STUDIA INFORMATICA 2011
Volume 32 Number 3A (98)

Tadeusz CZACHÓRSKI, Krzysztof GROCHLA
Instytut Informatyki Teoretycznej i Stosowanej Polskiej Akademii Nauk
Adam JÓZEFIOK, Tomasz NYCZ
Politechnika Śląska, Instytut Informatyki

# PORÓWNANIE METOD ANALIZY EFEKTYWNOŚCI NA PRZYKŁADZIE SERWERA APLIKACJI W SIECI LOKALNEJ

**Streszczenie**. W artykule przedstawiono model kolejkowy służący do oceny funkcjonowania dużego systemu sieciowego, w skład którego wchodzi rozbudowany system bazodanowy. Opisywany system jest systemem rzeczywistym. Zebrane wyniki pracy systemu posłużyły do budowy modelu działania aplikacji interakcyjnych, na podstawie łańcuchów Markowa, aproksymacji dyfuzyjnej i symulacji zdarzeń dyskretnych. Porównanie ich z rzeczywistymi wynikami umożliwiło sprawdzenie przydatności użytych metod w rzeczywistych warunkach.

**Słowa kluczowe**: modele kolejkowe, aproksymacja dyfuzyjna, ocena efektywności

# PERFORMANCE EVALUATION OF A MULTIUSER INTERACTIVE NETWORKING SYSTEM – A COMPARISON OF MODELLING METHODS

**Summary**. The article presents a queueing model for performance evaluation of a large database system at an assurance company. Measurements were collected inside the working system to construct a synthetic model of applications activities. We apply simulation, Markov and diffusion models – their comparison, based on real data, may better verify the utility of particular methods than usual academic examples.

**Keywords**: queueing theory, diffusion approximation, performance evaluation

## 1. Wprowadzenie

Poprawne i szybkie działanie dużych systemów informatycznych, obsługujących każdego dnia tysiące użytkowników, ma kluczowe znaczenie w wielu rodzajach działalności firmy. Ograniczenie prędkości pracy sieci komputerowej oraz mocy obliczeniowej serwerów prze-

twarzających dane znacząco wpływa na czas reakcji aplikacji podczas komunikacji np. z serwerem baz danych. Ocena pracy takiego systemu powinna w pierwszej kolejności zlokalizować jego „wąskie gardło" i podać wariantowe propozycje jego rozbudowy.

W niniejszym artykule przedstawiono porównanie wybranych metod analizy wydajności dużego systemu informatycznego, na przykładzie systemu pracującego w największej polskiej firmie ubezpieczeniowej. System przeanalizowano za pomocą pomiarów, modelu symulacyjnego i dwu modeli analitycznych opartych na łańcuchach Markowa i aproksymacji dyfuzyjnej. Celem analizy jest określenie czasu potrzebnego do obsłużenia klienta w czasie trwania jednej sesji pomiędzy serwerem a klientem oraz określenie, jak czas ten zmieni się w przypadku zwiększenia liczby klientów. Badany czas jest złożony z wielu operacji, przeprowadzanych na serwerze bazodanowym. System zawiera wiele różnego rodzaju serwerów bazodanowych podłączonych do lokalnej sieci komputerowej, w której pracują stacje robocze z zainstalowanymi, interakcyjnymi aplikacjami. Aplikacje umożliwiają pracownikom pobieranie danych z bazy danych lub wysyłanie zmienionych dokumentów. W systemie wszyscy użytkownicy mogą pracować równolegle i niezależnie od siebie. Każda z aplikacji ma własną charakterystykę i formę prezentowania otrzymywanych danych z bazy danych. Każda aplikacja zawiera również inne mechanizmy przetwarzania otrzymanych danych.

