

## Projeto 1

### Avaliação Experimental de Agentes Inteligentes

## Objetivos

1. Compreender os conceitos fundamentais necessários à aplicação da noção de agentes inteligentes na resolução de problemas em ambientes de tarefas de diversos tipos.
2. **Projetar programas de agentes artificiais reativos simples e reativos baseados em modelos, orientados por regras condição-ação, para resolver problemas em ambientes de tarefas determinísticos e parcialmente observáveis.**
3. Avaliar o desempenho de agentes inteligentes em ambientes de tarefas simulados em computador.

## Justificativas

Qual a importância do estudo dos conceitos sobre agentes inteligentes, do projeto, e da avaliação de desempenho de agentes reativos simples e baseados em modelos?

- O ponto de vista de agentes abrange muitas entidades no mundo, ou seja, que podem ser vistas como interagindo com um ambiente por meio de sensores e atuadores, visando realizar algum objetivo.
- A grande maioria dos sistemas computacionais são agentes artificiais concebidos para realizar objetivos em ambientes parcialmente observáveis.
- Alguns dos sistemas computacionais agentes artificiais, são concebidos como agentes reativos que tomam decisões baseados em percepções do ambiente e regras condição-ação. Outros sistemas são concebidos como agentes baseados em modelos, isto é, que tomam decisões considerando um estado interno a respeito do ambiente e um conjunto de regras condição-ação.
- A definição de inteligência artificial (IA) moderna considera um conceito ideal de inteligência, ou seja, a noção de racionalidade. Considerando este ponto de vista, o projetista do agente artificial precisa avaliar o desempenho do sistema e verificar se o mesmo é inteligente, ou seja, se o agente artificial é racional. A avaliação de desempenho deve ser realizada de maneira adequada visando perceber se o agente tem a capacidade de realizar o objetivo para o qual foi projetado.

## Metodologia

Para realizar os objetivos do projeto (em resumo: compreender, projetar e avaliar programas de agentes artificiais), **considerar um dos itens abaixo:**

- a. **visando aplicar os conceitos relacionados aos agentes inteligentes, utilizar o simulador de ambiente de medição de desempenho proposto por Jill Zimmerman (<http://phoenix.goucher.edu/~jillz/cs340/AI.zip>), do Goucher College, USA, para uma**

versão do mundo do aspirador de pó em que o agente deve limpar a sujeira contida em locais representados por retângulos, sem móveis ou outros obstáculos (ver observações ao final do texto do projeto para facilitar o uso).

- b. Considerando a linguagem disponível na plataforma NetLogo (<https://ccl.northwestern.edu/netlogo/>), implementar um simulador de ambiente de medição de desempenho para uma versão do mundo do aspirador de pó. Sua implementação deve ser modular, de forma que as características do ambiente (tamanho, forma, localização de sujeira) possam ser alteradas com facilidade.
- c. Considerando qualquer linguagem de programação, implementar um simulador de ambiente de medição de desempenho para uma versão do mundo do aspirador de pó, conforme proposto no item b.

Considerar duas medidas de avaliação para os agentes como, por exemplo: **medida oferece o prêmio de um ponto para cada quadrado limpo em cada período, e medida oferece o prêmio de um ponto para cada quadrado limpo e penaliza com um ponto a menos cada movimento.** Em seguida, implementar **um programa reativo simples e outro baseado em modelos para um agente artificial realizar uma tarefa em um ambiente determinístico e parcialmente observável.** Os agentes não conhecem o padrão inicial de sujeira, nem a geografia (extensão, limites e obstáculos) do ambiente. Os sensores dos agentes percebem localmente o ambiente.

Experimentar: fixe o tamanho do ambiente e execute o simulador de ambiente para várias configurações iniciais possíveis de sujeira, obstáculos e posições dos agentes. Registre a pontuação de desempenho dos agentes para cada configuração do ambiente e sua pontuação média global.

Ao final do desenvolvimento do projeto, cada grupo deverá:

- Entregar um relatório técnico apresentando os mecanismos de atualização de estado interno e de tomada de decisão de cada agente, uma breve descrição do comportamento de cada agente, tabelas, gráficos e uma análise dos resultados a partir do ponto de vista da racionalidade de cada programa.
- Entregar os códigos documentados do ambiente e dos agentes desenvolvidos.

A nota final atribuída a cada grupo de alunos será obtida como resultado do somatório dos pontos obtidos nos seguintes itens:

- Redação sobre Agentes Inteligentes: **2 pts.**
- Implementações ambiente e estratégias de seleção de ações nos agentes: **2 pts.**
- Documentação adequada dos códigos dos agentes implementados: **1 pt.**
- Realização dos experimentos e resultados obtidos: **2 pts.**
- Relatório técnico com descrição experimentos, resultados e implementações: **3 pts.**

## Observações

O quadro abaixo apresenta o método proposto para avaliação de desempenho de programas de agentes aspirador empregando o simulador proposto por Jill Zimmerman, proposto no Passo 4a.

```
/* This method uses the percept (stored in the Vector percept) and state values to determine
the agents next action. The member variable action is assigned a string (either "suck", "forward",
"turn left", "turn right", or "shut-off") to indicate the agent's next action */

// My Vacuum Agent package ai.worlds.vacuum;
import java.util.*;
class MyVacuumAgent extends VacuumAgent
{
// declare any state variables here

void determineAction() { }
}
```

Para realizar o experimento no simulador, nomeie cada agente com o seu nome seguido pelo nome de cada agente seguido por VacuumAgent, por exemplo, **PauloAgente1VacuumAgent**. A classe precisa estar contida no arquivo com o mesmo nome com a extensão java, por exemplo **PauloAgente1VacuumAgent.java**. Adicione esse arquivo a um projeto Eclipse que contém todos os arquivos java fornecidos por Jill Zimmerman. Você precisará fazer algumas pequenas modificações em um dos arquivos.

Faça o download dos arquivos do projeto em: <http://phoenix.goucher.edu/~jillz/cs340/ai/ai.zip>. Use o Eclipse para importar o projeto. Em seguida, compile os arquivos e execute a classe AI. O novo arquivo **<xx>VacuumAgent** deve ser colocado no pacote **ai.worlds.vacuum**. Além disto, selecione o arquivo **WorldCreatePanel.java** no pacote **ai.worlds** e faça as seguintes alterações para incluir o seu agente:

1. Adicionar um string com o nome de seu agente em:

```
String[] vacuumStrings = {"Random Vacuum Agent", "Reactive Vacuum Agent"};
```

2. No método createAgent, adicione um comando **else if** no final, tal como:

```
else if (agentName == "PauloAgente1 Vacuum Agent")
    return new PauloAgente1VacuumAgent();
```

Com essas modificações, você pode comparar o desempenho de seu agente com aqueles "dummy" fornecidos ("trials"). Basta selecionar os agentes e executá-los. As pontuações médias alcançadas nos experimentos serão exibidas.