



UNIVERSIDADE FEDERAL DO MARANHÃO
DEPARTAMENTO DE ENGENHARIA DA COMPUTAÇÃO

Curso: Engenharia da Computação	Ano / Semestre: 2024 / 2
Disciplina: Inteligência Artificial	Professor: Thales Levi Azevedo Valente
Tema: Agentes - Modificando o Modelo de Formigas em NetLogo	
Aluno:	Código:

SUMÁRIO

Atividade Criativa para os Alunos: Modificando o Modelo de Formigas em NetLogo	3
1. Objetivo da Atividade	3
2. Ferramenta	3
3. Descrição da Atividade.....	3
3.1 Seleção de Modificações	3
3.3.1 Introdução de Obstáculos no Ambiente	3
3.3.2 Implementação de Predadores	3
3.3.3 Diferenciação de Tipos de Alimento	3
3.3.4 Limitação de Capacidade de Carga.....	4
3.3.5 Comunicação Direta entre Formigas	4
3.3.6 Condições Climáticas Variáveis.....	4
3.3.7 Ciclo de Vida das Formigas.....	4
3.3.8 Hierarquia Social.....	4
3.3.9 Combate entre formigueiros	4
3.2 Implementação das Modificações.....	5
3.3 Análise e Documentação	5
3.4 Apresentação Oral:.....	6
4. Dicas.....	6
4.1 Introdução de Obstáculos no Ambiente.....	6
4.2 Implementação de Predadores	7
4.3 Diferenciação de Tipos de Alimento.....	7
4.4 Limitação de Capacidade de Carga	7
4.5 Comunicação Direta entre Formigas	7
4.6 Condições Climáticas Variáveis	8
4.7 Ciclo de Vida das Formigas	8
4.8 Hierarquia Social	8
4.9 Recursos do NetLogo que Facilitam Essas Implementações	9
5. Critérios de Avaliação	9
5.1 Criatividade e Originalidade (25%).....	9
5.2 Implementação Técnica (25%)	9
5.3 Análise e Reflexão (25%)	9
5.4 Comunicação (25%)	9

Atividade Criativa para os Alunos: Modificando o Modelo de Formigas em NetLogo

1. Objetivo da Atividade

Estimular a compreensão dos conceitos de Inteligência Artificial e modelagem baseada em agentes através da modificação criativa do modelo de formigas no NetLogo. Os alunos serão desafiados a alterar o código fornecido para incorporar novos comportamentos, elementos ambientais ou mecanismos de interação, analisando o impacto de suas mudanças no comportamento emergente da simulação.

2. Ferramenta

O NetLogo ([NetLogo Web: Ants](#)) é uma ferramenta poderosa e versátil para modelagem e simulação de sistemas baseados em agentes. Ele foi projetado especificamente para permitir que estudantes e pesquisadores explorem comportamentos emergentes em sistemas complexos, como colônias de formigas, ecossistemas, mercados econômicos e muito mais.

3. Descrição da Atividade

3.1 Seleção de Modificações

Os alunos devem escolher uma ou mais das seguintes sugestões ou propor suas próprias ideias para modificar o modelo:

3.3.1 Introdução de Obstáculos no Ambiente

- Adicionar obstáculos como rochas, rios ou áreas inacessíveis que afetam o movimento das formigas.
- Modificar o comportamento das formigas para que elas consigam contornar os obstáculos.

3.3.2 Implementação de Predadores

- Introduzir um novo agente, como um tamanduá, que se move pelo ambiente e captura formigas.
- As formigas devem desenvolver comportamentos de evasão ao detectar a presença do predador.

3.3.3 Diferenciação de Tipos de Alimento

- Criar fontes de alimento com valores nutricionais ou quantidades de energia diferentes.
- As formigas podem priorizar fontes de alimento mais nutritivas ou próximas.

3.3.4 Limitação de Capacidade de Carga

- Estabelecer um limite na quantidade de alimento que uma formiga pode carregar.
- Implementar formigas transportadoras ou trabalho em equipe para carregar alimentos maiores.

3.3.5 Comunicação Direta entre Formigas

- Permitir que as formigas troquem informações quando se encontram, como localização de fontes de alimento.
- Introduzir uma forma de memória ou aprendizado coletivo dentro da colônia.

3.3.6 Condições Climáticas Variáveis

- Simular mudanças climáticas que afetam a taxa de evaporação ou difusão dos feromônios.
- As formigas devem adaptar seu comportamento às novas condições ambientais.

3.3.7 Ciclo de Vida das Formigas

- Implementar envelhecimento nas formigas, onde elas têm uma vida útil limitada.
- Simular reprodução para manter ou aumentar a população.

