

# Sistemas Multiagentes

Claudio Cesar de Sá  
claudio.sa@udesc.br

Departamento de Ciência da Computação  
Centro de Ciências e Tecnologias  
Universidade do Estado de Santa Catarina

24 de fevereiro de 2017

# Sumário

---

## O Curso

- Ferramentas
- Metodologia e avaliação
- Dinâmica
- Referências

## Introdução aos SMAs

- Motivação aos SMAs
- Os Elementos de SMAs

## Agentes Racionais

- Tipos de Agentes
- Agentes Racionais

## Estratégias de Jogos

## Coordenação

- Exemplos de Coordenação SMAs

## Teoria de Jogos Aplicado a SMA

## Projetos de SMAs

- Implementação de SMAs

## Conclusão

# Disciplina

---

## Sistemas Multiagentes – OSIM001

- **Turma:**
- **Professor:** Claudio Cesar de Sá
  - `claudio.sa@udesc.br`
  - Sala 13 Bloco F
- **Carga horária:** 72 horas-aula • Teóricas: 36 • Práticas: 36
- **Curso:** BCC
- **Requisitos:** Vários – IA, LMA, TEC, SO, PRP, ...
- **Período:** 1º semestre de 2017
- **Horários:**
  - 6ª 10h10 (2 aulas) - F-104 – aula expositiva
  - 6ª 18h00 (2 aulas) - F-306 – lab

# Ementa

---

## Ementa

Motivação do paradigma. Agentes reativos e cognitivos. Teoria e arquitetura de agentes. Sistema multiagentes (SMA) reativo e cognitivo. Linguagens e protocolos de comunicação. Coordenação e negociação. Metodologias para desenvolvimento de SMAs. Ambientes de desenvolvimento.

# Objetivos (1)

---

- **Geral:** Apresentar o conceito de inteligência artificial distribuída: desenvolvimento de agentes e abordagens para coordenação de sistemas multiagentes, permitindo ao aluno ser capaz de modelar problemas de forma a modularizar sua solução de forma distribuída.

## Objetivos (2)

---

### ■ *Específicos:*

- Descrever o histórico e quadro atual da Inteligência Artificial – Moderna.
- Compreender a noção de Teoria de Problemas, computabilidade e complexidade na ótica de IA e IAD.
- Diferencia IAD (orientação a divisão de problemas) versus SMA (orientação a coordenação de agentes)
- Conhecer diferentes arquiteturas de agentes
- Modelar problemas computacionais através de aplicação de agentes.
- Descrever o processo de tomada de decisão e aprendizagem computacional baseado em sistemas multiagentes.
- Conceber, projetar e construir sistemas computacionais capazes de aplicar sistemas multiagentes como técnica de resolução.

# Conteúdo programático

---

- Conceitos de SMA (há muitos correlacionados há áreas diversas)
- Ferramentas: Netlogo e Picat
- Aplicação: vocês escolhem
- Um artigo  $\equiv$  projeto
- Um artigo OUTRO da área a ser apresentado: ficha técnica

# Ferramentas

---

- PICAT (com suporte)
- NETLOGO <http://ccl.northwestern.edu/netlogo/docs/>  
(escondido in WEB)



# Metodologia e avaliação (1)

---

## **Metodologia:**

*As aulas serão expositivas e práticas. A cada novo assunto tratado, exemplos são demonstrados utilizando ferramentas computacionais adequadas para consolidar os conceitos tratados. As aulas nas sextas-feiras a tarde poderão ser realizadas, também, na forma de estudo dirigido.*

## Metodologia e avaliação (2)

---

### Avaliação

- Duas provas (conceituais) –  $\approx 25\%$ 
  - $P_1$ : 25/mar
  - $P_2$ : 25/maio (provão: todo conteúdo)
- Exercícios de laboratório –  $\approx 10\%$
- Implementação de um protótipo –  $\approx 20\%$
- O artigo (resultados da implementação) –  $\approx 30\%$
- Para o artigo: muito material será fornecido em  $\text{\LaTeX}$ ...
- Apresentação de um artigo estudado sobre SMA –  $\approx 15\%$
- Presença e participação
- Média para aprovação: 5,0 (cinco)

