

Universitatea din București
Facultatea de Matematică și Informatică

CURS nr. 9 – TEHNICI DE SIMULARE

Modele de simulare

Lect. dr. Bianca Mogoș

Conținut

1. Funcționalități ale limbajului GPSS
 - 1.1 Operatori aritmetici/relaționali/logici
 - 1.2 Folosirea repartițiilor de probabilitate în GPSS
 - 1.3 Obținerea unor caracteristici ale entităților GPSS
 - 1.4 Predefinirea unor expresii constante sau variabile
 - 1.5 Definirea paramerilor tranzacțiilor și a variabilelor
2. Tipuri de modele care pot fi simulate folosind limbajul GPSS
 - 2.1 Sisteme de așteptare
 - 2.2 Sisteme de modelare a stocurilor

Operatori aritmetici/relaționali/logici

► *Operatori aritmetici:*

- ▶ “+” – adunare, “-” – scădere
- ▶ “#” – înmulțire, “/” – împărțire
- ▶ “\” – împărțire având ca rezultat o valoare întreagă
- ▶ “^” – putere, “@” – restul împărțirii

► *Operatori relaționali:*

- ▶ “G” – $>$, “GE” – \geq
- ▶ “L” – $<$, “LE” – \leq
- ▶ “E” – $=$, “NE” – \neq

► *Operatori logici:*

- ▶ “AND” – “și” logic
- ▶ “OR” – “sau” logic
- ▶ “NOT” – “negație” logică

Obținerea unor caracteristici ale entităților GPSS (1)

- ▶ *SNA – System Numerical Attributes* – reprezintă “variabile de stare” ale sistemului care pot fi interogate în cadrul unui model de simulare

O clasificare a atributelor din SNA:

- ▶ Variabile furnizând *informații despre sistem*:
 - ▶ C1 – timpul de simulare de la ultimul RESET
 - ▶ AC1 – timpul de simulare de la ultimul CLEAR
- ▶ Variabile furnizând *informații despre tranzacții*:
 - ▶ M1 – lungimea intervalului de timp scurs de la crearea unei tranzacții și până în momentul în care se evaluează atributul
 - ▶ PR – prioritatea tranzacției
 - ▶ XID1 – indice asociat tranzacției la generare

Obținerea unor caracteristici ale entităților GPSS (2)

- ▶ Variabile furnizând *informații despre blocuri*:
 - ▶ W\$etichetaBloc – numărul curent de tranzacții aflate în blocul având eticheta “etichetaBloc”
 - ▶ N\$etichetaBloc – numărul total de tranzacții care au intrat în blocul având eticheta “etichetaBloc”
- ▶ Variabile furnizând *informații despre facilitatea cu numele “resursă”*:
 - ▶ F\$resursă – indică starea curentă: 0 – liberă, 1 – ocupată
 - ▶ FC\$resursă – numărul de tranzacții care au ocupat facilitatea
 - ▶ FR\$resursă – procentul de timp (din timpul total de simulare) cât a fost ocupată facilitatea (ia valori între 0 și 1000)
 - ▶ FT\$resursă – timpul mediu de ocupare de către o tranzacție

Obținerea unor caracteristici ale entităților GPSS (3)

- ▶ Variabile furnizând *informații despre punctul de servire cu mai multe stații sau entitatea de depozitare cu numele "resurse"*:
 - ▶ R\$resurse – numărul de stații/unități libere (care pot fi ocupate de tranzacții)
 - ▶ S\$resurse – numărul de stații/unități ocupate
 - ▶ SA\$resurse – numărul mediu de stații/unități ocupate
 - ▶ SR\$resurse – procentul de utilizare
 - ▶ ST\$resurse – timpul mediu de utilizare a entității
- ▶ Variabile furnizând *informații despre coada de așteptare cu numele "coadă"*:
 - ▶ Q\$coadă – numărul curent de tranzacții din coada de așteptare
 - ▶ QA\$coadă – numărul mediu de tranzacții din coadă
 - ▶ QM\$coadă – numărul maxim de tranzacții aflate la un anumit moment al timpului de simulare în coadă
 - ▶ QC\$coadă – numărul total de tranzacții care au intrat în coada de așteptare
 - ▶ QT\$coadă – timpul mediu de așteptare în coadă
 - ▶ QX\$coadă – timpul mediu de așteptare în coadă calculat pentru tranzacțiile care au așteptat un timp strict pozitiv

