

Universidade Virtual Africana

INFORMÁTICA APLICADA: ITI 4303

SIMULAÇÃO E MODULAÇÃO

Fernando Sanha

Prefácio

A Universidade Virtual Africana (AVU) orgulha-se de participar do aumento do acesso à educação nos países africanos através da produção de materiais de aprendizagem de qualidade. Também estamos orgulhosos de contribuir com o conhecimento global, pois nossos Recursos Educacionais Abertos são acessados principalmente de fora do continente africano.

Este módulo foi desenvolvido como parte de um diploma e programa de graduação em Ciências da Computação Aplicada, em colaboração com 18 instituições parceiras africanas de 16 países. Um total de 156 módulos foram desenvolvidos ou traduzidos para garantir disponibilidade em inglês, francês e português. Esses módulos também foram disponibilizados como recursos de educação aberta (OER) em oer.avu.org.

Em nome da Universidade Virtual Africana e nosso patrono, nossas instituições parceiras, o Banco Africano de Desenvolvimento, convido você a usar este módulo em sua instituição, para sua própria educação, compartilhá-lo o mais amplamente possível e participar ativamente da AVU Comunidades de prática de seu interesse. Estamos empenhados em estar na linha de frente do desenvolvimento e compartilhamento de recursos educacionais abertos.

A Universidade Virtual Africana (UVA) é uma Organização Pan-Africana Intergovernamental criada por carta com o mandato de aumentar significativamente o acesso a educação e treinamento superior de qualidade através do uso inovador de tecnologias de comunicação de informação. Uma Carta, que estabelece a UVA como Organização Intergovernamental, foi assinada até agora por dezenove (19) Governos Africanos - Quênia, Senegal, Mauritânia, Mali, Costa do Marfim, Tanzânia, Moçambique, República Democrática do Congo, Benin, Gana, República da Guiné, Burkina Faso, Níger, Sudão do Sul, Sudão, Gâmbia, Guiné-Bissau, Etiópia e Cabo Verde.

As seguintes instituições participaram do Programa de Informática Aplicada: (1) Université d'Abomey Calavi em Benin; (2) Université de Ougadougou em Burkina Faso; (3) Université Lumière de Bujumbura no Burundi; (4) Universidade de Douala nos Camarões; (5) Universidade de Nouakchott na Mauritânia; (6) Université Gaston Berger no Senegal; (7) Universidade das Ciências, Técnicas e Tecnologias de Bamako no Mali (8) Instituto de Administração e Administração Pública do Gana; (9) Universidade de Ciência e Tecnologia Kwame Nkrumah em Gana; (10) Universidade Kenyatta no Quênia; (11) Universidade Egerton no Quênia; (12) Universidade de Addis Abeba na Etiópia (13) Universidade do Ruanda; (14) Universidade de Dar es Salaam na Tanzânia; (15) Université Abdou Moumouni de Niamey no Níger; (16) Université Cheikh Anta Diop no Senegal; (17) Universidade Pedagógica em Moçambique; E (18) A Universidade da Gâmbia na Gâmbia.

Bakary Diallo

O Reitor

Universidade Virtual Africana

Créditos de Produção

Autor

Fernando Sanha

Par revisor(a)

Arlete Ferrão

UVA - Coordenação Académica

Dr. Marilena Cabral

Coordenador Geral Programa de Informática Aplicada

Prof Tim Mwololo Waema

Coordenador do módulo

Florence Tushabe

Designers Instrucionais

Elizabeth Mbasu

Benta Ochola

Diana Tuel

Equipa Multimédia

Sidney McGregor

Michal Abigael Koyier

Barry Savala

Mercy Tabi Ojwang

Edwin Kiprono

Josiah Mutsogu

Kelvin Muriithi

Kefa Murimi

Victor Oluoch Otieno

Gerisson Mulongo

Direitos de Autor

Este documento é publicado sob as condições do Creative Commons

[Http://en.wikipedia.org/wiki/Creative_Commons](http://en.wikipedia.org/wiki/Creative_Commons)

Atribuição <http://creativecommons.org/licenses/by/2.5/>



O Modelo do Módulo é copyright da Universidade Virtual Africana, licenciado sob uma licença Creative Commons Attribution-ShareAlike 4.0 International. CC-BY, SA

Apoiado por



Projeto Multinacional II da UVA financiado pelo Banco Africano de Desenvolvimento.

Índice

Prefácio	2
Créditos de Produção	3
Direitos de Autor	4
Apoiado Por	4
Descrição Geral do Curso	6
Bem-vindo(a) a O objetivo deste curso é fornecer as noções básicas de	6
Pré-requisitos	6
Materiais.	6
Objetivos do Curso Após concluir este curso, o(a) aluno(a) deve ser capaz de: . . .	6
Unidades.	7
Unidades 0.	7
Avaliação.	8
Calendarização.	8
Leituras e outros Recursos.	10
Unidade 0: Avaliação Diagnóstica (Introdução aos sistemas)	11
Introdução à Unidade	11
Objetivos da Unidade	11
Termos-chave	11
Avaliação da Unidade	12
Unidade 1: Introdução à Modelagem e Simulação	14
Introdução à Unidade	14
Objetivos da Unidade	14
Actividades de Aprendizagem.	14
Actividade 1 Introdução à Modelagem	14
Termos-chave	14
Detalhes da actividade	15
Conclusão	22
Actividade 2 - Os formalismos simulação	23

Detalhes da atividade	23
Conclusão	25
Resumo da Unidade	25
Avaliação da Unidade	26
Unidade 2: Modelagem de Sistemas Contínuos Com Base em Agentes	27
Introdução à unidade	27
Objetivos da unidade	27
Atividades de aprendizagem	27
Atividade 1 – Descoberta de sistemas, alguns exemplos concretos	
Introdução	27
Detalhes de actividade	27
Conclusão	35
Atividade 2 – Os agentes de modelização baseado	36
Detalhes da actividade	36
Conclusão	42
Atividade 3 – Uma base Vs equipe um equações básicas	43
Detalhes da actividade	43
Resumo da Unidade	55
Avaliação da Unidade	56
Unité 3. Modelagem de Sistemas Dinâmicas (Dinâmica de Sistamas)	57
Introdução a unidade	57
Objective da unidade	57
Atividades de aprendizagem	57
Atividade 1 - Apresentação do módulo SD Netlogo	57
Termos-chave	57
Detalhes da atividade	58
Conclusão	59
Atividade 2 - A operação do módulo SD	59
Detalhes da atividade	59
Conclusão	62

Detalhes da atividade	62
Resumo da Unidade	69
Avaliação da Unidade	69
Unité 4: Modelagem de Agentes Descreto	70
Introdução a unidade	70
Objectivos da unidade	70
Actividade 1 – Definição de modelagem de agentes discreto	70
Termos-chave	70
Detalhes da atividade	71
Conclusão	74
Actividade 2 – Representação de todos os elementos de um modelo de eventos discretos	74
Detalhes de atividades	75
Conclusão	83
Resumo da Unidade	83
Avaliação da Unidade	84

Descrição Geral do Curso

Bem-vindo(a) a O objetivo deste curso é fornecer as noções básicas de

sistemas de modelagem e simulação

O principal objetivo é se apropriar dos fundamentos desta área de design para a realização de simulações. Você será mostrado os diferentes tipos de sistemas e modelos que possam surgir. Em seguida, software de simulação permitirá que você avance para a execução do modelo de análise estatística e visualização de resultados para uma melhor compreensão do sistema de destino.

Pré-requisitos

1. Cálculo Diferencial e Integral.
2. Introdução à probabilidade e estatística
3. Álgebra linear
4. Algoritmos e estruturas de dados
5. Projeto e Análise de Algoritmos

Materiais

1. Os materiais necessários para completar este curso incluem:
2. O software Netlogo: <https://ccl.northwestern.edu/netlogo/>
3. O software anylogic : <http://www.anylogic.com>

Objetivos do Curso Após concluir este curso, o(a) aluno(a) deve ser capaz de:

O objetivo deste curso é fornecer as noções básicas de sistemas de modelagem e simulação. O principal objetivo é se apropriar dos fundamentos desta área de design para a realização de simulações. Você será mostrado os diferentes tipos de sistemas e modelos que possam surgir. Em seguida, software de simulação permitirá que você avance para a execução do modelo de análise estatística e visualização de resultados para uma melhor compreensão do sistema de destino.

Unidades

Unidades 0

Unidade 0: Diagnóstico Avaliação Diagnóstica (Geral Introdução aos Sistemas)

Esta unidade fornece uma introdução geral ao conceito de sistema, sistema de entidades, relações e interações entre as entidades de sistemas, tipos de sistemas

Unidade 1: Introdução à Modelagem e Simulação

Esta unidade introduz os conceitos de modelagem e simulação, análise de sistemas, e fornece uma classificação das diferentes abordagens de modelização: modelagem de sistemas contínuos com base em agentes, modelagem de sistemas dinâmicos modelagem e eventos (Dinâmica de Sistemas) discreta.

Unidade 2: Modelagem de sistemas contínuos com base em agentes

Nesta unidade, se trata de estar interessado na modelagem de sistemas de tempo contínuo. As equações diferenciais (equações diferenciais ordinário, equações diferenciais parciais) são matemáticas ferramenta de modelagem clássicos tais sistemas, mas mostrar a implementação de tais sistemas no computador através do uso de uma simulação de software baseado em agentes, Netlogo.

Unidade 3: Modelagem de sistemas dinâmicos (Dinâmica de Sistemas)

Esta unidade trata de sistemas dinâmicos (Sistemas Dinâmica), que são um tipo de modelagem onde o objetivo é entender como os elementos estão em relacionamentos uns com os outros. Esta abordagem diferente, baseada em agentes que vamos ver na Unidade 2 ainda será processada pelo mesmo software. De fato, em Netlogo, há um módulo para simular os sistemas Dinâmica. Esta unidade é baseada em sistemas manuais do usuário Dinâmica NetLogo.

Unidade 4: Modelagem de Eventos Discretos

Esta unidade discute a última formalismo de modelagem; modelagem de eventos discretos. Ele lhe dará os conceitos-chave em torno de que a modelagem ao especificar as estruturas necessárias para realizar tal modelação (agentes, fila, relógio ...). Um exemplo será tratado para ajudar a identificar esse formalismo.

Avaliação

Em cada unidade encontram-se incluídos instrumentos de avaliação formativa a fim de verificar o progresso do(a)s aluno(a)s.

No final de cada módulo são apresentados instrumentos de avaliação sumativa, tais como testes e trabalhos finais, que compreendem os conhecimentos e as competências estudadas no módulo.

A implementação dos instrumentos de avaliação sumativa fica ao critério da instituição que oferece o curso. A estratégia de avaliação sugerida é a seguinte:

1	Mesa de plantão	25%
2	Projeto de equipe	25%
3	Mesa de exame	50%

Calendarização

Unidade	Temas e Atividades	Estimativa do tempo
Introdução à Modelagem e Simulação	Introdução à modelagem Compreender o processo de modelagem Descubrir os sistemas complexos Assimilar o modelo de conceito Reges baseados modelagem e leis baseadas em modelagem Os formalismos simulação Simulação discreta ou contínua, determinista vs. estocástica O tempo e espaço nas simulações Apresentação de três (3) grande de modelagem e simulação famílias Curso	30h

Modelagem de sistemas contínuos com base em agentes	Descobertas de sistemas, alguns exemplos concretos Classificar Sistemas seguintes entidades e interações. Veja os limites da resolução digitais Observe que o computador pode trazer Baseado no agente de modelos Agente Sistemas multi-agentes (MAS) Plataformas SMA Outros formalismo, meta-heurísticas e estruturas Informáticas Uma base Vs equipe um equações básicas Comparação das duas abordagens Exemplo Modelagem (presas / predadores)	30h
Modelagem de sistemas dinâmicos (Dinâmica de Sistemas)	Apresentação do módulo SD Netlogo Conceitos básicos Exemplos de modelos SD O funcionamento do módulo SD Diagramas Elementos do diagrama O código de procedimentos Implementação do modelo de rapina / Predadores	30h

Modelagem de Eventos Discretos	Definição de modelagem de eventos discretos Localize o evento discreto modelagem Algumas noções em torno de modelagem de eventos discretos Representação de todos os elementos de um modelo de eventos discretos Simulação de comparação Eventos Discretos Vs DC Descubra as filas e variáveis aleatórias discretas Exemplo Modelagem (caixas automáticos dos bancos)	30h
--------------------------------	--	-----

Leituras e outros Recursos

As leituras e outros recursos deste curso são:

Unidade 0

Leituras e outros recursos obrigatórios estão listados abaixo:

- P. A. Fishwick, Simulation Model Design and Execution, Englewood Cliffs, New Jersey: Prentice-Hall, 1995 (QA76.9.C65F57).
- J. Banks, J. C. Carson, II, and B. L. Nelson, "Discrete-Event System Simulation," 2nd Edition, Prentice Hall, 1996, ISBN 0-13-217449-9.
- W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, Numerical Recipes in C, Second Edition, New York: Cambridge University Press, 1992 (QA76.73.C15N865 1992).
- M. C. Jeruchim, P. Balaban, and K. S. Shanmugan, Simulation of Communication Systems, Plenum Press, 1992 (TK5102.5.J47).
- P. Bratley, B. L. Fox, and L. Schrage, A Guide to Simulation, New York: Springer-Verlag, 1987 (QA76.9.C65B73 1987).
- G. Gordon, System Simulation, Prentice Hall, 2nd edition, 1977 (QA76.5.G63 1978).
- M. Law and W. D. Kelton, Simulation Modeling & Analysis, New York: McGraw-Hill, 1991 (QA76.9.C65L38 1991).

Unidade 0: Avaliação Diagnóstica (Introdução aos sistemas)

Introdução à Unidade

O propósito desta unidade é verificar a compreensão dos conhecimento que possui relacionados com este curso.

Esta unidade permitirá que você verificar o conhecimento que você precisa antes de iniciar o curso. Você pode fazer a avaliação da unidade antes de atividades de aprendizagem para ajudar a atualizar seu conhecimento.

Objetivos da Unidade

Após a conclusão desta unidade, deverá ser capaz de:

1. No final dessa unidade, você deve ser capaz de:
2. Identificar vários elementos em interações ou relacionamentos
3. Conhecer as propriedades de um sistema

Termos-chave

Um conjunto de entidades relacionadas ou interativas.

Diagramas

Elementos do diagrama

O código de procedimentos Implementação do modelo de rapina / Predadores

Modelagem de eventos discretos.

Avaliação da Unidade

Verifique a sua compreensão!

Mesa de plantão 0

Instruções

- Responda às perguntas consultando os recursos livres
- Internet

Critérios de Avaliação

A escala é indicada seguintes perguntas

Avaliação

Perguntas de conhecimentos gerais (20 pts)

Um conjunto (0,5 pt)

O que é uma relação (0,5 pt)

O que é uma interação (0,5 pt)

Feedback, programado (0,5 pt)

Dá as definições dos termos:

Singularidade (1pt)

Adaptação (1pt)

Resiliência (1pt)

Bifurcação (1pt)

Estado crítico (1pt)

de transição de fase (1 pt)

Teleonomia (1pt)

Autopoiese (1pt)

Uma vez que esses termos definidos escolher um sistema exemplar e dizer para cada período para o qual ele está em seu sistema (10 pts)

Leituras e Outros Recursos

As leituras e outros recursos desta unidade encontram-se na lista de "Leituras e Outros Recursos do curso".

Unidade 1: Introdução à Modelagem e Simulação

Introdução à Unidade

Esta unidade introduz os conceitos de modelagem e simulação, análise de sistemas, e fornece uma classificação das diferentes abordagens de modelização: Modelagem de Eventos Discretos, modelagem de sistemas contínuos e modelagem de sistemas heterogêneos

Objetivos da Unidade

Após a conclusão desta unidade, deverá ser capaz de:

1. Entenda o processo de modelização.
2. Definir um sistema complexo.
3. Entenda o modelo conceito.
4. Assimilar conceitos de simulação.
5. classificar diferentes simulação formalismo.

Termos-chave

Modelagem: modelagem está projetando um modelo

Modelo: O que é dado para referência

Simulação: representação ou reprodução do comportamento

dado sistema

Actividades de Aprendizagem

Actividade 1 Introdução à Modelagem

Introdução

Esta actividade irá familiarizá-lo com os sistemas de modelagem. Ele irá apresentar as questões de modelagem e todo o seu processo. Você também vai descobrir os sistemas de conceitos e complexos modelo.

Detalhes da actividade

Compreender o processo de modelagem

Os modelos podem ser definidos como um processo, para um sistema dado, para a produção de um modelo que é (Fig. 1):

“Uma imagem simplificada da realidade forjada a partir de uma determinada seleção de dados observacionais e uma série de suposições”

Coquillard, P., Hill, D., 1997. Modelagem e ecossistema de simulação. Masson.

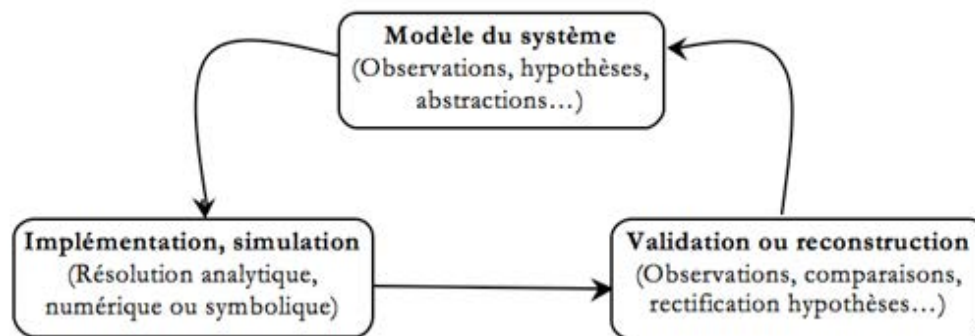


Fig. 1 - As principais etapas de um processo de sistemas de modelagem e simulação

O processo termo é usado aqui para significar que o modelo está a ser construída, colocando-o à prova por experimentos (simulações) para endossar ou para refiná-lo, verificando novamente observações ou suposições feitas.

Deve-se notar neste ponto que um modelo nunca é concluída. Ele simplesmente reflete o estado de conhecimentos e deverá colher melhorar estes por exemplo:

- O modelo geocêntrico de Ptolomeu e Aristóteles ao modelo heliocêntrico - provavelmente obsoleto - defendida por Copérnico, Kepler, Galileu e Newton.
- A lei da inércia de Newton, válido para os valores de referência de Galileu, relatividade geral de Einstein, ...
- para citar apenas estes exemplos bem conhecidos ...

Daremos os seguintes lembretes ou definições de certos conceitos:

- Os sistemas complexos (que muitas vezes requerem abordagem de modelagem e simulação)
- O reducionismo e o holismo,
- Um modelo de sistema,
- para nos permitir definir e re-especificar o nosso quadro de modelagem /
- simulação.

Explorar sistemas complexos

Etimologicamente, "sistema" vem do sistema palavra grega significa montagem ou composição e significa, em nosso contexto estudo, um conjunto de entidades que interagem.

Além disso, ao olhar para a palavra "complexo" que parece levantar uma tautologia, porque esta última, a partir da palavra latina complexus de complecti, significa algo que contenha ou combina vários elementos diferentes.

Adiciona-se então que estes elementos juntos e manter vários relatórios apresentam frequentemente aspectos emergentes difíceis de entender com a mente.

Podemos agora avançar uma definição ampla, mas que ainda é informal no campo de sistemas complexos pela natureza multidisciplinar combina várias ferramentas, abordagens e técnicas de exploração. No entanto, de acordo com a área de aplicação, os conceitos de "entidades", "interações" e sensação "emergente" será uma sensação mais formal e aplicar rigorosamente o tema escolhido.

Um sistema complexo é um conjunto de entidades que interagem relativamente simples, e em que emerge um comportamento imprevisível. exemplos atuais da literatura ajudar a ter uma idéia melhor sobre os sistemas complexos. Aqui estão algumas delas:

- O cérebro humano feito de neurônios trocando pulsos elétricos; o estudo detalhado da natureza bioquímica ou função dos neurônios não é uma informação clara sobre as características do cérebro, como memória ou consciência.
- Os sistemas sociais em que observamos uma grande variedade de interações entre os indivíduos e os comportamentos emergentes complexas (exemplo: Fig. 2).
- Uma colônia de formigas troca de feromônios para resolver problemas (3 fig.).
- O fluxo turbulento de partículas de fluido descritos pelas transportadoras vorticidade (Fig. 4).

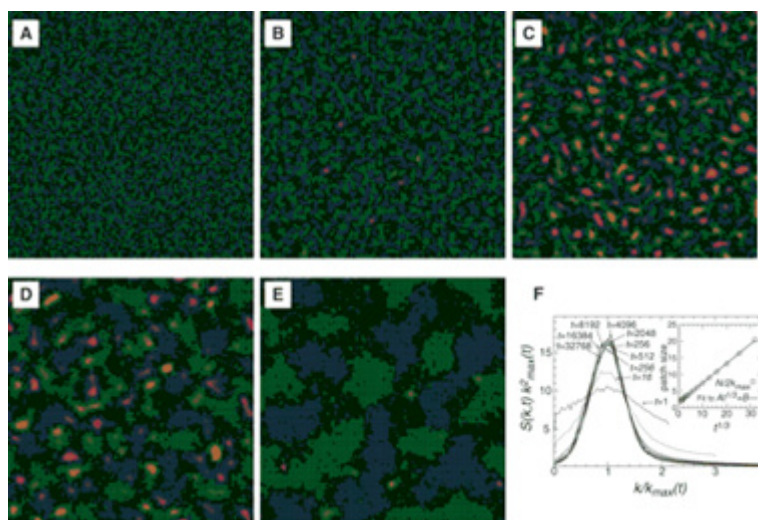


Fig. 2 - grupos de separação étnicas ou culturais e potenciais conflitos,

Lim, M. Metzler, R., Bar-Yam, Y., 2007. Global de formação de padrões e violência étnica / cultural. Science, Vol. 317. no. 5844, pp. 1540-1544.

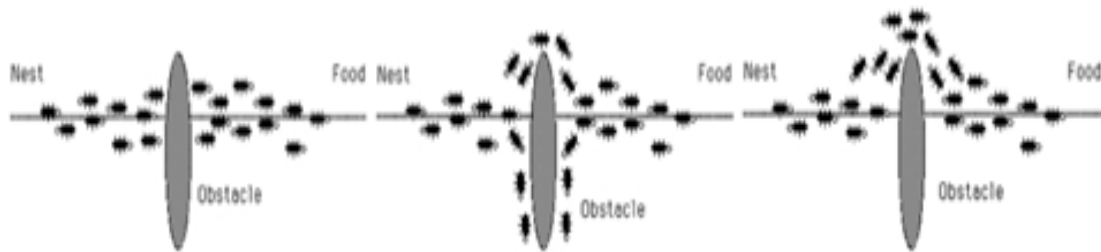


Fig. 3 - As formigas que determinam o caminho mais curto entre o seu ninho e alimentos através do depósito de feromônios

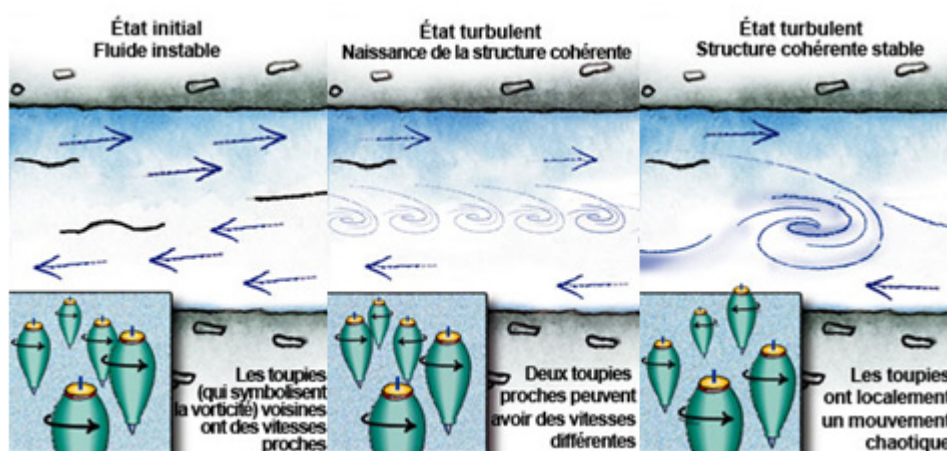


Fig. 4 - surgimento de uma estrutura de vórtice coerentes em um fluxo turbulento.

