

Examen Final Simulación - Resumen

Módulo 1 - Sistemas y Modelos

SIMULACIÓN

Proceso de diseñar un *modelo* de un *sistema* y llevar a cabo experiencias con él, a fin de analizar su comportamiento y/o evaluar diversas estrategias para un mejor funcionamiento.

SISTEMA

Conjunto de componentes relacionados que interactúan entre sí, o bien conjunto de reglas o principios sobre una materia vinculados entre sí.

- Caracterización

- **Parámetros:** Medidas o datos independientes que configuran la entrada y la estructura interna del sistema.
- **Variables:** Medidas o datos que dependen de los parámetros o de otras variables.
- **Estado:** Es el conjunto de valores que poseen los parámetros y variables del sistema en un determinado momento.

- Clasificación

- **Real vs Abstracto:** Un sistema real existe físicamente, uno abstracto está conformado por conceptos abstractos.
- **Natural vs Artificial:** Dependen de la intervención del hombre en el mismo.
- **Estático vs Dinámico:** El sistema es estático si su estado no cambia con el tiempo. Un sistema dinámico se clasifica a su vez pudiendo ser: Continuo, Discreto, o Combinado.
 - # **Continuo:** Las variables que dependen del tiempo cambian de valor en forma continua.
 - # **Discreto:** Las variables que dependen del tiempo cambian de forma discontinua.
 - # **Combinado:** Sistemas híbridos donde algunas variables cambian de forma continua y otras discontinua.

SISTEMAS DINAMICOS

- Elementos básicos:

- **Recursos físicos circulando:** En un sistema continuo suele ser flujo de materia, mientras que para sistemas discretos se habla de entidades que cambian de estado o se transforman con el transcurso del tiempo.
- **Repositorios donde se acumulan los recursos físicos del sistema:** En un sistema continuo son denominados niveles dado que existen cantidades no discretas de materia acumulada, mientras que en sistemas discretos los repositorios se denominan colas.

MODELOS

Un modelo es una representación del sistema, en su construcción se representan aquellos aspectos del sistema que se consideren relevantes para el estudio que se intenta efectuar.

- Tipos de Modelos:

- **Físicos:** Reproducciones del sistema que representan (pueden ser maquetas).
- **Gráficos:** Representaciones en dos o tres dimensiones del sistema original.
- **Simbólicos:** Representaciones abstractas del sistema (modelos matemáticos por ejemplo).

- Evaluación de Modelos

Para modelos físicos o gráficos la evaluación del modelo se puede realizar por observación o medición directa.

Para modelos simbólicos la solución está oculta dentro de la estructura del modelo abstracto, se puede resolver de dos formas:

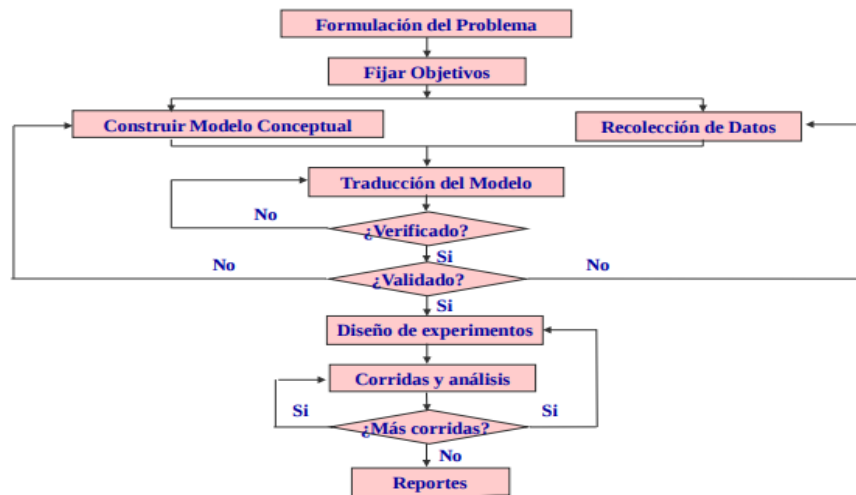
- **Métodos Analíticos:** Se suelen aplicar en modelos matemáticos, se fundamentan en el razonamiento deductivo de las teorías matemáticas que se emplearon en la construcción del modelo. Se caracterizan por ser precisos y obtener soluciones rápidamente. Sólidos y robustos. Pierden precisión o llegan a ser inaplicables en problemas complejos que poseen características que no se pueden modelar con teorías matemáticas ya disponibles. Ejemplos de este tipo de métodos: Técnicas de resolución de ecuaciones de integración numérica y derivación.
 - **Métodos Experimentales:** Tienen un enfoque simplificado, muchas veces basado en el sentido común. Se establece un grado de compromiso entre simplicidad y facilidad versus alto tiempo de cómputo e imprecisión en los resultados. En muchos problemas reales y complejos el enfoque experimental es el único factible.
-

