

Tomasz CHMIELEWSKI
Politechnika Śląska, Instytut Informatyki

SYMULACJA PRACY SIECI STANOWISK MASOWEJ OBSŁUGI WYKONYWANA W SYSTEMIE MS WINDOWS

Streszczenie. Wewnętrzne mechanizmy przepływu komunikatów w systemie MS Windows mogą znaleźć zastosowanie przy symulowaniu pracy sieci stanowisk masowej obsługi. W artykule prezentowane jest oprogramowanie wykorzystujące tę koncepcję, pozwalające na przeprowadzanie symulacji działania pewnych klas modeli kolejkowych.

THE SIMULATION OF QUEUEING NETWORKS IN MS WINDOWS SYSTEM

Summary. The internal mechanism of message flow in MS Windows system can be applied to queueing networks simulation. The article presents the software, that can take advantage of this idea, for some queueing network classes simulation.

1. Wprowadzenie

Sieci i systemy komputerowe różnego rodzaju są bardzo powszechnie stosowane w świecie współczesnym. Działają one w przemyśle, gospodarce, bankowości i w wielu innych dziedzinach życia społecznego. Są to systemy informatyczne coraz bardziej skomplikowane i złożone. Jest więc rzeczą oczywistą, że ich działanie zależy od bardzo wielu powiązanych ze sobą czynników. To pociąga za sobą konieczność przewidywania ich działania. Nie można tego dokonywać intuicyjnie ze względu na skomplikowaną budowę systemów. Trzeba więc stosować metody specjalistyczne, pozwalające zbadać zachowanie tych systemów. Przy realizacji tego celu bardzo pomocne okazują się modele matematyczne. Zawierają one wiele uproszczeń i nie wszystkie cechy rzeczywistego systemu znajdują w nich swe

odzwierciedlenie. Mimo to pozwalają wystarczająco dokładnie określić podstawowe parametry systemu i znaleźć jego słabe punkty.

Do najczęściej używanych modeli zaliczamy *modele kolejkowe*, czyli takie, w których system komputerowy jest przedstawiony w postaci sieci stanowisk obsługi. Analizą pracy sieci stanowisk masowej obsługi zajmuje się *teoria masowej obsługi*.

Zastosowanie modelowania z użyciem stanowisk obsługi może służyć do analizy wielu innych zagadnień, również spoza dziedziny systemów informatycznych. Dotyczyć to może na przykład analizy zagadnień transportu lub organizacji produkcji.

Istnieje wiele pakietów oprogramowania, realizujących wymienione funkcje, przeznaczonych do analizy działania nawet bardzo rozbudowanych modeli kolejkowych. Przykładem tego typu oprogramowania może być Analizator Modeli Kolejkowych AMOK[3], opracowany w Instytucie Informatyki Teoretycznej i Stosowanej PAN w Gliwicach.

W tej publikacji przedstawiono program przeznaczony do symulowania pracy pewnych klas modeli sieci stanowisk masowej obsługi. Program pozwala na interakcyjne tworzenie modelu przez użytkownika oraz interakcyjne przeprowadzanie symulacji, a następnie na wizualizację otrzymanych wyników.

Stworzona aplikacja działa w środowisku MS Windows. Zastosowana w niej koncepcja przeprowadzania symulacji została zrealizowana na podstawie przepływu komunikatów od zegara systemowego w tym środowisku.

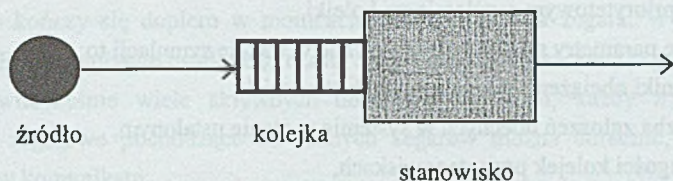
2. Zagadnienia wybrane do implementacji

2.1. Struktura modeli kolejkowych

Model kolejkowy przedstawia system komputerowy w postaci sieci stanowisk obsługi [1,2]. W skład podstawowego systemu rozpatrywanego w teorii masowej obsługi (rys. 1) wchodzi:

- źródło zgłoszeń od użytkowników,
- kolejkę zgłoszeń oczekujących na obsługę,
- stanowisko obsługi.

Źródło zgłoszeń reprezentuje zbiór użytkowników, którzy kierują swoje zgłoszenia (zadania do realizacji) na stanowisko obsługi. Ich zgłoszenia w razie potrzeby szeregowane są w kolejkę. Źródło jest charakteryzowane przez parametr rozkład czasów pomiędzy kolejnymi generowanymi w źródle zgłoszeniami.