Robert, R., 2001. O efeito borboleta é ido, para a ciência.

Reduccionismo e o holismo

Note-se que o que é colocado para a frente através dessas ilustrações, é o surgimento dentro de sistemas complexos de certas propriedades que podem ser descritas como imprevisível se uma parte da definição funcional de entidades tomados separadamente.

Esta não-linearidade entre as ações de entidades individuais e o comportamento coletivo resultante mina a teoria reducionista adotada por Descartes em seu famoso texto filosófico: Discurso do Método.

O raciocínio do reduccionismo é que o estudo e controle das entidades são suficientes para entender o comportamento do sistema em questão.

Tal não é o caso dos sistemas complexos que são bastante próxima, do ponto de vista de seus comportamentos emergentes, a teoria holística Smuts:

“A tendência para formar buracos na natureza que são maiores do que a soma das partes através da evolução criativa”.

Smuts, J. C., 1926. Holismo e Evolução. Macmillan & Co Ltd.

Teoria que é comumente expressa como dizendo:

“O todo é maior do que a soma das suas partes.”

Portanto, essa visão particular para considerar os sistemas como conjuntos para tentar descrever o comportamento. Os conceitos de antagonistas holismo e assim não pode ser reductionism uma dicotomia de abordagens para modelar sistemas complexos, e é necessário, neste contexto, para tirar partido dos dois para o todo e as partes estão ligadas como os destaques Bar -Yam [Bar-Yam, 1997]:

O observador pode prever a evolução do sistema complexo, com uma soma simples das entidades de conhecimento. É então necessário para simular o comportamento do sistema, utilizando um processo de modelagem. Que relatórios do holismo que está interessado em seu assunto como constituindo qualquer

“Qualitativamente, para entender o comportamento de um sistema complexo, devemos entender não apenas o comportamento do propósito ações como eles agem em conjunto para formar o comportamento do todo. É porque não podemos descrever o todo, sem a descrição de cada lado, e porque cada quota deve ser descrito em relação à --outras partes Isso sistemas complexos são dificuldade de entender. ”.

Bar-Yam, Y., 1997. dinâmica de sistemas complexos. Livros Grupo Perseus.

A abordagem aqui proposta pode ser descrito como sistêmica; uma forma inspirada por um sentido da cibernética de Wiener.

Devemos, portanto, usar essa abordagem possível neste curso

Além de um controle teleológico (feedback), instituído pela (ou através de) o fluxo de informações entre os componentes funcionais, abordagem de sistemas considera suas características estruturais e as trocas que não estão exclusivamente sob a forma de informação, mas também de matéria ou energia Consulte a trabalhar na teoria geral do sistema de Von Bertalanffy.

Assimilar o modelo de conceito

Como discutido abaixo, a natureza dos problemas abordados podemos optar por construir um modelo. Com efeito, é difícil ou muito caro para conduzir uma abordagem experimental diretamente sobre os sistemas estudados. O modelo desenvolvido em vez serve então não só como um “barômetro” do conhecimento, mas também um catalisador na compreensão do sistema que está sendo modelado.

Ciência usa os resultados da teoria de sinal e informação para desenvolver um método de análise e síntese de sistemas complexos, suas relações funcionais e mecanismos de controle, biologia, economia, informática, etc. (Fonte: Centro Nacional de Recursos Textuais e <http://www.cnrtl.fr/Lexical>)

Também ele fornece ao utilizador um encontro laboratório virtual com detalhes sobre os problemas difíceis de tratar experimentalmente; ou porque o sistema não é apenas conhecida ou porque as medições experimentais podem, em alguns casos de corrupção a integridade do sistema (se fenômenos turbulentos por exemplo).

Suplemento com a definição do modelo e da modelagem baseada em leis ou regras básicas. Este modelo conceito também foi introduzido em um pouco mais cedo este curso como sendo implicitamente ligada a abordagens holísticas, reducionistas ou sistêmicos no sentido de que o último ficar como desenvolver um. Mais formalmente, retornando à sua etimologia do modelo *Modellus Latina populares italiana modello* significa aquilo que serve, ou deveria servir como um objeto de imitação para reproduzir ou algo assim. Ele pode ser facilmente compreendido o seu significado em nosso contexto, mas com a seguinte declaração Minsky também notamos sua natureza subjetiva.

“Para uma observação B, um objeto A^* é um modelo de um objeto A na medida em que B pode usar A^* para responder a uma pergunta sobre Esse interesse $_{him_A}$ ”

Minsky, M., 1965. assunto, mentes e modelos. Federação Internacional de Informação do processamento do Congresso, 1: 45-49.

Com efeito, isto significa que o objecto A^*B é para o observador um modelo assim que é utilizável. Então, dependendo do campo de estudo da B e as perguntas que ele pede, então podemos para o mesmo objeto Uma vantagem para vários modelos.

Não podemos neste nível fornecer um procedimento geral para a construção de um modelo. É necessário em primeiro lugar para especificar as dimensões de interesse e os objectivo do sistema de estudo e aperte a definição da gama de modelos e optar por ferramentas e métodos de modelagem. Outra característica de um modelo é a sua simplicidade em comparação com o sistema estudado. Demasiado complexo um modelo é inútil se não muito perto do sistema modelado e ser inutilizável como o sistema em questão é difícil de entender. Como também um modelo simplista não é utilizável porque impertinente questões que podem ser levantadas pelo estudo. Um modelo deve, portanto, ser uma projeção da realidade efetivamente definir o interesse ou dimensões desejadas.

Modelagem baseada em regras e leis baseadas em modelos

A modelagem baseada em regras é a de considerar o sistema através de seus elementos estruturais e seguir a sua evolução de acordo com as regras de interações e interdependências que ligam os elementos. A evolução do sistema é discreto e depende de interações entre os elementos.

muitas vezes se opõem a esta modelagem baseada em regras que se baseia em leis. Estes modelos consistem em vez de considerar algumas características dos elementos como variáveis e estudo de formulação de equações que descrevem a evolução de toda a legislação do sistema. O resultado, como veremos na abordagem de modelagem matemática, equações geralmente contínuas (no tempo e no espaço) digitar equações diferenciais.

Além disso, a análise numérica, nós temos métodos de simulação de sistemas físicos que podem queridos geralmente lineares ou não-lineares, estacionárias e instáveis, laminar ou turbulento ... E como Bourdine observação:

"(...) Em menos de três séculos, os físicos matemáticos realmente colocar a equação mundo. (...) Desordem é a equação de Kepler em mecânica celeste, a equação da gravitação de Newton, a equações hidráulica Bernoulli, o princípio da d'Alembert na dinâmica, equações de Euler e outros mecanismos dinâmicos de fluidos, as equações de Lagrange que resumem as mecânica analítica, contribuições de Gauss em mecânica (princípio da menor restrição) em teoria do potencial, eletricidade e magnetismo, a equação de Fourier propagação do calor, as equações de Navier-Stokes de mecânica dos fluidos, equações do electromagnetismo e da teoria da luz de Maxwell, a equação de Boltzmann na mecânica estatística, equação de Schrödinger da mecânica quântica (...). »

Bullseye, J-P., 2001. Homo mathematicus. Matemática e nós. Vuibert.

Note-se que a comunidade matemática produz um monte de equações, ver concretamente como tais equações são construídos.

O equacionamento do modelo contínuo

Lembre-se de novo, e isso não é exclusivo de modelagem matemática, um modelo é uma abstração da realidade e dá apenas uma representação parcial e relevante do sistema real estudado. Este último, como já apontado, em seguida, passa por uma projeção sobre as dimensões escolhido como o mais revelador, mas também o mais acessível e interpretável.

A observação do sistema e a sua representação em seguida, correspondem a uma fase selectivo que destaca certas características do sistema (Fig. 6). Isso torna ainda uma vez em conta a natureza subjetiva de um modelo, mas, em contrapartida, deve opor-se a visão de um tal sistema não pode ser abrangente, e muito menos a sua representação.

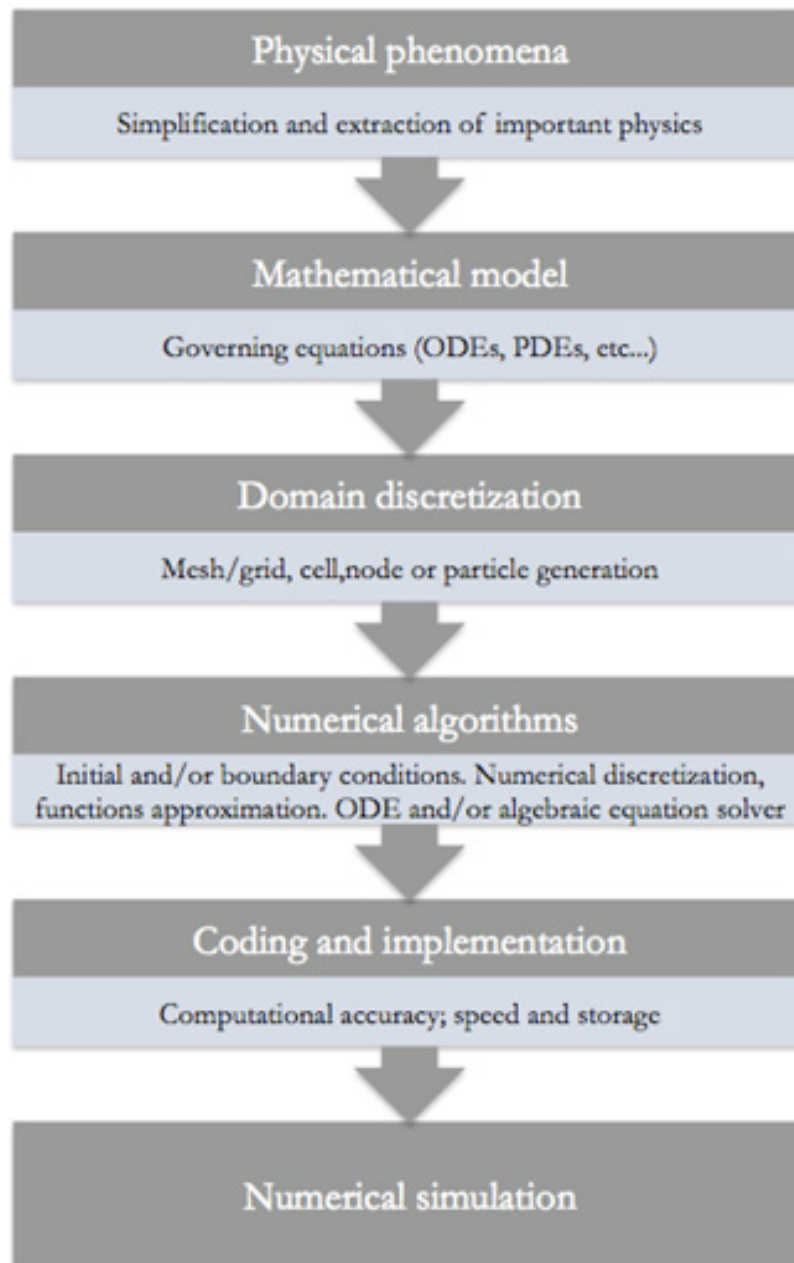


Fig. 5 - O procedimento para a realização de uma simulação digital

Um modelo de sistema físico dinâmico, em seguida, irá descrever de forma mais explícita a evolução das características selecionadas como aqueles que podem representar fielmente o sistema:

- Velocidade, altura, pressão ... em um fluxo de fluido,
- Movimentação e restrições para um problema de elasticidade,
- A concentração para um problema de transporte ...

Estes indicadores são então explicados como variável normalmente contínua no espaço e no tempo e cuja integração, de acordo com as leis de conservação de massa, quantidade de movimento, o fluxo de calor

geralmente fornece equações como equações diferenciais parciais ou equações diferenciais ordinárias (ODE).

Essas equações governantes de sistemas físicos dinâmicos exigem mais conhecimento das condições iniciais (CI) do fenômeno e as condições de contorno (CB) do sistema de domínio (Dirichlet, Neumann e Robin).

Uma vez que esses elementos representava a existência e unicidade da solução de equação governante pode ser avançado. Resolvendo esta equação determina o regime em seguida, equivalente a uma simulação do comportamento (ou dinâmico) do sistema.

Ele rastreia os valores numéricos de características selecionadas em qualquer ponto do sistema se uma solução analítica da equação é possível.

Nós aqui no risco de repetir-nos, nós entregamos o mesmo exercício da recapitulação como Bullseye, mas apenas alguns, no entanto, citando sistemas que regem EDP. Na física, e isso não é exaustiva:

- As equações de Poisson e Laplace para o equilíbrio ou fenômenos estacionária: uma membrana equilibrada
- A equação do calor para os fenômenos evolução: difusão de calor, espécies mistas
- A equação de onda para os fenômenos de propagação: membrana vibratória, ondas eletromagnéticas, ondas sísmicas
- O Korteweg-de Vries por ondas em águas rasas
- A equação de Schrödinger na física quântica não relativista
- Navier-Stokes para um fluido viscoso incompressível em biologia e química, reação-difusão equações do tipo A. Turing aos fenômenos da morfogênese.

Em dinâmica das populações equação Von Foerster, por exemplo, a evolução da densidade de uma população de uma certa idade. Etc.

Conclusão

Aqui, a primeira atividade deve dar-lhe uma ideia sobre os modelos de conceitos, sistemas complexos de abordagem de sistemas, o reducionismo e o holismo e modelagem regras básicas e as leis básicas.

Este é um primeiro teórico e os seguintes capítulos lhe dará uma ideia clara de todos esses elementos.

Avaliação

1. Dê a definição de modelo de Minsky O que te inspira a você isso?
2. O reducionismo e holismo?
3. Como ser um modelo para permanecer utilizável?
4. Para um sistema simples que você escolher, mostram que dá uma modelagem baseada em regras e na legislação com base modelagem.

Actividade 2 - Os formalismos simulação

Introdução

Agora que o conceito de modelo está posto, vamos passar para a simulação. Este passo efectivamente mostrar o valor do modelo de computador que vai ser submetido a uma execução em. A modelagem diferentes formalismos e simulação serão apresentados e, em seguida, faremos um foco em três abordagens principais que formarão as seguintes unidades.

Primeiro dar uma definição:

"A simulação consiste em fazer uma abstracção de um sistema ao longo do tempo para ajudar a compreender o funcionamento e comportamento do sistema, e para compreender algumas das suas características dinâmicas, com o objetivo de avaliar diferentes decisões"

Hill, D., 1993. análise orientada a objetos e simulação de modelagem. Addison-Wesley, 342P.

Detalhes da atividade

Primeiro dar uma definição:

"A simulação consiste em fazer uma abstracção de um sistema ao longo do tempo para ajudar a compreender o funcionamento e comportamento do sistema, e para compreender algumas das suas características dinâmicas, com o objetivo de avaliar diferentes decisões"

Hill, D., 1993. análise orientada a objetos e simulação de modelagem. Addison-Wesley, 342P.

A simulação discreta continua, determinista vs, Estocástica. -

Esta definição além construção do modelo liberta aspecto essencial neste caso é alterar o modelo em tempo. Em seguida, envolve a seguinte:

Tempo de Granularity contínuo e discreto de simulação

E seguindo os métodos de cálculo de uma simulação que também pode ser determinista ou probabilística.

simulação contínua geralmente entendida para descrever o sistema sob a forma de equações diferenciais e requer uma solução analítica se o modelo está à direita.

Como para simulação discreta, que simula a passagem de um tempo constante ou não no sistema, ele pode ser usado:

Na modelagem matemática como já vimos, a discretização das equações diferenciais que requerem resolução digital (Runge-Kutta e Leap-Frop para Equações Diferenciais Ordinárias, elementos finitos, volumes finitos e métodos de partículas para Equações derivada parcial.

Em um motor de inferência “computer modeling” baseado em regras (autômatos celulares, modelos de objeto orientado, modelos orientados processos, sistemas multi-agente, ...)

Em contraste com métodos determinísticos no estabelecimento de regras da evolução, podemos encontrar métodos probabilísticos (ou estatísticas), como a simulação de Monte Carlo para equações diferenciais estocásticas, sistemas de Markov em física estatística ou movimento browniano.

Tempo e espaço nas simulações

A natureza do espaço e do tempo também pode participar na classificação de formalismo simulação.

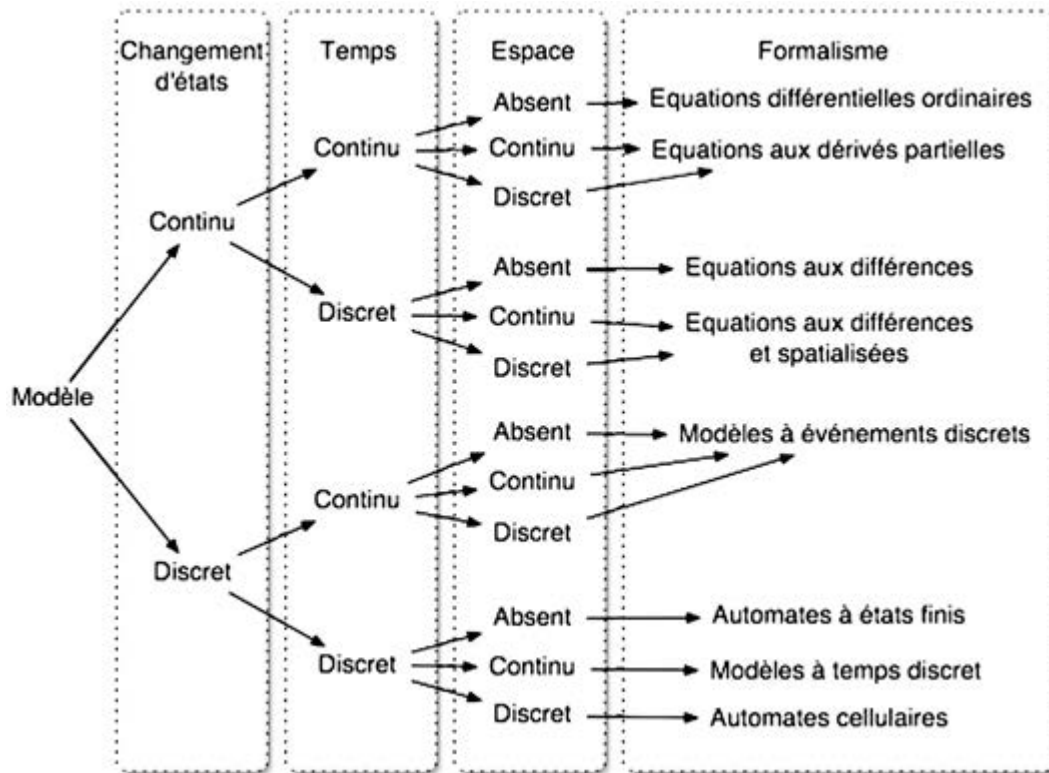


Fig. 6 - Classificação dos formalismos seguinte dinâmica, tempo e espaço.

Apresentação de 3 (três) grandes modelagens e simulação famílias Curso

Vamos agora dar as três principais classes de modelagem e simulação incluídas neste curso. Nós escolhemos estas três classes como eles dão uma visão geral sobre o campo da modelagem e simulação.

Modelagem de sistemas contínuos com base em agentes

Nós apresentamos nesta secção uma introdução à modelagem de sistemas contínuos. Para este tipo de problema, o tempo é assumido como sendo contínua, mesmo se, por um discretização resolução ao mesmo tempo, é executada. Computing contínua não existe! Unidade 2 voltará a estes modelos contínuos e mostrará-los no processamento de computador com software dedicado à simulação, incluindo plataforma Netlogo que permite implementar agentes.

A modelagem de sistemas dinâmicos (Dinâmica de Sistemas)

Aqui também serão discutidos sistemas contínuos, mas vamos ver outra abordagem diferente da abordagem baseada em agentes, onde é bastante questão de usar técnicas de programação baseada em fluxo. Deve ser dito que, ao contrário Unit 2, unidade 3, que será dedicado a estes sistemas dinâmicos (Dinâmica de Sistemas na literatura), em vez lidar com os problemas em que o espaço não é modelado explicitamente.)

Modelagem de eventos discretos

Finalmente, vamos ver a modelagem de eventos discretos ou não está interessado em o tempo que decorre mas os eventos que são mais importantes para o sistema. bem distingo será feita entre os sistemas contínuos e sistemas discretos e a plataforma uso anylogic para implementar um modelo simples.

Conclusão

Esta actividade tem descoberto que a simulação e ver as diferentes formalismos que existem. Responda as seguintes perguntas para testar seus conhecimentos

Avaliação

1. O que é uma simulação?
2. Deixar um ranking de métodos de simulação, dependendo do clima.
3. Lembre-se de modelar grandes famílias que serão apresentados e detalhar alguns princípios.

Resumo da Unidade

Isso conclui esta unidade de introdução à modelagem e simulação. Você também deve ter uma idéia do que esperar com as outras unidades. Tente testar o que você aprendeu.

Avaliação da Unidade

Verifique a sua compreensão!

Instruções

O projeto tem a duração de uma semana, procurar os recursos da Internet

Critérios de Avaliação

O ensaio é pontuado em 20

Avaliação

Fazer um ensaio de pelo menos 4 páginas para apresentar modelos e / ou simulações famosos no formalismo apresentado nesta unidade.

Leituras e outros Recursos

As leituras e outros recursos desta unidade encontram-se na lista de Leituras e Outros Recursos do curso.

Unidade 2: Modelagem de Sistemas Contínuos Com Base em Agentes

Introdução à unidade

Nesta unidade, se trata de estar interessado na modelagem de sistemas de tempo contínuo. As equações diferenciais (equações diferenciais ordinárias, equações diferenciais parciais) são matemáticas ferramenta de modelagem clássicos tais sistemas, mas mostrar a implementação de tais sistemas no computador através do uso de uma simulação de software baseado em agentes, Netlogo.

Objetivos da unidade

No final dessa união, você deve ser capaz de:

1. Identificar um sistema contínuo.
2. Conhecer o processo de modelagem baseada em equações.
3. Dominar o processo de modelagem baseada em equações.
4. replicar a operação de um sistema contínuo usando agentes

Atividades de aprendizagem

Actividade 1 – Descoberta de sistemas, alguns exemplos concretos

Introdução

Esta actividade vai permitir-lhe atingir os sistemas estudados. Irá incluir sistemas por meio de suas entidades e interações pode haver internamente. A resolução digitais depois de modelagem matemática também será abordada. Isto irá permitir-nos saber quando usar e quando usar uma equação infrantique um modelo.