Módulo 2: Introducción a la Simulación

SIMULACIÓN COMPUTACIONAL

Proceso de diseñar un modelo computacional de un sistema y llevar a cabo experimentos con él, a fin de evaluar el impacto de introducir cambios tanto en la estructura interna del sistema como en las condiciones vinculadas al entorno del mismo.

Ciclo de Vida de un Modelo Computacional para Simulación



MODELOS ESTOCÁSTICOS vs DETERMINÍSTICOS

Un modelo es estocástico o no determinístico si posee variables aleatorias (variables cuyos valores presentan cierto nivel de incertidumbre). Un modelo que no posee variables aleatorias es determinístico.

SIMULACIÓN DE MONTE CARLO

Es un experimento cuyo propósito es estimar la distribución de una variable de salida que depende de variables de entrada aleatorias. Se aplica sobre modelos estáticos.

Módulo 3: Modelos de Simulación Dinámica

MODELOS DE SIMULACIÓN PARA SISTEMAS DINÁMICOS

Las variables dinámicas incluidas en este tipo de modelos pueden cambiar de valor en alguna de las siguientes 4 formas:

- **Continuamente en cualquier punto del tiempo**
- **Continuamente en puntos de tiempo discretos**
- **Discretamente en cualquier punto del tiempo**
- **Discretamente en puntos de tiempo discretos**

SIMULACIÓN DE SISTEMAS DE COLAS

Es un conjunto de entidades que arriban a una o más colas del sistema en donde esperan ser atendidas por uno o más servidores. Los cambios en las variables dinámicas se producen discretamente en cualquier punto del tiempo.

SIMULACIÓN DE SISTEMAS DE INVENTARIO

Se estudia el crecimiento y decrecimiento del nivel de stock de un producto a lo largo del tiempo. El nivel es chequeado periódicamente a intervalos uniformes de tiempo a fin de establecer la necesidad o no de reaprovisionamiento. Los cambios en las variables dinámicas se producen discretamente y en puntos de tiempo discretos.

SIMULACIÓN DE SISTEMAS CONTINUOS

Este tipo de modelos poseen variables cuyos valores cambian en forma continua respecto del tiempo, dichas variables son denominadas variables de estado.

Módulo 4: Distribuciones Probabilísticas

VARIABLES ALEATORIAS

Una variable aleatoria es discreta cuando el conjunto de valores que puede tomar la variable es finito o infinito contable. Una variable aleatoria se dice continua cuando el conjunto de valores que puede tomar la variable es un intervalo o conjunto de intervalos formado por infinitos números.

DISTRIBUCIONES PROBABILISTICAS

- **Continuas:** se definen por la función de densidad de probabilidad.
 - **Discretas:** se definen por la función másica de probabilidad.
-

Módulo 5: Números Pseudo-Aleatorios y Variables Aleatorias

ALEATORIEDAD EN MODELOS DE SIMULACIÓN

El uso de este concepto se emplea en dos tipos de situaciones:

- Fenómenos que se aprecian realmente como aleatorios (ej, tirar un dado).
- Situaciones que se asumen como aleatorias por falta de mayor información, con el fin de simplificar la lógica del modelo de simulación. En estos casos si bien la variable podría ser calculada determinísticamente, modelarla como tal implica un costo que no se justifica en función de la calidad de los resultados de la simulación.

