



Tehnici de simulare – Curs 1

florentina.suter@fmi.unibuc.ro

florentina.suter@g.unibuc.ro



Evaluare

- Nota obținută din două proiecte 30%
- Nota examen scris cu întrebări bazate pe materia de curs 70%



Bibliografie:

- Ion Văduva (2004) *Modele de simulare*, Editura Universității din București;
- Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol (2005) *Discrete-Event System Simulation*, Pearson Education, Inc., Upper Saddle River, New Jersey.
- Sheldon M. Ross (1997) *Simulation*, Academic Press, San Diego.
- Averill M. Law (2013) *Simulation modelling and analysis* Fifth edition, McGraw Hill Education.
- Christopher Chung (2004) *Simulation Modeling Handbok – A practical Approach*, CRC Press, Boca Raton.



Introducere

- Exemple:
 - Traficul într-o intersecție – timpul de staționare;
 - Bancă – număr de ghișee;
 - Mic magazin: farmacie
 - Un farmacist se gândește să-și deschidă o nouă farmacie care să funcționeze între 9:00 și 17:00. În medie știe că îi sosesc 32 de clienți în acest interval orar. Durata servirii unui client este aleatoare cu o medie de 10 minute și o deviație standard de 4 minute. El plănuiește să nu mai accepte clienți după ora 17:00, dar să-i servească pe cei care sunt deja în farmacie. Întrebări:
 - Care este timpul mediu zilnic petrecut în farmacie?
 - Care este timpul mediu de servire a unui client de la sosirea lui în farmacie?
 - Câți clienți sunt serviți pe oră?
 - Dacă își schimbă politica de servire, câți clienți ar fi pierduți?



Introducere

- Exemple:
 - Mic magazin: farmacie
 - Pentru a raspunde la intrebari trebuie stabilite anumite ipoteze si creat un model
 - Model probabilist:
 - cu ce frecventa vin cei 32 de clienti?
 - Care este distributia timpului de servire?
 - Teoretic acest model se poate rezolva analitic, dar in practică, datorită complexității lui, nu este așa
 - Raspunsurile vor fi aflate prin simulare: este programat mecanismul probabilist folosind "numerele aleatoare" si programul este rulat pentru un număr mare de zile astfel încât, utilizând teoria statistică, se pot da estimari ale raspunsurilor. De exemplu, pentru 1000 de zile simulate exista 150 in care farmacistul inca lucreaza la ora 17:30.



Introducere

SIMULARE: imitare a lumii reale

- Domeniu interdisciplinar ce folosește Matematica, Statistica și Informatica;
- John von Neumann - ENIAC, Stanislaw Ulam;
- Los Alamos, 1944, studierea armelor nucleare: noi domenii de matematici aplicate care necesită utilizarea calculatorului:
 - Cercetări operaționale;
 - Teoria jocurilor;
 - Simulare (Metoda Monte Carlo): imitarea aleatorului.



Introducere

Definiție: Un sistem este o colecție de componente care interacționează pentru obținerea unui rezultat.

Un sistem poate fi afectat de schimbările care au loc în afara sistemului, numit mediul în care funcționează sistemul. În modelarea sistemelor trebuie stabilită limita dintre sistem și mediul său.

Definiție: Simularea este o tehnică de realizare a experimentelor cu calculatorul care implică utilizarea unor modele matematice și logice care descriu comportarea unui sistem real (sau a unor componente ale sale) de-a lungul unor perioade mari de timp.

Simularea

- Completează teoria matematică;
- Completează studiul unor experimente fizice.
- Instrument de analiza si instrument de proiectare.



Introducere

- Aplicații ale simulării:
 - Analizarea comportării utilajelor industriale;
 - Transporturi;
 - Operații desfășurate în aeroporturi;
 - Spitale;
 - Construcții;
 - Ingineria spațială;
 - Rețele de calculatoare.



Introducere

- Scopurile simulării:
 - Se pot obține informații despre ceea ce se petrece în interiorul sistemelor complexe;
 - Se pot dezvolta strategii de îmbunătățire a performanțelor sistemului;
 - Se pot testa noi concepte și sisteme înainte de implementare;
 - Se pot obține informații fără a interveni în evoluția sistemului fizic.



Introducere

- Avantajele simulării
 - Scurtarea timpului de experimentare;
 - Studiul dinamic al sistemelor;
 - Evoluția modelelor este ușor de urmărit (folosirea animației)
- Dezavantajele simulării:
 - Simularea nu poate da rezultate precise dacă datele de intrare nu sunt precise;
 - Simularea nu poate furniza rezultate simple la probleme complexe;
 - Simularea furnizează soluții posibile, nu rezolvă problema.