3.3.8 Hierarquia Social

- Introduzir diferentes papéis dentro da colônia, como operárias, soldados e rainhas.
- Cada tipo de formiga tem comportamentos específicos e contribuições para a colônia.

3.3.9 Combate entre formigueiros

- Introduzir diferentes papéis dentro da colônia, como operárias, soldados e rainhas.
- Cada tipo de formiga tem comportamentos específicos e contribuições para a colônia.

3.3.9.1 Criação de Formigueiros Competidores

a) Definição de Raças

- I. Utilize duas raças (breeds) distintas de formigas, por exemplo, breed [ants1 ant1] e breed [ants2 ant2] para representar os dois formigueiros.

b) Definição de Bases

- II. Adicione patches específicos que representem as bases ou ninhos de cada formigueiro, com variáveis como nest1? e nest2?.

3.3.9.2 Implementação do Combate

a) Detecção de Inimigos

- III. Quando uma formiga de uma raça encontra outra formiga da raça oposta (usando comandos como in-radius ou turtles-here), inicie o combate.

b) Mecânica de Combate:

- IV. Implemente variáveis como health para as formigas e reduza o valor de health durante o combate.
- V. Utilize uma probabilidade ou força baseada em variáveis como tamanho do formigueiro ou número de formigas próximas.
- VI. Elimine a formiga derrotada com o comando die.

3.3.9.3 Estratégias de Defesa e Ataque

a) Defesa do Ninho

- VII. As formigas próximas ao ninho devem priorizar a defesa do território.
- VIII. Defina comportamentos específicos para patrulhar e atacar inimigos que se aproximem.

b) Ataques em Grupo

- IX. Formigas podem se agrupar antes de atacar o ninho inimigo, utilizando comunicação ou sinais (ex.: pheromones).

3.3.9.4 Recompensa por Vitória

c) Captura de Recursos

- X. Permita que o formigueiro vencedor colete recursos do ninho inimigo ou aumente sua população.
- XI. Adicione variáveis globais para rastrear o progresso dos formigueiros.

3.2 Implementação das Modificações

3.2.1 Planejamento

- Desenhe um esboço das alterações planejadas e como elas serão incorporadas ao código existente.
- Identifique quais partes do código precisarão ser modificadas ou expandidas.

3.2.2 Codificação

- Implemente as alterações no código NetLogo, garantindo que o código esteja bem estruturado e comentado.
- Faça testes incrementais para verificar o funcionamento correto de cada nova funcionalidade.

3.3 Análise e Documentação

3.3.1 Relatório Escrito

a) Introdução

- Explique o objetivo das modificações e os conceitos teóricos envolvidos.

b) Desenvolvimento

- Descreva detalhadamente as alterações feitas no código.
- Inclua trechos de código relevantes com comentários explicativos.

c) Resultados

- Apresente os resultados da simulação após as modificações.
- Utilize gráficos, tabelas ou capturas de tela para ilustrar o comportamento observado.

d) Discussão

- Analise o impacto das modificações no comportamento das formigas e no ambiente.
- Relacione os resultados aos conceitos de IA estudados, como comportamento emergente, auto-organização e adaptação.

e) Conclusão

- Resuma as principais descobertas e aprendizagens obtidas com a atividade.
- Sugira possíveis extensões ou melhorias futuras.

3.4 Apresentação Oral:

3.4.1 Preparação

- Elabore uma apresentação de 10 a 15 minutos para compartilhar com a turma.
- Utilize slides para destacar os pontos principais, incluindo imagens e gráficos da simulação.

3.4.2 Demonstração

- Mostre o modelo modificado em execução, explicando os novos comportamentos e suas implicações.

3.4.3 Discussão

- Esteja preparado para responder perguntas e discutir suas escolhas e observações com os colegas.

4. Dicas

As sugestões de modificações que apresentadas na atividade são exemplos de como o NetLogo pode ser usado para estender e personalizar modelos existentes. Vamos detalhar como cada uma dessas propostas pode ser implementada no NetLogo:

4.1 Introdução de Obstáculos no Ambiente

4.1.1 Criação de Obstáculos

- Utilize patches para representar obstáculos no ambiente.
- Defina uma variável de patch, como `obstacle?`, para identificar se um patch é um obstáculo.
- Modifique os procedimentos de movimento das formigas para evitar patches onde `obstacle?` é verdadeiro.

4.1.2 Desenho de Obstáculos

- Use a interface do NetLogo para desenhar obstáculos manualmente.

- Ou gere obstáculos aleatoriamente durante a configuração do modelo.

4.1.3 Comportamento das Formigas

- Ajuste o procedimento wiggle ou crie um novo para que as formigas possam contornar os obstáculos.

