

МОДЕЛИРАНЕ НА ТРАФИКА

В КОМУНИКАЦИОННИ МРЕЖИ С GPSS.

Системата за симулации с общо предназначение (GPSS) на IBM Corp. е транзактно ориентирана симулационна среда, имитираща пространственото движение на обекти (транзакти) при фиксирана функционална структура. Основните обекти в симулационния модел могат да се разделят съответно на

- q динамични (транзакти), които имитират процеси, събития, потоци, и др.;

- q статични (апаратно – ориентирани), които съответстват на устройствата в моделираната система (процесор, опашка, многоканално устройство);

- q управляващи обекти, които представляват съответните блокове за управление на движението на транзактите.

Симулационните програми са реализирани със студентската версия на GPSS – **GPSS World Student Version**

4.2.1

. Поради ограниченията, налагани от студентската версия за обема на симулационните модели (допускат се до 150 оператора) в реализираните модели е използвано сравнително високо ниво на абстракция, което в най-общия случай е архитектурно ниво.

Принципите на изграждане и функциониране на симулационните модели са следните:

- q Процесорите се моделират като устройства (*facilities*). Заемането на устройство, обработката на задача и освобождаването на устройството се моделира с

тройката оператори:

SEIZE

; заемане на устройството

ADVANCE

RELEASE

; освобождаване на устройството

q Работно множество процесори, изпълняващи еднотипни задачи, както и комуникационните канали могат да се представят като многоканално устройство с капацитет, равен на броя на процесорите (комуникационните канали) в работното множество. Заемането на многоканалното устройство, обработката на задачите и освобождаването на многоканалното устройство се моделира с тройката оператори:

ENTER

; заемане на многоканалното *устройство*

ADVANCE

LEAVE

; освобождаване на многоканалното *устройство*

q Комуникационните канали се моделират с многоканални устройства. Времето за комуникация зависи както от дължината на маршрута, така и от обема на предаваните данни;

q За изпълнението на задачите на програмата се генерират транзакти, които се придвижват в системата с цел обработка. Блокът GENERATE създава транзакти, които ще бъдат активизирани в симулационния процес. Той има следния формат:

GENERATE A,B,C,D,E

Където операндите имат следното предназначение:

ï Операнд **A** съдържа средния интервал време за генериране на транзакти (опция).

ï Операнд **B** съдържа функционален модификатор (опция). Не могат да се използват параметри на транзакта.

ï Операнд **C** задава началния момент на генерирането на първия транзакт (опция). Не могат да се използват параметри на транзакта.

ï Операнд **D** задава лимита на броя на генерираните транзакти (опция). Не могат да се използват параметри на транзакта. По премълчаване лимит няма т.е. може да се генерират неограничен брой транзакти.

ÿ Операнд **E** съдържа приоритетното ниво. По премълчаване приоритетът е 0. Не могат да се използват параметри на транзакта.

Новогенерираните транзакти нямат параметри. Новогенериран транзакт фирмира ново асемблирано множество (Assembly Set) с един член.

q За генерирането на паралелни процеси се използва операторът SPLIT. Блокът SPLIT създава транзакции от групата на активния транзакт. Всички новосъздадени транзакти имат същите атрибути (вкл. приоритет) като транзакта – родител. Новите транзакти могат да бъдат изпращани към алтернативни дестинации чрез използването на операнда B. Блокът SPLIT има следния формат:

SPLIT A,B,C

Където:

A – броят на транзактите, които ще бъдат създадени; операндът може да бъде име, цяло число, израз или параметър.

B – номер на блок; Местоназначение за новосъздадения транзакт (опция).

C – номер на параметър; параметърът получава серийния номер на копието (опция). Операндът може да бъде нула, положително цяло число, израз, параметър.

q Синхронизацията на паралелните процеси се моделира с блока ASSEMBLE, който има следния формат:

ASSEMBLE A

Операндът A съдържа броя на транзактите, които подлежат на асемблиране. Например, при влизането на първия активен транзакт от асемблираното множество транзакти в блока

ASSEMBLE 2

той преминава в състояние на очакване.

При влизането на втори активен транзакт от асемблерното множество транзакти в блока, той се унищожават и чакащият транзакт се освобождава и продължава придвижването си в модела.

q Блокът ADOPT се използва да промени асемблерното множество на активния транзакт.

ADOPT A

Операндът A съдържа номера на асемблерното множество. Той може да бъде зададен с име, цяло положително число, израз или параметър. Например, блокът

ADOPT 2000

Активният транзакт става член на асемблерното множество с номер 2000.

q Блокът TRANSFER предизвиква преход на активния транзакт към нова дестинация. Този блок може да се изпълнява в 9 различни режима, всеки с различни

свойства. Когато активният транзакт влезе в блок TRANSFER, режимът на изпълнението му се определя от операнда му А. Интерпретацията на полетата В и С зависи от режимите на изпълнение на блока. Когато не е зададен операнд за блок към който да се извърши прехода, за дестинация се използва блока непосредствено следващ блока TRANSFER.

Блокът TRANSFER има следния формат:

TRANSFER A,B,C,D

A – режим на прехода (опция); Операндът може да бъде BOTH, ALL, PICK, FN, P, SBR, SIM, нула, положително цяло число, израз, параметър.