Rys. 1. Podstawowy system obsługi

Fig. 1. The basic queueing network

Kolejka spełnia w systemie rolę rejestru zgłoszeń. Czekają w niej na obsługę zgłoszenia, które są do niej kierowane, gdy stanowisko jest zajęte obsługą innego zgłoszenia. W kolejce obowiązuje pewien regulamin. Może on narzucać ograniczenie długości kolejki lub czasu oczekiwania.

Stanowisko obsługi jest złożone z jednego lub kilku identycznych i pracujących równolegle stanowisk, przy czym wszystkie te stanowiska otrzymują zgłoszenia z jednej wspólnej kolejki. Stanowisko jest charakteryzowane przez rozkład czasów obsługi poszczególnych zgłoszeń na stanowisku.

2.2. Przykłady systemów masowej obsługi

Istnieją różne warianty systemów stanowisk masowej obsługi. Najważniejsze cechy charakteryzujące taki system to:

- typ rozkładu odstępów czasu między zgłoszeniami w strumieniu wejściowym,
- typ rozkładu czasów obsługi zgłoszeń na stanowiskach,
- liczba równoległych stacji tworzących jedno stanowisko,
- wymiar źródła zgłoszeń,
- regulamin kolejki.

Ponadto systemy masowej obsługi możemy podzielić na otwarte i zamknięte. W skład systemów otwartych wchodzi źródła zgłoszeń i stanowiska obsługi. Systemy zamknięte natomiast nie zawierają źródeł zgłoszeń, a jedynie stanowiska, między którymi poruszają się znajdujące się w systemie zgłoszenia.

Najczęściej stosowane typy rozkładów odstępów czasu między zgłoszeniami w strumieniu wejściowym oraz rozkładów czasów obsługi to: rozkład wykładniczy, regularny, Erlanga oraz hiperwykładniczy. Przykładami systemów masowej obsługi mogą być:

- system typu $M/M/1$ o wykładniczych rozkładach odstępów czasu między zgłoszeniami w strumieniu wejściowym oraz czasów obsługi,
- system typu $M/M/1/RNT$ o ograniczonym czasie jednorazowego pobytu zgłoszenia w stanowisku obsługi,

— systemy z priorytetowym regulaminem kolejki.

Najważniejsze parametry systemu obserwowane w trakcie symulacji to:

- współczynniki obciążenia poszczególnych stanowisk,
- średnia liczba zgłoszeń obecnych w systemie w stanie ustalonym,
- średnie długości kolejek przy stanowiskach,
- średnie czasy oczekiwania przez zgłoszenia w poszczególnych kolejkach,
- średnie czasy obsługi zgłoszeń na poszczególnych stanowiskach,
- średnie czasy pobytu zgłoszeń w systemie.

3. Symulacja pracy modeli kolejkowych oparta na mechanizmach systemu Windows

3.1. Wykorzystanie mechanizmu obsługi meldunków od zegara systemowego na potrzeby symulacji

Podstawą każdego systemu obiektowo-zorientowanego jest przepływ komunikatów (zwanymi też meldunkami) między obiektami. Obiekt istniejący w takim systemie może być aktywny tylko wtedy, gdy jest związany ze strumieniem komunikatów. W systemie MS Windows komunikaty są generowane oraz otrzymywane z kilku źródeł. Najniższy poziom stanowią komunikaty generowane przez klawiaturę, mysz oraz zegar sprzętowy, które są umieszczane w systemowo-sprzętowej kolejce zdarzeń. Stąd komunikaty są wysyłane do kolejki aplikacji, do której są adresowane.

Mechanizm obsługi komunikatów stanowiący serce systemu MS Windows, jest skomplikowany. Umiejętne posługiwanie się komunikatami wymaga dokładnego zrozumienia sterowanej zdarzeniami architektury obiektowo-zorientowanego systemu MS Windows. Na potrzeby symulacji najistotniejsze jest wykorzystanie faktu, że każda akcja wykonywana przez aplikację jest związana z przesłanym do niej komunikatem pochodzącym z dowolnego źródła, w tym od zegara systemowego.

W systemie MS Windows rozróżnia się sto kilkadziesiąt różnych komunikatów, a niektóre z nich mają dodatkowo liczne warianty. Nazwy wszystkich komunikatów rozpoczynają się od liter WM (ang. windows message), po których występuje słowo opisujące w przybliżeniu, o jaki komunikat chodzi.

Zegar sprzętowy generuje tylko jeden typ komunikatu - WM_TIMER - ale komunikat ten należy do najbardziej przydatnych.

Funkcjonowanie zegara (ang. timer) w systemie Windows polega na regularnym wysyłaniu meldunków WM_TIMER co ustalonego odstępu czasu, nazywanego *okresem zegara*.