SECUENCIAS DE NÚMEROS ALEATORIOS UNIFORMEMENTE DISTRIBUIDOS

Una secuencia de números aleatorios R_1, R_2, \dots, R_n , debe satisfacer las dos siguientes propiedades:

- **Uniformidad:** los números de la secuencia siguen una distribución continua uniforme con valores comprendidos entre 0 y 1.
- **Independencia:** cada valor R_i de la secuencia es independiente de los restantes valores R_j .

CARACTERÍSTICAS DESEABLES EN UNA SUBROUTINA PARA GENERACIÓN DE NÚMEROS ALEATORIOS

- **Confiabilidad:** la secuencia debe cumplir en forma muy cercana las propiedades de los números aleatorios.
- **Eficiencia:** bajos run-times.
- **Portabilidad:** la confiabilidad no debe depender de la arquitectura de la máquina o del sistema operativo utilizado.
- **Ciclos largos:** la secuencia de números debe ser lo más extensa posible antes de empezar a repetirse.
- **Replicabilidad:** a veces es necesario poder efectuar corridas con la misma secuencia aleatoria (uso de semillas).

TÉCNICAS PARA GENERACIÓN DE NÚMEROS ALEATORIOS

- **Método Congruencial Lineal:**
Genera una secuencia de m números siguiendo la relación recursiva:
$$N_{i+1} = (a * N_i + c) \bmod m$$
- **Generador Multiplicativo Recursivo:**
Genera cada valor de la secuencia como una combinación lineal de los valores que le preceden:
$$N_{i+1} = (a_1 N_i + a_2 N_{i-1} + \dots + a_t N_{i-t+1}) \bmod m$$

SHUFFLING

Emplea un generador composicional y mezcla la salida. Resulta apropiado en generadores para aritmética de 16-bits los cuales poseen un período de secuencia inherentemente corto.

Ejemplo: Dados dos generadores congruenciales multiplicativos G_1 y G_2 :

1. Completar un vector V con los primeros k valores obtenidos por G_1 :

$$V = (V_1, V_2, \dots, V_k)$$

2. Usar G_2 para generar un valor entero i entre 1 y k .

3. Seleccionar el i -ésimo valor de V como valor aleatorio.

4. Generar un valor aleatorio con G_1 y sustituir V_i por ese valor.

5. Si se necesita otro valor aleatorio se vuelve al paso 2.

TESTS PARA NÚMEROS ALEATORIOS

Existen varios tests tendientes a establecer si una secuencia de números pseudo-aleatorios respeta las propiedades de los números aleatorios.

- **Tests de Frecuencia:** Evalúa la uniformidad de la distribución de los valores (verifica distribución uniforme).
- **Tests de Corridas:** Determina la existencia o no de subsecuencias dentro de la secuencia. Si existen corridas descendentes/ascendentes o por arriba/abajo de la media no se verifica la propiedad de independencia.
- **Tests de Autocorrelación:** Analiza la independencia entre los números de la secuencia para detectar la existencia de alguna correlación entre los valores generados.
- **Test de Gap:** Analiza el intervalo entre la repetición de un dígito y su próxima aparición buscando detectar un patrón general que viole la propiedad de independencia.
- **Test de Poker:** Busca verificar también la propiedad de independencia analizando cómo se repiten los dígitos a lo largo de los números que forman la secuencia pseudo-aleatoria.

GENERACIÓN DE VARIABLES ALEATORIAS

Distribuciones Uniformes

Los generadores congruenciales obtienen secuencias de números aleatorios distribuidos uniformemente entre 0 y 1, el problema es cómo generar valores aleatorios que sigan otras distribuciones.

Distribuciones no Uniformes

Existen diferentes técnicas empleadas:

- **Técnica de la Transformada Inversa**

Explota las propiedades de las funciones de acumulación de probabilidades de las distribuciones, dado que las funciones acumulativas son incrementales estas poseen inversa (es decir, que dado un valor r entre 0 y 1 existe un único valor x tal que $F(x)=r$). Luego, dado ese valor r , el valor aleatorio x se puede obtener como $x=F^{-1}(r)$.

