

Modelando complejidades

Simulación por procesos

Clase nro 14
CURSO 2010

Modelando complejidades

S.E.D. 2010

Introducción

- S.E.D. puede describir una gran variedad de sistemas. La descripción del sistema en cuestión debe ser lo más sencilla posible, se debe mantener pequeño el número de factores. De todos modos se deben incluir las complejidades intrínsecas al sistema que se está modelando.
- Muchos modelos de simulación requieren de distribuciones de llegadas que varían, ya que la tasa de llegadas varía según distintos períodos de tiempo.

S.E.D. 2010

Introducción

- En sistemas de filas humanas se deben incluir algunas conductas de tipo no retirarse o cambiar de fila.
- Las entidades en la mayoría de los sistemas pueden participar en más de una actividad a la vez, por lo que se deben implementar facilidades para tomar, unir, o desagendar entidades en el calendario.

S.E.D. 2010

Distintas tasas de arribo

- La mayoría de los sistemas que describen comportamiento humano, tienen períodos de tiempos activos (o flojos) que varían cíclicamente. Si no se incluyen estas variaciones la simulación puede resultar ilógica o falsa.
- Referente a las tasas de arribo se podría pensar en utilizar
 - la del período más activo,
 - la del período más suave o
 - el promedio de ambas.

S.E.D. 2010

Distintas tasas de arribo

Las dos primeras generan simulaciones sesgadas y la tercera deriva en la estimación de algunos resultados por debajo o por encima de lo debido según el período de tiempo que se esté simulando.

Todo depende de los objetivos planteados, ya que si estamos interesados en resultados correspondientes a períodos muy activos del sistema, entonces es válido que utilicemos la tasa correspondiente al período de interés.

S.E.D. 2010

Distintas tasas de arribo

- Solución inmediata: utilizar parámetros dependientes del tiempo en las distribuciones utilizadas.
- Problema: los eventos se agendan para ocurrir en un futuro, cuando los parámetros ya no son los mismos que al momento de agendarlos. Especialmente inadecuado cuando hay cambios importantes en los valores de los parámetros en un intervalo corto de tiempo.

S.E.D. 2010

Distintas tasas de arribo

- La solución más sencilla a este problema es el método llamado "thinning" (de adelgazamiento, acortamiento).
- Los cambios de la tasas a través del tiempo, se asumen como una función $f(t)$ continua a trozos (fig 11.1), con un valor máximo m .
- En este método los arribos se generan según la tasa máxima m y luego algunos de estos son podados ("pruned").
- La probabilidad de que un arribo sea podado depende del valor de $f(t)$ al tiempo del arribo.

S.E.D. 2010

Distintas tasas de arribo

- Las tasas de arribos se asumen Poissonianas y se muestrean en proceso iterativo en dos pasos:
 - 1) Se muestrea el valor u de una exponencial de media $1/m$, lo que determina el tiempo $t + u$ del próximo arribo, siempre que el último arribo haya ocurrido en el tiempo t .
 - 2) Para decidir si ese arribo se realiza o se poda (pruned out), se sortea un número uniforme entre 0 y 1, si este número es mayor que $f(t+u)/m$ el arribo se rechaza, si no, se acepta.
- Este proceso se repite hasta encontrar un tiempo de arribo que pueda ser aceptado. S.E.D. 2010

Distintas tasas de arribo

- El arribo va a ocurrir en el tiempo oportuno, no ocurre en la hora actual de la simulación.
- El método "thinning" reduce el muestreo en proporción a la diferencia entre m y $f(t)$.
- Existen otros métodos, pero este es eficiente y fácil de entender.

S.E.D. 2010

Distintas tasas de arribo

```
function thintime : real
  var  t, u: real;
  begin
    t:= tim;
    repeat
      u:= negexp(1/m, s);
      t:= t + u;
    until rnd(s) < (fun(t) / m);
    thintime := t - tim;
  end;
```

S.E.D. 2010

Conductas en Filas de espera

- Las distintas conductas de las personas en las filas son poco predecibles: una persona puede decidir irse del sistema sin ponerse en fila, cambiar de una fila a otra más corta o abandonar la fila.
- Si estas conductas se consideran poco frecuentes o si el estudio se interesa más por el uso de recursos que por los problemas propios de filas, (largo de las colas y tiempos de espera), se podrá no incorporar esas conductas en el modelo.