Folosirea repartițiilor de probabilitate în GPSS (1)

- ▶ Algoritmul de generare a numerelor (pseudo)aleatoare în GPSS se bazează pe *algoritmul multiplicativ – congruențial al lui Lehmer* cu o perioadă maximă de $\lambda = 2^{31} - 2$.
- ▶ *Atributul RNj* returnează o valoare întreagă în intervalul $[0, 999]$.
- ▶ Comanda *RMULT* setează sămânța generatorului pentru generatorii “RNj, $j = 1, \dots, 7$ ”.
- ▶ În absența comenzii “RMULT” sămânța coincide cu numărul j al generatorului “RNj”. De exemplu, RN2 pornește cu sămânța 2.

Folosirea repartițiilor de probabilitate în GPSS (2)

- ▶ *Forma generală a unei funcții de generare* de valori de selecție asupra unei variabile aleatoare având o repartiție de probabilitate dată este (de obicei):

numeRepartiție(nrGen, locație, scală, formă)

unde

- ▶ *nrGen* – reprezintă numărul generatorului de numere aleatoare; trebuie să fie o valoare ≥ 1 .
- ▶ *locație, scală, formă* – sunt parametri ai repartiției de probabilitate

Folosirea repartițiilor de probabilitate în GPSS (3)

- *Exemple* de funcții de generare de valori aleatoare

| <i>Repartiția de probabilitate</i> | <i>Funcția de generare</i> |
|---|---|
| $X \sim U(a, b)$ | UNIFORM(1, a , b) |
| $X \sim N(\mu, \sigma^2)$ | NORMAL(1, μ , σ) |
| $X \sim \text{Exp}(\alpha, \lambda)$ | EXPONENTIAL(1, α , $1/\lambda$) |
| $X \sim \text{Gamma}(\alpha, \lambda, \nu)$ | GAMMA(1, α , $1/\lambda$, ν) |
| $X \sim \text{Bin}(n, p)$ | BINOMIAL(1, n , p) |
| $X \sim \text{Poisson}(\lambda)$ | POISSON(1, λ) |

Folosirea repartițiilor de probabilitate în GPSS (4)

- ▶ *FUNCTION* - este o comandă care definește o entitate funcție. Forma generală este:

nume FUNCTION A,B
 $f_1, a_1 / f_2, a_2 / \dots / f_n, a_n$

- ▶ f_1, f_2, \dots, f_n trebuie să verifice condiția $f_1 < f_2 < \dots < f_n$
 - ▶ nume - este o etichetă obligatorie prin care va fi accesată valoarea returnată de funcție
 - ▶ A - este argumentul funcției (poate fi o expresie) ce va fi evaluat;
 - ▶ B - este un parametru format dintr-o literă care indică tipul funcției (poate fi C - continuă sau D - discretă) și o valoare numerică reprezentând numărul n de perechi (f_i, a_i) , $i = 1, 2, \dots, n$ din definiția funcției.
- ▶ Pentru a obține valoarea returnată de funcție se folosește *atributul* $FN\$nume$.

