



Planificaciones

7526 - Simulación

Docente responsable: CAIAFA CESAR FEDERICO

OBJETIVOS

Proveer los conocimientos necesarios para la implementación de técnicas de simulación numérica en un lenguaje de programación abierto (Octave/Matlab/Python) permitiendo al alumno no solo utilizar paquetes de simulación disponible sino también desarrollar herramientas numéricas a medida.

CONTENIDOS MÍNIMOS

-

PROGRAMA SINTÉTICO

UNIDAD 1: Introducción a la simulación numérica.

UNIDAD 2: Simulación de eventos aleatorios.

UNIDAD 3: Sistemas dinámicos.

UNIDAD 4: Simulación de eventos discretos y teoría de colas.

PROGRAMA ANALÍTICO

UNIDAD 1: Introducción a la simulación numérica.

Qué es y para qué necesitamos simular. Aplicaciones en Ciencias e Ingeniería. Modelos matemáticos de sistemas. Tipos de Modelos: determinísticos – estocásticos, continuos – discretos, estáticos – dinámicos, lineales – no lineales, analíticos – numéricos. Ejemplos en las ciencias e Ingeniería. Lenguajes generales de programación. Lenguajes orientados a la Simulación, estudio comparativo de diversos lenguajes para simulación de sistemas continuos (Python/Octave/Matlab/COMSOL) y eventos discretos (SimPy, GPSS, Matlab Toolbox, Arena, Simio).

UNIDAD 2: Simulación de eventos aleatorios.

Generación de números al azar. Números pseudoaleatorios y aleatorios. Algoritmo Congruente Lineal.

Tests estadísticos de aleatoriedad.

Métodos para distribuciones discretas y continuas (transformada inversa, aceptación y rechazo).

Métodos específicos de generación de números al azar para distribuciones habituales.

Procesos estocásticos:

Procesos de Poisson.

Cadenas de Markov.

UNIDAD 3: Sistemas dinámicos

Espacio de fases. Estabilidad. Puntos de equilibrio.

Sistemas lineales-no lineales. Sistemas complejos.

Resolución numérica de ecuaciones diferenciales.

Bifurcaciones.

Sistemas caóticos.

UNIDAD 4: Simulación de eventos discretos y teoría de colas.

Sistemas eventos discretos. Programación orientada a actividades, eventos y procesos.

Simulación mediante Simpy. Definición de la Estrategia de Simulación, su implementación y ejecución. Análisis, crítica y presentación de los resultados obtenidos.

Introducción a Teoría de colas. Fórmula de Little. Colas y cadenas de Markov. Colas con un servidor M/M/1 y M/M/1/K. Colas con múltiples servidores M/M/c, M/M/c/c y M/M/∞. Colas M/G/1. Teorema de Burke.

BIBLIOGRAFÍA

1- Steven I. Gordon and Brian Guilfoos, Introduction to Modeling and Simulation with MATLAB and Python, CRC Press, 2016.

2- Hiroki Sayama, Introduction to the Modeling and Analysis of Complex Systems, Open Suny, 2015.

3- Donald Knuth, The Art of Computer Programming, Third Edition, Addison-Wesley, 2011.

4- C. Moller, Numerical Computing with MATLAB, SIAM 2004.

5- Steven M. Kay, Intuitive Probability and Random Processes using MATLAB, Springer, 2006.

6- Svein Linge, Hans Petter Langtangen, Programming for Computations – Python. Springer, 2016.

7- A Discrete-Event Simulation Course Based on the SimPy Language - Norm Matloff, University of California, 2011.

8- <https://simpy.readthedocs.io/en/latest/> (Simpy documentation)

9- Hans Petter Langtangen, A Primer on Scientific Programming with Python, Springer, 2016.

10- A. B. Shiflet and G. W. Shiflet, Introduction to Computational Science: Modeling and Simulation for the Sciences, Princeton University Press, 2014.

11- S. Lynch, Dynamical Systems with Applications using MATLAB, Birkhauser Basel, 2014.

12- B. H. Hahn and D. T. Valentine, Essential MATLAB for Engineers and Scientists, Elsevier, 2013.