Detalhes de actividade

Classificar Sistemas seguintes entidades e interação

Nós vamos estar na forma de uma matriz de sistemas de classificação dynamques para ver como eles devem ser modelagem e simulação. Esta classificação será de acordo com o número de entidades, a natureza de suas interações eo comportamento geral do sistema

Exemplos de sistemas	Entidades	Interações	comportamento	Observação
Problema dos dois corpos (trajetórias de dois corpos A e B massa m_1 e M_2 submetido a forças gravitacionais)	Poucas entidades	Interações simples	Previsível	problema solvente ou integrado
Problema dos três corpos (trajetórias 3 corpo (ou N) sujeitos a forças gravitacionais.)	Poucas entidades	Interações simples	Imprevisível	Métodos de resolução aproximados A sensibilidade às condições iniciais (A teoria do caos).
Cristal ou gás (a formação de cristais (alinhamento ao nível molecular, por exemplo de diamante), ou características de um gás estável) (Pressão, temperatura)	Muitas entidades	Interações simples	Prévisibles	Estatísticas métodos para obtenção de parâmetros (temperatura, pressão, ...)

sistemas mecatrônicos (construção civil, forças aéreas,)	Muitas entidades	Interações complexas	Prévisibles	engenharia
redes complexas, comunidades espaciais, autômatos celulares, fluidos,	Beaucoup d'entités	Interações simples	Imprevisível	Modelagem e simulação (computador ou matemática)
Organizações, epidemias, economias, sociedades, ecossistemas ...	Muitas entidades	Interações complexas	Imprevisível	

Este quadro dá uma ideia sobre os sistemas que são cilbés por este curso. Na verdade, é as duas últimas linhas para as quais é necessária a modelagem e simulação matemática e computador.

Vamos verificar as entradas e desvantagens de modelagem de equações baseadas. Estes também têm sido pouco abordados na unidade 1. Na verdade, a equação de um sistema já foi levantada nesta unidade, vamos agora ver como é a solução numérica de uma equação que descreve o comportamento de um sistema.

Veja as limitações de resolução numérica

Sistemas, tendo em conta o espaço (EDP)

Um exemplo de sistemas que podem ser tomadas aqui é o fluxo de fluidos ou dynamques espaciais.

Deve-se notar, a maioria destes modelos matemáticos de sistemas dinâmicos, uma solução analítica é difícil - se não impossível - por que nós conduzimos uma solução numérica. Este último exige necessariamente uma fase de quantificação que acompanha as principais características do sistema contínuo (Fig. 1) - infinito - de uma forma discreta - over - usando uma área de malha (discretização eulerian) ou hardware (Lagrangian discretização).

A malha produzido um conjunto de células ou pontos em que os valores numéricos são aproximados características. O número de células (ou pontos) dependerá da potência de

cálculo disponível e terminar a malha é, melhor o sistema de reabastecimento.

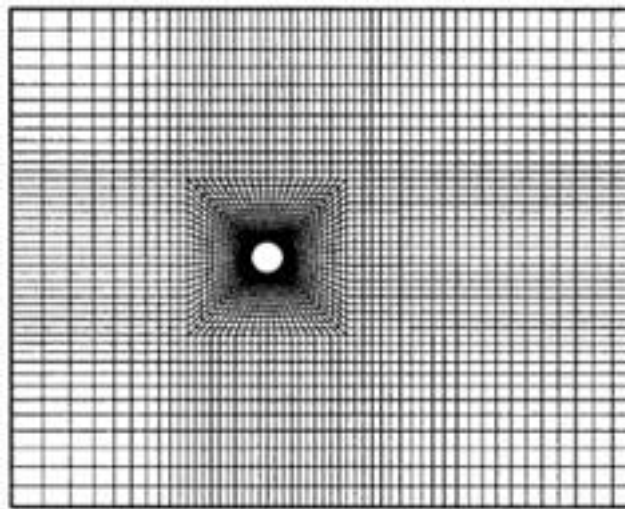
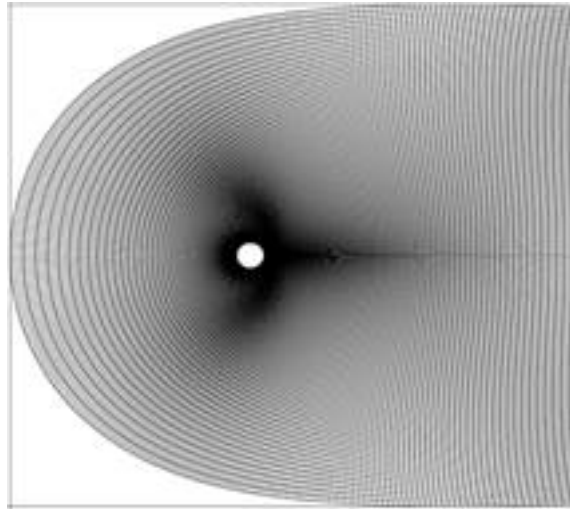


Fig.1 - Exemplos discretização problema euleriana num fluxo de um fluido em torno de um cilindro. À esquerda, uma diferença finita grid - direita, uma grade de elementos finitos.

Na simulação numérica, outra fase essencial da análise matemática deve se preocupar com a consistência da estabilidade e convergência do método utilizado.

Nós não ir para esta fase que cai dentro da análise numérica, mas simplesmente visitar os conceitos de métodos numéricos que podem nos interessar em nosso estudo.

O objetivo é expor mais destes métodos que ensinam um computador público. Também é interessante para diferenciar esse nível na resolução de abordagem as abordagens de Euler e Lagrange. Estas duas abordagens dar diferentes pontos de vista (e complementar) sobre o desenvolvimento de um (ou algébrico) Resolução do modelo discreto.

Abordagens eulerianas

proceder a uma subdivisão do espaço onde o fenómeno simulado ocorre, a produção de um conjunto finito de células fixadas na área que atingem as variáveis locais incluídos no estudo. Com efeito, o fluxo de informação (as variáveis estudadas) que passam através de cada célula, de acordo com os gradientes da equação que rege, dá uma vista a partir de um determinado ponto no espaço, a dinâmica do sistema.

Estes métodos podem, portanto, resistir distorções espaciais e temporais de largura no objecto (ou equipamento) simulado. Por contras, uma deficiência importante é a incapacidade de controlar a dinâmica de um ponto de vista específico que o sistema de malha não está ligado ao hardware. É, portanto, muito difícil de fazer a história da dinâmica de qualquer ponto ao longo de uma simulação.

Note-se, finalmente, que esses métodos perturbam a fase de malha, uma vez o campo tenha irregularidades geométricas ou fronteiras dinâmicas.

Abordagens de Lagrange

em frente às abordagens Euler vai subdividir o material e anexar o seu portão. As células assim obtidas movimento com o equipamento e o resultado no seu movimento de todas as características observadas neste caso as variáveis estudadas.

Esta forma de trabalho permitirá abordagens de Lagrange para superar as limitações de Eulerian aproxima pecando seus benefícios. Assim, podemos perfeitamente seguir qualquer qualquer elemento ao longo do desenvolvimento do fenómeno, aceitar geometrias complexas utilizando uma malha irregular. fronteiras dinâmicas também podem ser geridos por escolher uma malha especial na entrada da área. Propomos nos seguintes métodos de solução numérica esboço utilizadas na simulação de sistemas descritos por PDE. Eles serão classificados de acordo com seu uso ou não de uma malha, ignorando sua natureza determinística ou estocástica.

Métodos com malha

O método de diferenças finitas (MDF), que ilustra eulerian la'approche segue um princípio aproximação de funções. Isto envolve a substituição dos derivados nos pontos de malha por aproximações. Os derivados são então aproximadas por diferenças finitas centradas ou fora do centro no tempo e / ou espaço.

É obtida e vários esquemas numéricos (centrado, descentrado: Euler, Euler implícita) sobre a qual podemos interagir. O método dos elementos finitos (FEM), ao contrário do MDF, seguindo um princípio soluções aproximação que corta a área em subdomínios chamados elementos finitos e escolhe uma aproximação da solução aproximada em itens especiais do nós de elemento.

Esta aproximação é obtido por interpolação geralmente funciona como polinómios cujo grau dependerá do número de nós. Qualquer outro ponto do elemento é então determinada pelos valores da solução nos nós.

O leitor interessado pode consultar a documentação para obter mais detalhes:

O POURHIET, A., 1988. resolução Digital de equações diferenciais parciais: uma primeira abordagem. Edições Cépaduès.

Euvrard, D., 1990. Solução Numérica de equações diferenciais parciais da física, mecânica e ciência da engenharia: diferenças finitas, elementos finitos, método de singularidades. 2ª edição. Masson.

Os métodos sem malha

Elas podem ser distinguidos de acordo com a sua formulação fraca ou forte da EDP.

Para uma formulação fraca incluem o seguinte: Difuso Element, Elemento gratuito Galerkin (EFG), pontuação de Unity. Toutefois sem Malha local Petrov-Galerkin (MLPG), estes métodos não são completamente livre de uma malha que é necessária para a integração da formulação fraca.

Para os métodos que se seguem linguagem forte e proceder por colocação: Smoothed Particle Hidrodinâmica (SPH)

Métodos de partículas

De nossa parte, nós nos limitamos a perfeitos métodos particulares que usam um número finito de partículas discretas para formar o sistema. Eles estão livres das restrições impostas por uma malha e são definidos a partir de elementos discretos (ou partículas) que constituem o sistema de forma independente de qualquer malha. Eles são baseados em mais principalmente em funções de base radial para representar a distribuição e evolução de partículas, portanto, representam interpolação pontos variáveis modeladas locais. Estes métodos podem, então, ser agrupados de acordo com o nível de caracterização do fenómeno físico estudado. Escala que pode ser microscópica (ex. Dinâmica molecular clássica ou ab initio mecânica quântica, mesoscópicos (ex. Networks em controladores de gás, ou macroscópica (Smoothed Particle Hydrodynamics, Partícula na célula).

Os primeiros limites de métodos numéricos

Para falar de um métodos baseados em malha, introduzimos brevemente aquelas que permanecem populares no estudo de sistemas físicos por simulação numérica. Note-se no entanto, apesar de sua popularidade, que os limites persistem e estão provavelmente relacionados com a formulação e / ou a resolução destes modelos matemáticos com base em um campo homogêneo e isotrópico, muitas vezes e de aproximação uniforme de funções isto não ajuda a assimilação de uma geometria complexa do domínio (para MDF) ou aparecimento de estruturas heterogêneas (por MEF) no sistema.

Isto levou ao desenvolvimento de métodos que não utilizam malha perfeitamente e métodos de partículas. No entanto, eles não são sempre irrepreensível, especialmente no que diz respeito à consideração das condições de contorno complexas que afetam as aproximações radiais que são baseados nesses métodos.

Sistemas não levam em conta o espaço (EDO)

Um exemplo de um sistema que pode ser descrito por PDE é a propagação de uma doença numa população. Há uma série de modelos que tratam de fato essas epidemias.

Estes modelos consistem em duas partes: os compartimentos e regras.

Compartimentos são usados para dividir a população em sub populações consideradas entidades homogêneas para diferentes estados possíveis da doença ou infecção, daí o nome "modelo compartimental" utilizados na modelagem matemática de epidemias em geral.

Considerando a população humana, por exemplo, ela pode ser dividida em sub-populações de espécies sensíveis (S), de indivíduos infectados (I) de indivíduos recuperado (R), em quarentena (Q), morreu (D); como mostrado na figura a seguir.

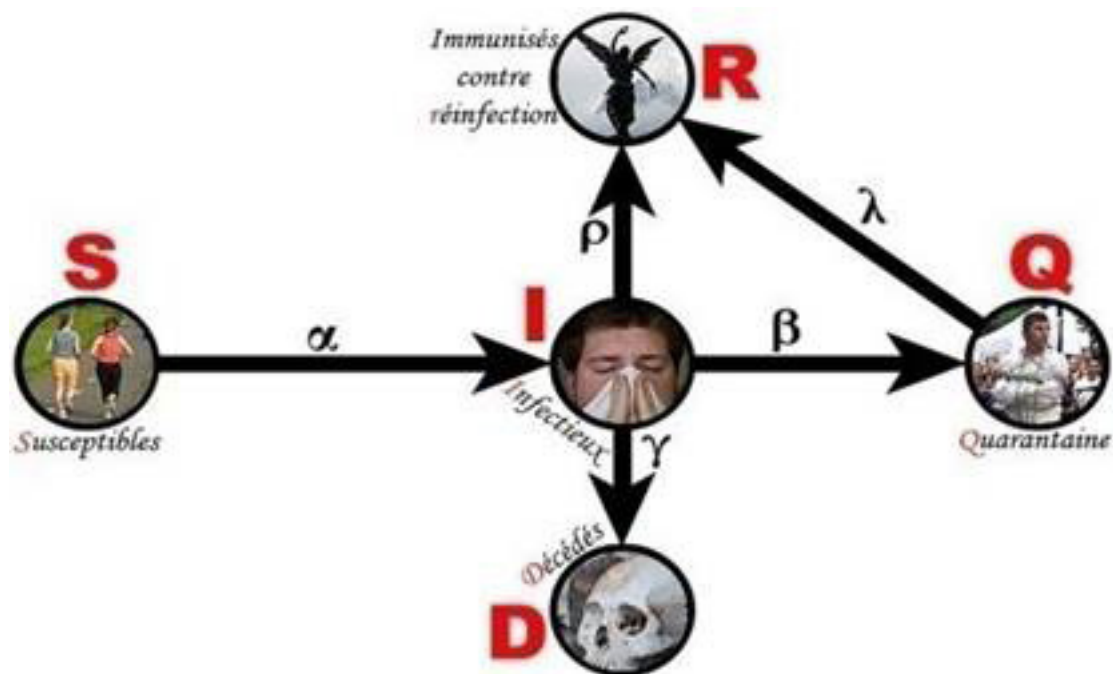


Fig.2 - O modelo compartimental de doenças infecciosas

As regras a dar-lhes as proporções de entidades que migram de um compartimento para outro. Exemplo, a regra $r(S \rightarrow I)$ expressa a proporção de indivíduos susceptíveis são infectadas.

Cada compartimento está representado no modelo por uma equação e regras associadas dar as proporções de entidades que entram e saem fazer o compartimento. Assim, o modelo matemático é um conjunto de equações e simulação numérica mostra a evolução quantitativa de cada compartimento, tal como a escala de tempo ou espaço-temporal considerado.

Embora muito eficaz, no entanto, alguns modelos conceituais limites. Na verdade, eles só permitem acompanhar a evolução de toda a população sem dizer que vai realmente ser afetado. Estes modelos também usam muitos especialmente médio para determinar as taxas de transição de um compartiment para outro.

Como a EDP, a EDO também precisam ser discretizada para ser resolvido. Esquemas iterativos, então, seguir passo a passo (sem tempo, sem tempo) a evolução das densidades compartiments.

Agora vamos ver o papel que pode desempenhar neste vasto campo de modelagem e simulação.

Observou que a informação pode trazer

Neste contexto interdisciplinar, é importante lembrar que a automatização ciência da computação moderna de armazenamento e processamento de informação é, acima de tudo intrinsecamente ligada à necessidade de computação numérica e modelagem / reprodução da realidade , o resumo e os vivos.

É neste perfeitamente como ele está envolvido em pesquisa em modelagem de sistemas dinâmicos.

Com o advento dos primeiros computadores a meados do século XX, testemunhamos o surgimento do resto arquiteturas, formalismos e programação de computadores línguas e, ao mesmo tempo para o nascimento de uma nova geração de modeladores digitais como Richard corrente, um dos precursores de elemento finito, ou John von Neumann ...

Esta última, cujo trabalho, especialmente em análise numérica e nuclear, requerem uma grande quantidade de tempo de computação, é um pioneiro na construção dos primeiros computadores. Ele também deve arquitetura que leva seu nome e que é tomada a partir do relatório sobre o EDVAC (EDVAC).

É também a origem do Monte Carlo para poupar tempo nas simulações e conceito de autômatos celulares sempre uma preocupação de abstração e modelagem do processo natural de reprodução.

O Turing Columbia também podem ser citados, entre outros, neste quadro. Famoso por seu trabalho em computability (máquina de Turing) que já é hora de atacar em inteligência e máquinas capacidades artificiais para "pensar"; o teste de Turing para além da representação da realidade, incluindo o homem e sua consciência, ele liderou uma tentativa de reprodução ... Para o primeiro arauto da contribuição potencial da TI nesta área, a sua em estreita colaboração com a modelagem é exibido por traçar a evolução histórica de linguagens de programação e paradigmas em relação às principais questões de pesquisa em modelagem.

Evolução das linguagens de programação para a modelagem e simulação

A evolução das linguagens de programação, juntamente com os problemas abordados e modelado objetos desde o advento dos computadores é notória. Após o binário (código de computador) e montador (mnemônicos) nos primeiros programas de computador, subir as línguas de terceira geração para amplamente já respondendo à necessidade de representar objetos mais complexos. Incluir o Cobol para a gestão e, especialmente, o Fortran (fórmula matemática Traduzindo System) com John Backus para IBM em 1957, que é bem adequado para computação científica e simulação, e é também a primeira linguagem de computador simbólica. Ele será usado amplamente, através dos seus vários upgrades, simulação de sistemas complexos, tais CFD: Computational Fluid Dynamics, DC: Química Computacional, CWM: Clima e Tempo Modeling). Ele vai ser sentida depois a necessidade de colocar à disposição dos cientistas dos programas de computador prerrogativas de produtividade tempo, manutenção e legibilidade que rapidamente atingem tamanhos impressionantes.

programação estruturada foi, portanto, a sua aparência com linguagens como Algol (Algorithmic Language) Naur a 1963, o Kurtz Básica e Kemeny 1964 ou Pascal Wirth em 1971. Lecionou em parte, a emergência do paradigma objeto muito usado hoje.

É claro então que a primeira língua que introduziu o conceito de uma simulação linguagem orientada a objetos; Simula Dahl e Nygaard, em 1966, que é baseado no Algol. Ele usado para modelar sistemas complexos com atividades paralelas. Isso mais uma vez reflete a interdependência entre o formalismo de computador construção e as necessidades de modelagem.

Na verdade, com os conceitos de objeto e classe, é fácil ser um elemento ativo com seus atributos e métodos e evoluir várias que são instâncias de uma ou mais classes.

Será, então, siga para o fortalecimento do conceito de objeto em idiomas, incluindo SmallTalk inova na comunicação entre objetos através da implementação de envio de mensagens. O legado vai seguir a corrente a 76 anos e confirma a necessidade de captação de TI de certos conceitos de real ou vivo.

O paradigma objeto será enriquecida progressivamente e primitivas como encapsulamento, polimorfismo, interfaces, redefinição surgirão e novas linguagens orientadas a objetos, como Ada, C ++ ou Java.

Também neste momento, agentes aparecerá em repelir o horizonte sobre os problemas que podem ser apreendidos. Com efeito, conceitos como entidades racionalidade modelada, a sua capacidade mais ou menos forte cognitiva, o fato de que eles podem ser localizados em um ambiente e a inclusão deste ambiente não são fortes linguagem objeto.

Ao contrário do objeto, um agente pode, portanto, ter uma certa percepção de seu ambiente, liberdade de iniciativas, uma pedidos opção de filtragem ou mensagens ... vivendo assim em várias áreas de aplicação que sofrem de uma falta de formalismo para simulação; sociologia, biologia, física ...

Conclusão

Esta atividade é dar uma idéia de sistemas contínuos pouvants ser tratado, bem como métodos matemáticos e princípios utilizados na maioria das vezes. Também chama a atenção do aluno sobre a contribuição incomensurável de dados neste campo crescente.

Avaliação

1. Quais são os sistemas que são desenvolvidos nesta unidade em termos das interações de entidades e comportamentos?
2. O que é a diferença entre EDO e EDP?
3. Quais são as diferenças entre os métodos de Euler e Lagrange?
4. Retomar história da evolução das linguagens de programação com base nas necessidades de modelagem.

Actividade 2 – Os agentes de modelização baseado

Apresentação

Vamos apresentar nesta atividade, o campo de sistemas multi-agente. O agente, como vai ser descoberto é uma tecnologia que melhora o objecto (no sentido de computador). Ele vai nos permitir continuar com os sistemas contínuos de modelagem, destacando indivíduos para representar e suas interações. Note-se que será uma abordagem baseada em regras, em vez de uma abordagem baseada em leis que conduzem a equações diferenciais.

Detalhes da actividade

O agente

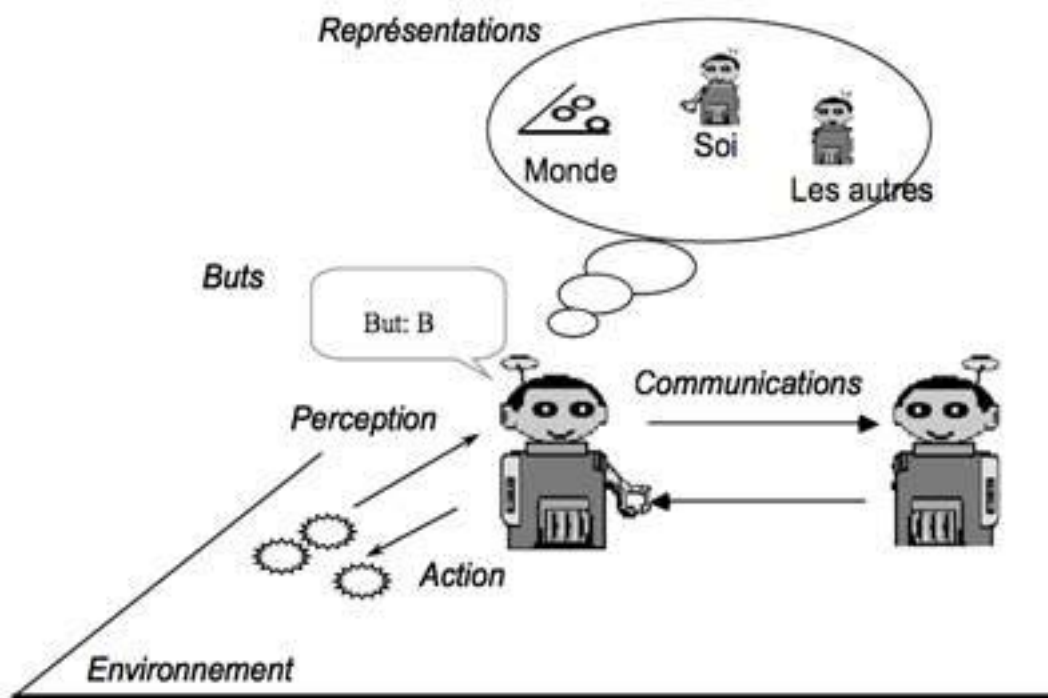


Fig.3 - Um universo multi-agente por Jacques Ferber.

É uma entidade física ou virtual, independente, recursos e capaz de intervir no seu ambiente. Tem uma percepção integral ou parcial do ambiente em que se opera e pode também trabalhar com outros agentes para conseguir uma tarefa ou objectivo conseguir um (ver Fig. 3).

Uma definição mais formal, mas não universal na comunidade SMA é dada em Jacques Ferber, em 1995. Com efeito, constituindo um relativamente novo paradigma, o agente ainda não admite uma única definição inequívoca e consensual.

No entanto, as principais características que podem ser aprendidas e emergentes a partir de suas várias definições pode ser resumido em quatro palavras:

- Autonomia: a capacidade de evoluir de acordo com o seu próprio comportamento, sem intervenção externa
- Reatividade: a capacidade de reagir a eventos externos
- Pró-atividade: a capacidade de tomar decisões mais ou menos elaborados, a fim de alcançar as metas
- sociabilidade: a capacidade de interagir com outros agentes
- Próximas habilidades implementadas pelo agente, além disso, uma distinção pode ser feita entre agentes cognitivos e reagentes.
- Agentes cognitivos incluem uma representação explícita de seu ambiente (a natureza, as propriedades, as leis de evolução ...) e outros agentes (comportamentos, objetivos, estratégias ...) e razão de que o conhecimento.

Agentes reativos, por sua vez, não têm essa representação do ambiente e outros agentes. Eles só respondem a estímulos recebidos e combinará as tarefas.