Folosirea repartițiilor de probabilitate în GPSS (5)

Algoritmul care stă la baza comenzii “FUNCTION”:

- ▶ Se evaluează expresia definită în argumentul funcției, notăm valoarea obținută cu A_0
- ▶ Se determină indicele i minim pentru care $A_0 \leq f_i$.
- ▶ Avem următoarele cazuri:
 - ▶ Dacă $f_n < A_0$ atunci $FN\$nume = a_n$
 - ▶ Dacă $A_0 < f_1$ atunci $FN\$nume = a_1$
 - ▶ Dacă $B = Cn$ și $f_{i-1} < A_0 \leq f_i$ atunci valoarea funcției se obține prin interpolare cu formula:
$$FN\$nume = a_{i-1} + (a_i - a_{i-1}) \# (A_0 - f_{i-1}) / (f_i - f_{i-1})$$
 - ▶ Dacă $B = Dn$ și $f_{i-1} < A_0 \leq f_i$ atunci $FN\$nume = a_i$

Folosirea repartițiilor de probabilitate în GPSS (6)

Folosirea entității funcție de tip continuu pentru simularea unei v.a. continue (nu este recomandată, se recomandă folosirea funcțiilor de generare).

- ▶ Fie v.a. continuă X având funcția de repartiție $F(x)$. În acest caz,
 - ▶ argumentul funcției va fi generatorul de numere aleatoare RN1 (generează valori între 0 și 0.999999 când este folosit de comanda "FUNCTION")
 - ▶ Parametrii din sintaxa comenzii "FUNCTION":
 $a_1 < a_2 < \dots < a_n$ și $f_1 < f_2 < \dots < f_n$ sunt astfel încât

$$f_i = F(a_i) = P(X \leq a_i), \forall i = 1, 2, \dots, n$$

- ▶ *Exemplu:* aproximarea simulării v.a. $X \sim \text{Exp}(1)$

varC FUNCTION RN1,C10

0.05,0.0513/0.15,0.1625/0.25,0.2877/0.35,0.4308/0.45,0.5978/
0.55,0.7985/0.65,1.0498/0.75,1.3863/0.85,1.8971/0.95,2.9957

Folosirea repartițiilor de probabilitate în GPSS (7)

- ▶ *Exemplu* de folosire a entității funcție de tip continuu pentru evaluarea unui argument și asocierea valorii corespunzătoare determinate prin interpolare

varC FUNCTION C1,C7

5,15/10,20/15,10/20,10/25,15/30,20/35,10

- ▶ Se evaluează argumentul funcției, C1 – timpul de simulare; de exemplu, dacă $C1 = 22$ atunci funcția va returna valoarea corespunzătoare lui $A_0 = C1 = 22$ prin interpolarea perechilor (20, 10) și (25, 15), și anume

$$FN\$varC = 10 + (15 - 10)\#(22 - 20)/(25 - 20) = 12$$

Folosirea repartițiilor de probabilitate în GPSS (8)

Folosirea entității funcție de tip discret pentru simularea unei v.a. discrete

- ▶ Fie v.a. discretă X definită prin repartiția

$$X : \begin{pmatrix} a_1 & a_2 & \dots & a_n \\ p_1 & p_2 & \dots & p_n \end{pmatrix}, \sum_{i=1}^n p_i = 1, a_1 < a_2 < \dots < a_n.$$

- ▶ argumentul funcției va fi generatorul de numere aleatoare RN1
- ▶ Parametrii din sintaxa comenzii "FUNCTION":
 - $\{a_1, a_2, \dots, a_n\}$ reprezintă valorile pe care le ia v.a. X
 - $\{f_1, f_2, \dots, f_n\}$ reprezintă frecvențele cumulate, adică

$$f_i = p_1 + p_2 + \dots + p_i, i = \overline{1, n}$$

- ▶ *Exemplu*

varD FUNCTION RN1,D5

0.1,5/0.35,10/0.7,20/0.8,25/1,30

Folosirea repartițiilor de probabilitate în GPSS (9)

- ▶ *Exemplu* de folosire a entității funcție de tip discret pentru evaluarea unui argument și asocierea valorii corespunzătoare

varD FUNCTION C1,D7

5,15/10,20/15,10/20,10/25,15/30,20/35,10

- ▶ Se evaluează argumentul funcției, C1 – timpul de simulare; de exemplu, dacă $C1 = 22$ atunci funcția va returna valoarea

$$FN\$varD = a_5 = 15$$

Predefinirea unor expresii constante sau variabile (1)

Predefinirea unor expresii constante

- ▶ *EQU* - este o comandă care definește o entitate asociată unei expresii având o valoare constantă. Forma generală este:
nume EQU X
 - ▶ X - este o expresie obligatorie având o valoare constantă
 - ▶ nume - este o etichetă obligatorie și reprezintă numele constantei ce va lua valoarea expresiei X.