- **Método de Aceptación-Rechazo**

Es aplicable a cualquier variable aleatoria con distribución continua que posea una función de densidad de probabilidad con rango bien definido, pero cuya función de distribución de probabilidad acumulativa no pueda ser expresada mediante integración directa. Desde un punto de vista práctico, se implementa como sigue:

1. Se calcula una constante C tal que $C*f(x) \leq 1$ para $a \leq x \leq b$.
2. Se obtienen dos valores aleatorios u_1 y u_2 .
3. Se calcula $x=a+u_1*(b-a)$
4. Si $u_2 \leq C*f(x)$, se acepta el par (u_1, u_2) y se retorna x como valor aleatorio, en caso contrario se vuelve al paso 2.

Módulo 6: Modelado de la Entrada de una Simulación

ETAPAS DEL MODELADO DE LA ENTRADA

1. Recolección de datos del sistema real.
2. Identificación del tipo de distribución que siguen los datos recolectados.
3. Determinación de los parámetros de la distribución seleccionada.
4. Efectuar tests de bondad de ajuste para determinar si la distribución establecida es realmente una buena aproximación para modelar la variable aleatoria. Si estos fallan se regresa a los pasos anteriores.

1. Recolección de Datos

Etapa más ardua de todo el proceso, es la que requiere más tiempo y mayor nivel de conocimiento y experiencia por parte del modelador.

Pautas: Analizar los datos mientras son recolectados, intentar combinar conjuntos de datos homogéneos, se debe evitar caer en la censura de datos, identificar dependencias entre datos, identificar autocorrelaciones, no confundir datos de entrada con datos de salida.

2. Identificación de tipo de Distribución

Con los datos recopilados sobre una variable se puede determinar que tipo de distribución probabilística se ajusta mejor a los datos mediante la construcción de histogramas (diagramas que permiten observar la frecuencia con que se repiten los distintos valores obtenidos en una muestra).

3. Estimación de Parámetros

Una vez que se tiene una hipótesis sobre el tipo de distribución que siguen los datos, es necesario establecer cuales son sus parámetros, para esto se calculan media y varianza.

4. Test de Bondad de Ajuste

Un procedimiento para testear la hipótesis de que una muestra aleatoria de tamaño n de una variable aleatoria X sigue una distribución específica es aplicar el test de bondad de ajuste de la Chi-Cuadrado, este test formaliza la idea intuitiva de comparar el histograma generado con la forma de la función de densidad de probabilidad o másica correspondiente a la distribución candidata.

Si bien estos tests proveen una buena guía en la selección de distribuciones para las variables aleatorias, no hay que confiar ciegamente... Es importante saber que estos tests son fuertemente influenciados por el tamaño de la muestra: cuando esta es pequeña se tiende a rechazar hipótesis nulas, mientras que si la muestra es muy grande suelen aceptar todas las hipótesis nulas.

Módulo 8: Aplicaciones del Método de Simulación de Monte Carlo

PROBLEMAS DE PLANIFICACIÓN

Consisten de determinar el orden óptimo en el que una serie de actividades deben ser efectuadas a fin de reducir el tiempo total transcurrido entre que empieza la primera tarea y finaliza la última.

En algunos de estos problemas existen ciertas restricciones y/o dependencias en lo que refiere al orden de realización de las tareas, que deben ser tenidas en cuenta a fin de construir una planificación válida.

Las restricciones de orden existentes entre las distintas tareas conforma un orden parcial, el cual puede ser expresado a través de un grafo dirigido acíclico conocido como grafo de dependencias.

En estos grafos se denomina camino crítico al camino más largo desde el nodo raíz hasta cualquiera de las hojas, donde cualquier demora en las actividades ocasionará una demora en la finalización de todo el proyecto.

Módulo 9: Conceptos Básicos de Simulación de Eventos Discretos (DEVS)

CONCEPTOS BÁSICOS:

- **Sistema:** Colección de entidades que interactúan en el tiempo a fin de alcanzar cierto objetivo o meta.
- **Modelo de Simulación basado en Eventos Discretos:** Abstracción que representa un sistema.
- **Estado del sistema:** Colección de variables que contienen toda la información para describir el sistema en cualquier instante del tiempo.
- **Entidades:** Objeto o componente del sistema que requiere explícita representación dentro del modelo.
- **Atributos:** Propiedades relevantes de las entidades que integran el modelo bajo construcción.