Não é, entre estas duas extremidades, agentes híbrido que combina a flexibilidade das arquiteturas reativos e cognitivas complexas.

Podemos concluir esta apresentação com uma comparação entre o agente e o objeto:

Semelhanças

- Estado Interno (campos como variáveis ou constantes)
- pode agir sobre sua condição (por meio de métodos)
- Unidades Modulares de comportamento
- Comunique-se através do envio de mensagens

Diferenças

Desde a autonomia de controle (métodos internos):

- Assunto = thread (aquele que chama, decide)
- Agente = unidade de comportamento (aquele que recebe, decide)

um controle é = Agente de controle da fonte (controle local)

conceitos ambientais de = um agente tem uma representação do ambiente.

Sistemas multiagentes (MAS)

Um sistema multi-agente (MAS) é um conjunto de agentes localizados em um ambiente e interagem em uma determinada organização. Esta declaração destaca, como foi o caso para a definição de um agente, um número de termos, ou seja, os agentes, ambiente, as interações, a Organização.

A organização é muito importante em sistemas multi-agente e incluem algumas propriedades (como Jaques Ferber):

- Uma organização consiste em agentes (indivíduos), que exibem comportamento
- A organização como um todo pode ser particionado, e partições diferentes pode possivelmente se sobrepõem
- Comportamento do agente é descrito por função ou papel dentro de sua organização. Esses papéis caracterizar os padrões (padrões) de interação que se conectam a outros papéis. Eles estipulam as obrigações, proibições e permissões que se aplicam a agentes que desempenham esses papéis
- aspecto organizacional de uma SMA é em dois níveis: no nível organizacional (social), que descreve o que persiste quando os agentes entrar ou sair da organização e nível concreto (ou agente), que é uma instanciação específica a estrutura organizacional.

A organização pode, portanto, constituir, por si só, o esqueleto do modelo desenvolvido. Existem também abordagens de modelagem muito comuns com base em organizacional. Aqui estão alguns principal para fornecer uma visão geral:

AGR - um metamodelo baseado em três abstrações; o agente, grupo e função. O agente pertence (a) um (a) papel (s) grupo (s) e peças (são).

Cassiopeia; através da definição de comportamento elementar (habilidades básicas de agentes), comportamentos relacionais (mostrando comportamento coletivo) e comportamento organizacional (que determina a dinâmica da organização).

GAIA com duas fases principais, a análise (papéis e interações) e design (agentes modelo, serviço e conhecidos) que une os terminais níveis de abstrações no processo de modelagem

Plataformas SMA

É difícil fornecer uma lista exaustiva das várias plataformas existentes para agentes baseados em simulação. Na verdade, os problemas-alvo, disciplinas relevantes e modelagem abordagens escolhida apenas contribuir para a explosão do número e da natureza das plataformas de simulação; por que não incluem as plataformas dedicadas ou correr o risco de privilegiar alguns que não necessariamente representam um estado do resultado art. plataformas genéricas que são amplamente utilizados em diferentes disciplinas e contextos são aqui apresentados:

MadKit (Multi-Agent Development Kit): Escrito em Java, é baseado no modelo AGR visto e permite a inclusão de arquiteturas heterogêneas de agentes, protocolos de comunicação desenvolvidos, extensões e ferramentas gestão e análise de simulações agentes baseados. Ele tem uma forte comunidade em torno dele.

Swarm oferece uma coleção completa de bibliotecas orientadas a objetos para desenvolver modelos em Objective C ou Java. O modelador pode então usar instâncias de classe existentes ou desenvolver uma especialização de definir suas entidades manipuladas. Uma característica a notar aqui é que um enxame, além de ser um agente, é também um conjunto de agentes com um programa de eventos de si próprio; Isso, então, permite a definição hierárquica dos agentes.

- Repasto (recursiva Porous Agent Simulation Toolkit) disponíveis para diferentes línguas (Repasto para Java, Microsoft .Net, Python Scripting) que atrai muitos SWARM e integra mais bibliotecas para algoritmos genéticos, regressão ...
- Ascape: também escrito em Java, ele também chama Swarm e agrupa os agentes nos caules (escapos), que são também agentes (estrutura hierárquica). comportamento do agente faz definir que por elas hastes. produto Ascape também coleções com frequência estatísticos sobre as hastes (todas as iterações).
- Mason (Multi-Agent Simulator dos bairros): uma biblioteca Java para multi-agentes de simulação de eventos discretos. Especialmente concebido para transportar um grande número de simulações (mais de 100.000) com, possivelmente, um grande número de agentes (mais de 10.000), a fim de otimizar as configurações ou realizar a aprendizagem, esta plataforma tem um núcleo muito simples e reduzida em tamanho. Está portanto limitado nas propriedades nativas dos agentes.
- Fichas de agente: um instrumento introdutório para não-programadores para desenvolver e compartilhar (exportação como applets) são modelos mais ou menos sofisticados, definindo um comportamento e objetivos muito simples para cada agente e, em seguida, combiná-los. Esta plataforma é usada para criar simulações e jogos interativos educacionais e ambientes virtuais.
- Star Logo: sua linguagem é baseado no logotipo e é traduzido em Java antes de ser executado. Ele define os agentes como tartarugas de passar definir suas entidades manipuladas. Uma característica a notar aqui é que um enxame, além de ser um agente, é também um conjunto de agentes com um programa de eventos de si próprio; Isso, então, permite a definição hierárquica dos agentes.
- Repasto (recursiva Porous Agent Simulation Toolkit) disponíveis para diferentes línguas (Repasto para Java, Microsoft .Net, Python Scripting) que atrai muitos SWARM e integra mais bibliotecas para algoritmos genéticos, regressão ...

Patches na aparência de um relógio que pode controlar tudo. Esta plataforma é adequado para a simulação de fenômenos emergentes que envolvem um grande número de agentes simples (reactivo).

Netlogo: é uma extensão do StarLogo populares implementado com uma infinidade de bibliotecas cooperativas. Tem, além de estar em um ciclo de desenvolvimento normal, várias melhorias ao longo StarLogo; "Contínuo" sistema de coordenadas, simulação participativa distribuída (hubnet) simulações de exportações na forma de applets ...

Redes neurais (RDN)

Eles são definidos a partir de um conjunto de neurônios artificiais inspirados neurônio humano. Cada neurônio tem um conjunto de entradas ponderadas pelos pesos sinápticos, uma função de combinação dos valores de entrada e uma função de ativação que produz a saída (ativação) de acordo com o valor do conjunto (Fig 5).

Por conseguinte, o RDN fica em uma sucessão de camadas de neurônios, cada um dos quais tem as suas entradas para as saídas da camada anterior.

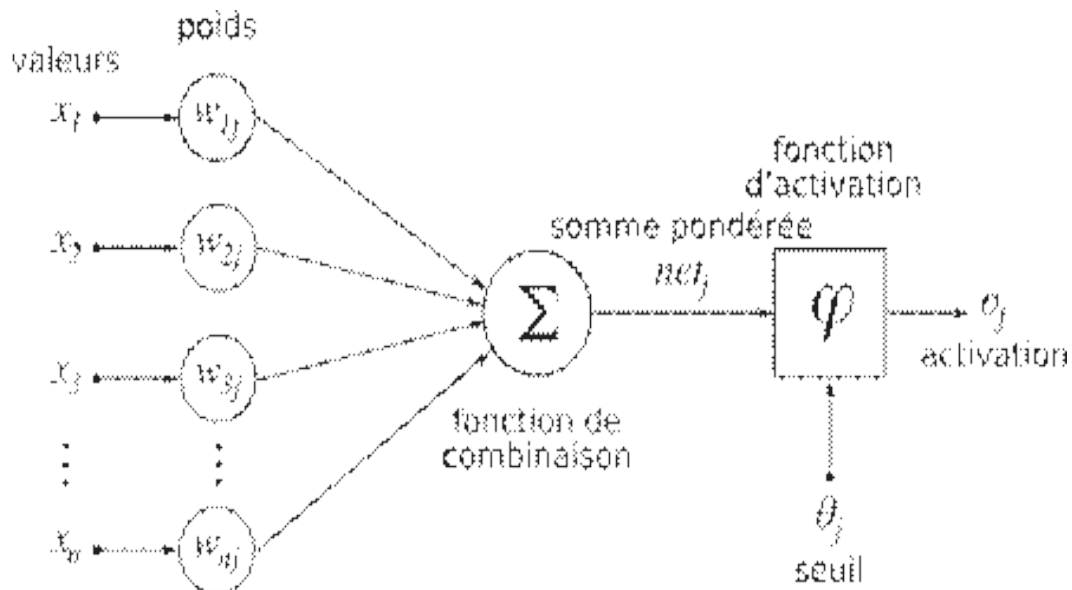


Fig. 5 - Estrutura de um neurônio artificial; entradas, função combinando, ativação e liberação

Um ponto essencial na RDN está atualizando os coeficientes sinápticos; operação que atribui sua capacidade de aprendizagem. É realizada por treino da rede por um número de exemplos de formação (conjunto) que intensificam ou inibem os pesos sinápticos retendo assim as informações.

Desde a sua primeira aplicação com Rosenblatt em 1962 RDN encontraram-se em processamento de sinais, reconhecimento de padrões, classificação, previsão de séries temporais, otimização ... Um bom resumo é dado por:

Hu, Y.H., Hwang, JIN., 2001. Handbook of Signal Processing Rede Neural. CRC Press.408p.

Metaheurísticas: abstrações verdadeiro processo e da vida

Metaheurísticas são uma família de métodos amplamente utilizados para problemas de otimização discreta ou continuamente. Eles desenham diferentes analogias com a biologia (algoritmos genéticos, busca tabu), etologia (algoritmos de colônia de formigas), sistemas sociais (algoritmos de otimização por enxame de partículas), física (algoritmos de recozimento simulado) ... e refletir a contribuição da ciência da computação que possibilite a exploração combinatória de diferentes fontes de inspiração para o tratamento de problemas do real e vivo.

Aqui estão os detalhes de alguns.

Algoritmos genéticos: eles estão surgindo com a Holanda trabalha em 1975, que tinha dois objectivos principais; explicar os processos naturais de evolução e adaptação (crossovers, mutações, selecção ...) e para conceber sistemas artificiais com propriedades semelhantes a estes processos. Para isso, considera que o património genético de uma população contém o (ou a) solução adaptativa para um problema. No entanto, esta solução exige a combinação genética, reprodução e possivelmente durante muitas gerações, porque ele é disperso implicitamente indivíduos da população. algoritmos genéticos são, então, com base neste princípio cadeias de caracteres e usando (representando por exemplo cromossomo) descrevem todas as soluções possíveis com notas (aptidão) e prossegue com as operações de seleção de cruzamento e mutação a convergir para uma solução melhor - uma nova população - por iteração. - Algoritmos genéticos: eles estão surgindo com a Holanda trabalha em 1975, que tinha dois objectivos principais; explicar os processos naturais de evolução e adaptação (crossovers, mutações, selecção ...) e para conceber sistemas artificiais com propriedades semelhantes a estes processos. Para isso, considera que o património genético de uma população contém o (ou a) solução adaptativa para um problema. No entanto, esta solução exige a combinação genética, reprodução e possivelmente durante muitas gerações, porque ele é disperso implicitamente indivíduos da população. algoritmos genéticos são, então, com base neste princípio cadeias de caracteres e usando (representando por exemplo cromossomo) descrevem todas as soluções possíveis com notas (aptidão) e prossegue com as operações de seleção de cruzamento e mutação a convergir para uma solução melhor - uma nova população - por iteração.

les algorithmes d'optimisation par essais particuliers[1] : très apparentés aux algorithmes à colonies de fourmis, ils furent créés au début pour modéliser des interactions sociales simples entre des agents autonomes sans superviseur, mais finalement ils montrèrent une meilleure applicabilité en tant qu'outil. Contrairement aux fourmis artificielles – mues par la sensibilité à la phéromone – les particules de ce système ont une dynamique régit par trois tendances ; suivre sa propre vitesse (tendance aventurière) revenir vers sa meilleure position (tendance conservatrice) ou aller vers la meilleure position fournie par ses informatrices (tendance panurgienne).

Conclusão

Aqui está o que termina esta segunda atividade onde apresentou o formalismo computador (agente) usaremos na próxima atividade para modelar nossos sistemas dinâmicos.

Avaliação

1. Defina o agente.
2. Qual é a contribuição do agente no que diz respeito ao assunto?
3. Dê exemplos de SMA metamodelo.
4. Citar Três plataformas SMA e suas propriedades.
5. O que é uma rede neural?
6. O que é um autômato celular?
7. Fornecer exemplos de aplicação de um autômato celular.

Actividade 3 – Uma base Vs equipe um equações básicas

Introdução

Esta última actividade irá introduzir um exemplo simples do sistema dinâmico. Antes de ligar para modelar este sistema, é preciso primeiro comparar as duas abordagens (agentes baseados em regras baseadas vs). Então, vamos descobrir como se obtém um modelo baseado em equações e como construir um outro agente à base.

Detalhes da actividade

A comparação das duas abordagens

Para comparar as duas abordagens, o melhor é elaborar uma tabela de resumo.

Vantagens e Desvantagens	
Equações do modelo base	Os agentes baseados em modelo
+ Muito formalizada	- Menos formalizada
+ Muito estudada	- Relativamente jovem
+ Algumas variáveis	- beaucoup paramètres
- Análise difícil	+ Fácil implementação
- Questões limitadas	+ Exploração Pressupostos
- Tempo de cálculo razoável	- Tempo de computação significativa
dificuldades com sistemas complexos	+ Gestão de sistemas complexos

Modelagem exemplo (Presa/predadores)

Vamos nos concentrar nesta última parte em um modelo simples que deve dar-nos uma ideia muito rapidamente em modelisaton matemática (com base em equações) e (com base agente) computador.

O modelo Lotka-Volterra de rapina / predador:

O modelo envolve duas pessoas cujos números no tempo t são, respectivamente, classificado A e segunda população B. La (predadores) é o primeiro nutritiva (a presa). Nós fazemos as seguintes premissas (simplificando!):

- Presa Para ter nouriture em quantidades ilimitadas, únicos predadores B opor o seu crescimento e na ausência de predadores da população de presas tem crescido exponencialmente.

O número de predadores é limitada pela quantidade de presa à sua disposição para se alimentar e na ausência de presa, a população predador tem um decaimento exponencial.

- O número de reuniões entre presas e predadores é ao mesmo tempo proporcional a A e B, portanto, proporcional ao produto AB.
- A taxa de desaparecimento das taxas de crescimento presa e predador devido a estes encontros são uma e outra proporcional ao número de partidas entre as duas populações.

Isso nos dá as seguintes equações:

$$\begin{aligned}\frac{dA}{dt} &= rA - \alpha AB \\ \frac{dB}{dt} &= -\delta B + \beta AB\end{aligned}$$

No mesmo sistema simples fornece uma equação relativamente complicado. Nós vamos construir um modelo de agentes com base, em vez especificando as regras que regem as interações.

Presa/predadores ... (agentes)

Presa (Leuk)

- Mova aleatoriamente no espaço.
- Verifique se é possível reproduzir.
- Verifique se ele atingiu a idade máxima.

Predador (bouki):

- Mova aleatoriamente no espaço.
- Verifique se é possível reproduzir.
- Verifique se ele atingiu a idade máxima.
- Verifique se há uma presa para devorá nas proximidades
- Verifique se há energia suficiente

Este modelo já parece mais realista do que o primeiro e vamos implementá-lo para ver se ele pode dar nos escores mais próximo da realidade (equilíbrio ecológico entre as duas populações). Então, vamos usar a plataforma Netlogo já apresentado anteriormente.

Apresentação Netlog

Três águias

- Interface: Esta é a parte onde ele interage com o modelo acabado. Pode-se ver a evolução de agentes e visualizar os dados da simulação. Ele contém botões, barras deslizantes, interruptores, caixas de monitor e o “mundo”, onde agentes de mudança (grande janela preta).
- Informações: Contém informações textuais no modelo para dizer como usar o programa, ele fornece uma documentação do modelo.
- Procedimentos: Ele contém o código para especificar o comportamento dos agentes. É contitué, como outro programa através de uma parte declarativa, uma série de procedimentos, incluindo um especial que veremos mais adiante.

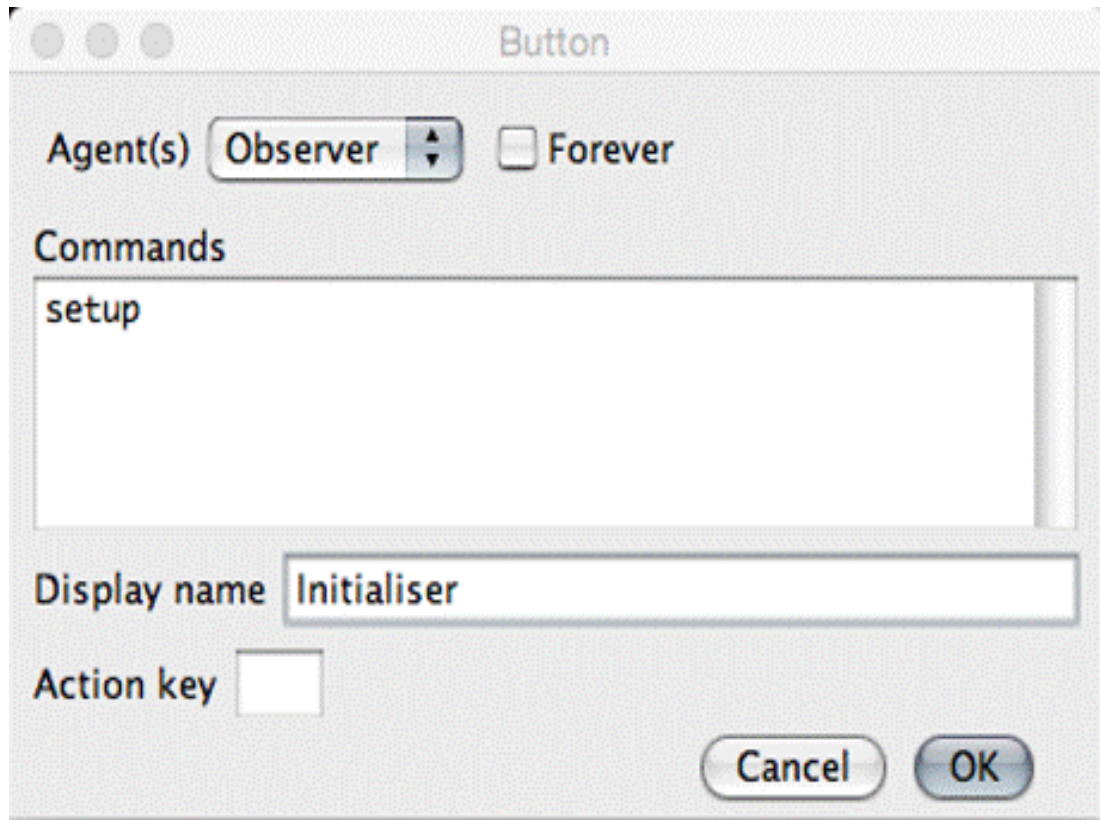
Modelo de desenvolvimento Presa / Predadores

O Botão Configurar

Abra um novo modelo no Netlogo

(Arquivo / Novo)

- Criar o botão “Configurar” (botão / e clique na parte branca)



- Na janela "Comandos" tipo "setup". Este é o procedimento que irá inicializar o modelo
- Na janela "Display Name" digite o nome que deseja que apareça "Iniciar".
- Clique em OK.

O botão Go

- Criar o botão "Go" (botão / e clique na parte branca)
- Na janela "Comandos" digitação "go". Este é o procedimento que irá inicializar o modelo
- Na janela "Display Name" digite o nome que deseja que apareça "Iniciar".
- Ativar o "para sempre" para o comando a ser executado para arrêter com STOP.
- Clique em OK.

O vão botões de configuração e são vermelho (porque as funções correspondentes ainda não estão definidos).

Os parâmetros do modelo

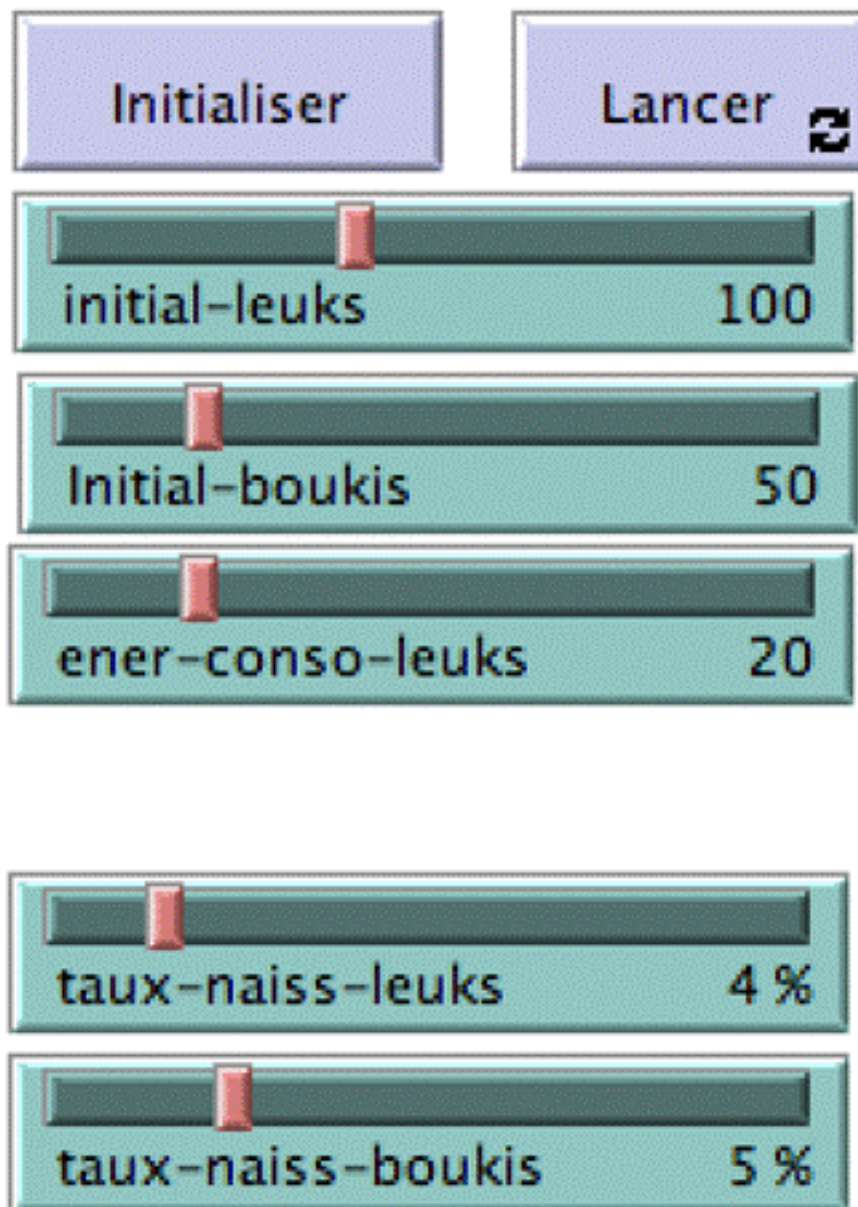
Selecione "slider" e clique no espaço em branco

- Defina o nome da variável (initial-leuks) o seu limite inferior, o limite superior eo padrão

Os parâmetros do modelo (bis)

	Min	Max	valor
inicial-leuk	0	250	100
Inicial-Bouki	0	250	50
Nascimento de taxa-leuks (%)	1	20	4
Nascimento-rate-Bouki (%)	0	20	5
Energia-conso-leuk	0	100	20

Assim que você criou para essas variáveis "slider" e obter



Os agentes e seus atributos

Em Netlogo as tartarugas são utilizados (tartarugas) nomear agentes. Se houver vários tipos de agentes deve criar categorias de agentes, raças

Déclaration	raça [substantivo plural substantivo no singular] [substantivo singular nome no plural]
Declare Netlogo	raça [Leuk Leuk] raça [Bouki Bouki]
para os parâmetros	boukis-próprio [de energia] leuks-próprio [de energia]

Os primeiros procedimentos

Configuração: dizer o que precisa ser feito na inicialização:

1. Gerar muitos leuks e boukis (initial-leuks, initial-boukis)
2. Coloque-os de forma aleatória na área (janela preta)
3. Sua forma "forma"
4. Fixar os valores de atributo

Aqui está o código, o tempo necessário para entender !!!

```
to setup
clear-all

ask patches [set pcolor white]

set-default-shape leuks "circle"

create-leuks initial-leuks [set color red
set size 1.5
set energy random 2
setxy random-xcor random-ycor ]

set-default-shape boukis "circle 2"

create-boukis initial-boukis [ set color black
set size 2
set energy random 2
setxy random-xcor random-ycor ]

end
```

Escrever procedimento de vácuo para o botão Go

```
to go
end
```

Tudo é verde, em princípio,!!!!!!

