

Modelowanie systemów liczących. Ćwiczenie 4

Badanie zamkniętej sieci kolejek

1. Wprowadzenie do systemu WinPEPSY

System WinPEPSY (*Performance Evaluation and Prediction System*) jest ogólnie dostępnym narzędziem na zasadach *shareware*. Powstał on na Uniwersytecie Erlangen. Zawiera on ok. 30 algorytmów dla zamkniętych, otwartych i mieszanych sieci kolejek. Klasy prac zamkniętych (*closed job classes*) są opisywane przez liczbę prac w każdej klasie, podczas gdy klasy prac otwartych są opisywane przez intensywność zgłoszeń prac w każdej z klas. Miary wydajności dla sieci kolejek obejmują takie parametry, jak **przepustowość**, (ang. *throughput*), **wykorzystanie stanowiska obsługi** (ang. *utilization*) oraz **średni czas odpowiedzi** (ang. *response time*) itp. Do ich obliczenia są stosowane zarówno metody algorytmiczne przybliżonego rozwiązywania zamkniętych sieci kolejek, jak i symulacje, podające dokładne wartości. Najbardziej rozbudowaną wersją systemu jest jego implementacja dla systemu okien X11 uruchamiana w systemie operacyjnym UNIX. Wersja dla MS Windows zawiera ograniczoną liczbę algorytmów i metod.

System WinPEPSY umożliwia opis każdego z węzłów (-/M/m - FCFS, -/G/1 - PS, -/G/ - IS (*Infinite Server*), -/G/1 - LCFS oraz także wielu innych typów węzłów. W systemie znajduje się szereg algorytmów, do których należą:

- 1) algorytm konwolucji dla sieci o postaci iloczynowej,
- 2) MVA (*Mean-value algorithm*) dla sieci o postaci iloczynowej,
- 3) algorytmy przybliżonego rozwiązywania dla sieci o postaci iloczynowej,
- 4) metody przybliżonego rozwiązywania dla sieci bez postaci iloczynowej,
- 5) automatyczne generowanie ciągłych łańcuchów Markowa (*Continuous-Time Markov Chain*) i ich rozwiązywanie dla stanu ustalonego
- 6) zastosowanie symulacji dyskretnej.

Dla każdego rozwiązania jest generowany plik wyjściowy z danymi wyjściowymi, który nazywa się **name_xx**. Słowo **name** jest nazwą pliku wyjściowego, a **xx** jest skrótem metody rozwiązania sieci kolejek. Plik wyjściowy zawiera krótki nagłówek podający metodę oraz wyniki obliczeń lub symulacji.

2. Przebieg ćwiczenia

Po skopiowaniu systemu WinPEPSY wraz z *manuałem* w postaci pliku z rozszerzeniem .pdf należy go rozpakować w katalogu **/tmp**. Następnie należy przeprowadzić ćwiczenie wprowadzające do systemu, które jest opisane w podręczniku. Stanowi ono ilustrację możliwości systemu WinPEPSY. Kolejność działań jest następująca:

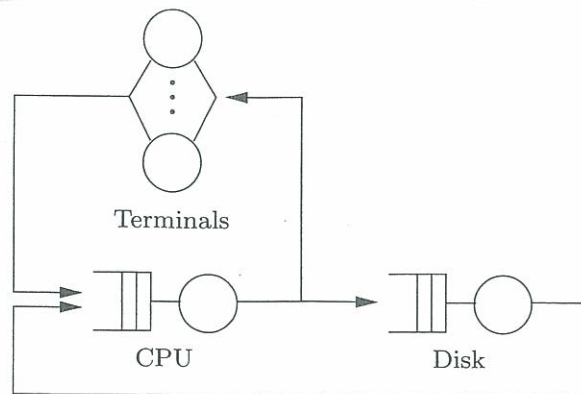
- 1) Z głównego menu wybrać **File** i nazwać nowy eksperyment.
- 2) Z menu do konstrukcji sieci wybrać liczbę klas prac - pole **Options**.
- 3) Z menu do konstrukcji sieci zaznaczyć węzeł, a następnie przenosić węzły, np. **trzy** węzły na planszę.
- 4) Z menu do konstrukcji sieci wybrać klasę prac (zamknięte lub otwarte) do konstrukcji sieci. W ćwiczeniu wprowadzającym należy wybrać klasę zamkniętych prac.
- 5) Z menu do konstrukcji sieci wybrać punkty zbierania (**collecting points**), a następnie umieścić je na planszy. W ćwiczeniu wprowadzającym należy wybrać tylko jeden punkt.
- 6) Z menu do konstrukcji sieci należy wybrać tranzycje (**transitions**), a następnie przenieść je na planszę, aby uzyskać sieć o topologii podanej w podręczniku.
- 7) Z menu konstrukcji sieci wstaw w przypadku zamkniętych sieci kolejek referencję, względem którego węzła jest rozwiązywany model. Uwaga, jeśli to nie będzie zrobione, to system sam wybierze węzeł pierwszy jako referencyjny. Dla otwartych sieci kolejek należy zaznaczyć źródło (**source**) i ujście (**sink**).