4.2 Implementação de Predadores

4.2.1 Criação de Predadores

- Utilize uma nova raça (breed) de agentes, por exemplo, breed [predators predator].
- Defina comportamentos específicos para os predadores, como perseguir formigas.

4.2.2 Interação com Formigas

- Quando um predador encontra uma formiga, você pode remover a formiga (usando die) ou implementar outra lógica.

4.2.3 Movimento dos Predadores

- Defina padrões de movimento para os predadores, que podem ser aleatórios ou direcionados.

4.3 Diferenciação de Tipos de Alimento

4.3.1 Tipos de Alimento

- Defina diferentes valores de food nos patches, representando o valor nutricional.
- Use cores distintas para cada tipo de alimento na visualização.

4.3.2 Comportamento das Formigas

- Modifique as regras de decisão das formigas para que elas priorizem certos tipos de alimento.
- Utilize variáveis adicionais nas formigas para armazenar preferências ou estados.

4.4 Limitação de Capacidade de Carga

4.4.1 Variável de Capacidade

- Adicione uma variável às formigas, como load, para representar a quantidade de alimento carregado.

4.4.2 Regras de Carregamento

- Ajuste o procedimento look-for-food para permitir que as formigas carreguem até um limite.
- Implementar lógica para formigas trabalharem juntas se necessário.

4.5 Comunicação Direta entre Formigas

4.5.1 Detecção de Formigas Próximas

- Utilize in-radius ou turtles-here para identificar formigas próximas.

4.5.2 Troca de Informações

- As formigas podem compartilhar informações através de variáveis ou sinais.
- Implementar procedimentos como communicate para gerenciar a troca de dados.

4.5.3 Memória ou Aprendizado Coletivo

- Adicione variáveis ou estruturas para armazenar informações coletivas, como locais de fontes de alimento.

4.6 Condições Climáticas Variáveis

4.6.1 Variáveis Ambientais

- Introduza variáveis globais como temperature ou humidity.

4.6.2 Impacto no Ambiente

- Modifique a taxa de evaporação ou difusão dos feromônios com base nas condições climáticas.

4.6.3 Adaptação das Formigas

- Ajuste os comportamentos das formigas para reagir às mudanças climáticas, talvez alterando padrões de atividade.

4.7 Ciclo de Vida das Formigas

4.7.1 Variável de Idade

- Adicione uma variável age às formigas.

4.7.2 Envelhecimento e Morte

- Incrementar age a cada tick e remover a formiga após atingir uma idade máxima.

4.7.3 Reprodução

- Implementar condições sob as quais formigas podem gerar novas formigas, talvez ao retornar ao ninho.

4.8 Hierarquia Social

4.8.1 Definição de Papéis

- Utilize breeds para criar diferentes tipos de formigas, como workers, soldiers, queens.

4.8.2 Comportamentos Específicos

- Defina procedimentos específicos para cada tipo de formiga.
- Por exemplo, soldiers podem patrulhar e defender, enquanto workers coletam alimento.

4.8.3 Interações entre Papéis

- Implementar lógica para como diferentes tipos de formigas interagem e colaboram.

4.9 Recursos do NetLogo que Facilitam Essas Implementações

4.9.1 Patches e Turtles

- O NetLogo diferencia entre agentes móveis (turtles) e estáticos (patches), permitindo modelar tanto agentes quanto o ambiente.

4.9.2 Variáveis de Agentes

- É possível adicionar variáveis personalizadas a patches e turtles para armazenar informações necessárias.

4.9.3 Procedimentos e Regras

- Você pode definir procedimentos para especificar comportamentos complexos dos agentes.

4.9.4 Comunicação entre Agentes

- Agentes podem interagir usando comandos como ask, in-radius, turtles-here.

4.9.5 Visualização e Interface

- O NetLogo permite visualizar facilmente o estado do modelo e fornece ferramentas para criar interfaces interativas com sliders, botões e monitores.

4.9.6 Extensões e Bibliotecas

- O NetLogo suporta extensões que permitem funcionalidades adicionais, como suporte a gráficos avançados ou integração com outras ferramentas.

5. Critérios de Avaliação

5.1 Criatividade e Originalidade (25%)

- Inovação nas modificações propostas.
- Relevância das alterações em relação aos conceitos de IA.

5.2 Implementação Técnica (25%)

- Qualidade e organização do código.
- Funcionamento correto e eficiente do modelo modificado.

5.3 Análise e Reflexão (25%)

- Profundidade na análise dos resultados e comportamentos observados.
- Conexão entre a prática realizada e os conceitos teóricos.

5.4 Comunicação (25%)

- Clareza e coerência no relatório escrito.
- Eficácia e organização na apresentação oral.

BOA SORTE 🤗

Dúvidas: grupo da disciplina ou thales.l.a.valente@gmail.com