Agora, prima repetidamente o botão de configuração para ver as botas

Este é apenas o começo, o nosso modelo, não importa, você deve programar os procedimentos na chamada go

```
to go

if not any? turtles [ stop ]

ask leuks [move

try-to-reproduce-leuks

chekh-death ]

ask boukis [move

set energy energy - 1

try-to-catch-leuks

try-to-reproduce-boukis

chekh-death ]

tick

end
```

Para que esse recurso funcione, devem ser definidos:

```
move,

try-to-reproduce-leuks,

try-to-reproduce-boukis,

try-to-catch-leuks,

chekh-death
```

Damos os códigos na página seguinte:

```
to move

rt random 50

lt random 50

fd 1

end
```

Para testar essa função movimento, desativar todas as outras funções ainda não definidos colocar; frente

Ex; try-to-reproduce-leuks

Uma vez que este está escrito, iniciar a simulação com bota e lançamento

Os agentes se mover, mas sem interação / relacionamento

to try-to-reproduce-leuks

```
if random-float 100 < taux-naiss-leuks [  
  set energy (energy / 2)  
  hatch 1 [ rt random-float 360 fd 1 ]  
]  
end
```

Uma vez que este está escrito, iniciar a simulação com bota e lançamento

Os agentes leukus multiplicar ...

```
to try-to-reproduce-boukis  
if random-float 100 < taux-naiss-boukis [  
  set energy (energy / 2)  
  hatch 1 [ rt random-float 360 fd 1 ]  
]  
end
```

Uma vez que este está escrito, iniciar a simulação com bota e lançamento

Todo mundo é multiplicado

```
to try-to-catch-leuks  
let prey one-of leukus-here  
if prey != nobody  
[ ask prey [ die ]  
set energy energy + ener-conso-leuks ]  
end
```

Uma vez que este está escrito, iniciar a simulação com bota e lançamento

Os boukis assumir ...

```
to chekh-death  
  if energy < 0 [ die ]  
end
```

Uma vez que este está escrito, iniciar a simulação com bota e lançamento

Os leuks estão fazendo

Saída gráfica

Na interface, selecione botões na parte "pad"

Dê o nome de seus gráficos e o nome dos eixos

Adicionar um "Pens enredo" já existe um padrão (cada ponto corresponde a pena uma nova curva)

Mudar o nome padrão por leuks e definir a cor para vermelho

Adicionar um boukis trama caneta e deixe a cor preta

Marque a opção "Mostrar legenda"

Adicionar este procedimento para GO

to update-plot

set-current-plot "boukis-leuks"

```
set-current-plot-pen "leuks"  
plot count leuks  
set-current-plot-pen "boukis"  
plot count boukis  
end
```

O que se observa é a população Leuques e boukis?

Então, adicionar uma restrição de energia em Leuk de seu alimento (capim)

Gramma conjunto com manchas de Netlogo.

Em declarações adicionar patches-own [compteur]

Na configuração adicionar ervas verdes ask patches [set pcolor green]

Dans les variables rajouter

	Min	Max	Valor
herbe-temps-repouss	0	100	30
ener-conso-herb	0	50	4

Na configuração

```
ask patches [set compteur random herbe-temps-repouss  
set pcolor one-of [green white]  
]
```

Em to go (rajouter cette directive aux leuks)

```
set energy energy - 1
```

```
try-to-eat-grass
```

(e esta ação)

```
ask patches [if pcolor = white [ifelse compteur <= 0  
[ set pcolor green  
set compteur herbe-temps-repouss ]  
[ set compteur compteur - 1 ]]]
```

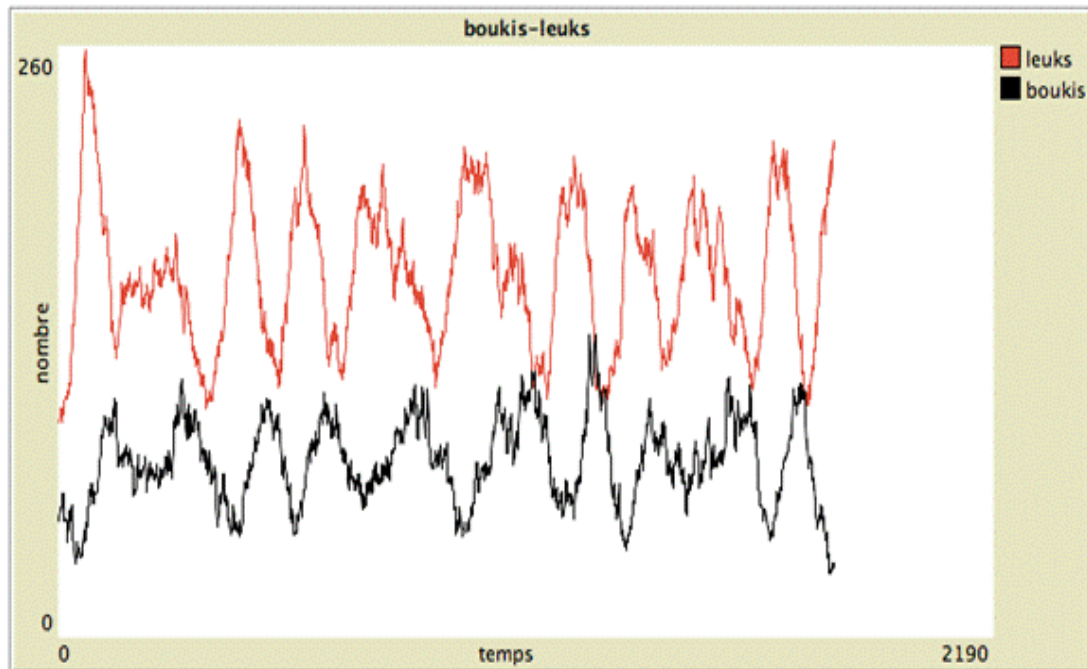
Então, adicionar uma função try-to-eat-grass

A função try-to-eat-grass

```
to try-to-eat-grass
```

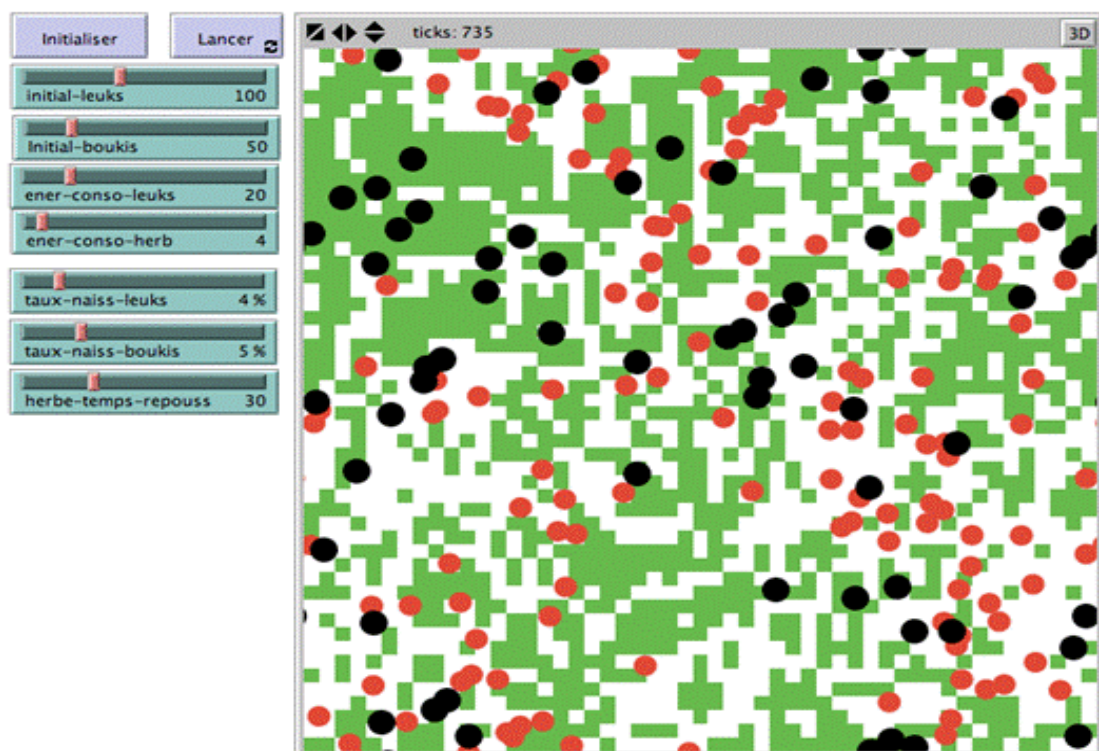
```
if pcolor = green [  
set pcolor white  
set energy energy + ener-conso-herb  
]  
end
```

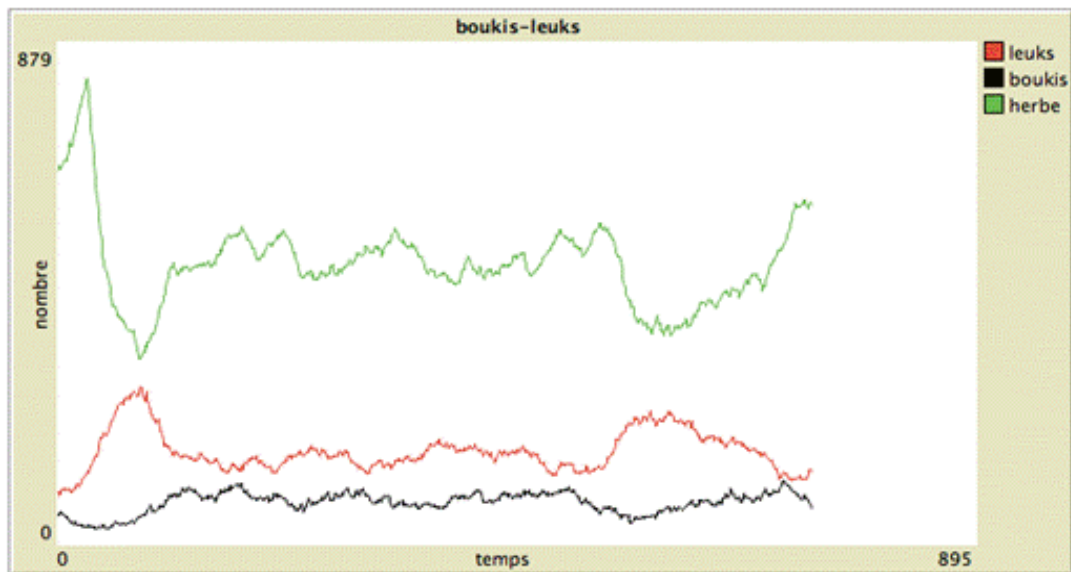
Então, se você seguiu o tutorial, você deve agora ter um modelo estável, com um equilíbrio ecológico.



Commentez le modèle avec vos camarades de classes.

Rajouter l'affichage de la quantité d'herbe pour avoir le résultat de la figure suivante :





Conclusão

Aqui estamos no final desta atividade nos permitiu construir o nosso primeiro modelo baseado em agentes. Sinta-se livre para enriquecer e melhor descobrir.

Para mais informações sobre Netlogo consulte a ajuda que vem com o software!

Avaliação

à Faça um ensaio de 2 páginas para comparar a modelagem abordagens baseadas em equações e o agente à base de.

Resumo da Unidade

Estamos no final desta unidade, que contou com a modelagem de sistemas contínuos. Um tutorial foi desenvolvido para implementar um modelo (presas / predadores), que não conta para o espaço. Para problemas de dinâmica espacial, procure a documentação Netlogo para encontrar os modelos que já foram desenvolvidos.

Avaliação da Unidade

Verifique o seu entendimento

Projecto para fazer

Instruções

O projeto dura uma semana

Sistema de classificação

O projeto está marcado sobre 20

Avaliação

Adicionar no modelo dos caçadores que buscam predadores (boukis), de modo a sempre ter um sistema equilibrado.

Considere estes caçadores no display.

Leituras e outros recursos

Leituras e outros recursos para esta unidade são de nível de leituras e outros recursos do curso.

1. <http://www.particleswarm.info/>

Unité 3. Modelagem de Sistemas Dinâmicas (Dinâmica de Sistemas)

Introdução a unidade

Esta unidade trata de sistemas dinâmicos (Dinâmica de Sistemas), que são um tipo de modelagem onde o objetivo é entender como os elementos estão em relacionamentos uns com os outros. Esta abordagem diferente, baseada em agentes que vamos ver na Unidade 2 ainda será processada pelo mesmo software. De fato, em Netlogo, há um módulo para simular os sistemas Dynamics. Esta unidade é baseada em sistemas Dynamics manuais de usuário Netlogo

Objective da unidade

1. No final dessa união, você deve ser capaz de:
2. Compreender os Sistemas Dinâmicos
3. Dominando modelagem de fluxos
4. Implementar um modelo com o módulo SD Netlogo

Termos-chave

Dinâmica de Sistemas: uma abordagem de modelagem para a compreensão do comportamento não-linear de sistemas complexos ao longo do tempo usando os estoques, fluxos, feedback interno laços, atrasos

Atividades de aprendizagem

Atividade 1 - Apresentação do módulo SD Netlogo

Introdução

Com agentes da unidade anterior, que escreveu o seu comportamento e interações individuais para o comportamento geral.

Com o módulo SD, não será a programação do comportamento dos agentes individuais. Veremos sim como as pessoas se comportam em geral.

Programação do módulo SD é feito pelo desenho de um diagrama de definir um lado, em populações jogo (ou "stocks") e em segundo lugar como essas populações interagem uns com os outros.

O módulo SD, em seguida, lê o seu diagrama e gera o código NetLogo apropriado (variáveis globais, procedimentos e repórteres) para fazer “transformar” o seu sistema dinâmico em NetLogo.

Detalhes da atividade

Os conceitos básicos

Um diagrama do sistema dinâmico consiste em quatro tipos de elementos:

- Stocks,
- Variável,
- O fluxo
- Link

Um estoque é uma coleção de entidades. Por exemplo, um banco pode representar uma população de ovelhas, a água de um lago ou o número de máquinas de uma fábrica.

Um fluxo (flow) carrega essas entidades, encher ou esvaziar um estoque. Fluxos olhar como tubos com uma torneira, a torneira para controlar a quantidade de coisas que fluem no tubo.

Uma variável é um valor utilizado no diagrama. Pode ser quer uma equação que depende de outras variáveis ou uma constante.

Uma ligação (link) permite que um valor presente numa parte do diagrama também é fornecida em outra parte. Um link transmite um número de variáveis ou estoque a um outro estoque ou de um fluxo.

O módulo SD calcula como os valores de inventário mudar ao longo do tempo, avaliando regularmente.

A aproximação nem sempre é perfeito, mas podemos jogar na precisão, alterando o valor do tempo dt . Quando dt diminui, os valores do modelo são avaliados com mais frequência, de modo a que a simulação torna-se mais preciso. No entanto, a diminuição do valor de DT também retarda a velocidade da simulação.

Alguns exemplos de modelos SD

A biblioteca NetLogo modelos “NetLogo Modelos Biblioteca” contém em sua seção “Modelos de Amostra” exemplos que usam o módulo SD.

Estes modelos explorar o crescimento da população (e, nos modelos com a predação, como o declínio da população)

Aqui está uma breve apresentação, ter tempo para explorá-los:

“Crescimento Exponencial” (crescimento exponencial) e “Crescimento logística” (crescimento logístico), que são meros exemplos do crescimento de um estoque.

“Lobo Carneiro Predação (agregada)” (A predação do lobo-Carneiro) é um exemplo de um sistema com várias unidades populacionais que se influenciam mutuamente. Ele modela um ecossistema predador-presa usando o módulo SD.

Conclusão

Aqui dá uma ideia sobre a modelagem SD. É, por conseguinte, não é uma abordagem mesmo como agentes para já. Precisamos saber a próxima pergunta, o método a utilizar.

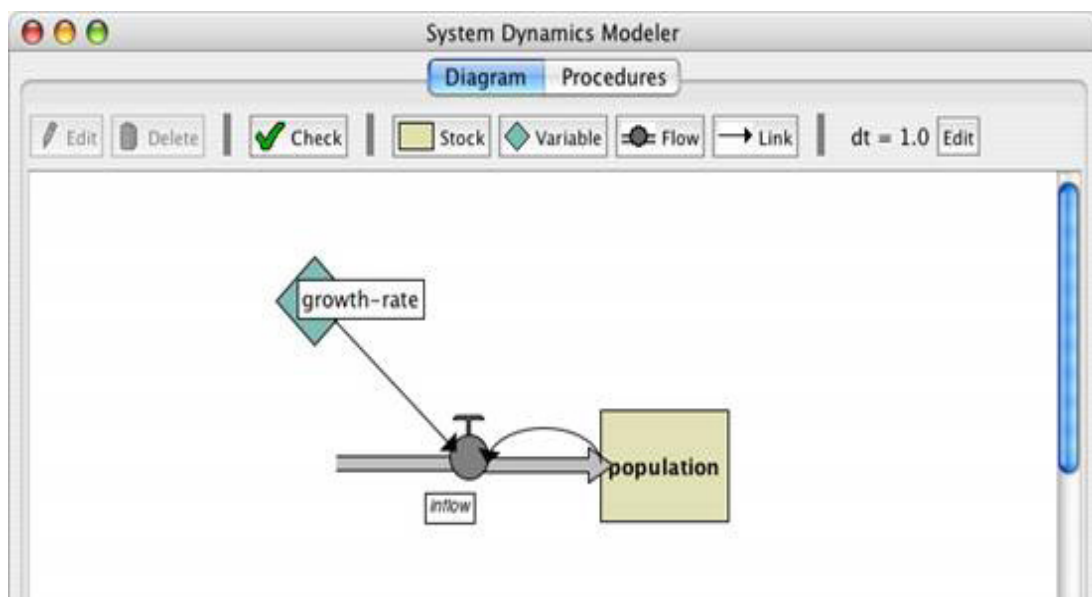
Avaliação

à Procurar na literatura de outro software que faz o SD.

Atividade 2 - A operação do módulo SD

Apresentação

Vamos agora ver como exatamente o módulo Netlogo SD. No menu “Ferramentas”, clique em “System Dynamics Modeler”.



Detalhes da atividade

Diagramas

O painel “Diagrama” (esquema) é onde traçar o diagrama do sistema dinâmico. Sua barra de ferramentas contém botões para editar, apagar ou criar itens no diagrama.

Elementos do diagrama



O diagrama de um sistema dinâmico é composta por quatro tipos de componentes: estoques, variáveis, fluxos (fluxos) e links (ligações).

O estoque

Para criar uma ação, pressione o botão "da" na barra de ferramentas e clique na área de desenho localizado abaixo. Um novo estoque aparece. Cada solicitação estoque um nome único, que se tornará uma variável global. Stocks também exigem um valor inicial ("Valor Inicial"). Este último pode ser um número, uma variável, uma expressão NetLogo complexo, ou uma chamada para um repórter NetLogo.

Variáveis

Para criar uma variável, pressione o botão "Variável" e clique na área de desenho diagrama. Cada variável de um sistema dinâmico deve receber um nome único, que se tornou o nome de um procedimento ou uma variável global. Variáveis também exigem uma expressão. Esta expressão pode ser um número, uma variável, uma expressão NetLogo complexo, ou uma chamada para um repórter NetLogo.

O fluxo

Para criar um feed, pressione o botão "Flow". Clique e segure onde o fluxo deve começar - ou em uma bolsa ou uma área vazia - e arraste para onde você quer que ele seja bem sucedido - em uma ação ou um espaço em branco. Cada fluxo deve ter um nome único, que passará a ser a de um repórter NetLogo. Fluxos requerem também a expressão, a qual deve constituir o "fluxo" dos fluxos a partir da entrada para a saída do "tubo". Esta expressão pode ser um número, uma variável, uma expressão NetLogo complexo, ou uma chamada para um repórter NetLogo. Se o valor for negativo, a corrente flui na outra direção

Quando mais do que um fluxo está ligado a um estoque, é importante ter em conta a forma como tais fluxos irá interagir uns com os outros. O NetLogo não requer fluxo emergente do estoque para fazer em uma ordem específica.

Além disso, NetLogo não parece assim todos os fluxos que saem de um estoque é inferior ou igual ao conteúdo do estoque. Estes comportamentos podem ser implementadas de forma explícita na expressão exigida pelo fluxo.

Por exemplo, se um fluxo é definida por um valor constante, digamos 10, você pode ter certeza que nunca drena mais do que o valor do estoque usando o min primitiva: min (lista de ações 10).

Se eu quiser um vácuo de fluxo de um estoque antes de fluxo B é calculado, eu posso ligar o fluxo de A para B, fluxo e refluxo mudança B para subtrair o valor transmitir A Stock: min (lista (max (lista 0 (estoque - flow-a))) 10).

as ligações

Para criar um link, clique e segure no link para começar - uma variável, uma ação ou fluxo - e puxar para cima ou o destino de fluxo variável.

Os procedimentos de código

O módulo SD gera variáveis e procedimentos NetLogo com base no conteúdo do diagrama construído. Estes são, de facto, estes procedimentos que permitam NetLogo para realizar os cálculos necessários para a simulação descrita pelo diagrama. O painel “Procedimentos” da janela do módulo SD exibe os procedimentos NetLogo gerados.

O conteúdo do painel “Procedimentos” não é editável. Para alterar o sistema dinâmico, você deve editar o gráfico.

Vamos olhar um pouco mais de perto como o código gerado relaciona-se com o diagrama:

Stocks correspondem a variáveis globais que foram inicializados com o valor ou expressão fornecida no “valor inicial”. Cada estoque é atualizado a cada passo, atualizadas com base nas entradas e saídas.

Eles correspondem a processos que contêm os termos previstos no campo “Expressão”.

As variáveis podem ser ou variáveis ou procedimentos globais. Se o “Expressão” é proporcionada uma constante, tornou-se uma variável global e inicializado com este valor. Se você usou uma “expressão” mais complexo para definir a variável, foi transformado em um procedimento, como para o fluxo.

As variáveis e os procedimentos definidos neste painel são acessíveis a partir da janela principal do NetLogo da mesma maneira como os procedimentos e variáveis definidas no painel “Procedimentos” main NetLogo. Você pode chamar esses procedimentos a partir do painel “Procedimentos” principais, a partir do Centro de Comando ou com os botões no painel “Interface”. Você pode se referir a variáveis globais a partir de qualquer lugar, inclusive a partir do painel “Procedimentos” e monitores principais.