- **Repositorios:** Colecciones permanentes o temporales de entidades que circulan por el sistema.
- **Eventos:** Situación que ocurre en un instante de tiempo y modifica el estado del sistema. Pueden ser exógenos o endógenos, los exógenos se generan en el exterior del sistema (por ejemplo tiempos de arribo), mientras que los endógenos se generan en el interior (por ejemplo los tiempos de finalización de servicio).
- **Aviso de evento:** Registro de un evento que está ocurriendo o que ocurrirá en un futuro. Mantiene la información necesaria para ejecutarlo.
- **Lista de eventos:** Lista formada por los avisos de eventos que ocurrirán en el sistema.
- **Actividad:** Representa un tiempo de servicio, un tiempo entre arribos, o cualquier tiempo de procesamiento que el modelador haya especificado.
- **Proceso:** Lista de eventos asociada a una entidad del sistema.
- **Demora:** Lapso de tiempo indefinido en el que una entidad se encuentra detenida dentro del sistema.
- **Reloj de Simulación:** Variable que representa el tiempo simulado.

TIPOS DE ESPERAS

Las demoras se conocen como esperas condicionales ya que están asociadas a la liberación de un recurso, mientras que las actividades se denominan esperas incondicionales.

TIPOS DE EVENTOS

La terminación de una actividad constituye un evento primario, y se maneja ubicando un aviso de evento en la lista de eventos de la simulación. La terminación de una demora suele denominarse evento condicional o secundario, y no es administrada en la lista de eventos.

VISTAS DEL MUNDO: BARRIDO DE ACTIVIDADES

Se concentra en las actividades del modelo y en las condiciones necesarias para llevarlas a cabo. El reloj avanza en intervalos fijos de tiempo, cuando este avanza se chequean las condiciones requeridas para realizar cada actividad. Resulta muy útil en la simulación de sistemas de inventario dado que estos generalmente se chequean en intervalos fijos de tiempo.

Tabla de simulación: Las filas representan el estado del sistema para un dado período.

VISTAS DEL MUNDO: INTERACCIÓN DE PROCESOS

Se define el modelo en término de las entidades u objetos del sistema y en como fluye el ciclo de vida de cada una a través del sistema, demandando y consumiendo recursos. Un proceso corresponde al ciclo de vida de una entidad y se conforma de eventos y actividades.

Tabla de simulación: Cada fila representa el ciclo de vida de un cliente.

VISTAS DEL MUNDO: PLANIFICACIÓN DE EVENTOS

Se administra una lista de eventos donde en un instante t la lista contendrá n eventos con tiempos t_1, t_2, \dots, t_n . En ese momento el reloj de simulación tendrá tiempo t y se debe verificar que: $t < t_1 \leq t_2 \leq \dots \leq t_n$

La simulación avanza tomando siempre el evento más inminente. Luego se lo remueve de la lista, se avanza el reloj al tiempo t_1 , se ejecuta el evento actualizando el estado del sistema, y nuevos eventos pueden o no ser generados e incorporados a la lista preservando el orden cronológico, por último se actualiza la información estadística.

Tabla de simulación: Cada fila representa la ocurrencia de un evento en el sistema.

VISTAS DEL MUNDO: ENFOQUE DE LAS TRES FASES

Combina características de planificación de eventos con barrido de actividades para lograr un avance de tiempo variable y evitar el barrido de condiciones cuando no es necesario.

Es útil en sistemas complejos donde la combinación de varios recursos es necesaria para llevar adelante distintas tareas.

En este método los eventos se consideran como actividades con una duración de 0 unidades de tiempo.

Las actividades se clasifican en dos categorías:

- **Actividades B:** Ligadas a un suceso. Se mantienen en la lista de eventos. La integran todos los eventos primarios y actividades no condicionales.
- **Actividades C:** Se producirán cuando se verifiquen ciertas situaciones.