# Dinâmica de Aula

---

- Teoria na parte da manhã – 10:00 hrs – F-104
- *Ralação* a tarde – LAB – estudar o NetLogo – vídeo-aulas

# Bibliografia (1)

---

## Básica:

- ALVARES, L. O., SICHMAN, J. *Introdução aos Sistemas Multiagentes*, Anais do EINE – Escola de Informática do Nordeste, Sociedade Brasileira de Computação – SBC, Brasil, 1997.
- FERBER, J. *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*. Harlow, England, Addison-Wesley, 1999.
- WOOLDRIDGE, M.. *An introduction to MultiAgent Systems*, John Wiley, 2001
- [https://github.com/claudiosa/CCS/tree/master/https://github.com/claudiosa/CCS/tree/master/sistemas\\_multiagentes](https://github.com/claudiosa/CCS/tree/master/https://github.com/claudiosa/CCS/tree/master/sistemas_multiagentes)

## Complementar:

- Nikos Vlassis, *A Concise Introduction to Multiagent Systems and Distributed Artificial Intelligence* Synthesis Lectures on Artificial Intelligence and Machine Learning 2007, 71 pages – guia dos tópicos destes slides
- O'HARE, G.; JENNINGS, N. (Editors) *Foundations of distributed artificial intelligence*, New York, NY: John Wiley, 1996.
- WEISS, G. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. London, MIT Press, 2001.
- Russell, S., Norvig, Peter; "Inteligência Artificial", Ed. Campus-Elsevier; Brasil, 2010 – em inglês.

## Bibliografia (2)

---

- Bittencourt, G.; "Inteligência Artificial, ferramentas e teorias"; 3. ed. UFSC; Florianópolis, SC; 2006.
- Barreto. J.M.; "Inteligência Artificial, uma abordagem híbrida"; 3a. ed.; RoRoRo; Florianópolis, SC; 2001
- Eberhart, R; Simpson, P.; Dobbins, R.; "Computational Intelligence PC Tools"; AP Professional; 1996; ISBN 0-12-228630-8.
- Fausett, Laurene; Fundamentals of Neural Networks; Prentice Hall Ind.; N. Jersey; 1994.
- Freeman, J. A.; Skapura, D. M.; "Neural networks – Algorithms, Applications and Programming Techniques"; Addison- Wesley Pub. Co.; New York; 1991.
- Luger, George F.; Inteligência Artificial; Artmed Ed. S.A.; P. Alegre; 2004.
- Mitchell, M.; "An introduction to genetic algorithms"; The MIT press; London; 1966.
- Rabuske, R. A.; Inteligência Artificial; UFSC; Florianópolis; 19??
- Resende, Solange O., Sistemas Inteligentes - Fundamentos e aplicações, Ed. Manole ([www.manole.com.br](http://www.manole.com.br)), 200?
- Rich, E.; "Artificial Intelligence"; McGraw-Hill Book Company; USA; 1983.
- Material didático disponível em: [www.inf.ufsc.br/~falqueto](http://www.inf.ufsc.br/~falqueto)

# Capítulo 1 – Contexto e Motivação aos SMAS

# Histórico e Contexto da IA versus IAD versus SMA

---

- IA cresceu muito nos anos 70 → 80... modelando a inteligência individual.
- Advento das redes de computadores modificou as necessidades!
- Inteligência como a integração dos processos de *raciocinar*, *decidir*, *aprender* e *planejar*.
- O *Modelo de Agente* aparece então como catalisador...

## Em verdade:

---

- Mundo onde informações e conhecimentos crescem (e mudam) rápido demais!
- O crescimento da Internet trás desafios constantes que incluem:
  - Acesso a informações relevantes
  - Identificação de oportunidades
  - Ação no momento preciso
  - Manipulação de grandes volumes de informação
- Ubiquidade, Gerenciamento e Inteligência



# Encaminhando aos SMAs

---

- Vários problemas não podem mais serem tratados de modo **centralizado**, por exemplo:
  - Controle de linha de trens (Brasil) metrô (hum SP talvez)
  - Monitoramento de Redes de Computador
  - Diagnóstico Médico
  - Compra e Venda

# Encaminhando aos SMAs

---

- Vários problemas não podem mais serem tratados de modo **centralizado**, por exemplo:
  - Controle de linha de trens (Brasil) metrô (hum SP talvez)
  - Monitoramento de Redes de Computador
  - Diagnóstico Médico
  - Compra e Venda
- Como Resolvê-los?