Introducere

- Tipuri de simulare:
 - **Metoda Monte Carlo:** o simulare statică (fără axă a timpului) care folosește numerele aleatoare pentru a modela fenomene probabiliste care nu își schimbă caracteristicile în timp sau pentru a evalua expresii matematice, al căror rezultat nu poate fi obținut prin metode analitice (integrale, ecuații sisteme de ecuații).
 - **Simularea bazată pe traiectorie:** traiectoria este o înregistrare ordonată în timp, a evenimentelor care au loc într-un sistem real.
 - Este folosită în analizarea funcționării sistemelor de calcul;
 - Analizarea și îmbunătățirea algoritmilor de gestionare a resurselor;
 - Algoritmi de organizare a operațiilor unui procesor;
 - Algoritmi de prevenire a blocajelor



Introducere

- Folosește un model al sistemului bazat pe stări discrete (există și simulare bazată pe evenimente continue în care stările sistemului se modifică continuu în timp - folosită mai ales în chimie, biologie, medicină).

Model de simulare: model matematic + algoritm;

Modelul de simulare: trebuie să descrie corect evoluția sistemului și să permită efectuarea de experiențe (prin rulări ale algoritmului pe calculator), care să înlocuiască experiențele pe sistemul real.



Model matematic

Model matematic: reprezintă realitatea folosind elemente sau noțiuni abstracte.

Elemente constitutive ale unui model matematic:

- Variabile și parametri;
- Relații funcționale;
- Caracteristici operative;
- Tehnica de rezolvare;
- Scopul modelului matematic este de a exprima variabilele și parametrii de ieșire în funcție de variabilele și parametrii de intrare.



Model matematic

- Tehnica de rezolvare:
 - Tehnică matematică care realizează exprimarea elementelor de ieșire în funcție de elementele de intrare:
$$(VE, PE) = f(VI, PI);$$
 - De cele mai multe ori sunt necesare ipoteze simplificatoare;
 - Uneori problema nu poate fi rezolvată prin tehnici de rezolvare.



Model matematic

Clasificări ale modelelor matematice:

- După tipul variabilelor: modele continue sau discrete, modele statice sau dinamice, modele deterministe sau stocastice;
- După structura determinată de părțile modelului: modele cu o componentă sau cu mai multe componente.



Model matematic

Exemplu:

Sistemul de așteptare: parte a lumii reale în care se produc aglomerări.

Componentă:

- Resurse: una sau mai multe stații de servire care servesc
- Entități: clienții care sosesc în sistem și care formează
- Cozi de așteptare

Scop: realizarea unui echilibru între pierderile datorate așteptării clienților și pierderile datorate lenevirii stațiilor de servire.



Model matematic

Teoria matematică a cozilor sau teoria așteptării cu aplicații în:

- Economie;
- Comunicații și transport;
- Rețele de calculatoare.

Un model matematic de așteptare:

- Variabile de intrare cunoscute VI:

AT= timpul între sosiri succesive ale clienților

ST=timpul de servire a unui client

sau

NA=numărul de clienți sosiți în unitatea de timp

NS=numărul de clienți serviți în unitatea de timp;



Model matematic

- Variabile de ieșire necunoscute VE:

WT=timp de așteptare

WL=lungimea cozii

sau

TID=timp de lenevire

NID=numărul de stații care lenevesc



Model matematic

Scopul modelului este realizat astfel:

- cunoscând repartițiile de probabilitate ale AT (NA) și ST (NS) se determină informații despre WT (WL) sau TID (NID) și se stabilesc condițiile pe care trebuie să le îndeplinească ST pentru ca o anumită funcție de cost să fie optimă.
- se studiază procesul stocastic discret $N(t)$ = numărul de clienți din sistemul de așteptare la momentul t - proces de naștere și deces



Algoritmul modelului de simulare

Utilizarea calculatorului -- îmbunătățirea performanțelor modelelor matematice prin aplicarea metodelor numerice și ale simulării.

Pentru un sistem de așteptare:

- Componentele sistemului din punct de vedere al simulării:
 - entitati
 - attribute
 - activitati
 - stari
 - evenimente



Algoritmul modelului de simulare

Exemplu: activitatea intr-o banca: entitati – clientii, atribut – suma depusa in cont, activitate – crearea unui depozit, evenimente – sosirea, plecarea unui client, variabile de stare – numarul de functionari ocupati, numarul de clienti care asteapta.