El método consiste en repetir las siguientes tres fases:

- **Fase A:** Remover de la lista de eventos el próximo evento y avanzar el tiempo de reloj al tiempo de dicho evento. Si hay más eventos con el mismo tiempo también removerlos.
- **Fase B:** Ejecutar los elementos removidos (son actividades B). Se puede modificar el estado del sistema ante la ejecución.
- **Fase C:** Chequear el cumplimiento de las condiciones para cada una de las Actividades C. Ejecutar aquellas cuyas condiciones hayan sido verificadas. Repetir hasta que ninguna actividad adicional de clase C pueda ser ejecutada, entonces retornar a la **Fase A**.

Módulo 10: Sistemas de Colas

CARACTERÍSTICAS DE SISTEMAS DE COLAS

- **Clientes:** Entidades que arriban al sistema. Arriban a la cola en forma aleatoria y avanzan hasta ser atendidas generalmente en el orden en que arribaron.
- **Servidores:** Recursos del sistema que se encargan de atender a los clientes. Sus tiempos de servicio suelen ser aleatorios y siguen una distribución independiente del tiempo.
- **Estado del Sistema:** Determinado por el número de entidades que están circulando por el sistema y el estado del servidor (ocioso/ocupado).
- **Evento:** Conjunto de circunstancias que causan un cambio instantáneo en el estado del sistema.
- **Reloj de la simulación:** Para marcar el progreso del tiempo.

POBLACIÓN DE POTENCIALES CLIENTES

La principal diferencia entre un modelo de población infinita y uno de población finita está dada por como se define la tasa de arribos.

Con **población infinita** la tasa de arribos es el número promedio de arribos por minuto y es independiente del número de clientes que ya estén en la cola o hayan dejado la misma.

Con **población finita** la tasa de arribos depende de la cantidad de clientes que se encuentran en la cola y la de los que ya fueron atendidos. En estos casos la tasa de arribo varía a lo largo de la simulación en función del estado del sistema.

CAPACIDAD DEL SISTEMA

Es la cantidad límite de clientes que pueden esperar en la cola, esta limitación generalmente está dada con la capacidad de la cola en el sistema real que se está modelando (para un cine por ejemplo es potencialmente infinita, mientras que la cantidad programas esperando en un buffer de memoria tiene un tope).

PROCESOS DE ARRIBOS

Los procesos de arribos para modelos de población infinita están generalmente caracterizados por los tiempos entre los arribos de los sucesivos clientes, estos pueden ocurrir en forma planificada o aleatoriamente. En este ultimo caso siguen alguna distribución aleatoria.

En los modelos de población finita los arribos son caracterizados en forma diferente: Se define un cliente como pendiente cuando es miembro de la potencial población de clientes y está fuera del sistema de colas, deja de estar pendiente mientras está en el sistema de colas. Un mismo cliente arriba y sale varias veces a lo largo de la simulación. Se denomina tiempo de corrida al lapso de tiempo transcurrido entre que un cliente deja la cola y vuelve a arribar a la misma. Este tipo de modelos es utilizado en problemas de reparación de maquinas donde un arribo equivale al momento de fallo de una máquina.

COMPORTAMIENTO DE COLAS

Refiere a las acciones que llevan adelante los clientes mientras esperan en la cola:

- un cliente arriba a una cola demasiado larga y decide abandonarla inmediatamente
- un cliente espera un tiempo en la cola y decide irse dado que se mueve muy lentamente
- un cliente cambia de una cola a otra en función de como avanza cada una.

DISCIPLINA DE COLAS

Refiere al orden lógico en que los clientes se sitúan dentro de la cola y a la política utilizada para determinar que cliente tomar de la misma:

- **FIFO**
 - **LIFO**
 - **SIRO:** aleatorio
 - **SPT:** procesa primero al cliente con menor tiempo de servicio
 - **PR:** según ordenes de prioridad.
-

Módulo 11: Modelos Conceptuales para DEVS

METODOLOGÍA DE DISEÑO

- Identificar las clases de entidades más relevantes
- Determinar las actividades en que las entidades están involucradas
- Enlazar las actividades a fin de establecer la secuencia dinámica de tiempo en que estas se producen

CATEGORÍAS DE ENTIDADES

- **Entidades:** Elementos individuales del sistema cuyo comportamiento es explícitamente registrado con el avance del tiempo.
- **Recursos:** Elementos individuales del sistema que no son modelados individualmente, son tratados como objetos contables y su comportamiento individual no es registrado.

