

МОДЕЛИРАНЕ НА ПРОЦЕСИ В ОПЕРАЦИОННИТЕ СИСТЕМИ**Валентина Кукенска¹, Петър Минеv¹, Матъо Динеv¹**¹*Технически университет - Габрово***PROCESS MODELING IN THE OPERATING SYSTEMS****Valentina Kukenska¹, Petar Minev¹, Matyo Dinev¹**¹*Technical University of Gabrovo***Abstract**

This paper focuses on the processes in the operating systems. It looks at their possible states. The study proposes a probability analytical model based on Markov chains. The model can be used to define the state of a given process in a specific moment of time, and to evaluate the probability of the process entering a certain state.

Keywords: process, modeling, probability analytical model

ВЪВЕДЕНИЕ

Съвременните операционни системи (ОС) са мултизадачни системи. В тях съществуват много процеси. Всеки процес от създаването до завършването си преминава през различни състояния. Преминаването на процесите от едно в друго състояние се променя във времето и се разглежда като случаен процес с дискретни състояния. Този процес може да се представи с граф на състоянията. Всяко състояние се означава с кръгче, а всеки възможен преход от състояние в състояние със стрелка.

Последователност от състояния на процеса образува верига. Тъй като броят на състоянията е ограничен, то веригата е крайна. Преходите от състояние в състояние се извършват в предварително неизвестни моменти от време. Те не зависят от това, по какъв начин и кога процесът е попаднал в предходното състояние.

В този доклад се разглеждат процеси и техните състояния, като ресурси на операционните системи. Предложен е вероятностен аналитичен модел на процес. Той е базиран на марковски вериги [1,2]. Чрез него е определена вероятността за попадение на процесите във възможните състояния в да-

ден момент от време. Предложеният модел е симулиран с програмната среда GPSS.

СЪСТОЯНИЯ НА ПРОЦЕСИТЕ

В многопроцесните ОС един процес може да премине многократно през различни дискретни състояния, като прехода от състояние в състояние се определя от настъпването на определени събития.

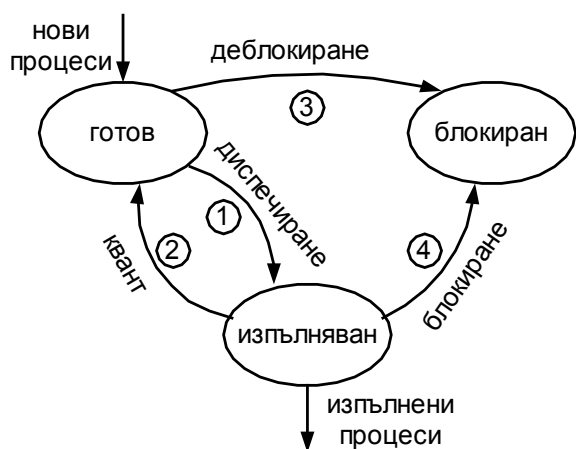
Един процес е изпълняван (*Running*), ако заема процесора. В еднопроцесорна система само един процес може да се намира в състояние изпълняван.

В състояние готов (*Ready*) се намира процес, който може да заеме свободен процесор. Операционната система определя на кой процес от списъка на готовите да предостави освободения процесор.

Процесът се намира в състояние блокиран (*Blocked*) при настъпването на събитие (например В/И операция, процес с по-висок приоритет и др.).

Състоянията на процесите се разделят на две групи – активни и пасивни. Тези състояния и условията за преход от едно в друго състояние могат да се представят с диаграми [4,5,6].

Всеки един процес от момента на неговото създаване до момента на неговото завършване се намира в едно от активните състояния отразени в диаграмата на фиг. 1.



Фиг. 1. Диаграма на състояния на процесите

В еднопроцесорните системи във времето може да се изпълнява само един процес. Останалите са в състояние на готовност или блокиран. Тъй като е възможно няколко процеса да са в състояние на готовност или блокиран, се налага поддържане на списъци на готовите и на блокираните процеси.

Всеки нов процес се включва в списъка на готовите процеси. Той е подреден по определен признак, например приоритет, процесорно време и др., в зависимост от избраната стратегия за диспечирание на процесора.

От опашката на готовите процеси се избира процес за изпълнение. Избраният процес преминава в състояние изпълняван.

Изпълняваният процес заема процесора докато не настъпи едно от следните събития:

- Изтичане на определения му интервал време (квант, quantum) за използване на процесора. Ако това се случи, операционната система прави изпълнявания процес готов и предоставя процесора на друг от списъка на готовите.
- Процесът, преди да е изтекъл кванта, сам освобождава процесора поради необходимост от изчакване на някакво събитие. Тогава изпълнявани-

ят процес преминава в състояние блокиран.