- 13- R. E. White, Computational Mathematics Models, Methods and Analysis with MATLAB, Chapman&Hall, 2004.
- 14- A. Papoulis, Probability, Random Variables and Stochastic Processes, Mc Graw Hill, 2002
- 15- Alberto Leon-Garcia, Probability, Statistics, and Random Processes for Electrical Engineering, Pearson Prentce Hall, 2008.
- 16- Mor Harchol-Balter, Performance Modeling and Design of Computer Systems: Queueing Theory in Action, Cambridge University Press, 2013.
- 17- Arnold O. Allen, Probability, Statistics and Queueing Theory with Computer Science Application, Academic Press, 1990.
- 18- Sheldon M. Ross, Simulation, Elsevier, 2006.
- 19- Sheldon M. Ross, Introduction to Probability Models, 9th edition, Elsevier, New York, 2007.

RÉGIMEN DE CURSADA

Metodología de enseñanza

Clases teóricas y prácticas. En las clases teóricas, se seguirán textos propuestos por el profesor donde se expondrán los conceptos teóricos y se ilustrarán los mismos a través de la implementación de simulaciones utilizando lenguajes de programación (Octave / Matlab, Python) y paquetes de software de simulación (por ej. SimPy). En las clases prácticas, los alumnos, guiados por los docentes auxiliares, resolverán ejercicios correspondientes a las unidades temáticas del programa. Se simularán sistemas en problemas encontrados en distintas aplicaciones científicas y de la Ingeniería.

Durante el curso se planteará uno o dos trabajos prácticos que los alumnos deberán desarrollar en grupo.

Modalidad de Evaluación Parcial

Modalidad de Evaluación Parcial:

Se evaluará en todo momento del curso el manejo de conceptos, aplicación de conocimientos y dominio de técnicas mediante:

1. Resolución de problemas (práctica):
 - a. Evaluaciones parciales y
 - b. Evaluación de trabajos prácticos individuales o en grupo, donde se evaluará: (i) Presentación en tiempo y forma, (ii) Método de desarrollo y (iii) Cumplimiento de los objetivos del programa.
2. Evaluación Integradora (coloquio) o presentación de trabajo final basado en un artículo científico.

CALENDARIO DE CLASES

Semana	Temas de teoría	Resolución de problemas	Laboratorio	Otro tipo	Fecha entrega Informe TP	Bibliografía básica
<1> 09/03 al 14/03	Unidades 1 y 2: Introducción a la Simulación. Utilización de computadoras para la simulación.	GUIA 0: Probabilidad y variables aleatorias	Herramientas computacionales (software) Lenguajes de programación (Matlab, Python). Paquetes o toolboxes (GPSS, SimPy, Arena, Simio, Simulation Matlab Toolbox, COMSOL, etc.) Introducción GUIA 0 (repaso probabilidades)			[1, Cap 1] [2, Cap 2] [5, Cap 2]
<2> 16/03 al 21/03	Unidad 3: Probabilidad y Variables aleatorias. Generación de números al azar. Tests de aleatoriedad	GUIA 1: Generación de números aleatorios y tests de aleatoriedad. Ejercicios en Matlab / Octave o Python.	Repaso de Probabilidades. Métodos generación número aleatorios. Tests de aleatoriedad: método visual, Chi2, Kolmogorov-Smirnov y otros tests empíricos.			[14, Cap 1-4] [15, Caps 2-3] [3, Cap 3] [4, Cap 9] [15, Caps 8.7]
<3> 23/03 al 28/03	Unidad 3: Generación variables discretas y continuas (transformada inversa, aceptación y rechazo).	GUIA 1: Generación de números aleatorios y tests de aleatoriedad. Ejercicios en Matlab / Octave o Python.	Ejemplos de generación de variables continuas y discretas.			[5, Cap 10] [15, Cap 4] [16, Cap 4] [19, Cap 11] [6, Cap 3.7.3] [9, Cap 8.5] [18, Cap 3.2]
<4> 30/03 al 04/04	Unidad 3: Procesos estocásticos: Procesos de Poisson.	GUIA 2: Procesos estocásticos - Procesos de Poisson. Ejercicios en Matlab / Octave y Python.	Ejercicios y ejemplos simulación procesos de Poisson.			[5, Cap 21] [15, Cap 9] [16, Cap 11] [19, Cap 5]
<5> 06/04 al 11/04	Unidad 3: Cadenas de Markov.	GUIA 3: Cadenas de Markov. Ejercicios en Matlab / Octave y Python. Ejemplo	Ejercicios y ejemplos simulación Cadenas de Markov. PageRank (Google)		Vencimiento TP1	[5, Cap 22] [15, Cap 11] [16, Cap 8-10] [19, Cap 4]
<6> 13/04 al 18/04	Unidad 4: Espacio de fases. Estabilidad. Puntos de equilibrio. Sistemas lineales-nonlineales. Sistemas	GUIA 4: Simulación Sistemas Dinámicos Nonlineales. Ejercicios en Matlab / Octave y Python.	Ejemplos con análisis de estabilidad y puntos fijos en sistemas lineales y no-lineales			[1, Cap 9] [2, Cap 3, 6 y 7] [6, Cap 4] [10, Cap 2-4] [11] [12, Cap 12]