OPERACIONES DE LAS ENTIDADES

Las entidades cooperan en la concreción de las operaciones que producen los cambios de estado en el sistema. Estas operaciones y su representación en el tiempo pueden ser descritas en términos de:

- **Eventos:** representan los instantes de tiempo en que se producen los cambios de estado, y van a estar asociados al inicio y fin de las operaciones.
- **Actividades:** representan las operaciones que producen las transformaciones en las entidades.

Para cada entidad del modelo se construye un DCA en donde se describe la secuencia de actividades que conforma el ciclo de vida de dicha entidad.

Estos diagramas poseen dos tipos de símbolos básicos:

- **Estados Activos:** representan actividades cooperativas cuya duración puede determinarse en avance, ya sea determinísticamente o mediante una distribución de probabilidades.
- **Estados Muertos:** Modelan actividades no cooperativas y generalmente representan estados en donde la entidad se queda esperando a que algo suceda.

Clase 17, 18 y 19: Construcción de Modelos de Simulación de Eventos Discretos usando VIMS

SIMULACIÓN UTILIZANDO VIMS

Se debe especificar el modelo conceptual a través de los constructores brindados por el paquete, el usuario solo debe preocuparse de especificar la topología de los procesos del sistema y la correcta parametrización de las unidades que los componen. La traducción al modelo computacional es realizada internamente por el software de forma totalmente transparente.

SIMUL8 - CONSTRUCTORES DE MODELOS

- **Puntos de Entrada de Trabajo:** Lugares por donde arriban las entidades que circulan por el sistema. *DEVS: Tiempo entre Arribos*
- **Áreas de Almacenamiento:** Lugares donde las entidades esperan para ser procesadas. *DEVS: Repositorios*
- **Centros de Trabajo:** Modelan el trabajo realizado por los servidores del sistema. *DEVS: Actividades*
- **Punto de Salida:** Lugares donde las entidades dejan el sistema una vez finalizado su proceso.

SIMUL8 - ENTIDADES DE LOS MODELOS

- **Items de Trabajo:** Entidades que circulan por el sistema, siendo procesadas por las estaciones de trabajo. Son generadas en los *puntos de entrada* de manera automática.
- **Recursos:** Entidades necesarias cuando las estaciones de trabajo compiten por recursos.

MAPEO AL MODELO COMPUTACIONAL

- **Regla 1:** Los estados activos son mapeados a constructores activos (puntos de entrada y estaciones de trabajo)
- **Regla 2:** Los estados muertos son mapeados a constructores pasivos (repositorios y puntos de salida)
- **Regla 3 (excepción a regla 1):** Los estados activos que representan el traslado de un servidor a un repositorio o punto de salida son mapeados como enlaces.
- **Regla 4 (excepción a regla 2):** Los estados muertos asociados a situaciones de bloqueos quedan embebidos en los estados activos correspondientes a los servidores bloqueados.

Módulo 20: Verificación y Validación de Modelos

VERIFICACIÓN

Compara el modelo conceptual con la representación computacional. Se responde a las siguientes preguntas:

- ¿Está el modelo implementado correctamente?
- ¿Los parametros de entrada y la estructura lógica del modelo están correctamente representados?

Existen dos tests estadísticos que indican la razonabilidad del modelo operacional:

- **Contenidos Actuales:** Refiere a la cantidad de items en cada componente del sistema en un dado instante de tiempo. Si esta cantidad es alta hay una gran cantidad de entidades demoradas en el sistema, si esta crece a lo largo del tiempo en un sistema de colas se está ante un sistema inestable lo que requeriría agregar más servidores al sistema.

- **Contador Total:** Refiere a la cantidad total de items que han entrado a cada componente del sistema en un determinado instante de tiempo. Si el contador total para un componente del sistema es 0, es que nunca ha entrado ningun item en este componente. Si el contenido actual de un componente y el contador total es 1 puede ser que haya capturado un recurso y no lo haya liberado.