- 8) Poprzez dwukrotne kliknięcie w lewy klawisz myszy uzyskuje się podmenu pozwalające na nazwanie każdego węzła (np. **węzeł 1** i wybranie dla niego modelu systemu obsługi. W ćwiczeniu wprowadzającym należy wybrać dla wszystkich węzłów M/M/m/ - FCFS.
- 9) W podmenu opisującym każdy węzeł wybierz **klasę** (*class*) zadania, a następnie intensywności obsługi i prawdopodobieństwa przejść pomiędzy węzłem 1 a 2 oraz 1 i 3. Tutaj jest to 0.5.
- 10) Zakończenie konstrukcji sieci potwierdź klawiszem **Refresh tree**, które po prawej stronie wyświetli drzewo dla poszczególnych węzłów oraz dostępne metody. Uwaga można teraz zmienić niektóre parametry.
- 11) Z głównego menu należy wybrać **Analysis**, a następnie metodę rozwiązania sieci. Zaleca się tu MVA (*mean-value analysis*). Po wyświetleniu menu analizy sieci należy podać parametry analizy oraz czas analizy. Naciśnięcie buttonu **Start analysis** rozpocznie analizę, która zakończy się wyświetleniem wyników w 3D.

Powtórzyć ćwiczenie dwa razy. Pierwszy raz dla tych samych danych, ale zmieniając metodę rozwiązywania sieci. Drugi raz należy do badań wziąć otwartą sieć kolejek. Należy użyć źródła oraz ujście.

3. Badanie modelu systemu liczącego

Dla modelu prostego systemu liczącego, który zawiera procesor, terminale, urządzenia we/wy oraz jest opisywany topologią, jak na rysunku poniżej użyj następujących parametrów: $\rho_{CPU} = 10$, $\rho_{disk} = 3$, $\rho_{terminal} = 0.2$, liczba klasa = 1, liczba prac = 10, $squared_coeff_of_variation = 1$. Rozwiąż model różnymi metodami.



NODE SPECIFICATION

node	name	type
1	cpu	-/G/1-PS
2	disk	-/M/1-FCFS
3	terminal	-/G/0-IS

CLASS SPECIFICATION

class	arrival rate	number of jobs
1	-	10

CLASS SPECIFIC PARAMETERS

CLASS 1

node	service_rate	squared_coeff._of_variation
cpu	10	1
disk	3	1
terminal	0.2	1

ROUTING PROBABILITIES

from/to	outside	cpu	disk	terminal
outside	0.000000	1.000000	0.000000	0.000000
cpu	0.000000	0.000000	0.700000	0.300000
disk	1.000000	0.000000	0.000000	0.000000
terminal	1.000000	0.000000	0.000000	0.000000

W przypadku wybrania metody **marie** uzyskane wyniki mają postać, jak to podano poniżej.

```

PERFORMANCE_INDICES FOR NET: simple_example
description of the network is in file 'e_simple_example'
the closed network was solved using the method 'marie'
jobclass 1
marie | lambda   e   1/mu   rho   mvz   maa   mwz   mwsl
-----|-----
cpu   | 3.922  1.000  0.100  0.392  0.156  0.613  0.056  0.221
disk  | 2.746  0.700  0.333  0.915  1.276  3.503  0.943  2.588
terminal | 1.177  0.300  5.000  0.000  5.000  5.883  0.000  0.000

characteristic indices:
marie | lambda   mvz   maa
-----|-----
      | 3.922  2.549 10.000

legend
e   : average number of visits           mu   : service rate
rho : utilisation                         lambda: mean throughput
mvz : average response time
maa : average number of jobs
mwz : average waiting time
mwsl: average queue-length
    
```