Proces ten kończy się dopiero w momencie dezaktywowania zegara. Wysłanie meldunku WM_TIMER, po upływie ustalonego czasu, nazywane jest *zdarzeniem zegarowym* [7]. Może istnieć równocześnie wiele aktywnych obiektów tego typu, każdy z innym okresem. Zdarzenia zegarowe pochodzące od różnych zegarów można odróżnić, badając jeden z parametrów komunikatu.

Okres czasomierza podaje się jako liczbę typu WORD z rozdzielczością 1 milisekundy. Teoretycznie można więc operować przedziałem czasu od 0,001 do 65 sekund. Praktycznie nie da się jednak uzyskać tak dużej rozdzielczości. Fizyczną przyczyną meldunków czasomierza są bowiem przerwania zegara systemowego (INT 8). W sprzęcie PC przerwania INT8 występują z częstotliwością 18,206 Hz, czyli co 0,05492 sek. W rzeczywistości najmniejszym kwantem czasu jaki może odmierzyć czasomierz, jest ok. 0,055 sekundy [6]. Wszystkie większe przedziały będą wielokrotnością tego kwantu. Ustalony dla danego zegara przedział czasowy jest automatycznie zaokrąglany w dół do najbliższej wielokrotności kwantu. Z jednym wyjątkiem: podanie wartości mniejszej od kwantu spowoduje przyjęcie przedziału długości kwantu. Jak widać koncepcja czasomierzy wybiega w przyszłość, przy założeniu optymistycznej rozdzielczości czasomierzy, mimo braku jej sprzętowej realizacji. Jednak w przyszłości platforma sprzętowa PC będzie zapewne unowocześniona i pojawi się szybszy zegar sprzętowy.

W środowisku DOS przerwanie zegarowe ma wysoki priorytet, a dostęp do niego uzyskuje się poprzez przerwanie programowe (INT 1CH). W środowisku MS Windows zdarzenia zegarowe również są obsługiwane asynchronicznie w programie obsługi SYSTEM.DRV, jednak ich priorytet nie jest już tak wysoki. Czasomierz przesyła komunikaty do kolejki komunikatów i na tym jego rola się kończy. Komunikaty od czasomierza są ustawiane na końcu kolejki. Może więc zaistnieć sytuacja, że zanim meldunek dotrze na czoło kolejki i zostanie obsłużony, upłynie czas większy niż założony. Ze względu na ograniczoną długość kolejki w systemie Windows nie dopuszcza się do akumulacji w kolejce meldunków od jednego czasomierza.

Mimo pewnych niedogodności, mechanizm zdarzeń czasowych w MS Windows w połączeniu z obiektowo-zorientowaną strukturą tego systemu może znaleźć swe zastosowanie przy symulacji działania różnego rodzaju modeli systemów, w tym systemów masowej obsługi.

Modele systemów masowej obsługi można bowiem rozumieć jako zbiór trwałych obiektów, czyli źródeł zgłoszeń oraz stanowisk obsługi zgłoszeń. Między tymi obiektami poruszają się nietrwałe obiekty - klienci systemu. Z każdym elementem trwałym systemu można łączyć czasomierz, który wysyłając komunikaty będzie aktywował element modelu. W przypadku źródła, otrzymanie komunikatu od odpowiedniego czasomierza będzie

powodować wygenerowanie kolejnego zgłoszenia i przekazanie go na odpowiednie stanowisko obsługi. W przypadku stanowiska obsługi, czasomierz z nim związany będzie aktywowany w chwili nadejścia zgłoszenia na stanowisko i rozpoczęcia jego obsługi. Po pewnym czasie (czasie obsługi zgłoszenia na stanowisku), czasomierz będzie wysyłał do stanowiska meldunek, co z kolei spowoduje zakończenie obsługi zgłoszenia i opuszczenie przez nie stanowiska.