Exemplu care ilustrează folosirea constantei predefinite prin blocul "EQU"

- ▶ timpServire EQU (150#72 + 36)
GENERATE 5

.....
SEIZE punctServire
ADVANCE timpServire
RELEASE punctServire

.....
TERMINATE 1

Predefinirea unor expresii constante sau variabile (2)

Predefinirea unor expresii variabile

- ▶ *BVARIABLE/VARIABLE/FVARIABLE* - este o comandă care definește o entitate asociată unei expresii a cărei valoare poate să varieze când se reevaluează expresia. Forma generală este:
nume BVARIABLE X ; X este o expresie logică
nume VARIABLE X ; X este o expresie ce ia o valoare întregă
nume FVARIABLE X ; X este o expresie ce ia o valoare reală
 - ▶ nume - este o etichetă obligatorie și reprezintă numele entității ce va lua valoarea expresiei X
 - ▶ X - este o expresie obligatorie având o valoare care poate varia când se reevaluează expresia.
- ▶ *Expresia X se evaluează* folosind
 - ▶ atributul BV\$nume pentru comanda "BVARIABLE"
 - ▶ atributul V\$nume pentru comanda "VARIABLE"
 - ▶ atributul V\$nume pentru comanda "FVARIABLE"
- ▶ BV\$nume(V\$nume) = valoarea expresiei X.

Predefinirea unor expresii constante sau variabile (3)

- *Exemplu – folosirea blocurilor “VARIABLE” și “FVARIABLE”*

| | |
|-----------------------|------------------------|
| timp VARIABLE RN1/100 | timp FVARIABLE RN1/100 |
| GENERATE 5 | GENERATE 5 |
| SEIZE servire | SEIZE servire |
| ADVANCE V\$timp | ADVANCE V\$timp |
| RELEASE servire | RELEASE servire |
| TERMINATE 1 | TERMINATE 1 |

Predefinirea unor expresii constante sau variabile (4)

- *Exemplu – folosirea blocului “BVARIABLE”*

timp BVARIABLE (25<M1'AND'M1<30)

astept VARIABLE RN1@16+20

GENERATE 30

SEIZE serviciu

ADVANCE V\$astept

RELEASE serviciu

SAVEVALUE 1,M1

TEST E BV\$timp,1,termin

TERMINATE 1

termin TERMINATE 1

Definirea paramerilor tranzacțiilor și a variabilelor (1)

- ▶ *ASSIGN* – permite atribuirea/modificarea unei valori/valorii unui parametru asociat tranzacției care accesează blocul. Are forma generală:

ASSIGN param[\pm],valoare

- ▶ param – este identificatorul asociat parametrului/caracteristicii tranzacției; valoarea parametrului poate fi obținută folosind atributul P\$param (dacă “param” este șir de caractere) sau Pparam (dacă “param” este număr)
 - ▶ valoare – este valoarea atribuită sau cu care se incrementează sau decrementează parametrul tranzacției
- ▶ [\pm] – parantezele pătrate nu fac parte din sintaxa blocului; indică opționalitatea parametrilor “+” sau “-”.
 - ▶ Dacă operatorii “+” sau “-” lipsesc atunci P\$param = valoare
 - ▶ Dacă blocul *ASSIGN* are sintaxa: “*ASSIGN* param+,valoare” atunci P\$param = P\$param + valoare
 - ▶ Dacă blocul *ASSIGN* are sintaxa: “*ASSIGN* param-,valoare” atunci P\$param = P\$param - valoare

Definirea paramerilor tranzacțiilor și a variabilelor (2)