S.E.D. 2010

Conductas en Filas de espera

- El fracaso de un ingreso al sistema de filas (*balking*) es una conducta muy común que se puede implementar p.ej. condicionando al largo de la fila el incluir una entidad a esa fila.
- Decidir cuál es largo de la cola adecuado es otro problema a considerar. Podríamos pensar en realizar mediciones reales, pero esta conducta también es difícil de medir ya que una misma persona puede cambiar de conducta de un día para el otro; de este modo corremos el riesgo de contabilizarlo dos veces, e inflando la cantidad de potenciales arribos al sistema.

S.E.D. 2010

Conductas en Filas de espera

El cambio de fila (*swopping*) ocurre cuando la persona tiene la posibilidad de elegir la fila en la que esperará, una vez que está ubicado en una fila particular.

Esta conducta no puede ser ignorada ya que puede derivar en una subestimación del uso de los recursos o una elevada valoración de los tiempos de espera.

Es importante además si queremos realizar simulación comparativa tomando como factor el número de filas de espera.

S.E.D. 2010

Conductas en Filas de espera

- El swopping se puede implementar en el método de las tres fases, mediante el agregado de un evento condicionado.

Ejemplo: *queue_swop*: acomodará a los clientes según ciertas reglas a definir (si un servidor queda libre, si la diferencia entre largos de filas es 2 o más unidades, etc.)

Este evento condicionado debe ser el último en el ejecutivo para asegurarnos de que se realiza al final de todo cambio del reloj.

- En el método de eventos se deberá escribir un procedimiento (p.ej *queue_swop*) que será llamado desde eventos fijos cuya ejecución resultará en cambio de fila.

S.E.D. 2010

Conductas en Filas de espera

- La probabilidad de abandono de una fila (*reneging*) antes de ser servido, depende del tiempo que se ha esperado y debe estar directamente relacionada con cada entidad en particular.
- La implementación de esta conducta se puede realizar mediante un evento fijo, llamado p.ej. “*renege*”, cuya función es quitar la entidad de la fila, cuando llegue su tiempo o momento.
- Al poner en fila una entidad, la misma se agenda en el calendario con el evento “*renege*”. Si la entidad es servida antes que este evento suceda, entonces la entidad agendada a retirarse se quitará del calendario.

S.E.D. 2010

Conductas en Filas de espera

- La acción de remover una entidad del calendario se llama *fetching* .
- Cuando una actividad debe ser terminada antes de tiempo o interrumpida (cosa muy común), las entidades deben ser des-agendadas o removidas del calendario.
- Esta acción se puede implementar mediante un simple procedimiento de búsqueda en el calendario y retiro de la entidad cuando ella es encontrada (pseudocódigo en libro, pág. 195).

S.E.D. 2010

Conductas en Filas de espera

- Cuando una entidad es quitada del calendario, debe ser puesta en cola a la espera de poder continuar con la actividad, o comenzar una nueva, o liberar recursos.
- Si la actividad se va a continuar realizando, después de una interrupción, entonces se debe anotar y recordar el tiempo que queda por "gastar" en esa actividad; este dato puede ser anotado como atributo de la entidad y llamado cuando la actividad se va a continuar.

S.E.D. 2010

Conductas en Filas de espera

- Por ejemplo en el taller de reparaciones, la hora de almuerzo, como evento fijo, puede ocurrir cuando se está reparando la máquina, actividad que se puede o no interrumpir, o que se puede suplir por una nueva actividad "almorzando".
- La búsqueda en listas pueden enlentecer considerablemente la simulación, sobre todo si son listas largas. Esto se puede mejorar proveyendo la información sobre la ubicación de la entidad en la lista. Esto hace más compleja la estructura de la entidad (ver solución propuesta en libro, pág. 196).

S.E.D. 2010

Entidades de tipo sombra (shadow)

Es común que entidades participen de varias actividades simultáneamente.

Ejemplos:

- La sala de operaciones puede estar participando de la actividad *operando* pero también estar agendada para *abrirse* o *cerrarse*.
- Si las máquinas del taller incluyeran también sus actividades en ocasión de estar trabajando tendríamos 1) *procesando* 2) *romperse* (*trabajando*).

S.E.D. 2010

Entidades de tipo sombra (shadow)

En casos simples se puede controlar la simultaneidad mediante una variable booleana, (la sala estará cerrada, pero se permite continuar con la operación que está en proceso).

En otros casos, p.ej. cuando la máquina se rompe y la actividad “en proceso” debe interrumpirse, se modela mediante dos tipos de entidades separadas pero relacionadas entre sí mediante punteros. Esas entidades las llamamos "padre" y "sombra", deben comunicarse entre sí, ya que lo que ocurre a un padre afecta a la sombra y viceversa.