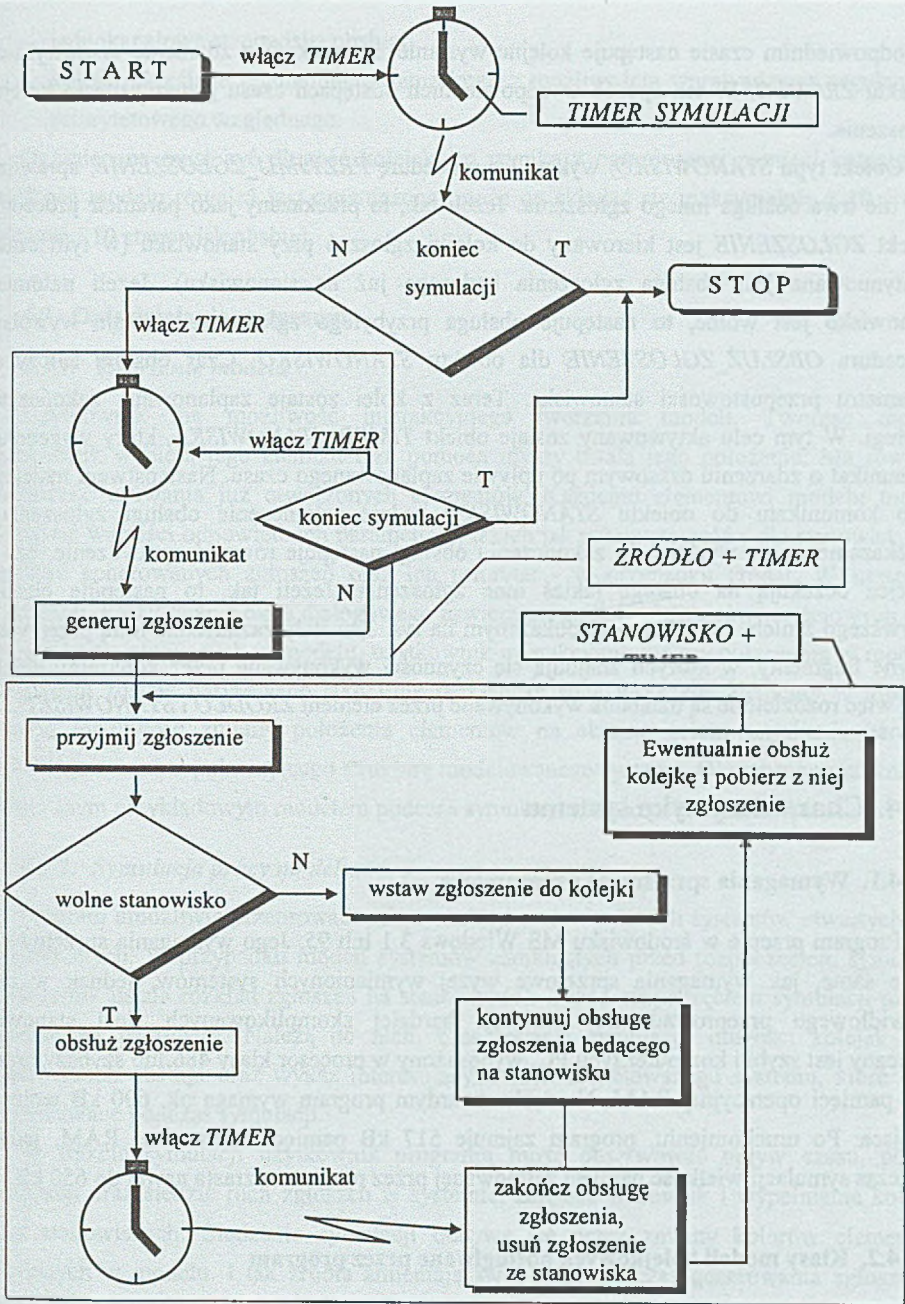
W celu zbudowania modelu należy więc stworzyć dostatecznie dużo obiektów typu *TIMER*. Każdy z tych obiektów będzie związany z jednym trwałym elementem modelu systemu. Oprócz tego można użyć czasomierzy do innych celów, np. do odmierzenia czasu trwania symulacji, czy przeprowadzania co pewien czas pomiarów pewnych parametrów pracy modelowanego systemu.

3.2. Algorytm symulacji

Sposób przeprowadzenia symulacji działania modelu sieci masowej obsługi polega na tworzeniu w programie obiektów odpowiadających trwałym obiektom modelu oraz powiązanych z nimi obiektów typu *TIMER*, a także tworzenie obiektów nietrwałych, jakimi są zgłoszenia. Tak więc liczba stworzonych obiektów typu *STANOWISKO* odpowiada liczbie stanowisk obsługi występujących w modelu. Ta sama zasada dotyczy źródeł zgłoszeń. Dodatkowo, z każdym tego typu obiektem będzie związany obiekt typu *TIMER*. Będzie on, wysyłając komunikaty o zdarzeniach czasowych, sterował pracą elementu modelu.

W celu wyjaśnienia działania algorytmu symulacji zostanie on omówiony na przykładzie najprostszego modelu, składającego się z jednego źródła zgłoszeń oraz jednego stanowiska obsługi (rys. 1). Źródło przesyłać będzie zgłoszenia na stanowisko obsługi, a to z kolei po zakończeniu ich obsługi kierować je będzie na wyjście z systemu. Schemat działania mechanizmu symulacji został pokazany na rys. 2.