Semana	Temas de teoría	Resolución de problemas	Laboratorio	Otro tipo	Fecha entrega Informe TP	Bibliografía básica
	complejos.					
<7> 20/04 al 25/04	Unidad 4: Simulación de Sistemas no-lineales continuos	GUIA 4: Simulación Sistemas Dinámicos No-lineales. Ejercicios en Matlab / Octave y Python.	Integración numérica, Runge Kutta. Ejemplos solución sistemas no-lineales.			[2, Cap 6] [3, Cap 7] [6, Cap 4] [10, Cap 6]
<8> 27/04 al 02/05	Unidad 4: Bifurcaciones	GUIA 4: Simulación Sistemas Dinámicos No-lineales. Ejercicios en Matlab / Octave y Python.	Ejemplos con bifurcaciones.			[2, Cap 8] [3, Cap 7]
<9> 04/05 al 09/05	Unidad 4: Sistemas caóticos	GUIA 4: Simulación Sistemas Dinámicos No-lineales. Ejercicios en Matlab / Octave y Python.	Ejemplos sistemas caóticos			[2, Cap 8] [3, Cap 7]
<10> 11/05 al 16/05	Repaso Unidades 1 - 4	Primer PARCIAL				
<11> 18/05 al 23/05	Unidad 5: Sistemas eventos discretos. Programación orientada a actividades, eventos y procesos.	GUIA 5: Ejercicios Simpy	Intro Simpy. Ejemplos.			[7, Cap 1-4] [8]
<12> 25/05 al 30/05	Unidad 5: Introducción a Teoría de colas. Fórmula de Little.	GUIA 5: Ejercicios Simpy	Ejemplos de Colas con Simpy			[15, Cap 12] [16, Cap 1-2] [17, Cap 5]
<13> 01/06 al 06/06	Unidad 5: Colas M/M/1. Colas y cadenas de Markov.	GUIA 5: Ejercicios Simpy	Simulación de Colas con Simpy		Vencimiento TP2	[15, Cap 12] [16, Cap 1-2] [17, Cap 5]
<14> 08/06 al 13/06	Unidad 5: Colas M/G/1. Teorema de Burke y the Jackson.	Primer Recuperatorio GUIA 5: Ejercicios Simpy	Simulación de Colas con Simpy			[15, Cap 12] [16] [17, Cap 5]
<15> 15/06 al 20/06	Repaso Unidades 1 - 5	Repaso GUIAS 1-5	Repaso			
<16> 22/06 al 27/06	Repaso Unidades 1 - 5	Repaso GUIAS 1-5	Repaso			

CALENDARIO DE EVALUACIONES

Evaluación Parcial

Oportunidad	Semana	Fecha	Hora	Aula
1º	10	13/05	19:00	
2º	14	10/06	19:00	
3º				
4º				
Observaciones sobre el Temario de la Evaluación Parcial				
En la evaluación parcial entran los contenidos de las unidades 1 a 4. Existen 2 fechas de recuperatorio.				
Otras observaciones				
Teórica: 3hs/semana (Viernes 19-22hs) Práctica: 3hs/semana (Miércoles 19-22hs) Días y horarios a confirmar por el Depto				