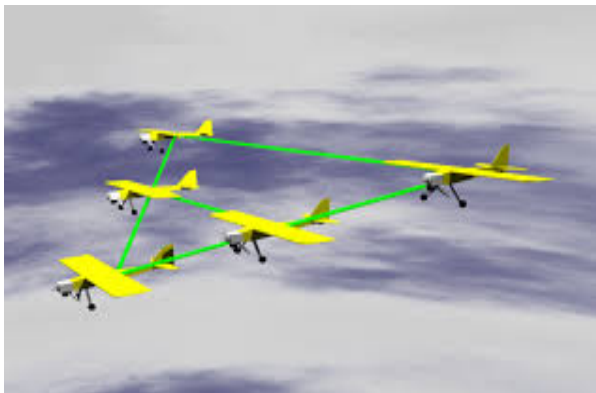
# Encaminhando aos SMAs

---

- Vários problemas não podem mais serem tratados de modo **centralizado**, por exemplo:
  - Controle de linha de trens (Brasil) metrô (hum SP talvez)
  - Monitoramento de Redes de Computador
  - Diagnóstico Médico
  - Compra e Venda
- Como Resolvê-los?  
Inteligência Coletiva  $\Rightarrow$  IA Distribuída  $\Rightarrow$ 
  - Resolução Distribuída de Problemas (RDP)
  - Sistemas Multiagentes (SMA)  $\Leftarrow$  **foco deste curso**

## Motivando aos SMAs

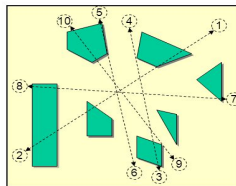
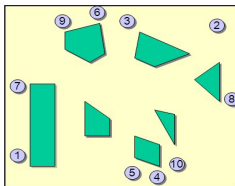
---



**Figura:** Observe o sentido das flechas – e o foco da missão

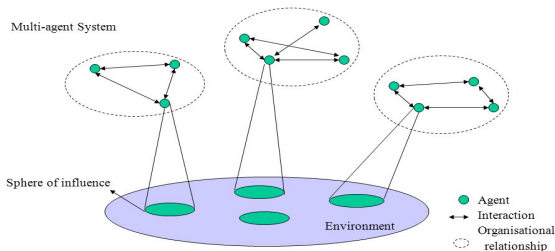
# Motivando aos SMAs

## Path Planning for Multiple Robots



# Motivando aos SMAs

## Multi-agent Systems (MAS)



8

**Figura:** Arquitetura clássica – comunidade de agentes  $\equiv$  SMA

# Motivação I

---

Projetar e construir sistemas multiagentes é uma tarefa difícil, pois:

# Motivação I

---

Projetar e construir sistemas multiagentes é uma tarefa difícil, pois:

- Apresenta todos os problemas já conhecidos dos sistemas distribuídos e concorrentes.



# Motivação I

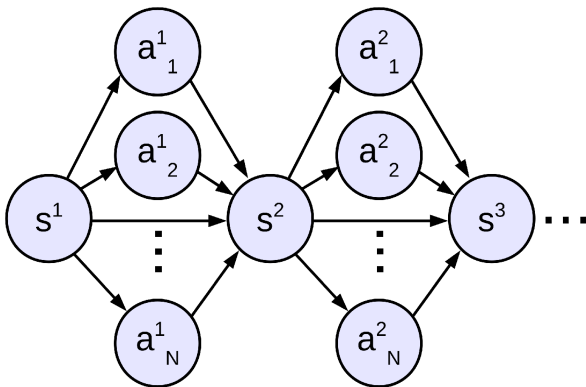
---

Projetar e construir sistemas multiagentes é uma tarefa difícil, pois:

- Apresenta todos os problemas já conhecidos dos sistemas distribuídos e concorrentes.
- Dificuldades adicionais surgem da flexibilidade e complexidade das interações

Exemplificando a complexidade por um DFD por **agentes**  $\times$  **ações**:

---



**Figura:** Complexidade via DFD de um SMA (agentes)  $\times$  ações  $\equiv$  um único estado

## Motivação II

---

Dois principais impedimentos técnicos, pois:

## Motivação II

---

Dois principais impedimentos técnicos, pois:

- Inexistência de uma metodologia sistemática para claramente especificar e estruturar aplicações SMA.