Structura algoritmică a unui model de simulare (care depinde de timp) are două concepte de baza:

- Ceasul simulării;
- Agenda simulării;



Algoritmul modelului de simulare

Eveniment: modificarea valorilor uneia sau mai multor variabile care se calculează sau se generează prin instrucțiunile modelului.

Ceasul simulării: asigură eșalonarea corectă în timp a evenimentelor create de model și uneori determină terminarea simulării.

Ceas cu

- creștere constantă;
- creștere variabilă.



Algoritmul modelului de simulare

Creșterea variabilă: se face cu valoarea care corespunde apariției primului eveniment următor.

Creșterea constantă: se face cu o cuantă de timp constantă.

Terminarea simulării: se impune condiția ca ceasul să ajungă la un T_{max} sau să se fi prelucrat un anumit număr de evenimente.

Agenda simulării: organizează prelucrarea evenimentelor.

- Agenda evenimentelor curente AEC: evenimentele cu timpul de apariție valoarea curentă a ceasului;
- Agenda evenimentelor viitoare AEV: evenimentele cu timpul de apariție mai mare decât valoarea curentă a ceasului



Algoritmul modelului de simulare

Prelucrarea unui eveniment (in algoritmul simulării): determinarea apariției unui nou eveniment (care se memorează în AEC), modificarea unei stări sau distrugerea unui eveniment (ștergerea) din agendă.

Algoritmul simulării:

1. Se inițializează ceasul cu valoarea 0;
2. Se selectează din agendă evenimentele care fac parte din AEC;
3. Se prelucrează evenimentele din AEC până când aceasta devine vidă. Dacă este îndeplinită condiția de oprire algoritmul se termină, altfel
4. Se crește ceasul simulării și se reia pasul 2.



Metodologia simulării

Etapele realizării unui experiment de simulare:

- Formularea problemei prin precizarea:
 - întrebărilor la care trebuie să răspundă modelul;
 - domeniului lumii reale ce trebuie analizat;
 - formei răspunsului la întrebări (grafice, tabele, rapoarte).
- Colectarea de date preliminare: realizarea unor experimente preliminare (dacă sunt posibile): pe baza observațiilor și a datelor se stabilesc variabilele și parametri de intrare sau de ieșire;
- Prelucrarea (interpretarea) primară a datelor preliminare:
 - Se disting variabilele aleatoare;
 - Se estimează parametri;
 - Se testează ipotezele statistice;



Metodologia simulării

- Formularea unui model matematic preliminar
 - Se precizează relații funcționale și ipoteze de lucru;
 - Se identifică relațiile care nu pot fi exprimate matematic și dificultățile care trebuie înlăturate;
- Evaluarea modelului:
 - Evaluarea complexității modelului (dacă poate răspunde în timp real și complet la întrebări);
 - Revizuirea răspunsurilor din etapele precedente prin simplificări sau completări.
- Construcția modelului de simulare
 - Scrierea unui algoritm detaliat care să cuprindă cazul cel mai general al problemei;



Metodologia simulării

- Se va ține cont de limbajul în care se va programa algoritmul: limbaj specializat pentru simulare sau nu.
 - Folosirea unui limbaj de simulare (GPSS, SIMULA, Arena):
 - Modelele se construiesc rapid;
 - Experiențele se desfășoară repede;
 - Au implementate entități specifice simulării cum ar fi ceasul și agenda simulării;
 - Nu sunt foarte flexibile;
 - Nu se poate controla foarte bine ce se întâmplă în interiorul modelului.
 - Folosirea unui alt tip de limbaj:
 - Rezultate precise și controlabile;
 - Se construiesc mult mai greu.



Metodologia simulării

- Verificarea si validarea modelului.
- Planificarea experiențelor de simulare (experimental design).
Ce fel de experiente sunt facute: lungimea etapei de initializare, lungimea rularilor si numarul de repetari ale fiecărei rulari.
- Prelucrarea și interpretarea experiențelor de simulare prin rularea programului și determinarea valorilor statisticilor construite cu ajutorul valorilor de selecție obținute.
- Realizarea documentatiei si a rapoartelor.
- Implementarea finala



Elemente de implementare

Implementarea agendei de evenimente

- Necesitatea organizării evenimentelor: agenda evenimentelor. Cum este ea implementată?
- Agenda simulării: o listă ordonată înlănțuită a evenimentelor. Fiecare element al listei trebuie să memoreze timpul de apariție a evenimentului și să aibă un pointer către codul care trebuie executat la acel moment de timp.
- Operații frecvente: inserarea unui nou eveniment și găsirea și eliminarea evenimentului care a avut loc.
- Alegerea structurii de date folosite pentru memorarea acestei liste afectează timpul de rulare.