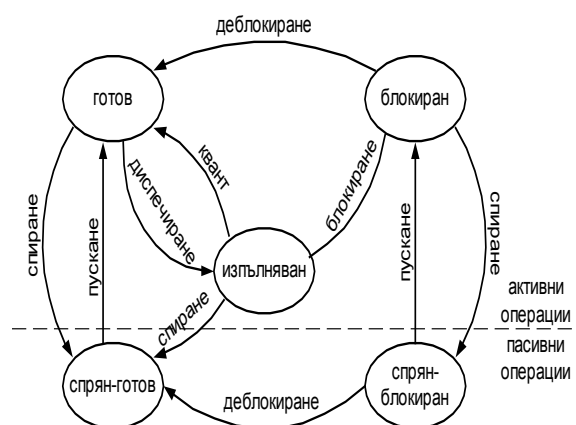
- По време на изпълнение процесът инициализира входно-изходна операция, той освобождава процесора и блокира сам себе си, чакайки завършването на операцията.

Когато завърши входно-изходната операция (или някое друго събитие, очаквано от процеса), процесът отново преминава в състояние готов. Ако по време на изпълнение един изпълняван процес бъде прекъснат, той преминава от състояние "изпълняван" в състояние "блокиран".

Когато очакваното от един блокиран процес събитие настъпи, то процесът преминава в състояние готов.

В някои приложения е желателно да се въведат и други състояния. Например, много операционни системи имат състояние на преустановяване. Процесът преустановява сам себе си или друг процес го преустановява. От такова състояние процесът излиза само, ако друг процес го възобнови.

Ако се добавят пасивните състояния на процесите и операциите за преход по тях се получава разширена диаграма на състоянията (фиг. 2).



Фиг. 2. Разширена диаграма на състояния на процесите

Разширената диаграма на състоянията включва операциите спиране (*suspend*) и пускане (*resume*). В този случай се добавят и две нови състояния - *спрян_готов* и *спрян_блокиран*. Това са пасивни състояния на процесите.


```

; MS/1/
PS1  GENERATE          90
      QUEUE           QSS1
      SEIZE           SS1
      DEPART          QSS1
      ADVANCE         1
      RELEASE         SS1
      TRANSFER        90,,PS1
PS2  QUEUE           QSS2
      SEIZE           SS2
      DEPART          QSS2
      ADVANCE         1
      RELEASE         SS2
      TRANSFER        40,,PS1
      TRANSFER        40,,PS2
      TRANSFER        0,,PS4
PS3  QUEUE           QSS3
      SEIZE           SS3
      DEPART          QSS3
      ADVANCE         1
      RELEASE         SS3
      TRANSFER        90,,PS3
      TRANSFER        ,PS1
PS4  TERMINATE
; MS/1/
      GENERATE          100
      TERMINATE         1

```

При симулация на модела са получени статистически данни за състоянието на процесите за зададен интервал от време.

При изследване на един процес част от получените резултати са:

FACILITY	ENTR.	UTIL.	AVE. TIME	AVAIL.	OWNER
SS1	4	0.040	1.000	1	0
SS2	4	0.030	0.750	1	1
SS3	3	0.030	1.000	1	0

QUEUE	MAX CONT.	ENTRY	ENTRY(0)	AVE.CONT	
QSS1	1	0	4	4	0.000
QSS2	1	0	4	4	0.000
QSS3	1	0	3	3	0.000

За зададения временен интервал изследваният процес е бил четири пъти в състояние S_1 , четири пъти в състояние S_2 и три пъти в състояние S_3 . В края на интервала е в състояние изпълнение (S_2).

При изследване на десет процеса се получават следните данни:

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER
SS1	82	0.810	0.988	1	9
SS2	75	0.740	0.987	1	3
SS3	75	0.740	0.987	1	6

QUEUE	MAX CONT.	ENT.	ENT.(0)	AV.CONT.	AV.TIME	
QSS1	6	6	87	24	0.940	1.080
QSS2	4	3	77	52	0.360	0.468
QSS3	4	0	75	34	0.910	1.213

ЗАКЛЮЧЕНИЕ

Предложен е аналитичен модел на процеси в операционните системи. Чрез него може да се определи състоянието на процесите във всеки момент от време на тяхното съществуване.

Моделът е симулиран с програмната среда GPSS.

Предложеният модел може да се използва за моделиране и изследване на процеси с дискретни състояния.

REFERENCE

- [1] Clymer, J. System Analysis Using Simulation and Markov Models, Prentice-Hall, N. J., 1990.
- [2] Ferreira F., A. Pacheco, Simulation of semi-Markov processes and Markov chains ordered in level crossing, Next Generation Internet Networks, 2005, pp. 121-128.
- [3] Romanski R., Tehnologija na computarnoto modeliranje, Sofia, 2008.
- [4] Tanenbaum A., Modern Operating Systems, International Version, N.Y., 2008.
- [5] Stallings W., Operating Systems: Internals and Design Principles, N.Y., 2008.
- [6] Tanenbaum A., A. Woodhull, Operating Systems: Design and Implementation, 2008.
- [7] Silberschatz A., Operating System Concepts, N.Y., 2009.