## Motivação II

---

Dois principais impedimentos técnicos, pois:

- Inexistência de uma metodologia sistemática para claramente especificar e estruturar aplicações SMA.
- Inexistência de ferramentas e ambientes de desenvolvimento de SMA com qualidade industrial.

# Os Elementos de SMAs (1)

---

O que abordaremos neste curso:

Projeto de Agente:

Ambiente:

Percepção:

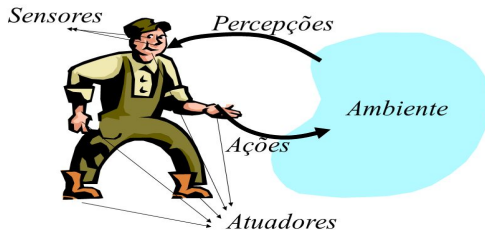
Controle:

Conhecimento:

Comunicação:

# Exercício

## Exemplo: agente humano



1A 13/10/10

6

**Figura:** Exercício: enumere domínios para este agente identificando os itens de SMAs

# Capítulo 2 – Agentes Racionais



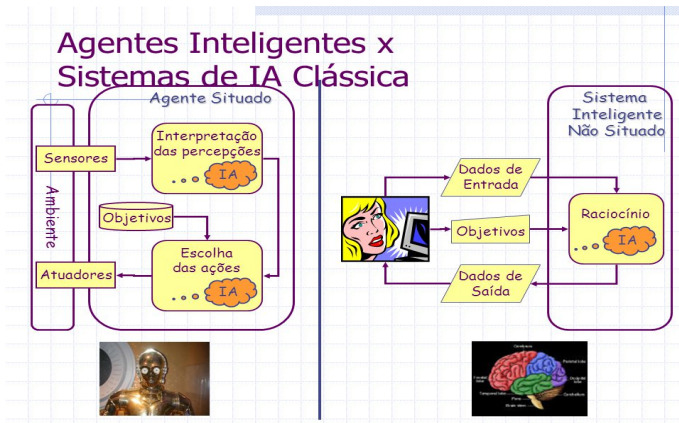
# O que é um Agente? (1)

---

Qualquer entidade (humana ou artificial) que:

- está **imersa** ou **situada** em um ambiente (físico, virtual/simulado)
- **percebe** ou **sente** seu ambiente através de sensores (ex. câmeras, microfone, teclado, finger, ...)
- **age** sobre ele através de atuadores (ex. vídeo, auto-falante, impressora, braços, ftp, ...)
- **possui objetivos** próprios: explícitos ou implícitos
- **escolhe** suas ações em função das suas percepções para atingir seus objetivos

# Agente Situado x Não-Situado



**Figura:** Agente situado versus a visão clássica de sistemas inteligentes

# O que é um Agente Racional?

---

## ■ Agente Racional

- faz a melhor coisa possível
- segue o princípio da racionalidade:  
dada uma seqüência perceptiva, o agente escolhe, segundo seus conhecimentos, as ações que melhor satisfazem seu objetivo
- Limitações de:  
sensores  
atuadores  
raciocinador (conhecimento, tempo, etc.)

# Outras propriedades freqüentemente associadas aos Agentes (1)

---

- Autonomia:
  - raciocínio, comportamento guiado por objetivos ou reatividade
  - Requer máquina de inferência e base de conhecimento
  - Essencial em sistemas especialistas, controle, robótica, jogos, agentes na internet ...
- Adaptabilidade & aprendizagem
  - Capacidade de adaptação a situações novas, para as quais não foi fornecido todo o conhecimento necessário com antecedência
  - Duas implementações: sistema com aprendizagem e/ou programação declarativa
  - Essencial em agentes na internet, interfaces amigáveis ...