Existem três procedimentos importantes para relatar, são: Sistema de dinâmica de configuração, sistema-dinâmica-go e do sistema-dinâmica-do-enredo.

system-dinâmica de configuração inicializa o modelo dinâmico do sistema. Ele dá valor à DT, chama de reposição-carrapatos e inicializa stocks e conversores. Os conversores de valores constantes são inicializados em primeiro lugar, seguido de ações a valor constante. Outros estoques são então inicializado em ordem alfabética.

system-dinâmica-go executa o sistema dinâmico de nenhuma unidade de tempo dt. Ele calcula os valores de fluxos e variáveis e atualiza o valor das ações. Ele também chama tick-antecipadamente com o valor dt. Conversores e fluir com “expressões” não-constantes são calculadas apenas quando este procedimento é chamado, no entanto, a ordem de avaliação é indefinido.

system-dinâmica-do-gráfico mostra os valores dos stocks de sistemas dinâmicos. Para utilizar este procedimento, você deve primeiro criar uma plotter no painel “Interface” NetLogo. Devemos definir um lápis para cada ação que você deseja representar o gráfico. Este procedimento usa o enredo atual, que você pode alterar usando o set-atual-enredo de comando.

Conclusão

Esta actividade está concluído. Vamos passar para a implementação do modelo de presa / predador.

Avaliação

1. O que é um procedimento Netlogo.
2. Quais são os elementos essenciais para Netlogo SD?
3. Dê a cada como configurar ou defini-las.

Actividade 1.3 – Implementação de modelo de ave rapina/predador

Introdução

Esta última actividade nos permitirá implementar nosso modelo, mas com um formalismo diferente da unidade anterior.

Detalhes da atividade

Reprodução de rapina (ovelhas)

- Abra um novo modelo no NetLogo.
- Inicie a "Dinâmica de Sistemas

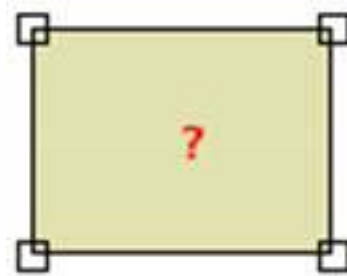


- Pressione o botão "Stock" na barra de ferramentas



- Clique na área de diagrama.

Você vê um estoque com um ponto de interrogação vermelho no centro.



- Clique duas vezes nele para editar estoque.
- Nome da ovelha estoque (ovelhas).
- Defina o valor "valor inicial" inicial para 100.
- Limpar a caixa de seleção "Permitir valores negativos" (permitir valores negativos). Um número negativo de ovelhas não faz sentido!

A população de ovinos pode aumentar se uma ovelha nasce. Para adicionar este fato ao nosso diagrama, criar um fluxo de entrar no estoque ovelhas.

Clique no botão "Seguir" na barra de ferramentas e pressione o botão do mouse em uma área vazia à esquerda do estoque de ovelhas. Puxe o fluxo para a direita até que ele se conecte com o estoque ovelhas. Solte o botão do mouse.

Editar o nome que fluxo e sheep-nascidos.

O número de ovelhas nascido durante um período de tempo dt depende do número de carneiros vivos: mais de carneiros vivos, mais recém-nascidos.

Desenhe um link a partir do estoque e ovelhas indo para o fluxo de ovelhas nascimentos.

Por agora, digite uma constante, por exemplo 1, no campo "Expressão".

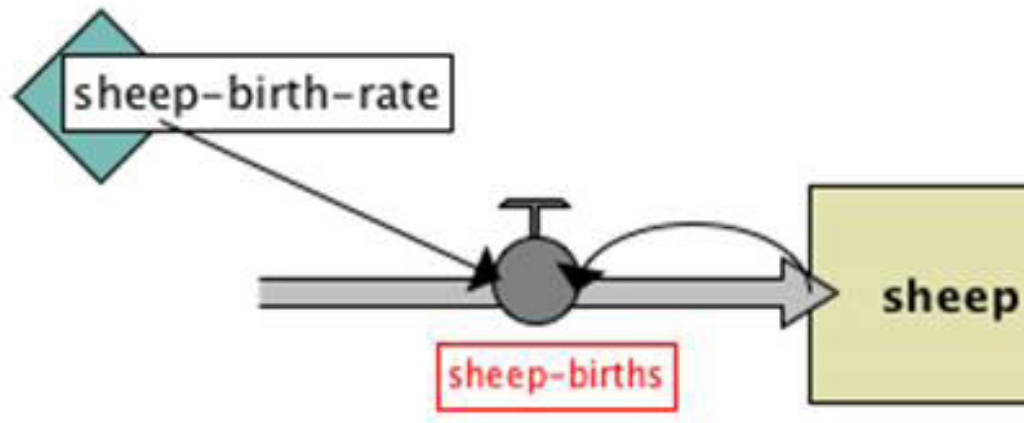
Taxa de natalidade ovelhas também depende de alguns fatores constantes que a discussão está além do escopo deste modelo, como a taxa de reprodução, por exemplo.

criar uma variável e atribua-lhe ovelhas-birth-rate.

Dê-lhe um valor de 0,04

Desenhar um-birth-rate ovelhas ligação variável no fluxo de ovelhas nascimentos.

Seu diagrama agora deve ter esta aparência::

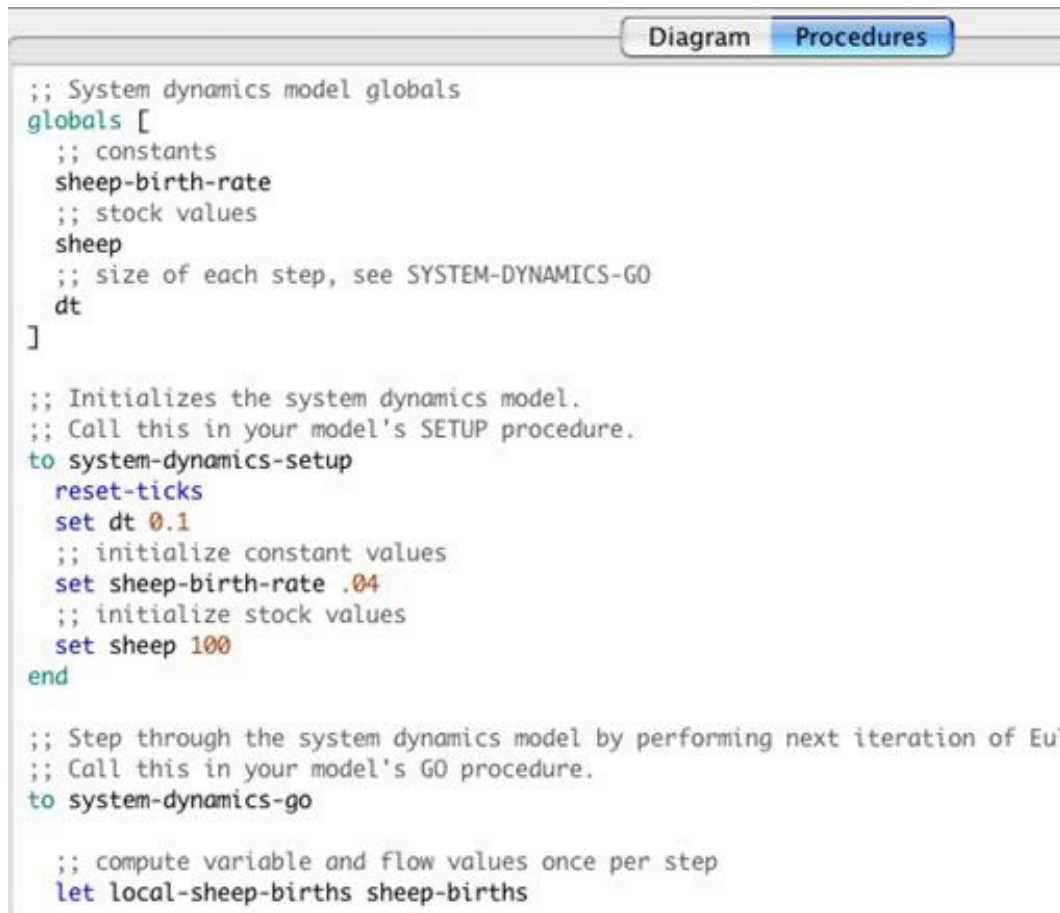


O rótulo do fluxo de ovelhas nascimentos é vermelho porque ainda não deram o seu discurso. A cor vermelha indica sempre que algo está faltando nesta parte do diagrama.

O número de ovelhas entrar em nosso estoque dependem positivamente o número de ovelhas e as taxas de natalidade ao nascer taxa de ovelhas.

Edite o sheep-nascidos de fluxo e escrever a seguinte expressão:-birth-rate ovelhas * ovelhas.

Temos agora um diagrama completo. Para ver o código NetLogo gerado por este esquema, clique em "Procedimentos" ongleL da janela do módulo SD. Este código deve ser:



```

;; System dynamics model globals
globals [
  ;; constants
  sheep-birth-rate
  ;; stock values
  sheep
  ;; size of each step, see SYSTEM-DYNAMICS-GO
  dt
]

;; Initializes the system dynamics model.
;; Call this in your model's SETUP procedure.
to system-dynamics-setup
  reset-ticks
  set dt 0.1
  ;; initialize constant values
  set sheep-birth-rate .04
  ;; initialize stock values
  set sheep 100
end

;; Step through the system dynamics model by performing next iteration of Eu
;; Call this in your model's GO procedure.
to system-dynamics-go
  ;; compute variable and flow values once per step
  let local-sheep-births sheep-births

```

Uma vez que o modelo criado com o módulo SD, podemos interagir com o modelo através da janela "Interface" NetLogo. agora a construir o modelo NetLogo que nos permite executar o código gerado pelo diagrama.

Precisamos da configuração e vá botões para chamar os procedimentos do sistema-dinâmica-configuração e do sistema-dinâmica-go criado pelo módulo SD.

Além disso, a fim de acompanhar a evolução das populações também precisamos de um monitor e um plotter.

Escolha a janela principal NetLogo.

No "Procedimentos", escrevem:

```

to setup

ca

system-dynamics-setup

end

```

```
to go
system-dynamics-go
system-dynamics-do-plot
end
```

Vá para o painel de "Interface".

Criar um botão de configuração.

Criar um botão Go (não se esqueça de fazer um botão "para-sempre" na caixa "para sempre").

Criar um monitor de ovelhas.

Criar uma etiqueta chamada "populações" com uma caneta com o nome ovelhas

Agora estamos prontos para executar o nosso modelo.

Pressione o botão "setup".

Não pressione o botão "ir" agora.

Em vez disso, escrever quatro ou cinco vezes ir na Linha Centro de Comando Command pressionando Retorno após cada ir.

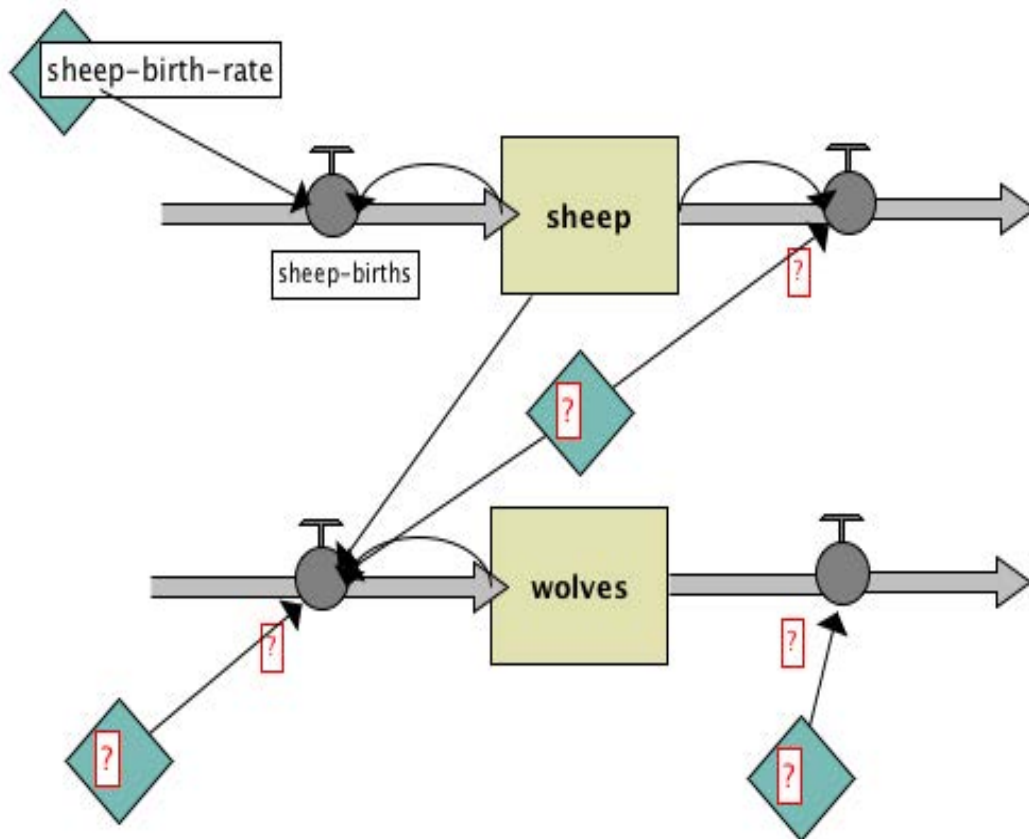
Observe o que acontece. A população de ovinos está crescendo exponencialmente. Depois de quatro ou cinco iterações, temos uma enorme quantidade de ovelhas. Terminamos esta situação, porque nós temos a criação de ovinos, mas nunca morrem. Para resolver este problema, terminar o nosso diagrama através da introdução de uma população de lobos que comem ovelhas.

Voltar para a janela "System Dynamics"

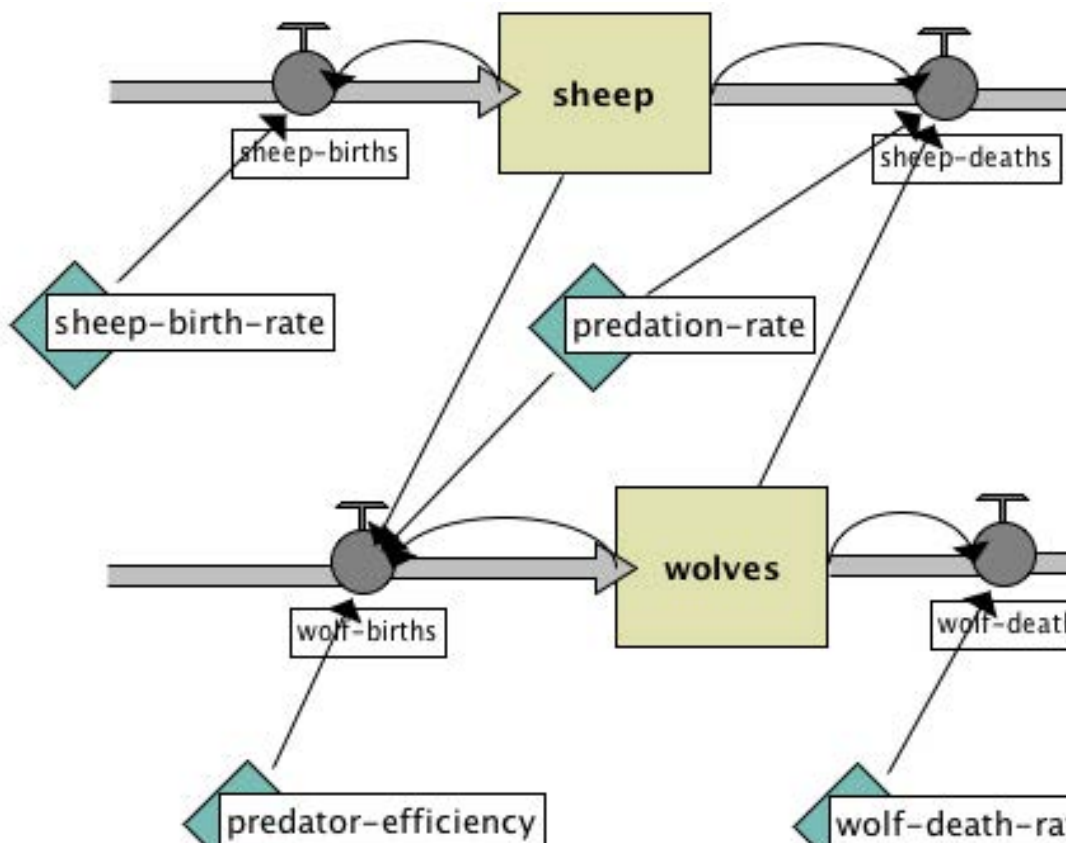
Adicionar um estoque de lobos chamados "wolfes".

Adicionar feeds, variáveis e propriedades que o diagrama se parece com:

Adicionar feeds, variáveis e propriedades que o diagrama se parece com:



Nome dos elementos do diagrama da seguinte forma:



onde o valor inicial de lobos (lobos) é 30,

wolf-deaths $\text{vaut } \text{wolves} * \text{wolf-death-rate}$,

wolf-death-rate $\text{vaut } 0.15$,

predator-efficiency $\text{vaut } 0.8$,

wolf-births $\text{vaut } \text{wolves} * \text{predator-efficiency} * \text{predation-rate} * \text{sheep}$,

predation-rate $\text{vaut } 3.0\text{E-}4$,

et sheep-deaths $\text{vaut } \text{sheep} * \text{predation-rate} * \text{wolves}$.

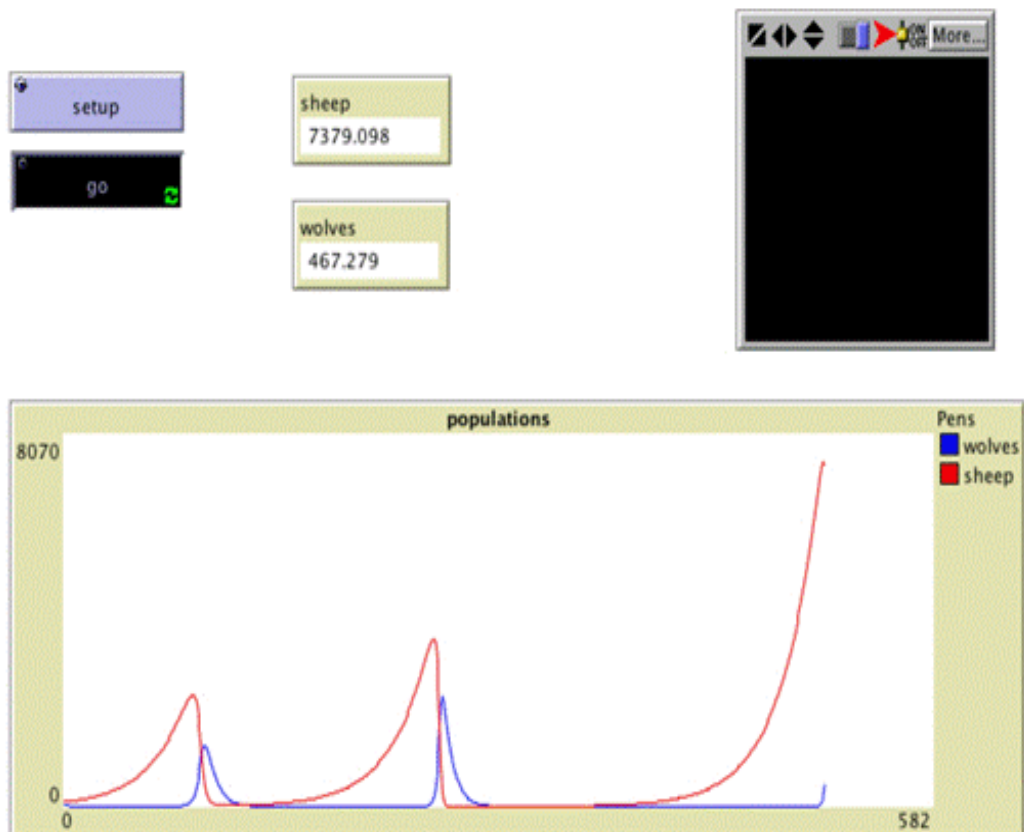
Agora temos realmente feito com o gráfico, mas ainda colocar o modelo NetLogo à data.

Voltar para a janela principal do NetLogo.

Adicionar uma caneta com o nome "lobos" plotter "populações".

Carregue a configuração e vá para ver o diagrama do sistema dinâmico em ação.en action.

Você deverá ver um gráfico da população que se parece com isso:



Conclusão

Aqui é o seu primeiro modelo SD. Nós vamos ter que melhorá-lo e adicionar mais funcionalidades, como fizemos na unidade anterior.

Avaliação

- Dizer por que as curvas destes modelos são mais suaves do que os modelos baseados em agentes.
- Quando é que estes dois formalismos são equivalentes para você?

Resumo da Unidade

Você acabou de ver a segunda forma de simular um sistema, Dinâmica de Sistemas. Então você pode comparar as duas abordagens e escolher o melhor de acordo com as perguntas feitas no sistema.

Avaliação da Unidade

Verifique o seu entendimento

Projecto para fazer

Instruções

O projeto dura uma semana

Sistema de classificação

O projeto está marcado sobre 20

Avaliação

Adicionar no modelo de estoque grama e fazer depender a sobrevivência das ovelhas sobre o estoque grama.

Leituras e outros recursos

Leituras e outros recursos para esta unidade é do nível de leituras e outros recursos do curso.

Unité 4: Modelagem de Agentes Discreto

Introdução a unidade

Esta unidade abrange o primeiro formalismo de modelagem; modelagem de eventos discretos. Ele lhe dará os conceitos-chave em torno de que a modelagem ao especificar as estruturas necessárias para realizar tal modelação (agentes, fila, relógio ...). Um exemplo será tratado para ajudar a identificar esse formalismo.

Objectivos da unidade

No final desta unidade, serás capaz de :

1. Compreender e aplicar os princípios básicos da simulação de eventos discretos;
2. O uso de métodos de simulação para modelar e analisar sistemas de eventos discretos;
3. Aplicar as ferramentas e técnicas de simulação em um projeto prático que envolve vários cenários.

Termos-chave

Événement: Evento: importante ou famoso Feito no decurso de um processo.

Fila: uma fila, um agrupamento sequencial de indivíduos ou objetos esperando algo tão organizado.

Actividade de aprendizagem

Actividade 1 – Definição de modelagem de agentes discreto

Introdução

Cette activité va nous permettre de bien comprendre la modélisation à événements discrets. Un bref historique sera donné et certaines notions seront bien présentées pour cerner ce domaine

Detalhes da atividade

Localize a modelagem de eventos discretos

Modelagem de eventos discretos remonta à mesma época que a modelagem de sistemas dinâmicos (sistemas dinâmicos). Em 1961, a IBM introduziu engenheiro GPSS Geoffrey Gordon, considerada a primeira implementação da modelagem de eventos discretos. Hoje, uma série de programas - incluindo as versões modernas de GPSS - oferecer modelagem de eventos discretos.

modelagem evento discreto requer considerando o sistema como um processo de: uma sequência de operações que realizam agentes.

Operações pode ser possível, um serviço que pode vir de diferentes recursos, processo de seleção, especificações, etc ... Enquanto agentes para realizar o processo estão competindo por recursos limitados, podem ser colocadas em espera. As filas se tornam, assim, parte de quase todos os modelos, com eventos discretos.

O modelo é especificado como graficamente um diagrama de fluxo de processo onde os blocos representam operações. A organização geralmente começa com os blocos "fonte" agentes geradores e injecta-se o processo e termina por blocos "sumidouros" que removem.

Agentes - primeiro chamado por transações em GPSS ou entidades em outro software de simulação - podem representar clientes, pacientes, chamadas telefônicas, materiais físicos e eletrônicos, peças, produtos, paletes, operações TI, veículos, tarefas, projetos, idéias, e assim por diante.