Po rozpoczęciu symulacji uaktywniany jest obiekt *TIMER_ŹRÓDŁA*. Wysłanie przez ten obiekt komunikatu do obiektu *ŹRÓDŁO* będzie powodowało wykonanie przez nie procedury *GENERUJ*. Jak łatwo się domyślić, zadaniem tej procedury będzie stworzenie obiektu typu *ZGŁOSZENIE*, nadanie mu odpowiednich cech oraz wysłanie go do przyporządkowanego mu stanowiska. Wysłanie to odbywa się poprzez wywołanie procedury *PRZYJMIJ_ZGŁOSZENIE* dla obiektu *STANOWISKO*. Jako parametr tej procedury jest przekazywany adres obiektu *ZGŁOSZENIE*. Następnie zaplanowane zostaje kolejne wywołanie procedury generującej zgłoszenie. W tym celu zostaje „włączony” *TIMER_ŹRÓDŁA*, którego okres jest zależny od częstości generowania zgłoszeń przez źródło.



Rys. 2. Schemat działania mechanizmu symulacji
 Fig. 2. The schematic diagram of simulation mechanism

Po odpowiednim czasie następuje kolejne wysłanie komunikatu o zdarzeniu czasowym do obiektu *ŹRÓDŁO*. W ten sposób w odpowiednich odstępach czasu generowane są kolejne zgłoszenia.

Obiekt typu *STANOWISKO*, wykonując procedurę *PRZYJMIJ_ZGŁOSZENIE*, sprawdza, czy nie trwa obsługa innego zgłoszenia. Jeżeli tak, to przekazany jako parametr procedury obiekt *ZGŁOSZENIE* jest kierowany do kolejki zgłoszeń przy stanowisku (w tym czasie kontynuowana jest obsługa zgłoszenia będącego już na stanowisku). Jeżeli natomiast stanowisko jest wolne, to następuje obsługa przybyłego zgłoszenia. Zostaje wywołana procedura *OBSŁUŻ_ZGŁOSZENIE* dla obiektu *STANOWISKO*. Czas obsługi zależy od parametru przepustowości stanowiska. Teraz z kolei zostaje zaplanowane zakończenie obsługi. W tym celu aktywowany zostaje obiekt *TIMER_STANOWISKA*, który wygeneruje komunikat o zdarzeniu czasowym po upływie zaplanowanego czasu. Następnym nadejściem tego komunikatu do obiektu *STANOWISKO* będzie zakończenie obsługi zgłoszenia i przekazanie go na wyjście. Po zakończeniu obsługi następuje również sprawdzenie, czy w kolejce oczekują na obsługę jakieś inne zgłoszenia. Jeżeli tak, to następuje obsługa pierwszego z nich. Na schemacie pokazanym na rys. 3 zostały zaznaczone linią przerywaną pewne fragmenty, w których znajdują się czynności wykonywane przez konkretny obiekt. Tak więc rozdzielone są działania wykonywane przez element *ŹRÓDŁO* i *STANOWISKO*.

4. Charakterystyka systemu

4.1. Wymagania sprzętowe i programowe

Program pracuje w środowisku MS Windows 3.1 lub 95. Jego wymagania sprzętowe są takie same, jak wymagania sprzętowe wyżej wymienionych systemów, jednak w celu prawidłowego przeprowadzania symulacji bardziej skomplikowanych sieci stanowisk zalecany jest szybki komputer typu PC, wyposażony w procesor klasy 486 lub szybszy oraz 2 Mb pamięci operacyjnej RAM. Na dysku twardym program wymaga ok. 600 kB wolnego miejsca. Po uruchomieniu, program zajmuje 517 kB pamięci operacyjnej RAM, jednak podczas symulacji wielkość pamięci zajmowanej przez program wzrasta nawet do 650 kB.

4.2. Klasy modeli kolejkowych obsługiwane przez program

Program jest przeznaczony do przeprowadzania symulacji działania pewnej klasy modeli kolejkowych. Klasa ta obejmuje modele systemów o następujących cechach:

- wykładniczy rozkład odstępów między zgłoszeniami w strumieniu wejściowym,
- wykładniczy rozkład czasów obsługi zgłoszeń na stanowiskach,

- jednokanałowe stanowiska obsługi,
- kolejki określane regulaminem naturalnym z możliwością wprowadzenia regulaminu priorytetowego względnego.

Ograniczona musi być długość kolejek, co wynika z ograniczonej pamięci komputera. Wielkość modelu również jest ograniczona, może on składać się maksymalnie z 10 źródeł zgłoszeń i 10 stanowisk obsługi.