## Outras propriedades freqüentemente associadas aos Agentes (2)

---

- Comunicação & Cooperação (Sociabilidade)
  - Protocolos padrões de comunicação, cooperação, negociação
  - Raciocínio autônomo sobre crenças e confiabilidade
  - Arquiteturas de interação social entre agentes
- Personalidade
  - IA + modelagem de atitudes e emoções
  - Essencial em entretenimento digital, realidade virtual, interfaces amigáveis
  - ...
- Continuidade temporal (persistência)
  - Requer interface com sistema operacional e banco de dados
  - Essencial em filtragem, monitoramento, controle, ...
- Mobilidade (caso internet)
  - Requer itens como:

## Outras propriedades freqüentemente associadas aos Agentes (3)

---

1. Interface com rede
  2. Protocolos de segurança
  3. Suporte a código móvel
- Essencial em agentes de exploração da internet, ...

## Exemplo: Agente de Polícia

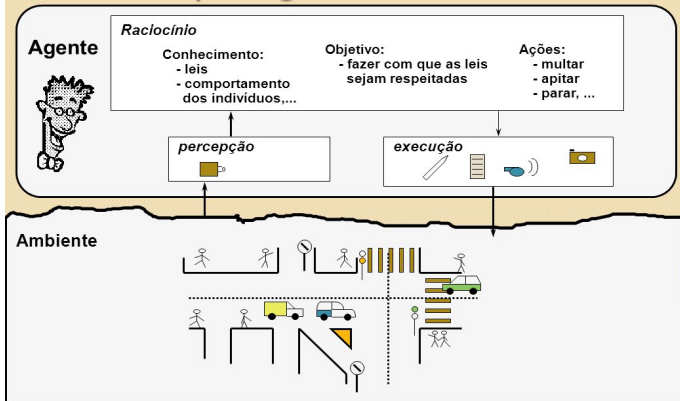


Figura: Exercício: enumere as características citadas a este exemplo

## O que é um Agente? (1)

---

Qualquer entidade (humana ou artificial) que:



# Características aos SMAs

---

# Características aos SMAs

---

# Características aos SMAs

---



■ cap 2

Os agentes podem ser de dois tipos:

**Agentes Reativos (ou reflexivos):** geralmente são agentes simples, escolhem suas ações baseados exclusivamente nas percepções que têm do ambiente. Normalmente possui representação do conhecimento implícito no código, por não possuírem memória, não tem histórico dos fatos e das ações que executou.

**Agentes Cognitivos:** têm uma representação simbólica explícita do seu ambiente, no qual eles podem argumentar e prever eventos futuros. São dirigidos por intenções, isto é, por metas explícitas que conduzem seu comportamento e os tornam capazes de escolher entre possíveis ações. Engloba as características: percepção, ação, comunicação, representação, motivação, deliberação, raciocínio e aprendizagem.