Recursos representam os funcionários, médicos, comerciantes, trabalhadores, servidores, processadores, memória do computador, equipamento e transporte

tempo de serviço e agentes de hora de chegada são geralmente estocástica. Uma vez que eles são extraídos de uma distribuição de probabilidade, modelos de eventos discretos são eles próprios estocástica. Em termos simples, isso significa que um modelo deve funcionar por um período de tempo específico ou executar um número de iterações antes de produzir saída significativa.

Algumas noções em torno de modelagem de eventos discretos

Virando-se para a definição de alguns conceitos importantes sobre a modelagem de eventos discretos.

Eventos discretos:

A grande maioria dos processos que observamos no mundo consistem em mudanças contínuas (ver próxima unidade em modelos contínuos).

No entanto, quando tentamos analisar esses processos, muitas vezes faz sentido dividir um processo contínuo em partes discretas para simplificar a análise. Técnicas de modelagem de eventos discretos approximent processos do mundo real que são contínuas com os eventos não-contínuos que você definir.

Aqui estão alguns exemplos de eventos:

A grande maioria dos processos que observamos no mundo consistem em mudanças contínuas (ver próxima unidade em modelos contínuos).

No entanto, quando tentamos analisar esses processos, muitas vezes faz sentido dividir um processo contínuo em partes discretas para simplificar a análise. Técnicas de modelagem de eventos discretos aproximam processos do mundo real que são contínuos com os eventos não-contínuos que você definir.

Aqui estão alguns exemplos de eventos:

- Um cliente entra em uma loja,
- Um caminhão terminou descarga,
- Um novo produto é lançado,
- Os níveis de estoque ter atingido um certo limiar, etc.

Em eventos discretos modelar o movimento de um trem do ponto A ao ponto B será modelado com dois eventos, nomeadamente um evento de início e um evento de chegada. O movimento real do trem é modelado como um período (intervalo) entre os eventos de partida e de chegada. Isso não significa que você não pode modelar o trem em movimento. Na verdade, com anylogic (discutido mais tarde), você pode produzir animações visuais contínuas para eventos logicamente discretos.

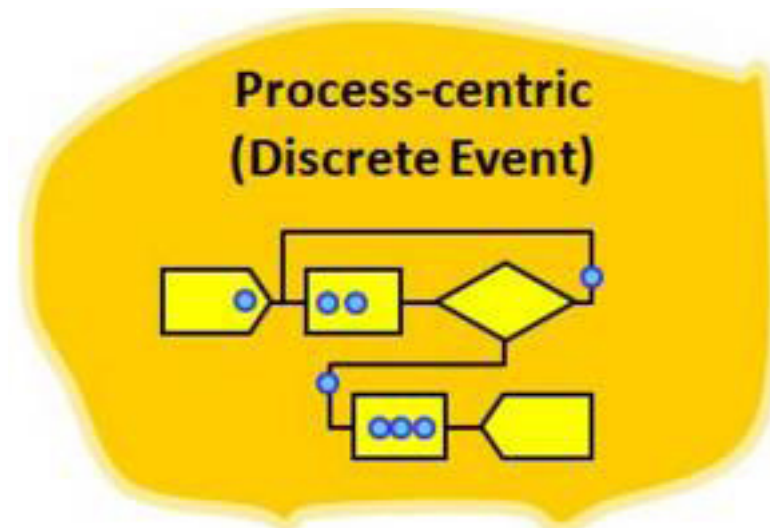


Fig. 1 - uma representação de um processo de passo a passo discreto.

O termo é usado de eventos discretos no sentido restrito para denotar modelagem "centrada em torno do processo", que sugere para representar o sistema como uma sucessão de operações em execução em entidades. Entidades como clientes, documentos, partes do sistema, os pacotes de dados, veículos, ou telefonemas.

As entidades são passivas, mas podem ter atributos que afetam a maneira como eles são manipulados ou podem mudar seus atributos à medida que evoluem.

Modeling “centrada em torno do processo de” segue uma abordagem de modelagem de nível de abstração meio. Embora cada objeto é modelado individualmente como uma entidade, geralmente o modelador ignora muitos detalhes de “fisicamente” como a geometria exata, aceleração e desaceleração. Modeling “centrado em torno do processo” é amplamente utilizado nos campos de fabrico, logística, e saúde ...

Agente :

Ao modelar e simulação, o modelador:

Identificar entidades activas, agentes (que podem ser indivíduos, empresas, projetos, ativos, veículos, cidades, animais, navios, produtos, etc.)

Define o seu comportamento (principais impulsionadores, reações, memória, relatórios, ...)

Coloca-los em um determinado ambiente,

Estabelece conexões,

E é executado a simulação.

O comportamento global surge como o resultado de interações de muitos comportamentos individuais (Uma definição mais formal de um agente será dado na próxima unidade).

A fila de espera :

Uma fila é “um tipo de dados abstrato, em que os elementos são adicionados e removidos da parte traseira para a frente;. O” primeiro a chegar, primeiro a sair “(FIFO)”

A principal diferença entre uma fila e uma pilha é que os elementos de uma fila são no final e no início recuperado. Para um exemplo de fila todos os dias, considere uma linha do cliente em um banco esperando para ser servido.

Cada novo cliente recebe a cauda na parte de trás. Quando o caixa está pronta para ajudar um novo cliente, o cliente em frente da linha é servido. Esta é uma fila - o primeiro cliente on-line primeiro a ser servido.

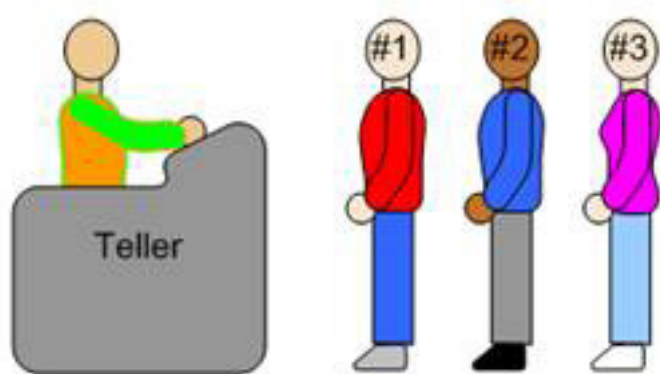


Fig. 2 - Uma fila de espera ao banco.

O relógio :

Na simulação temos de manter a noção do tempo, independentemente das unidades de medida que são adequados para serem sistema modelado. Nas simulações a eventos discretos, ao contrário de simulações em tempo real, o tempo de "saltos" porque os eventos são instantâneos - o relógio se move para o próximo evento como e enquanto a simulação roda.

A lista de eventos :

A simulação também manter pelo menos uma lista de eventos da simulação. Isso às vezes é chamado o conjunto de eventos esperando por ele lista os eventos que estão pendentes, como resultado do evento previamente simulado, mas ainda a ser eles mesmos simulados.

Um evento é descrito no momento em que ocorre e um tipo, que indica o código a ser usado para simular o evento. Muitas vezes o código de evento a ser definido, neste caso, a descrição do evento também contém as configurações do código do evento.

Condição de término

Porque os eventos estão acorrentados, teoricamente, uma simulação de eventos discretos poderia durar infinitamente. Assim, o designer da simulação é decidir quando a simulação vai acabar.

Conclusão

Esta primeira atividade dá uma ideia sobre a modelagem de eventos discretos. Agora você deve saber todas as palavras que se relacionam com esta área. Verifique se o seu conhecimento agora.

Avaliação

1. Que tipo de sistemas devemos usar a modelagem de eventos discretos
2. O que é uma fila
3. Explicar o conceito de um pequeno relógio em uma simulação de eventos discretos

Actividade 2 – Representação de todos os elementos de um modelo de eventos discretos

Apresentação

Esta actividade vai permitir-nos para saber quando e como usar uma simulação de eventos discretos. É em primeiro lugar na comparação de uma simulação de eventos discretos com a continuação antes de tratar um caso único (serviço de um caixa automático) com software de computador (anylogic).

Detalhes de atividades

Compare Eventos Discretos Simulação Vs DC

Por exemplo, considere o seguinte problema:

Há três caminhões trazendo produtos da planta.

Em média, eles levam 3 dias para chegar.

Cada caminhão leva entre 10 e 20 produtos (todos igualmente prováveis)

Adicionalmente, nós temos cinco distribuidores que recuperam produto da planta de acordo com as encomendas.

Normalmente eles querem 5 a 25 produtos (todos igualmente prováveis).

Cada distribuidor tem uma necessidade média de 2 dias para voltar ao mercado, e uma média de 5 dias de entregar as mercadorias.

E nós Queremos saber: QUANTOS Produtos Vendidos Serão Desta Maneira?

Neste Caso, Não vamos usar Simulação Contínua. Porque:

Nós não queremos controlar o número de dias em tempo real,

Nós nem sequer se preocupam com nós todos os dias,

As coisas não podem ser sincronizados,

Nós, então, usar a simulação em tempo discreto

No laço do tempo:

Em uma simulação de eventos discretos: Não há laço do tempo:

Há eventos que são planejadas;

A cada passo, o próximo evento programado com o melhor tempo será processado.

Os agentes não levam a ações contínuas:

Em vez disso, eles esperam que os eventos ocorrem,

Eles definir novos eventos para coincidir com a próxima coisa que eles vão fazer.

Os agentes podem ser bloqueados

Os agentes não podem fazer o que quer fazer:

Se eles querem um produto (por exemplo), e não há outro, eles travam (Eles não podem agendar novos eventos até que eles são desbloqueados)

Muitos agentes podem ser bloqueados espera para o mesmo recurso (mais do que um distribuidor pode ser aguardando a chegada dos caminhões).

Descubra as filas e variáveis aleatórias discretas

É necessário compreender o processo de fila de espera e as variáveis aleatórias discretas. Realizar uma pesquisa na internet sobre estes.

Exemplo Modelagem (Automated Banking Machines)

Este tutorial é a partir do website da anylogic. Ele vai continuar com a sua implementação:

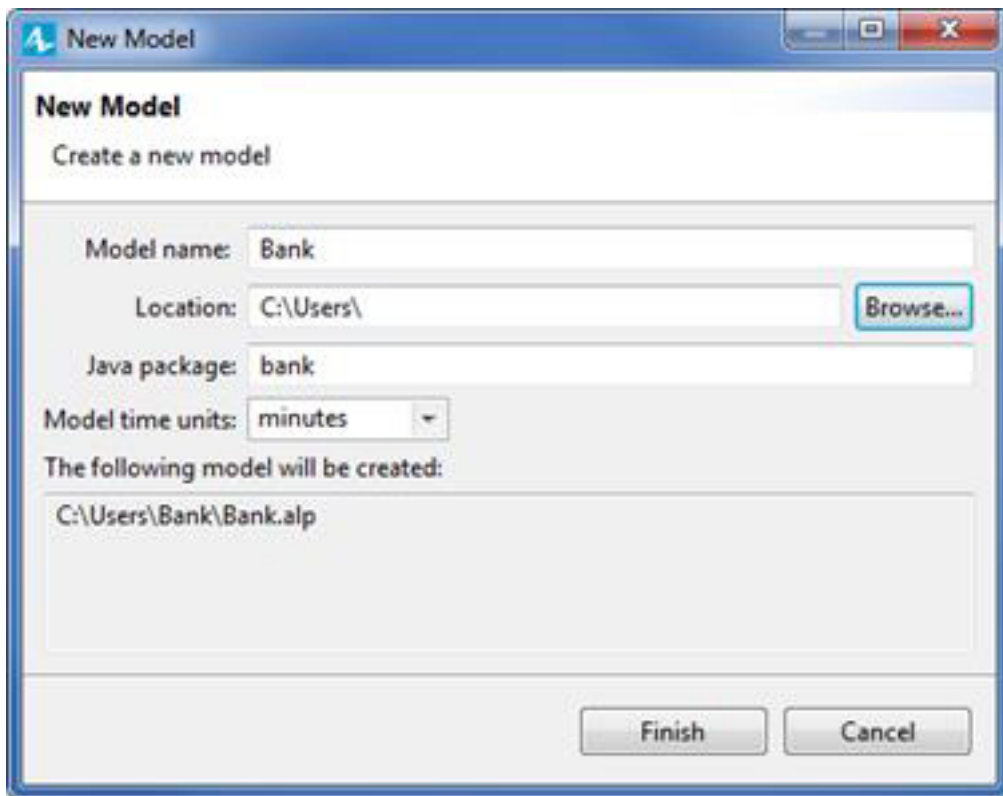
Primeiro, crie um modelo simples que os clientes espesso a um caixa eletrônico.

Criar módulo

Clique no botão New  para a janela de criação do modelo.

Escolha um nome (Banco)

Clique no botão



Escolha onde você deseja armazenar seus arquivos no modelo.

Procure o arquivo existente usando o botão Procurar ou digite o nome da pasta que você deseja criar na caixa local.

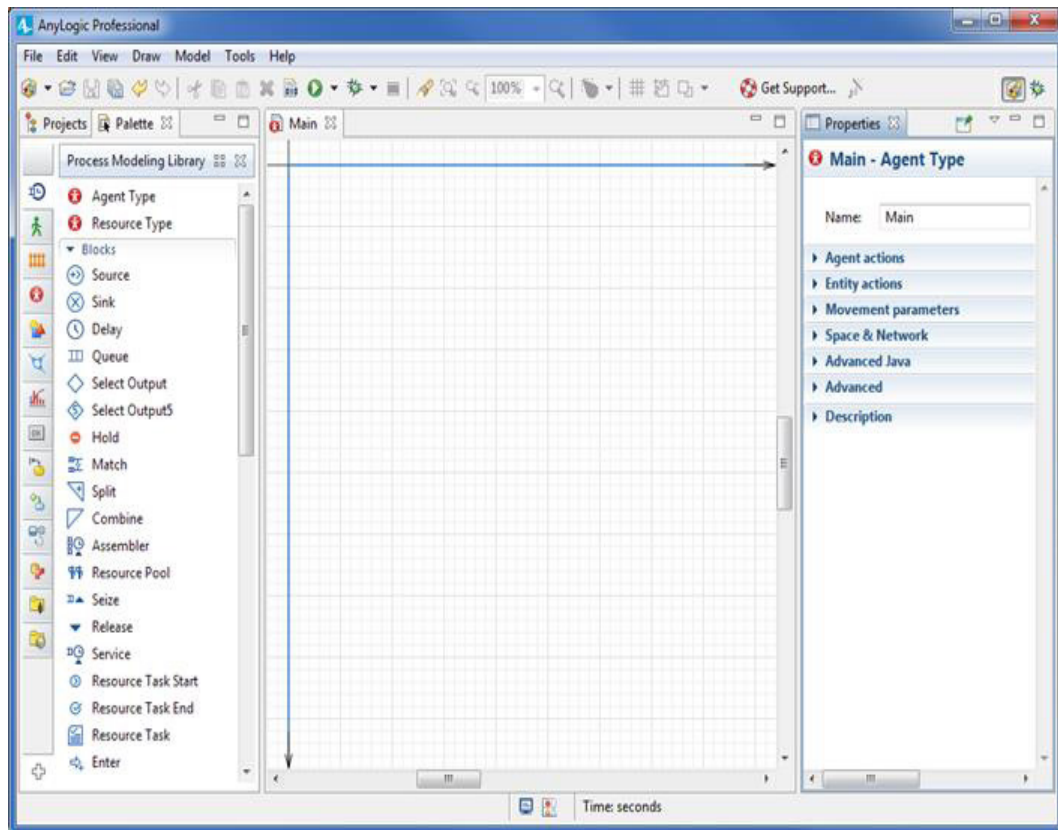
Selecione minutos para unidades de tempo modelo.

Clique em Concluir para concluir o processo.

Um novo modelo é criado. Ele já tem um tipo de agente ligado um experimento chamado principal e Simulação.

Os agentes são os principais elementos de um modelo anylogic. No nosso caso, o agente principal será a entidade na qual vamos definir toda a lógica do modelo: aqui vamos colocar o fluxograma do processo, e definir uma animação simples.

No centro do espaço de trabalho, você vai ver o editor gráfico. Ela mostra o diagrama do Tipo principal.



À esquerda do editor de gráfico, você pode ver a vista Projetos e visualização Paleta que compartilham a mesma área.

Projetos vista fornece acesso aos modelos anylogic atualmente abertas na área de trabalho.

A árvore da área de trabalho fornece uma navegação fácil através de modelos.

view La Palette contém todos os gráficos que podem ser adicionados no editor de gráficos do seu agente simplesmente arrastando e soltando. os elementos do modelo são agrupados em categorias num certo número de paletes.

No lado direito da área de trabalho, você pode ver a vista em Propriedades. A visualização Properties é utilizado para visualizar e modificar as propriedades dos elementos do modelo selecionado.

Quando você seleciona alguma coisa - por exemplo, na visualização Projetos ou no editor gráfico - a visualização Propriedades exibe as propriedades da seleção. Agora podemos começar a desenvolver o modelo.

Criando o gráfico de definição do processo

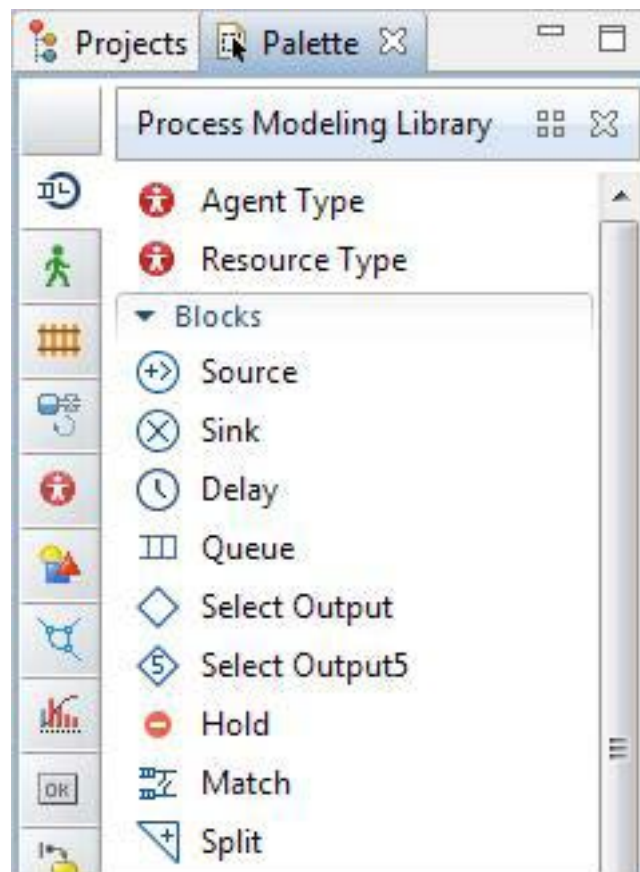
Agora vamos definir o processo com uma organização composta de Modelagem Biblioteca blocos de Processos.

Cada bloco define aqui uma operação que vai ser levada a cabo por agentes à medida que passam através deste bloco.

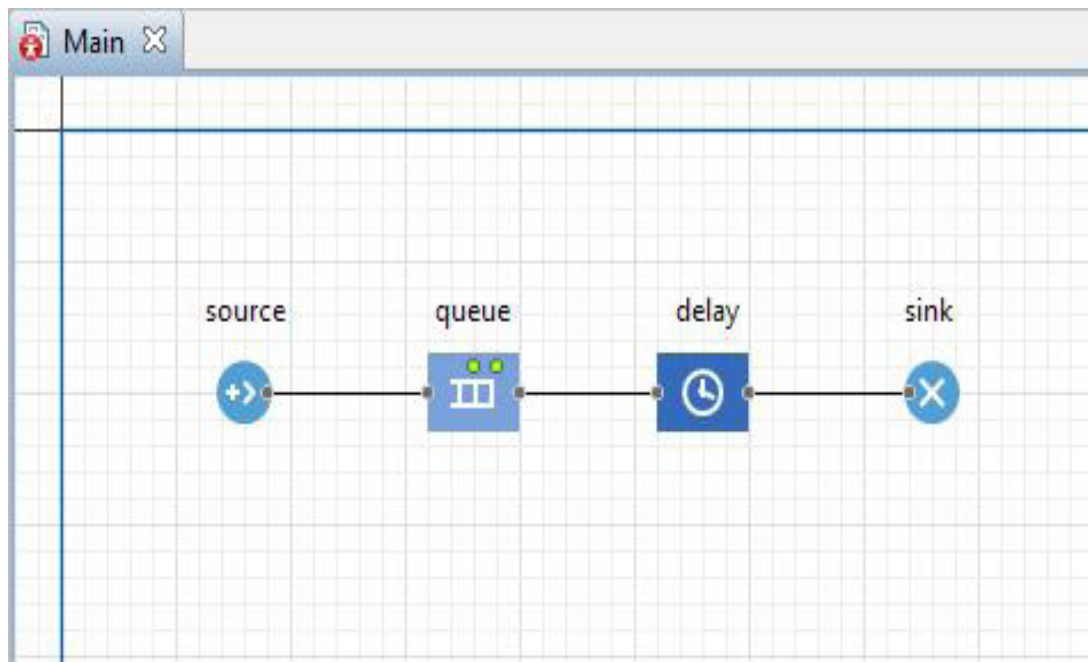
Em anylogic você cria fluxogramas adicionando blocos da biblioteca palette sobre o padrão gráfico, conectando os blocos juntos ao ajustar os parâmetros dos blocos.

Criar um modelo organizacional

Depois de criar o modelo, anylogic vai passar para o ponto de vista da palette eo Processo de exibição palette Modeling Library:



Adicionar Modeling Biblioteca blocos de processo no diagrama e conectá-los como mostrado na figura a seguir. Para adicionar uma unidade organizacional, clique no item e mantenha e colocá-lo no editor gráfico.



Enquanto blocos deslizantes e colocando-os próximos uns dos outros, você pode ver as linhas que ligam os blocos que aparecem.

Note-se que esses conectores deve apenas conectar as portas nas fronteiras esquerda e direita dos ícones de bloco.

O fluxograma indicado na figura anterior, mostra o modelo simples uma fila consistindo de uma fonte de agentes, um atraso (e uma fila que antes) e objeto receptor final.

uma fila consistindo de uma fonte de agentes, um atraso (e uma fila que antes) e objeto receptor final



Source Fonte deste bloco gera agentes. Ele é geralmente utilizado como o ponto de partida de um processo. No nosso caso, os modelos da chegada dos clientes



Queue «Cauda» Fila deste bloco representa uma fila. No nosso modelo, há uma fila de espera de clientes introduzir o distribuidor.



Delay «Atraso» este bloco representa o atraso associado com o distribuidor.



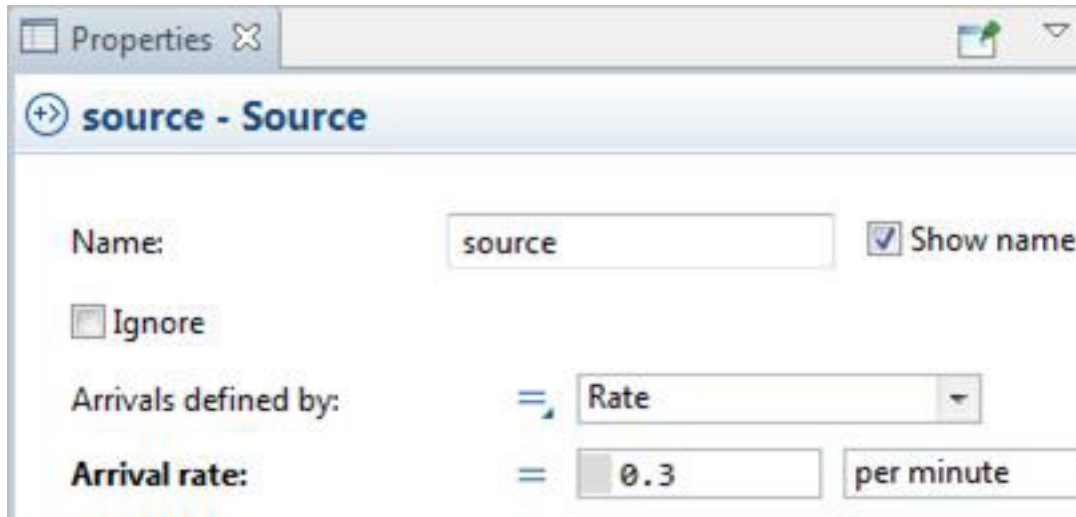
Sink Afundar este bloco representa o fim do processo e remove agentes acessá-lo.