4.3. Opis działania programu

4.3.1. Tworzenie modelu

Użytkownik ma możliwość interakcyjnego tworzenia modeli. Tworząc model, użytkownik wybiera jego element i za pomocą myszy ustala jego położenie. Ma również możliwość usuwania już utworzonych elementów. Każdemu elementowi modelu można przypisać wartości odpowiednich parametrów, takich jak przepustowość - dla stanowisk, czy częstość generowanych zgłoszeń oraz ich priorytet - w przypadku źródeł. W następnej kolejności, korzystając z okna dialogowego zawierającego listy elementów wyjściowych oraz wejściowych istniejących w modelu, użytkownik interakcyjnie tworzy połączenia w modelu. Połączenia zostają narysowane jako linie łączące ze sobą wskazane elementy w modelu. Istnieje możliwość zmiany położenia elementów na ekranie celem uzyskania bardziej czytelnego rysunku, pokazującego strukturę modelowanego systemu. Okno programu wraz ze stworzonym przykładowym modelem podczas symulacji zostało pokazane na rys. 3.

4.3.2. Symulacja pracy modelu

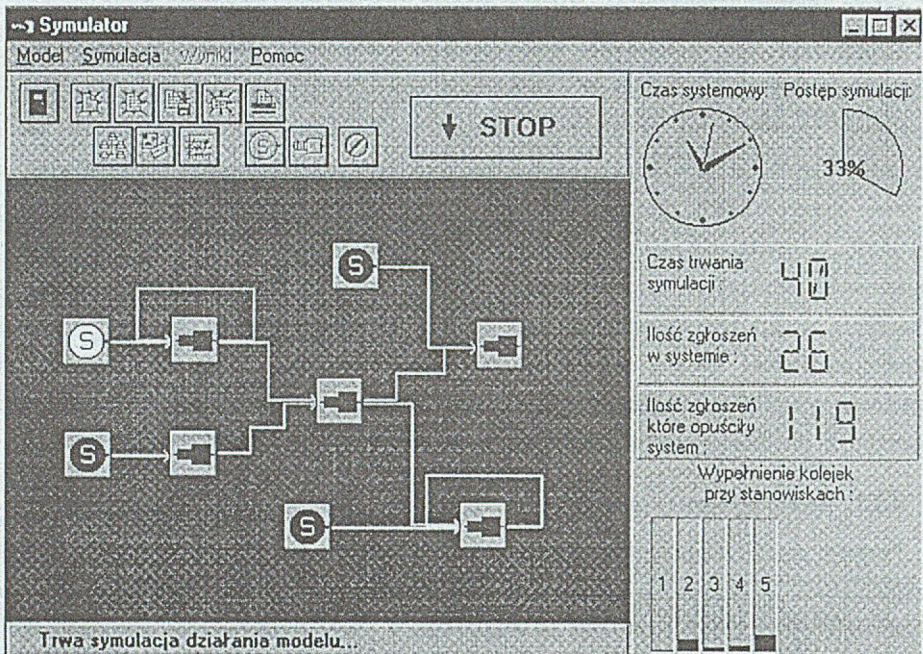
Program umożliwia przeprowadzanie symulacji zarówno modeli systemów otwartych, jak i zamkniętych. W przypadku modeli systemów zamkniętych przed rozpoczęciem symulacji użytkownik ustala rozkład zgłoszeń na stanowiskach. Przed rozpoczęciem symulacji należy określić jej parametry. Należą do nich: czas trwania symulacji, długości kolejek przy stanowiskach obsługi oraz wykaz interesujących cech modelowanego systemu, które będą obserwowane podczas symulacji.

W trakcie symulacji użytkownik programu może obserwować upływ czasu, postęp symulacji oraz śledzić ruch zgłoszeń w systemie, zajętość stanowisk i wypełnienie kolejek przy stanowiskach. Śledzenie symulacji odbywa się przez zmiany kolorów elementów aktywnych w modelu. I tak źródła zmieniają swój kolor podczas generowania zgłoszenia. Kolor przyjmowany przez źródło jest zależny od klasy zgłoszeń, które źródło generuje. Kolor jest różny dla zgłoszeń o różnych priorytetach.

Również stanowiska zmieniają swój kolor w momencie, gdy są zajmowane przez zgłoszenie. Kolor jaki przyjmuje zajęte stanowisko jest zależny od tego, jaki priorytet ma

zgłoszenie, które jest aktualnie na stanowisku obsługiwane. W ten sposób można obserwować poruszanie się zgłoszeń w sieci.