Elemente de implementare

Implementarea agendei de evenimente

- Necesitatea organizării evenimentelor: agenda evenimentelor. Cum este ea implementată?
- Agenda simulării: o listă ordonată înlănțuită a evenimentelor. Fiecare element al listei trebuie să memoreze timpul de apariție a evenimentului și să aibă un pointer către codul care trebuie executat la acel moment de timp.
- Operații frecvente: inserarea unui nou eveniment și găsirea și eliminarea evenimentului care a avut loc.
- Alegerea structurii de date folosite pentru memorarea acestei liste afectează timpul de rulare.



Elemente de implementare

- Liste ordonate dublu înălțuite (GPSS): primul element din listă este următorul cel mai recent eveniment. Inserția se face căutând locul potrivit pentru noul eveniment.
- Liste indexate: mulțimea evenimentelor viitoare este împărțită în mai multe submulțimi. Fiecare submulțime este asociată unui anumit interval de timp de lungime Δt din timpul total de simulare. Un vector de indici, asociază fiecărui indice i lista cu evenimentele programate în intervalul $[(i-1)\Delta t, i\Delta t)$
- Structuri arborescente: arbori binari de sortare cu rădăcina fiind cel mai recent eveniment din listă.



Verificarea și validarea modelului

- Implementare corectă: verificare;
- Ipoteze corecte: validare;

Modelul de simulare: program de dimensiuni mari.

- Verificarea modelului este facilitată de:
 - Proiectarea modulară “top-down”: modelul dezvoltat într-o structură ierarhică în care programul este format dintr-o serie de module care comunică prin interfețe bine stabilite;
 - Includerea de verificări pe parcursul rulării programului și determinarea de rezultate parțiale;
- Verificarea de face prin:
 - Rularea de cazuri simplificate;
 - Rularea programului pentru valori ale parametrilor care diferă foarte puțin (test de continuitate);



Verificarea și validarea modelului

- Rularea programului pentru cazuri extreme (testul valorilor degenerate);
- Verificarea dacă modelul produce rezultate asemănătoare pentru aceleași date de intrare (test de consistență);
- Verificarea independenței de valoarea de plecare a generatorului de numere aleatoare;
- Validarea se face pentru:
 - ipoteze;
 - valori și distribuții ale parametrilor de intrare;
 - valorile de ieșire și concluzii
cu ajutorul
 - intuiției expertului;
 - măsurărilor asupra sistemului real;
 - rezultatelor teoretice.



Terminarea simulării

- Criteriile de oprire trebuie să țină cont de faptul că:
 - o durată prea scurtă implică rezultate imprecise;
 - o durată prea lungă implică irosirea resurselor de calcul.
- Trebuie să ia în considerare și observațiile rezultate:
 - independente și cu o anumită repartiție.



Erori care se pot face în simulare

- **Nivel de detaliere neadecvat:** simularea permite ca sistemul să fie studiat în detaliu, nivelul de detaliu fiind limitat doar de timpul alocat simulării. Detalierea nu determină neapărat calitatea modelului de simulare
 - Probabilitatea de eroare crește;
 - Lipsa de informație precisă despre parametrii de intrare;
 - Necesitatea unui timp prea îndelungat pentru a obține rezultate.
- **Limbaj de programare nepotrivit.**
- **Modele neverificate:** programele care implementează un model de simulare sunt în general mari și atunci probabilitatea de eroare crește.



Erori care se pot face în simulare

- **Modele imprecise:** programul modelului de simulare poate să nu reprezinte în mod corect sistemul simulat din cauza unor ipoteze greșite asupra comportării sistemului. Un model de simulare trebuie confirmat de modele analitice, observații sau intuiție.
- **Prelucrarea incorectă a condițiilor inițiale:** de obicei rezultatele inițiale ale simulării nu sunt relevante pentru evoluția sistemului.
- **Durata prea scurtă a rulării modelelor de simulare:** dacă simularea nu se face un timp suficient de îndelungat, rezultatele pot depinde prea mult de condițiile inițiale și pot să fie irelevante pentru evoluția unui sistem real. Durata corectă a simulării este dată de precizia dorită și de dispersia mărimilor observate.



Erori care se pot face în simulare

- **Generatoare de numere aleatoare neperformante.**
- **Alegerea unor valori nepotrivite** de plecare pentru generatorii de numere aleatoare.