S.E.D. 2010

Entidades de tipo sombra (shadow)

- Si una máquina tiene una entidad padre ocupada en una actividad (“procesando”) y la entidad sombra está agendada para romperse en un tiempo determinado, entonces cuando ocurre la ruptura, también la entidad padre debe ser removida del calendario.
- En algunas simulaciones complejas pueden existir varias entidades de tipo sombra, deben existir punteros que relacionen las entidades que se ven afectadas por las actividades que se realizan simultáneamente.
- El padre se agenda a la actividad más frecuente.

S.E.D. 2010

Simulación celular

- En el método de las tres fases todos los eventos condicionados se examinan en cada "time beat", lo cual es muy ineficiente ya que muchas veces solamente algunos de ellos deben ser activados.
- En el método de Spinello de Carvalho, se ejecutan solo aquellos relevantes para el "time beat" que se considera. Desde los eventos fijos se abren o cierran los eventos condicionados que afectan su ejecución. El ejecutivo examinará solamente aquellas condiciones de los eventos que estén abiertos (ver pág. 199).

S.E.D. 2010

Simulación celular

<i>Evento fijo</i>	<i>Evento condicionado que abre</i>
arribo_pac_t1	inicio_estadía
arribo pac_t2	inicio_estadía_preoperatoria
final_estadía	inicio_estadía inicio_estadía_preoperatoria
final_estadía_preoperatoria	inicio_estadía
fin_operación	inicio_operación inicio_estadía_posoperatoria
final_estadía_posoperatoria	inicio_estadía inicio_estadía_preoperatoria
abre_sala	inicio_operación
cierra_sala	S.E.D. 2010

Simulación por procesos

S.E.D. 2010

Simulación a Procesos

- La simulación a procesos se clasifica a su vez en dos tipos:
 - Simulación mediante interacción de procesos (SIP).
 - Simulación mediante descripción de procesos (SDP) (o flujo de transacciones).
- SDP proviene del lenguaje GPSS desarrollado por IBM en 1950.
- SIP comienza con el lenguaje SIMULA desarrollado por el centro noruego de Computación a mediados de 1960.

S.E.D. 2010

Simulación a Procesos

Conceptos básicos

- Las acciones realizadas por cada clase de entidades se proyectan como un Proceso. El proceso es la descripción del ciclo de la entidad.

Ejemplo: en el caso del hospital tenemos 3 clases de entidades cuyos procesos se describen en la pág. 201.

S.E.D. 2010

Simulación a Procesos

Conceptos básicos

- En cada etapa de su proceso una entidad se va a encontrar en dos estados:
 - a) demorada (*delayed*) ocupada en alguna actividad (internación),
 - b) bloqueada (*blocked*) en espera para comenzar alguna actividad, ocupar algún recurso o recibir algún mensaje de otra entidad.

S.E.D. 2010

Simulación a Procesos

Conceptos básicos

- Notar que no existe manejo explícito de ninguna cola.
- Pueden existir una o múltiples instancias de un determinado tipo de entidad.
- Cada entidad fluye a través de su proceso hasta que abandona la simulación o la simulación termina.

S.E.D. 2010

Simulación a Procesos

Conceptos básicos

- Debe existir interacción entre procesos, que se puede modelar mediante envíos de mensajes entre entidades.

P.ej, cuando la sala de operaciones se abre, entonces manda un mensaje a las entidades que esperan para operarse.

S.E.D. 2010

Simulación mediante descripción de Procesos (PD)

- La mayoría de los lenguajes orientados a procesos no permiten comunicación directa entre procesos. Esa interacción entonces debe ser descripta mediante datos que se comparten como ser mediante variables globales.
- Ejemplo: El valor de la var. global booleana "*sala_de_operaciones*" puede ser cambiada desde el proceso *theatre*, y puede ser leída desde el proceso *patient2* para determinar si la sala de operaciones está abierta o cerrada.

S.E.D. 2010

Simulación mediante descripción de Procesos (PD)

- Esta forma de interacción entre procesos es propia de la Sim mediante descripción de procesos. Es decir, se describen los procesos pero las entidades de distintos procesos no pueden interactuar explícitamente entre sí.
- El ejecutivo de las simulaciones “a procesos”, en general, es bastante más complejo que el de “a eventos” o tres fases, ya que se encarga del manejo de las distintas colas y sus cambios, así como también de la “alocación” de recursos.

S.E.D. 2010

Simulación mediante descripción de Procesos (PD)

El ejecutivo de PD trabaja con 2 listas ordenadas:

- 1) el calendario de eventos futuros, y
- 2) una cadena de entidades suspendidas, ordenada según tiempo de arribo o prioridades. Las entidades suspendidas están bloqueadas en alguna etapa de su proceso, esperando en alguna cola.