Na bieżąco jest również podawana liczba zgłoszeń aktualnie przebywających w systemie oraz liczba zgłoszeń, które już system opuściły. Użytkownik przeprowadzający symulację może również podczas jej trwania śledzić długości kolejek zgłoszeń przy poszczególnych stanowiskach. W połączeniu z możliwością zmiany parametrów elementów modelu podczas symulacji pozwala to na regulowanie przepustowości elementów modelowanego systemu, celem np. uzyskania jego stabilności.



Rys. 3. Okno programu ze stworzonym modelem, w trakcie symulacji
 Fig. 3. The window with model during the process of simulation

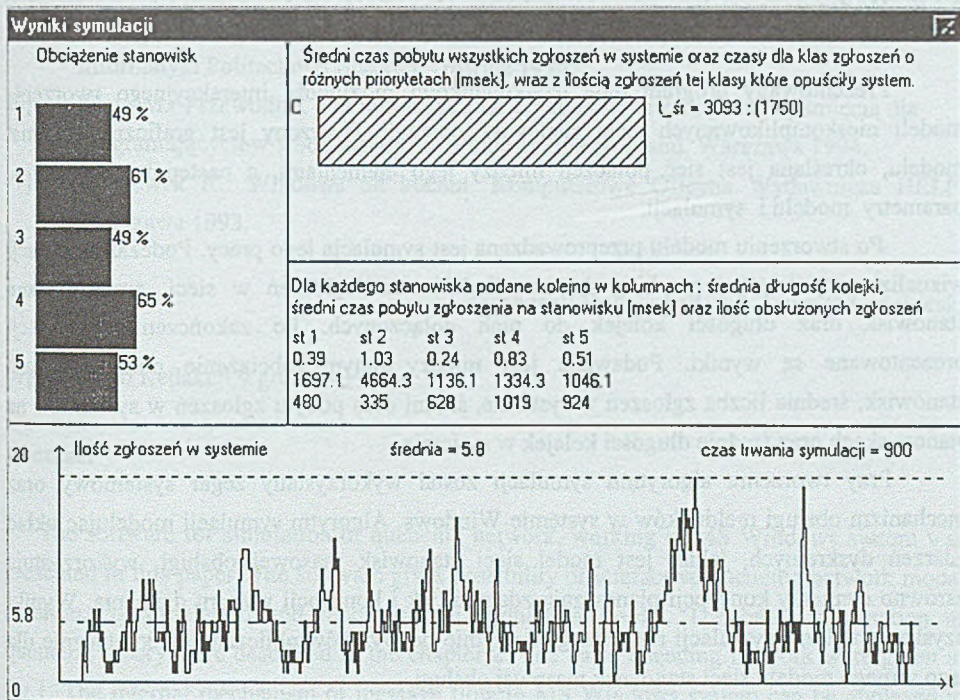
Symulacja zostaje zakończona po upływie czasu, który został na nią przeznaczony. Użytkownik ma również możliwość wcześniejszego zakończenia symulacji poprzez jej przerwanie. Może również nakazać zakończenie symulacji w momencie przepełnienia kolejki przy jednym ze stanowisk obsługi.

4.3.3. Prezentacja wyników symulacji

Po zakończeniu symulacji prezentowane są jej wyniki. Na podstawie zbiorów danych zebranych w trakcie symulacji zostają obliczone pewne wskazane przez użytkownika

parametry badanego systemu oraz zostają narysowane wykresy statystyczne. Okno prezentacji przykładowych wyników zostało pokazane na rys. 4. Prezentowane są następujące wyniki:

- wyrażone w procentach obciążenia stanowisk w systemie, wyliczone jako stosunek czasu przeznaczzonego na obsługę zgłoszeń na stanowiskach, do całkowitego czasu trwania symulacji; różnice obciążeń poszczególnych stanowisk są ilustrowane wykresem słupkowym;
- na wykresie pokazana jest liczba zgłoszeń aktualnie przebywających w systemie w funkcji czasu; obliczona zostaje oraz wyszczególniona na wykresie wartość minimalna, maksymalna i średnia, liczby zgłoszeń przebywających w systemie;



Rys. 4. Okno z przykładowymi wynikami symulacji
 Fig. 4. The window with an example of simulation results

- podane zostają wartości średnie czasów przebywania zgłoszeń w systemie, osobno dla zgłoszeń o odmiennych priorytetach oraz dla całego zbioru zgłoszeń; dodatkowo przedstawiony zostaje wykres ilustrujący różnice między średnimi czasami pobytu w systemie, dla zgłoszeń o różnych priorytetach,
- średnie długości kolejek zgłoszeń przy poszczególnych stanowiskach,

- średnie czasy pobytu zgłoszeń na stanowiskach obsługi,
- liczby zgłoszeń obsłużonych na poszczególnych stanowiskach oraz liczba zgłoszeń każdego priorytetu, które opuściły system.