## Agente em seu ambiente

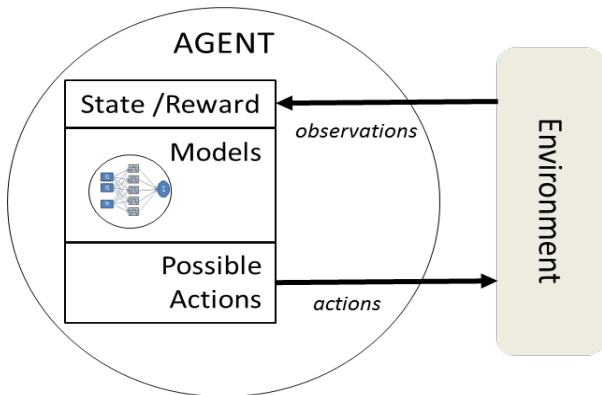


Figura: Ciclo do agente

# Arquitetura clássica de um agente reflexivo

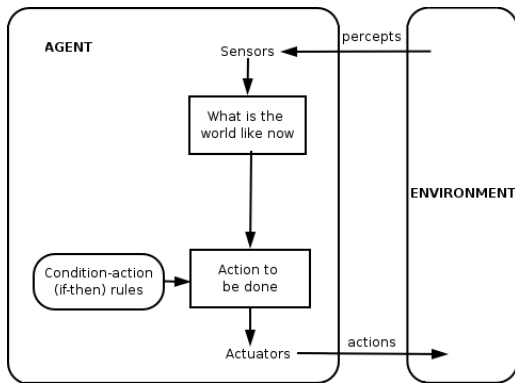


Figura: Arquitetura clássica

## Arquitetura clássica de um agente que *aprende* – desejável

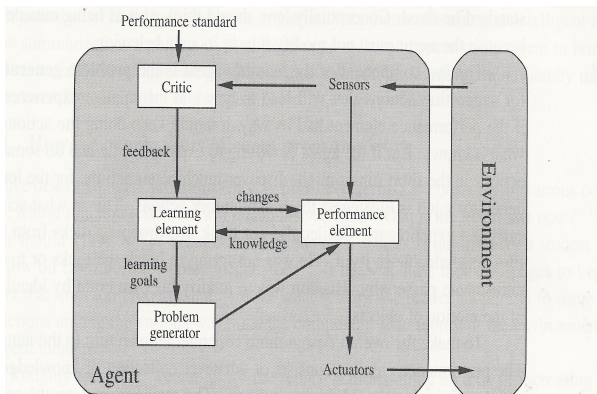


Figura: Arquitetura agente com aprendizagem

# Teoria de Jogos

---



# Teoria de Jogos

---

# Teoria de Jogos

---



■ cap 3

# Coordenação

---

# Coordenação

---

# Coordenação

---



■ cap 4

## Exemplo de Coordenação SMAs

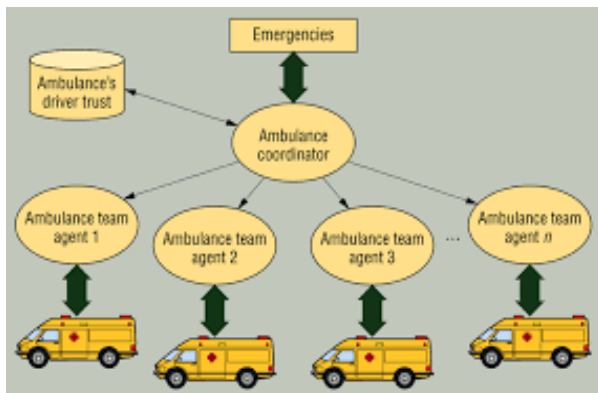


Figura: Coordenação de agentes  $\equiv$  SMA

## Exemplo de Coordenação SMAs

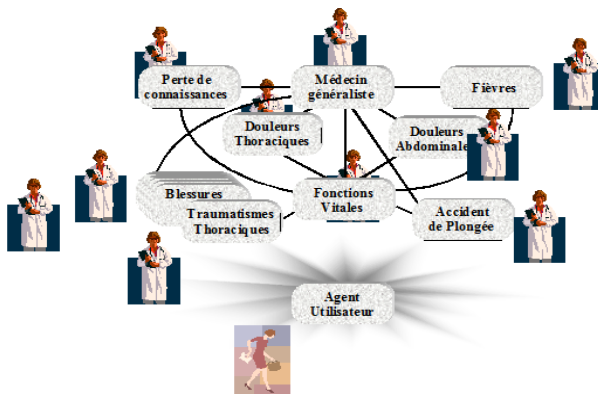


Figura: Coordenação de agentes  $\equiv$  SMA

# Teoria de Jogos Aplicado a SMA

---



# Teoria de Jogos Aplicado a SMA

---

- $\prod_{x=1}^n \neq \prod_{x=1}^{n+1}$
- <https://www.codecogs.com/latex/eqneditor.php>
- <http://www.hostmath.com/>

# Teoria de Jogos Aplicado a SMA

---

- $\prod_{x=1}^n \neq \prod_{x=1}^{n+1}$
- <https://www.codecogs.com/latex/eqneditor.php>
- <http://www.hostmath.com/>
- cap 6

# Mecanismos de Projetos

---

# Mecanismos de Projetos

---

# Mecanismos de Projetos

---



■ cap 6

# Implementação de Agentes

---

# Implementação de Agentes

---

# Implementação de Agentes

---



■ xxxxxxxxxxxxxxxx



# Conclusão

---

