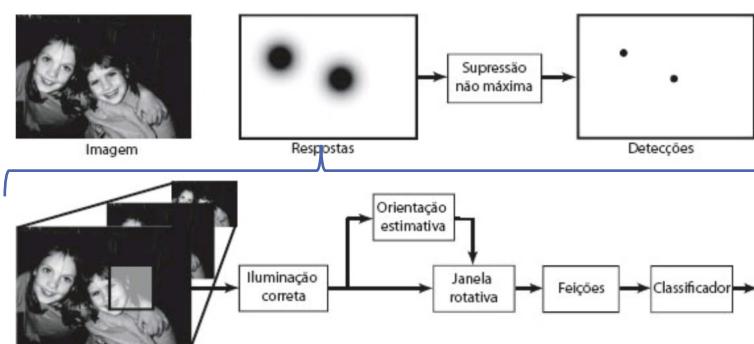
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

277

277

## Reconhecimento de Objetos por Aparência

- Supressão não máxima: supressão de pixels que não são máximos locais de forma transversal, causando afinamento da borda



- Dados de treinamento para um classificador são facilmente encontrados em bases públicas

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

278

278

139

## Reconhecimento de Objetos Complexos.

- Histograma de orientação



Imagen



Histogramas de orientação

Prof. Dr. Razer A N R Montaño

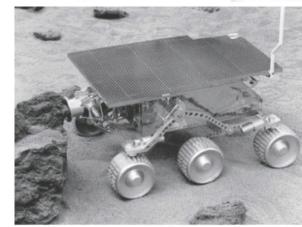
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

279

279

## Robótica

- **Robôs:** agentes físicos que manipulam o mundo físico
- Possuem **efetuadores** que exercem força física sobre o ambiente: pernas, rodas, articulações, garras
- São equipados com **sensores:** permite ao robô perceber o ambiente



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

280

280

## Sensores

- Interface de interação com o ambiente
- Passivos:
  - Ex. Câmeras
  - Observam o ambiente
  - Captam sinais de outras fontes
- Ativos:
  - Ex. Sonar
  - Enviam energia ao ambiente, aguardando o retorno (reflexão)
  - Podem fornecer mais informações, mas têm custo alto e sofrem interferência

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

281

281

## Sensores

- Sonar
  - Emitem ondas sonoras direcionais
  - Tempo e intensidade do retorno dão a distância até o objeto
- Radar (Radio Detection And Ranging)
  - Detecta objetos e calcula distâncias
  - Usa ondas eletromagnéticas
- Câmera Estéreo
  - Múltiplas câmeras, para retratar o ambiente sob vários ângulos
  - Noção de profundidade
- LIDAR (Light Detection And Ranging)
  - Laser pulsado
  - Mede a distância por reflexão



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

282

282

## Sensores

- Sensores Táteis
  - Sensíveis ao toque
  - Ex. Botões de elevadores
- Sensores de Localização
  - GPS (Sistema de Posicionamento Global)
  - Mede a distância entre os satélites, que emitem sinais pulsados
  - Com 3 sinais, há a triangulação e o cálculo do posicionamento



Localização exata



Possíveis localizações

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

283

283

## Sensores

- Sensores Proprioceptivos : informam o estado do próprio robô
  - Decodificadores de eixo: saber a configuração exata de uma articulação
  - Odometria: informam as revoluções das rodas
  - Sensores Inerciais (giroscópio, acelerômetro)
  - Sensores de Torque e Sensores de Força: manipulação de objetos frágeis

Prof. Dr. Razer A N R Montaño

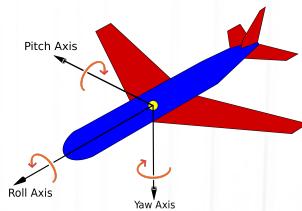
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

284

284

## Efetuadores

- Interação com o ambiente
- Grau de Liberdade: cada direção independente em que um robô ou efetuador pode se mover
  - Robô Rígido de Movimentos Livres
    - AUV (autonomous underwater vehicle)
    - 6 graus de liberdade
    - 3 para sua posição ( $x, y, z$ )
    - 3 para orientação angular (yaw, roll, pitch)
    - Estado Cinemático (pose) do robô
    - Estado Dinâmico inclui a dimensão velocidade
  - Robôs não-rígidos
    - Graus de liberdade dentro do próprio robô
    - Ex, um cotovelo: 2 graus de liberdade
    - Pulso : 3 graus de liberdade