S.E.D. 2010

Simulación mediante descripción de Procesos (PD)

El ejecutivo es similar al de las tres fases:

- A:** Avanza el tiempo
- B:** Ejecutar los eventos fijos del calendario
- C:** Examinar las entidades suspendidas

S.E.D. 2010

Simulación mediante descripción de Procesos (PD)

Cuando una entidad es quitada del calendario, fase B, se mueve secuencialmente a través de las acciones especificadas en la descripción de su proceso hasta que:

- a) abandona la sim,
- b) es demorada e ingresa al calendario nuevamente o
- c) es suspendida nuevamente por alguna razón (pasa a la cadena correspondiente)

S.E.D. 2010

Simulación mediante descripción de Procesos (PD)

En la fase C se recorre la cadena de entidades suspendidas tratando de ejecutar las partes siguientes de las descripciones del proceso correspondiente a la entidad en cuestión. Cuando se encuentra una entidad cuyo proceso puede continuar, continúa hasta que una de las tres condiciones anteriores se cumple.

- Usualmente la cadena de entidades suspendidas posee disciplina FIFO, aunque se puede implementar otro tipo de disciplinas. (pág. 204)

S.E.D. 2010

Simulación mediante descripción de Procesos (PD)

El hecho de tener una sola lista de entidades suspendidas en lugar de distintas colas, una por actividad, es mas fácil de programar pero tiene muchas desventajas. P.ej. es ineficiente cuando la lista es muy larga. Además dificulta la recolección de datos para procesamiento estadístico, así como la programación de la salida visual.

- Lenguaje de Simulación GPSS. (Pág. 205)

S.E.D. 2010

Simulación mediante interacción de procesos (PI o SIP)

- Un lenguaje que permite interacción de procesos, como SIMULA, provee las siguientes facilidades:
 - 1) Facilidad de comunicación entre entidades de distintos procesos, sin necesidad de uso de estructuras de datos intermedios.

En SIMULA un proceso puede *suspender (passivate)* o activar directamente otros procesos.
 - 2) Facilidad de co-rutinas que permite que los atributos de las entidades puedan ser declarados localmente y retenidos entre distintas activaciones de los procesos.

S.E.D. 2010

Simulación mediante interacción de procesos (PI o SIP)

3) Herencia, un proceso puede heredar atributos y rutinas locales de otros procesos . Cualquier tipo de paciente puede heredar de una clase paciente con todos los atributos comunes a todos los pacientes.

- Para la simulación son más importantes las dos primeras, comunicación entre entidades y manejo de co-rutinas.

S.E.D. 2010

Comparación entre tipos de simulación

- El uso de determinado tipo de simulación depende muchas veces de la educación recibida (USA eventos, UK tres fases).
- Muchos de los lenguajes tienen implementados más de una visión del mundo que se modela.

S.E.D. 2010

Comparación entre tipos de simulación

- Descripción de procesos es más adecuado para simulaciones de sistemas de filas simples, ya que pone énfasis en el modelado de las entidades, es fácil de entender y no tiene manejo de colas.
- Interacción de procesos es adecuado para sistemas basados en envíos de mensajes, donde las entidades envían mensajes y actúan según los mensajes recibidos.

S.E.D. 2010

Comparación entre tipos de simulación

El hecho de que la simulación a procesos no tenga fácil acceso a las colas, ya que están incorporadas en la cadena de eventos suspendidos, dificulta p.ej.

- la asignación de recursos a distintos tipos de entidades por prioridades,
- la implementación de salida visual y
- la inclusión de complejidades como ser entidades con sombras.

S.E.D. 2010

Método de las tres fases

- Es el más apropiado para modelar el manejo complejo de recursos.
- Toda decisión es tomada en un solo lugar, en el evento condicionado apropiado. La colas son de fácil acceso y manipulación, permite la implementación fácil de salida visual, ya que la pantalla se puede actualizar en puntos específicos del desarrollo de la simulación (ev. fijo, ev. condicional o golpe de reloj).
- Desventaja: ineficiencia. Los sistemas complejos pueden incluir un número grande de eventos condicionados, que se llaman sin ejecutarse ninguna acción, se puede evitar con simulación celular a costo de mayor programación. (obs. a procesos también puede ser ineficiente cuando la cadena es larga).

S.E.D. 2010

Método a eventos

- El Método a eventos es más eficiente pero menos modular.
- La ineficiencia se resuelve pasándole la tarea pesada de programación al programador.
- Este método debe ser utilizado cuando la eficiencia es una condición indispensable del sistema.

S.E.D. 2010