VALIDACIÓN

Es utilizada para determinar si el modelo construido representa en forma precisa el sistema real que se desea simular. La validación se logra a través de la calibración del modelo, el cual es un proceso iterativo de comparación entre el modelo y el sistema real que finaliza cuando se alcanza un nivel de precisión apropiado.

La comparación entre el modelo y la realidad se lleva adelante mediante varios tests, que pueden ser objetivos o subjetivos. Los subjetivos involucran personas con conocimiento sobre el sistema que emiten juicios de valor sobre el modelo y sus salidas. Los objetivos requieren datos sobre el comportamiento del sistema y sobre los producidos por el modelo, luego con tests estadísticos se comparan los datos.

FASES DE LA VALIDACIÓN DE MODELOS

- Construir un modelo con alta **Face Validity**
- Validar suposiciones del modelo
- Comparar transformaciones de entrada/salida efectuadas por el modelo con las transformaciones de entrada/salida efectuadas por el sistema real.

FACE VALIDITY

El modelo debe resultar razonable para los usuarios del sistema y para las personas con conocimiento sobre el sistema real a ser simulado. Los potenciales usuarios deben estar estrechamente involucrados en la construcción del modelo, desde su conceptualización hasta su implementación operacional.

Esto debería asegurar un modelo con alto grado de realismo construido sobre la base de apropiadas suposiciones de la estructura del sistema.

VALIDACIONES DE SUPOSICIONES DEL MODELO

Las suposiciones pueden ser estructurales o de datos.

Las estructurales involucran cuestiones sobre como opera el sistema, incluyendo simplificaciones y abstracciones.

Las suposiciones de datos se basan en el análisis estadístico de datos confiables obtenidos del sistema.

TRANSFORMACIONES DE ENTRADA-SALIDA

Es el ultimo test del modelo y el unico completamente objetivo. Evalúa la capacidad del modelo para predecir el comportamiento del sistema real. Esto es, cuando los valores de entrada del modelo coinciden con los del sistema real, los valores de salida obtenidos por el modelo deberían reflejar los del sistema real.

En esta fase del proceso de validación el modelo es visto como una caja negra que transforma datos de entrada en datos de salida. Se compara estadísticamente los datos de salida del sistema real versus los valores obtenidos por el modelo.

- **Entrada-Salida utilizando Datos Históricos**

Cuando se puede obtener la salida del sistema real para cualquier combinación de valores de entrada, los datos de entrada para la validación se generan en forma aleatoria. En tanto, para algunos sistemas sólo es posible trabajar con datos de entrada y salida históricos por lo que es necesario asegurar que la comparación entre el sistema real y el modelo sea justa en términos de las condiciones bajo las cuales se obtuvieron los datos históricos.

- **Entrada-Salida utilizando el Test de Turing**

Cuando los test estadísticos no son aplicables, personas con conocimiento sobre el comportamiento el sistema pueden ser empleadas para validar mediante tests de Turing. Se le dan X ejemplares a ellos donde $X/2$ son generados por el modelo operacional y $X/2$ obtenidos por el sistema real, si estos pueden identificar buena parte de los generados por la simulación el modelo debe ser corregido.

Módulo 22: Simulación de Sistemas de Computación

NIVELES DE ABSTRACCIÓN

Los sistemas de cómputo poseen un alto nivel de complejidad dado el comportamiento dinámico que poseen: se requiere trabajar con distintas escalas de tiempo lo que hace más difícil el modelado.

Para lidiar con esta complejidad se los puede analizar de manera jerárquica y composicional, lograndose así distintos niveles de abstracción. El sistema es visto a través de capas haciendo una descomposición en subsistemas cada vez más pequeños:

- **Nivel de Sistema:** Estaciones de trabajo, tareas que circulan por el sistema, colas de servicios pendientes, etc.
- **Nivel de Procesador:** CPU, cache, memoria principal, registros, bus, etc.
- **Nivel de CPU:** Unidad de interfaz de memoria, unidad aritmético-lógica (ALU), etc.
- **Nivel de Compuertas:** Componentes eléctricas, señales débiles, etc.