

Introduzione alla Simulazione

Discrete-event simulation (DES)

Modelli Stocastici e Dinamici

Le variabili di stato cambiano valore solo in corrispondenza di eventi discreti, determinati a loro volta da attività e ritardi.

Tempo in Simulazione

- Tempo
 - **reale** : tempo del sistema da simulare
 - **simulato** : tempo rappresentato nel modello (clock : variabile del modello il cui valore è il tempo simulato corrente)
 - **di esecuzione** : tempo di elaborazione del programma di simulazione
- Nota: il tempo di esecuzione dipende dalle componenti del modello e dalla complessità dei cambiamenti di stato rappresentati, e non dalla scala del tempo simulato.
- Contrazione / espansione temporale
 - Rappresentazione dell'evoluzione temporale (clock)

Avanzamento del tempo

- Meccanismi di avanzamento del tempo in simulazione
 - **per intervalli fissi** (*unit-time*): si incrementa il clock di una quantità fissa Δ e si esamina il sistema per determinare gli eventi che devono aver luogo per i quali si effettuano le necessarie trasformazioni
 - tratta tutti gli eventi con tempo di occorrenza $t_i \in (t, t+\Delta]$
 - Problema: scelta dell'incremento Δ
- Eventi con diversi tempi di occorrenza possono essere trattati come eventi simultanei
- Possibili intervalli vuoti

Avanzamento del tempo

- Meccanismi di avanzamento del tempo in simulazione
 - **per eventi** (*event-driven*): si incrementa il clock fino al tempo di occorrenza del prossimo (primo) evento
 - incrementi irregolari
 - eventi simultanei solo se hanno lo stesso tempo di occorrenza
 - evita tempi di inattività

Schemi di simulazione

Strutture di modeling

Esistono essenzialmente quattro metodologie

Interazione tra **Processi**

Scheduling di **Eventi**

Scansione di **Attività**

“Tre fasi”

Processi

Simulazione per **Interazione tra Processi**

Il flusso di esecuzione di un **processo** in esecuzione emula il flusso di un **oggetto** attraverso il sistema

L' esecuzione procede finchè il flusso non viene bloccato o entra in una nuova attività

Attesa in coda, servizio (ritardo), sink

Quando il flusso di un' entità viene bloccato, il tempo simulato avanza al tempo di inizio previsto dalla prima successiva entità in esecuzione

Eventi

Simulazione per **scheduling di eventi**

Si avanza il tempo simulato al tempo dell' **evento** successivo (di solito il termine o l' inizio di un'attività)

Lista ordinata di eventi e scheduler di eventi

Il termine di un' attività coincide con la nuova allocazione di risorse rilasciate tra le entità in attesa e con lo scheduling di nuove attività causalmente determinate

Eventi

scheduler di eventi

mantiene struttura di lista ordinata per tempo simulato (multi-linked) di eventi futuri (schedulati, cancellati, rinviati, bloccati).

Gestisce l'avanzamento del tempo simulato

event-driven:

`clock` <-(tempo del prossimo evento il lista)

unit-time:

`clock`<- `clock`+ Δ Sono accaduti eventi?

Struttura dati Evento=(time, puntatore al codice della routine di evento)

La routine di evento aggiorna le var. di stato e la lista di eventi (inserisce, cancella, o rinvia eventi)

Introduzione alla Simulazione

routine di **inizializzazione**:

chiamate per prime, definiscono lo stato iniziale del sistema, gli stream pseudo-casuali, ecc.

Routine di **gestione degli eventi**

Report generator:

procedure che al termine della simulazione generano i dati di stima

routine di trace:

procedure per notificare eventi o stime a run-time

gestione dinamica della memoria e garbage collection

Attività

Simulazione per **scansione attività**

“due fasi” simili alla rule-based programming

Condizione ok -> esecuzione azione

Esiste un insieme di moduli in attesa di esecuzione, uno per **attività**

Periodicamente si esegue test sulle condizioni che determinano esecuzione di eventi

Esecuzione di eventi -> aggiorna variabili di stato

Tre fasi

Simulazione: metodologia in “**tre fasi**”

Fase1: avanzamento del tempo simulato

Fase2: rilascio risorse mantenute dalle attività che risultano terminate dopo l' avanzamento

Fase3: esecuzione attività per le quali siano disponibili le risorse

Trace-driven simulation

Trace-driven simulation:

simulazione basata su eventi ordinati per tempo registrati dal sistema reale

pro:

credibile: non basata su ipotesi e distribuzioni di input

semplice validazione

nessuna assunzione sul workload e correlazione

buona analisi di sensitività

minori effetti di “randomizzazione” (varianza minore)

Il confronto tra sistemi è certamente fair

Trace-driven simulation

Tipi di simulazione

Trace-driven simulation:

contro

complessità del modello è maggiore (di solito)

il trace è riduttivo in termini di rappresentazione dei carichi possibili (soprattutto per sistemi diversi)

il trace è lungo e legato alla situazione in cui è registrato

occorrono molti trace diversi per motivare e dimostrare i risultati

comportamenti transienti devono essere registrati nel trace

un trace diverso per ogni workload