- ▶ *SAVEVALUE* – permite atribuirea/modificarea unei valori/valorii unei variabile (globale) atunci când o tranzacție accesează blocul. Are forma generală:

SAVEVALUE var[±],valoare

- ▶ var – este identificatorul asociat variabilei; valoarea variabilei poate fi obținută folosind atributul X\$var (dacă “var” este șir de caractere) sau Xvar (dacă “var” este număr)
 - ▶ valoare – este valoarea atribuită sau cu care se incrementează sau decrementează variabila
- ▶ [±] – parantezele pătrate nu fac parte din sintaxa blocului; indică opționalitatea parametrilor “+” sau “-”.
 - ▶ Dacă operatorii “+” sau “-” lipsesc atunci $X\$var = \text{valoare}$
 - ▶ Dacă blocul *SAVEVALUE* are sintaxa: “*SAVEVALUE* var+,valoare” atunci $X\$var = X\$var + \text{valoare}$
 - ▶ Dacă blocul *SAVEVALUE* are sintaxa: “*SAVEVALUE* var-,valoare” atunci $X\$var = X\$var - \text{valoare}$

Sisteme de aşteptare – Exemplu 1

- ▶ *Modelarea şi simularea rezolvării unor proiecte diferite* (considerate în ordinea apariţiei sau ordonate în funcţie de priorităţi) într-o firmă având mai multe departamente. Câteva modalităţi de abordare a proiectelor având ca scop rezolvarea acestora:
 - ▶ *rezolvarea secvenţială a unui proiect*: proiectul va trece prin fiecare departament în mod secvenţial în sensul că Departamentul 2 depinde de partea realizată de Departamentul 1, Departamentul 3 primeşte proiectul de la Departamentul 2, etc.
 - ▶ *rezolvarea în paralel a părţilor proiectului*; se presupune că pentru a se finaliza proiectul Departamentul 2 şi Departamentul 3 trebuie să discute părţile realizate separat.

Sisteme de aşteptare – Exemplu 2

- ▶ *Modelarea şi simularea dezvoltării unui produs* într-o fabrică având mai multe departamente
 - ▶ *construirea produsului în mod secvenţial*: componenta construită în Departamentul 2 depinde de componenta construită în Departamentul 1, etc.
 - ▶ *construirea produsului în paralel*: fiecare departament construieşte o anumită componentă a produsului, în final “Departamentul asamblare” va realiza asamblarea componentelor şi va construi produsul final.

Sisteme de modelare a stocurilor – Exemplu

- ▶ Presupunem că sucursala unei bănci deține 2 bancomate. Presupunem că suma maximă cu care este alimentat un bancomat este de 10000 de lei. Se realizează următoarele operațiuni:
 - ▶ la un interval repartizat exponential de medie 20 de minute se extrag sume având valori repartizate uniform între 50 și 2000 de lei. Durata unei tranzacții este de 5 ± 2 minute.
 - ▶ La fiecare 2 ore un angajat al băncii verifică dacă bancomatul conține o sumă mai mică de 5000 de lei. În acest caz, soldul bancomatului va fi completat până la capacitatea maximă.
- ▶ Să se simuleze activitatea sistemului în primele 8 ore.

Bibliografie I



T. Schreiber, *An Introduction to Simulation Using GPSS/H*, Disponibil la:

<http://webuser.bus.umich.edu/schreiber/gpssh/gpssh.htm>,
Ultima accesare: decembrie 2015



I. Văduva (2004), *Modele de simulare: note de curs*, Editura Universității din București, București



K. Velten (2009), *Mathematical Modeling and Simulation: Introduction for Scientists and Engineers*, WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim



Generaliști GPSS, site [http://andrei.clubcisco.ro/cursuri/f/f-sym/5master/md/MS_03%20\(GPSSH\).PDF](http://andrei.clubcisco.ro/cursuri/f/f-sym/5master/md/MS_03%20(GPSSH).PDF) Ultima accesare: decembrie 2015