Wyniki symulacji przeprowadzanych przy użyciu przedstawianego oprogramowania są bardzo zbliżone do wartości wyznaczonych teoretycznie. Różnice wynikają z tego, że losowy charakter mają zarówno odstepy czasu między zgłoszeniami nadchodzącymi ze źródła, jak i czasy trwania obsługi poszczególnych zgłoszeń. Wyniki przykładowych symulacji, przeprowadzonych dla modeli kolejkowych różnego typu, zostały przedstawione w pracy [5].

5. Podsumowanie

Prezentowany program daje użytkownikowi możliwość interakcyjnego tworzenia modeli nieskomplikowanych sieci stanowisk obsługi. Tworzony jest graficzny schemat modelu, określana jest sieć połączeń między jego elementami, a następnie ustalane są parametry modelu i symulacji.

Po stworzeniu modelu przeprowadzana jest symulacja jego pracy. Podczas symulacji wizualizowany jest jej przebieg. Można śledzić ruch zgłoszeń w sieci, zmiany stanu stanowisk oraz długości kolejek do nich dołączonych. Po zakończeniu symulacji prezentowane są wyniki. Podawane jest między innymi obciążenie poszczególnych stanowisk, średnia liczba zgłoszeń w systemie, średni czas pobytu zgłoszeń w systemie i na stanowiskach oraz średnie długości kolejek w systemie.

Przy tworzeniu algorytmu symulacji został wykorzystany zegar systemowy oraz mechanizm obsługi meldunków w systemie Windows. Algorytm symulacji modelując układ zdarzeń dyskretnych, jakim jest model sieci stanowisk masowej obsługi, wykorzystuje zarówno elementy koncepcji planowania zdarzeń, jak i koncepcji wyboru działania. Wyniki uzyskane podczas symulacji nie odbiegają zbytnio od wyników obliczanych teoretycznie dla tego samego modelu sieci stanowisk masowej obsługi.

Dzięki stworzeniu biblioteki obiektów używanych w programie, istnieje możliwość jej wykorzystania przez innych programistów w aplikacjach pisanych przy użyciu pakietu Borland Delphi. W przyszłości będzie więc można tworzyć kolejne wersje tego typu oprogramowania. Aktualna wersja oprogramowania jest dostępna w sieci Internet, na stronie WWW pod adresem http://sun_zo.iinf.polsl.gliwice.pl/~tchmiel.

LITERATURA

- [1] Węgrzyn S.: Podstawy informatyki. PWN, Warszawa 1982.
- [2] Czachórski T.: Modele kolejkowe systemów komputerowych. Skrypt Politechniki Śląskiej nr 1844, Gliwice 1994.
- [3] Praca zbiorowa :Modelowanie i ocena pracy systemów komputerowych za pomocą AMOK-u. Skrypt Politechniki Śląskiej nr 1626, Gliwice 1993.
- [4] M.Skowronek : Modelowanie cyfrowe układów dynamicznych. Skrypt Politechniki Śląskiej nr 1776, Gliwice 1993.
- [5] Chmielewski T.: Napisać program pracujący w środowisku MS Windows, symulujący działanie sieci stanowisk masowej obsługi. Praca dyplomowa magisterska. Instytut Informatyki Politechniki Śląskiej, Gliwice 1996.
- [6] Klain M.: Przewodnik po bibliotekach DLL i sposobach zarządzania pamięcią dla programujących w środowisku MS Windows. Intersoftland, Warszawa 1994.
- [7] Wacławek R.: Windows od kuchni. Komputerowa Oficyna Wydawnicza HELP, Warszawa 1993.

Recenzent: Doc. dr hab. inż. Tadeusz Czachórski

Wpłynęło do Redakcji 9 grudnia 1996 r.

Abstract

The software for simulation of queueing network, working in MS Windows system was presented in this paper. The software gives possibility of interactive queueing network model creation, its simulation and report of the simulation results. The basic information of queueing theory were described in the chapter 2. The basic queueing network were given in fig.1. The internal mechanism of message flow in MS Windows system can be applied to queueing network simulation. The idea of simulation mechanism and simulation algorithm were described in the chapter 3. The base of this idea is a service of messages from the system timer. The schematic diagram of simulation mechanism were given in fig.2. The software characteristic were described in the chapter 4. The examples of user interface were given in fig.3 and fig.4. The software is available in Internet at the following location: http://sun_zo.iinf.polsl.gliwice.pl/~tchmiel.