B – номер на блок или ред, име или номер на параметър в режим P (опция). Операндът може да бъде нула, положително цяло число, израз, параметър.

В случаите, когато операндът А липсва, блокът TRANSFER се изпълнява в *“безусловен режим”* т.е. активната транзакция винаги преминава безусловно към дестинацията, определена от операнда В:

TRANSFER ,New_Place

Когато операндът А не се използва като ключова дума, блокът TRANSFER се изпълнява във вероятностен режим. При него активният транзакт преминава към блока, определен от операнда В с вероятност, зададена в операнд А.

TRANSFER .75,,New_Place

При влизането на активният транзакт в този блок, той се придвижва към дестинацията New_Place с вероятност .75. В останалите случаи за дестинация се използва блока

непосредствено следващ блока TRANSFER.

В режим BOTH (ключовата дума в операнда A е BOTH) първо се тества блока в операнд B. Ако той откаже да приеме активния транзакт, се тества операнда C. В случай, че и той откаже, транзактът чака в блок TRANSFER, докато получи разрешение от една от двете дестинации.

TRANSFER BOTH, First_Place, Second_Place

Когато активният транзакт влиза в този блок, се тества блока First_Place. В случай, че транзактът може да влезе в него, той се препраща. В противен случай се тества блока Second_Place. Ако и при двете дестинации има отказ, транзактът остава в очакване при блока TRANSFER до получаване на разрешение от една от двете дестинации.

В режим ALL (ключовата дума в операнда A е ALL) първо се тества блока в операнд B. Ако той откаже да приеме активния транзакт, се тестват последователно блоковете непосредствено от блока в операнд B до блока, зададен в операнд C. Номерата на блоковете между B и C се инкрементират със стъпката, зададена в операнд D. Ако операнд D не се използва, всички блокове между B и C се тестват последователно. Ако операнд C не се използва, се тества само един блок.

TRANSFER ALL, First_Place, Last_Place, 2

При влизането на активния транзакт в този блок се тества блокът First_Place. Ако транзактът не може да влезе в него, се проверяват последователно всички блокове между First_Place и Last_Place със стъпка 2. Транзактът се препраща към първия свободен блок в зададения диапазон. В случай, че никой от проверяваните блокове не приеме транзакта, той остава да чака в блока TRANSFER.

При *функционален режим* дестинацията на прехода се определя чрез изчисляване на функцията, зададена в операнда B и като се прибави инкремента в операнд C (опция).

TRANSFER FN,Func1,5

При влизането на активния транзакт в този блок се изчислява функцията Func1, като към тази стойност се прибавя 5 за определянето на дестинацията на прехода.

При ключова дума в полето P блокът TRANSFER се изпълнява в *параметричен режим*. При него дестинацията на прехода се определя от сумата на стойността на зададен параметър и съдържанието на операнд C.

TRANSFER P,Placemaker,1

При влизането на активния транзакт в този блок веднага се препраща към блока, непосредствено следващ този, зададен от параметъра на транзакта с име Placemaker.

В режим *“подпрограма”* операнд A съдържа ключовата дума SBR. Активният транзакт винаги извършва преход към блока, определен от операнд B. Блокът на прехода се записва в параметъра, указан в операнд C.

TRANSFER SBR,New_place,Placemaker

При влизането на активния транзакт в този блок веднага се препраща към блока New_place. Адресът на блока

TRANSFER се записва в параметъра с име Placemaker на активния транзакт. За връщане от подпрограмата може да се използва блок TRANSFER в параметричен режим.

q Блокът **TEST** сравнява стойности и на основата на получения резултат управлява дестинацията на активния транзакт:

```
TEST G Q$Teller1,Q$Teller2,Newteller
```

В този случай блокът **TEST** функционира в алтернативен режим. Ако съдържанието на опашката **Q\$Teller1** е по-голямо от съдържанието на опашката **Q\$Teller2**, транзактът продължава към следващия блок в симулационния модел. В противен случай, транзактът се отправя към блок **Newteller**.

q Блокът **GATE** променя потока транзакти в зависимост от състоянието на устройство или многоканално устройство.

```
GATE SNF POOL,OTHERWAY
```

В случая, ако многоканалното устройство **POOL** не е пълно (**STORAGE NOT FULL**) транзактът продължава към следващия блок в симулационния модел. В противен случай, транзактът се отправя към блок **OTHERWAY**

q Унищожаването на процесите в паралелната система се моделира с блока **TERMINATE**, който има формата:

TERMINATE A

При влизането на активния транзакт в този блок се изчислява съдържанието на операнд **A**, което може да бъде нула, име, положително цяло число, израз или параметър. Броячът на унищожените транзакти **TG** се декрементира (намалява) с толкова единици, колкото е съдържанието на операнд **A**. Този брояч се инициализира от командата **START**. Нулирането му прекратява симулацията. Например, при влизането на активния транзакт в блока

TERMINATE 1

транзактът се унищожавя и напуска имитационния модел. Броячът на унищожените транзакти TG се декрементира с 1.

q Всички времезакъснения се представят в относителни моделни единици, в случая машинни цикли.