Consulte o Guia de Modelagem de Processos Biblioteca de Referência para obter detalhes.

Configurar os blocos do fluxograma

Para alterar as propriedades de um item, selecione-o clicando no editor gráfico da visualização Projetos. Isso faz com que a abertura do item na visualização Propriedades.

Selecione o bloco de origem na visualização Properties, e especificar quantas vezes os clientes chegam. 0.3 Tipo e selecione Chegada tarifa por minuto.



Properties

source - Source

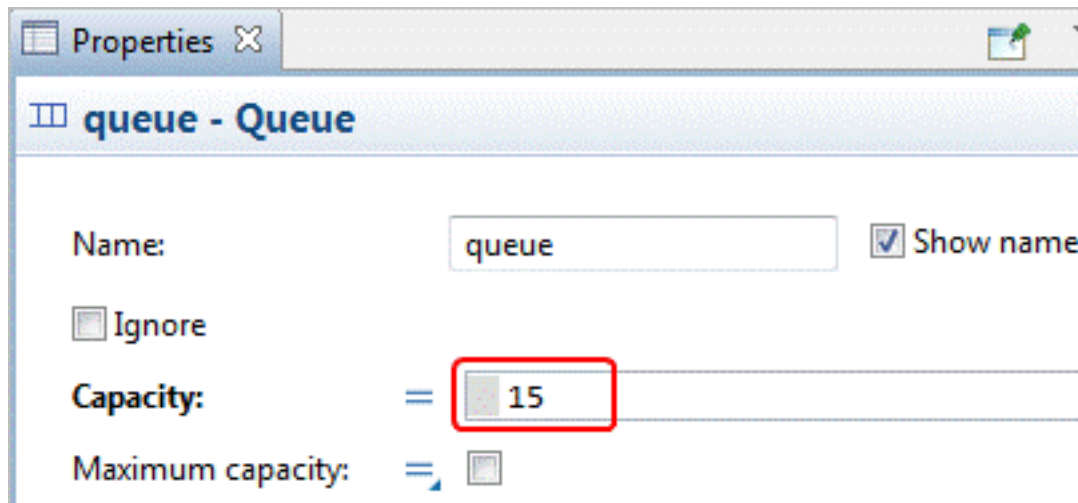
Name: ☒ Show name

☐ Ignore

Arrivals defined by:

Arrival rate:

Editar as propriedades da unidade de cauda. Corrigir a capacidade de 15 agentes. Mais de 15 trabalhadores podem esperar nesta fila.



Properties

queue - Queue

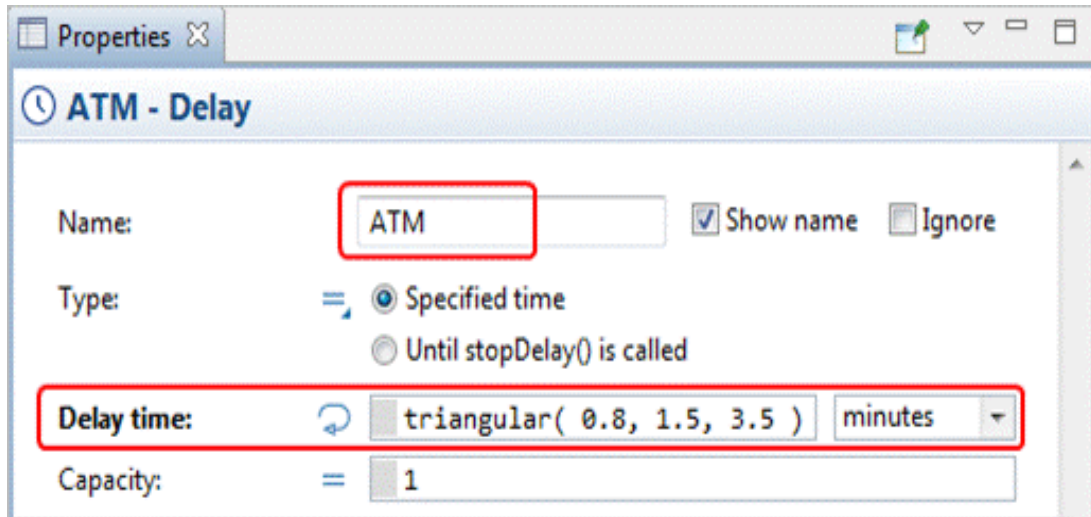
Name: ☒ Show name

☐ Ignore

Capacity:

Maximum capacity:

Alterar as propriedades de atraso. Nomear o objeto ATM (distribuidor). Especifique o tempo de processamento. Suponha-se que o tempo é descrita por uma distribuição triangular da média de 1,5, 0,8 mínimos e máximos de 3,5 minutos.



Função triangular () é um gerador de números aleatórios padrão de anylogic.

Anylogic também tem outras distribuições aleatórias, como a distribuição normal, Poisson, ou exponencial.

Rever essas distribuições utilizando anylogic Execute o modelo.

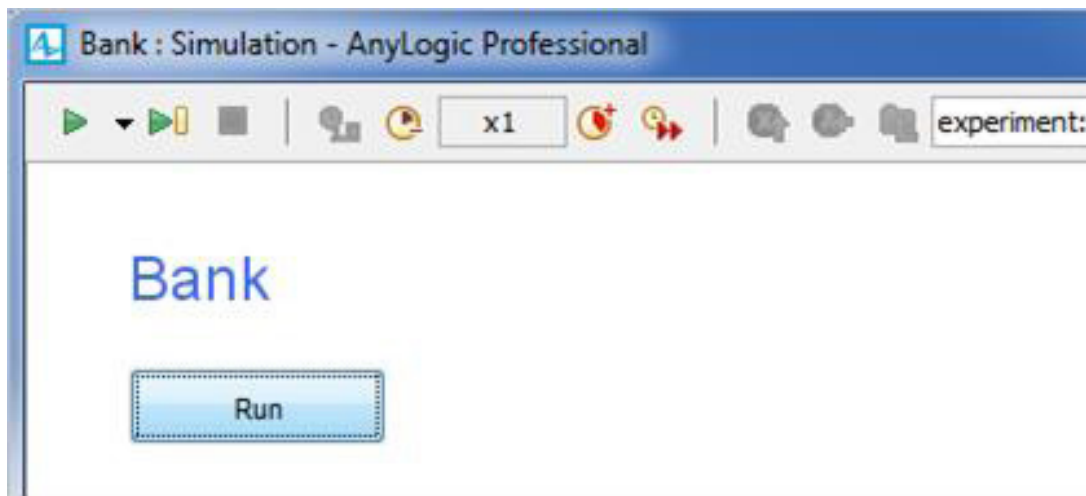
Agora que terminamos nosso modelo simples, podemos executá-lo. Mas primeiro você tem que compilar o modelo clicando em Criar Modelo.

Se houver erros, a compilação falha e os erros serão exibidos. Clique duas vezes o erro e corrija-lo.

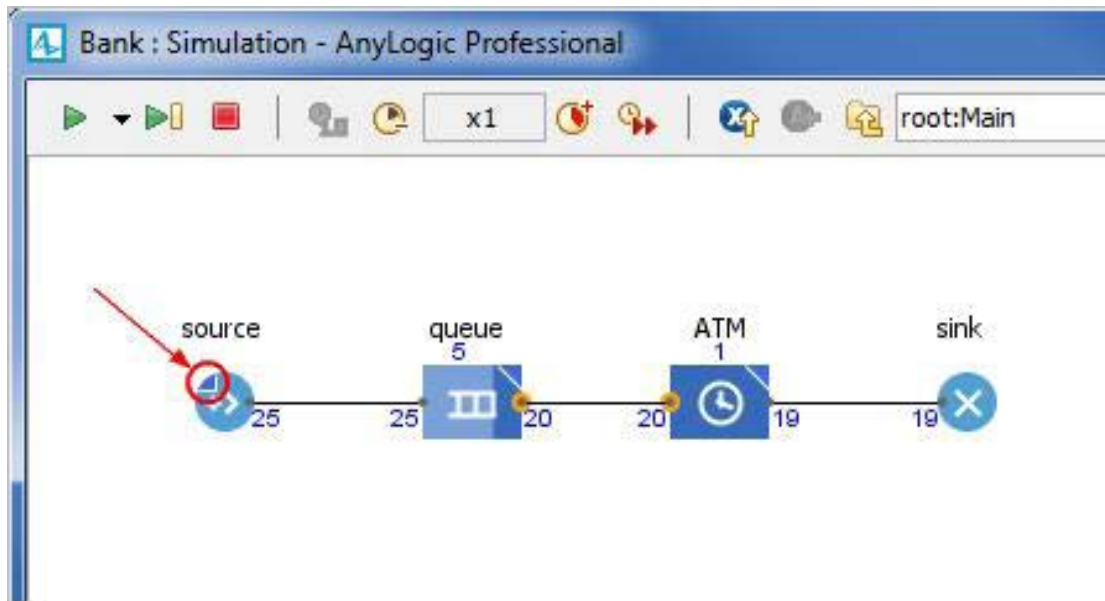
Uma vez corretamente modelo compilado, você pode executar.

Lançar o modelo

Clique no botão Run  e escolher a experiência que você deseja iniciar a partir da lista suspensa. A simulação é chamado de Banco / Simulação.



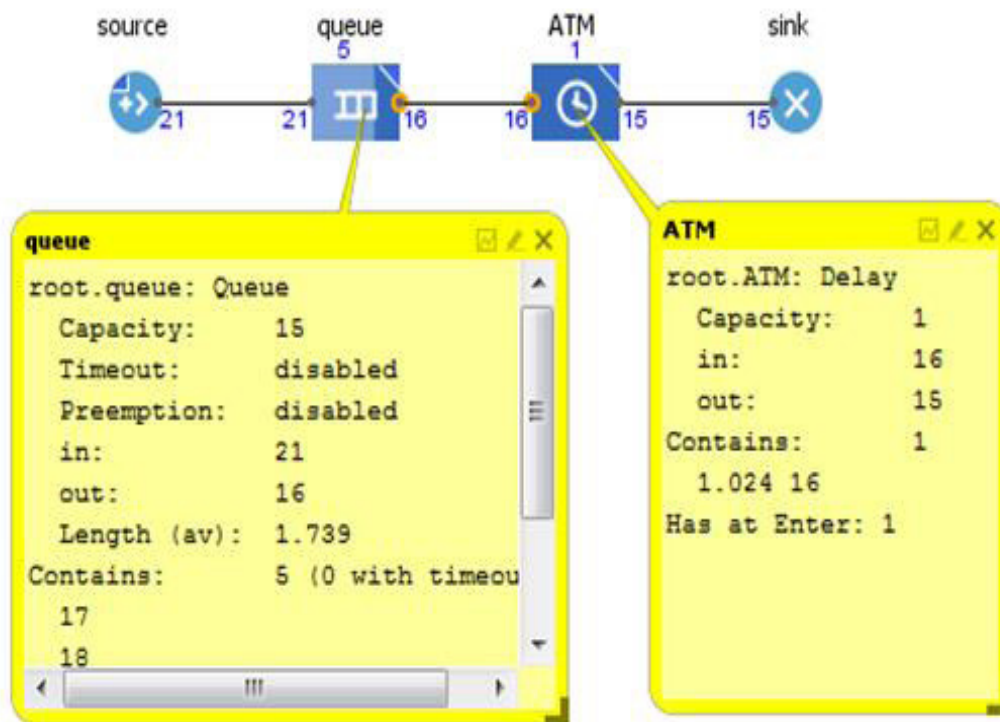
Clique no botão Executar e observe os processos dinâmicos. Você vai ver um gráfico animado com informações como o tamanho da fila, o número de agentes restantes e assim por diante!



necessário, ajustar a velocidade de execução, utilizando abrandar e acelerar na barra de ferramentas.

Você pode inspecionar os objetos do fluxograma para obter informações detalhadas sobre o seu estado atual.

Clique no objeto para abrir a janela que exibe estatísticas inspecionar o objeto, por exemplo, para o inspetor objeto Fila mostra a capacidade da fila, o número de funcionários passou por um dos ports do objeto e também se a opção de tempo limite está habilitado para essa fila. A cadeia contém exibe o número de agentes actualmente em progresso no assunto, bem como os identificadores destes agentes.



Conclusão

Aqui é o seu primeiro modelo com um modelo de eventos discretos. Esta não é uma introdução, mas você vai levar um projeto para completar esta fila modelo na frente do distribuidor.

Avaliação

Deixar o código Java em uma simulação de filas

Faça uma dissertação de 4 páginas sobre variáveis aleatórias discretas

Resumo da Unidade

Esta unidade coberto a última formalismo de modelagem nomeadamente evento discreto. Você já viu três grandes famílias de modelos que constituíam o curso.

Avaliação da Unidade

Verifique o seu entendimento

Projecto para fazer

Instruções

O projeto tem duração de uma semana, você vai encontrar o tutorial Bank Office (Process Centric) no local de anylogic e continuar.

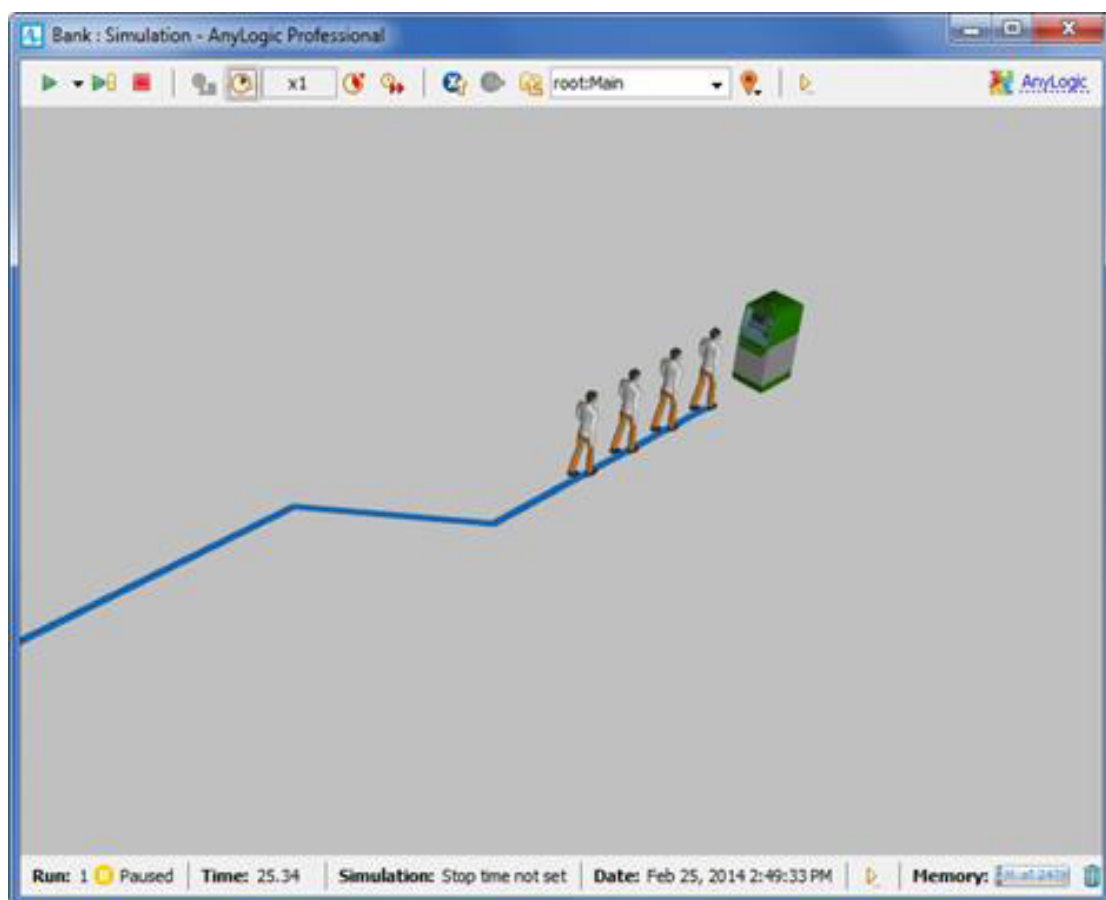
Sistema de classificação

O projecto é classificado para 20

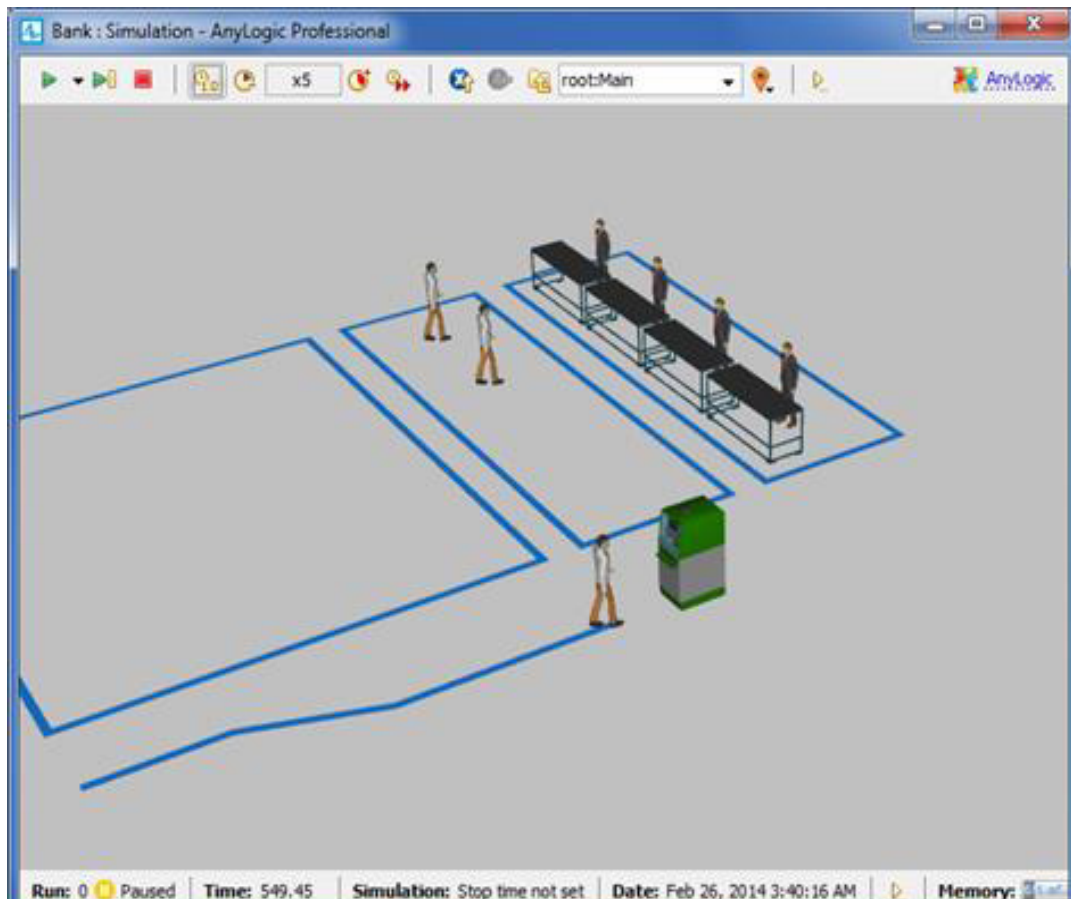
Avaliação

Continue o modelo desenvolvido no tutorial a considerar:

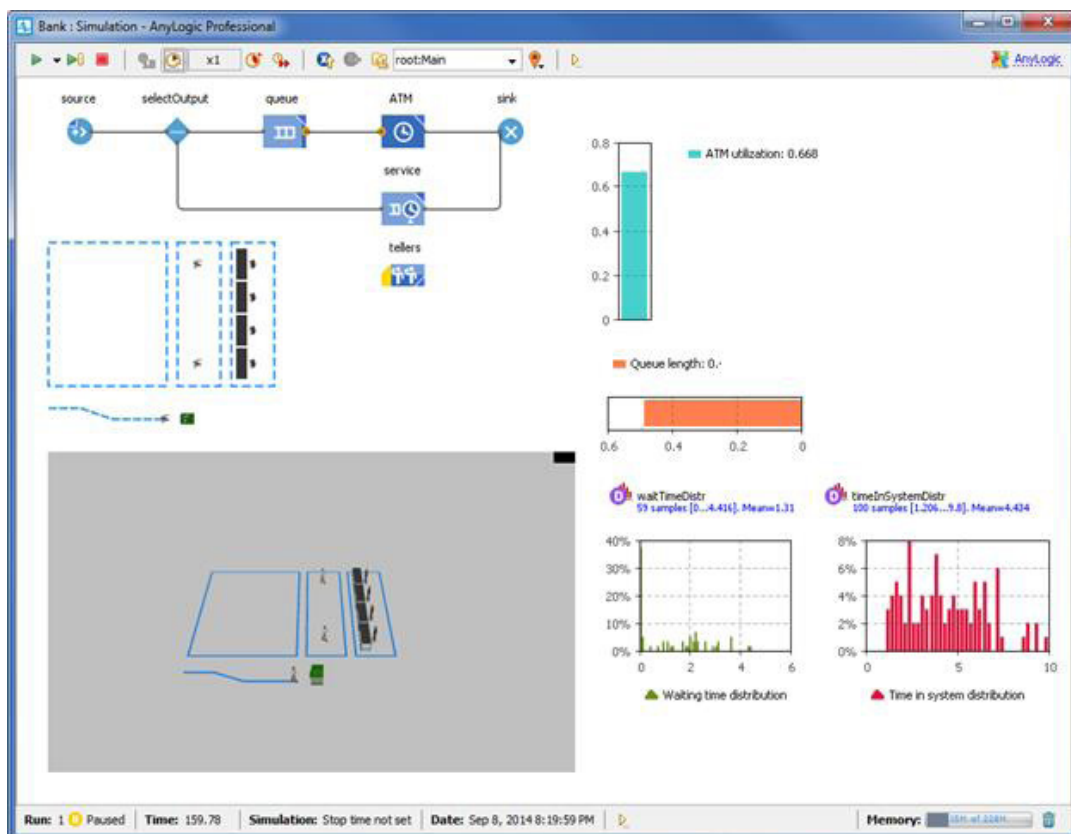
Display gráfico



Adição de gerentes de contas e fundos



E fazer estatísticas



Mostrar como adicionar recursos

Leituras e outros recursos

Leituras e outros recursos para esta unidade são leituras de nível de leituras e outros recursos durante o curso.

Sede da Universidade Virtual africana

The African Virtual University
Headquarters

Cape Office Park

Ring Road Kilimani

PO Box 25405-00603

Nairobi, Kenya

Tel: +254 20 25283333

contact@avu.org

oeer@avu.org

Escritório Regional da Universidade Virtual Africana em Dakar

Université Virtuelle Africaine

Bureau Régional de l'Afrique de l'Ouest

Sicap Liberté VI Extension

Villa No.8 VDN

B.P. 50609 Dakar, Sénégal

Tel: +221 338670324

bureauregional@avu.org