Prof. Dr. Razer A N R Montaño

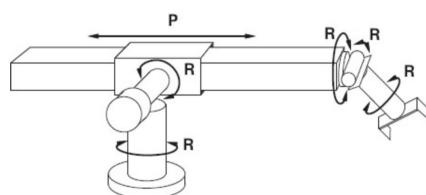
Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

285

285

## Efetuadores

- São necessários 6 graus de liberdade para colocar um objeto em um ponto específico com orientação específica



- R : Articulação de Revolução – movimentos de Rotação
- P: Articulação Prismática – movimentos de Deslizamento

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

286

286

143

## Efetuadores

- Mecanismos de Locomoção
  - Tração diferencial: duas rodas/esteiras independentes (como um tanque)
    - Se ambas forem na mesma velocidade, se move em linha reta
  - Tração sincronizada: cada roda pode se mover ou girar em torno do próprio eixo
    - São bem coordenadas
- Pernas
  - Percorrer terrenos acidentados
  - Mais lentos em superfícies planas
  - Construção difícil



• <https://exame.abril.com.br/tecnologia/robo-corre-sobe-escadas-e-da-salto-mortal-em-video/>

## Efetuadores

- Outros métodos de movimentação
  - Hélices
  - Turbinas
  - Propulsores
- Atuadores: Fonte de Energia para alimentar os Efetuadores
  - Efetuadores são montados na ponta dos Atuadores
  - Motor Elétrico
  - Atuador Pneumático: gás comprimido
  - Atuador Hidráulico: fluídos pressurizados

## Percepção Robótica

- Mapear os sensores em representações internas do ambiente
  - Ruídos
  - Observações parciais
  - Dinamismo
  - Etc
- Localização: descobrir onde os objetos estão, inclusive o robô
- Como robôs físicos são imprevisíveis
  - Distribuição Gaussiana
  - Modelo do movimento do robô
- Usar modelo de sensores também
  - Pode usar marcos identificáveis no ambiente

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

289

289

## Planejamento do Movimento

- Espaço de Configuração: espaço de estados do robô: posição, orientação e ângulos de articulação
- Planejamento de Caminho:
  - Encontrar um caminho de uma configuração até outra
  - No espaço de configuração
- É um problema de planejamento
  - Diferença: espaços contínuos
- Deve-se criar uma representação apropriada do problema
- Mapear coordenadas do Espaço de Configuração no Espaço de Trabalho

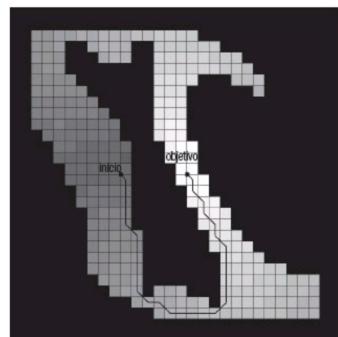
Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

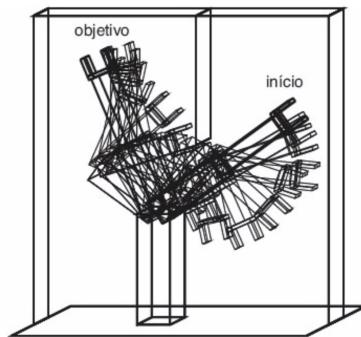
290

290

## Planejamento do Movimento



Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

291

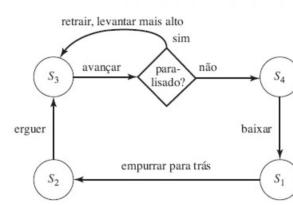
291

## Outras Questões

- Movimentos incertos
- Controle do Movimento
- Agente Reflexo : controle reativo



(a)



Máquina de estados de controle de uma perna

Prof. Dr. Razer A N R Montaño

Especialização em Inteligência Artificial Aplicada  
SEPT / UFPR

292

292

## Domínios de Aplicação da Robótica..

- Indústria e Agricultura: Ex, Linhas de montagem, colheita
- Transporte: Ex, Helicópteros, drones autônomos
- Carros Autônomos
- Cuidados com a Saúde: Ex, Ajuda a cirurgiões
- Ambientes Perigosos: Ex, Limpeza de resíduos nucleares
- Exploração: Ex, Marte
- Serviços Pessoais: Ex, Aspirador de pó
- Entretenimento: Ex, Futebol de robôs
- Ampliação da Capacidade Humana: Ex, Transporte de pessoas, como cadeira de rodas, braços robóticos, membros robóticos