

LEIM – Modelação e Simulação de Sistemas Naturais

Exame Primeira Época

24 janeiro 2017

Grupo I – Sistemas Dinâmicos (5.0 val.)

Considere o seguinte modelo que descreve a evolução de uma determinada população e onde se tem:

$$[\text{nascimentos}] = [\text{taxa de natalidade}] * [\text{População}]$$

$$[\text{densidade}] = [\text{População}] / [\text{Capacidade de carga}]$$

$$[\text{taxa de natalidade}] = [\text{taxa de natalidade máxima}] * (1 - [\text{densidade}])$$

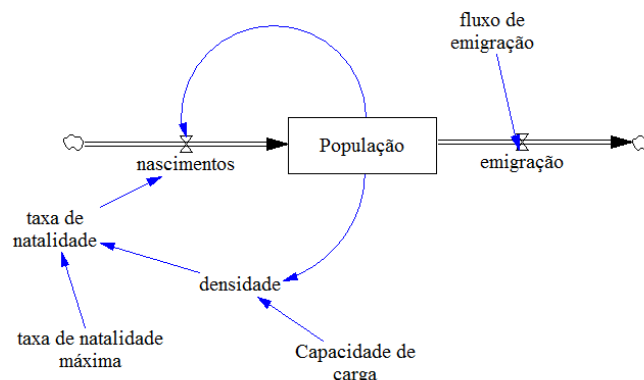
$$[\text{emigração}] = [\text{fluxo de emigração}]$$

$$[\text{taxa de natalidade máxima}] = 0.02$$

$$[\text{Capacidade de carga}] = 1000$$

$$[\text{fluxo de emigração}] = 0$$

Nota: a unidade de tempo é [mês]



- Complete o diagrama marcando cada seta com a polaridade respetiva. Quantos ciclos de feedback contém o sistema? Identifique-os e classifique-os. [1.0]
- Descreva o sistema através da respetiva equação diferencial. Quantos pontos fixos tem o sistema? Classifique-os quanto à estabilidade. Justifique. [0.8]
- Como evoluirá o sistema ao longo do tempo? Esboce um gráfico com essa evolução, assumindo duas situações: i) população inicial pequena; ii) população inicial grande. Para cada zona do gráfico identifique qual o ciclo de feedback que é dominante. Justifique. [0.7]
- Considere agora que a variável [fluxo de emigração] é igual a 5. Explicite as unidades em que é expressa esta variável e repita a alínea b) [1.0]
- Nas condições da alínea anterior, simule a evolução da população nos primeiros dois meses, assumindo uma população inicial de 800 pessoas. [0.7]
- Suponha um modelo mais realista para a emigração, deixando de ser uma constante (variável exógena) mas passando a depender da densidade populacional. Acrescente a respetiva seta ao modelo, identificando a sua polaridade. O sistema passou a ter mais um ciclo de feedback? Justifique. [0.8]

Grupo II – Autómatos Celulares (5.0 val.)

Responda às seguintes questões relativas a autómatos celulares:

- Considere um autómato celular 1D, totalístico, quaternário, com vizinhança $r=2$. Qual a dimensão da tabela de regra? Quantas são as diferentes regras existentes? [1.0]
- E se o autómato anterior fosse totalístico externo? Qual seria agora a dimensão da tabela de regra? Justifique. [1.0]
- Considere a forma compacta de expressar o funcionamento do Jogo da Vida, dizendo que se trata da aplicação da regra 23/3. Explique o significado desta expressão. Será que todos os autómatos celulares binários e totalísticos externos podem ser expressos desta forma compacta? Justifique. [1.0]
- Considere o Jogo da Vida, versão clássica, regra 23/3, com a seguinte configuração inicial. Determine a evolução do autómato. Comente o resultado obtido. [2.0]

				1			
		1		1			
						1	
	1	1	1	1	1		
						1	
		1		1			
				1			

Grupo III – Agentes Autónomos (5.0 val.)

Considere cinco agentes autónomos cujos atributos, para $t=0$, estão expressos na seguinte tabela. Além disso, sabe-se que cada agente tem visão omnidirecional, com raio igual a 4, e uma velocidade máxima igual a 5.

	Atributos ($t=0$)
Agente 1	Localização: (10,10) Velocidade: (4,-2) Massa: 2
Agente 2	Localização: (10,13) Velocidade: (4,-2) Massa: 10
Agente 3	Localização: (14, 12) Velocidade: (-4, -2) Massa: 6
Agente 4	Localização: (8, 6) Velocidade: (2, -4) Massa: 1
Agente 5	Localização: (13, 10) Velocidade: (5, 0) Massa: 4

- Represente graficamente, no espaço 2D, as posições e velocidades dos agentes. Assinale também o campo de visão do agente 1. [1.0]
- Determine a força de guiamento, aplicada no agente 1, supondo que ele executa o método SEEK, sendo o seu alvo o agente 2. Justifique todos os cálculos realizados. [1.2]
- Determine a força de guiamento, aplicada no agente 1, supondo que ele executa o método COHESION. Justifique os cálculos realizados. [1.3]

- d) Nas condições da alínea anterior e supondo um ritmo de simulação de 10 fps, determine a nova localização e velocidade do agente 1, na frame seguinte. Justifique. [1.5]

Grupo IV – Programação (5.0 val.)

Atente ao seguinte código para a classe **LSystem**, implementada nas aulas, bem como ao subsequente programa principal.

1	class LSystem {
2	String sentence;
3	Rule[] ruleset;
4	LSystem(String axiom, Rule[] ruleset){
5	sentence = axiom;
6	this.ruleset = ruleset;}
7	void nextGeneration() {
8	...
9	}
10	}

1	LSystem s1, s2;
2	String axiom = "A";
3	void setup() {
4	Rule[] rule = new Rule[1];
5	rule[0] = new Rule('A', "ABA");
6	rule[1] = new Rule('B', "AC");
7	s1 = new LSystem(axiom, rule);
8	// axiom = "B";
9	// s2 = s1;
10	for (int i=0;i<2;i++) s1.nextGeneration();
11	println(s1.sentence);}

- a) O método **setup** tem um erro. Identifique-o e corrija-o. Esse erro manifestar-se-ia em tempo de compilação ou em tempo de execução? Justifique. [1.0]
- b) Tendo corrigido o erro, determine qual a mensagem que é enviada para a consola durante a execução do programa [1.0]
- c) Repita a alínea anterior mas descomentando as linhas 8 e 9 do método **setup**. Quantos objetos da classe **LSystem** foram criados durante a execução? [1.0]
- d) Escreva um método para a classe **LSystem** – **numberOfAs** – que retorne o número de 'A' contidos na sequência de caracteres. [1.0]
- e) Escreva o código do construtor da classe **Rule** que seja compatível com a funcionalidade pretendida (**Nota**: escreva apenas o código do construtor) [1.0]