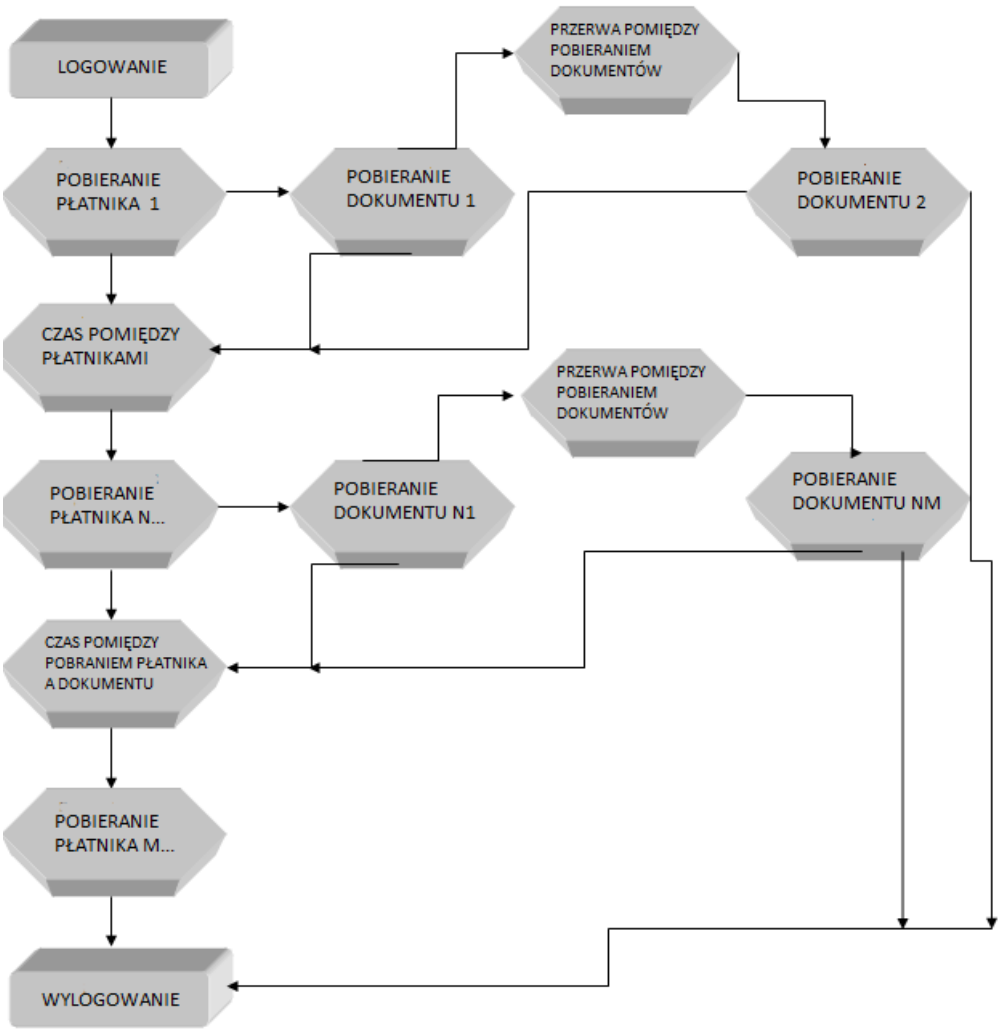
Analizę rozpoczęto od pomiarów działania rzeczywistego systemu, podczas jego normalnej pracy. Zebrane dane pozwoliły na skonstruowanie syntetycznego modelu aktywności każdej aplikacji, który posłużył do późniejszego zbadania zachowania systemu w przypadku wzrostu liczby użytkowników.

Cały system każdego miesiąca wykonuje ponad 40 miliardów obliczeń, w tym ponad 100 milionów operacji księgowych w specjalnie do tego stworzonych hurtowniach danych. Badano działanie jednego z oddziałów systemu. W badanej sieci działa 30 przełączników Cisco, połączonych ze sobą światłowodem. Ponadto, w sieci lokalnej działa około 10 serwerów, przeznaczonych do różnych celów, związanych z utrzymaniem środowiska produkcyjnego. Serwery odpowiedzialne są za przetwarzanie danych związanych z działaniem aplikacji interakcyjnych. Każdy serwer może odpowiadać za prace jednej lub wielu aplikacji. Komputery klientów przesyłają zapytania do serwerów poprzez przełączniki rozmieszczone na każdym piętrze firmy, natomiast te podłączone są do przełącznika szkieletowego.

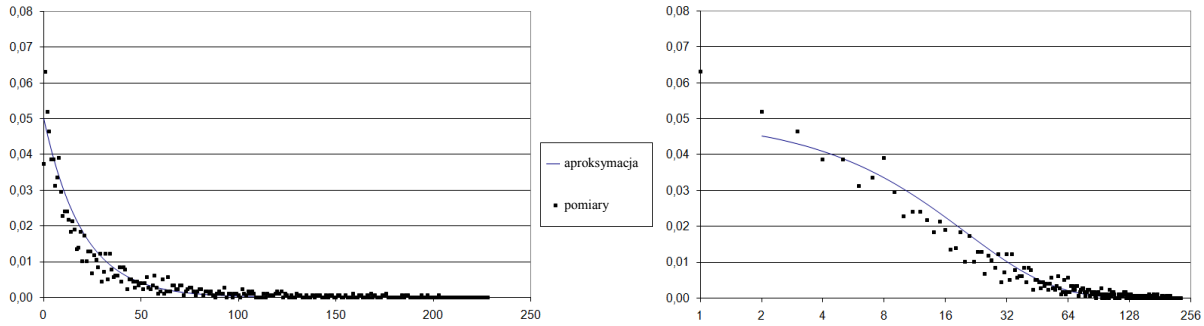
## 2. Działanie aplikacji interakcyjnych i dane pomiarowe

Typowe działanie aplikacji interakcyjnej pokazane jest na rys. 1. W pierwszej fazie użytkownik loguje się do aplikacji i wybiera odpowiedniego płatnika, w kontekście którego chce pracować. Aplikacja łączy się z bazą danych za pośrednictwem protokołu TCP i odpowiednio zdefiniowanego portu. Następuje wyszukanie danych w bazie danych i prze-

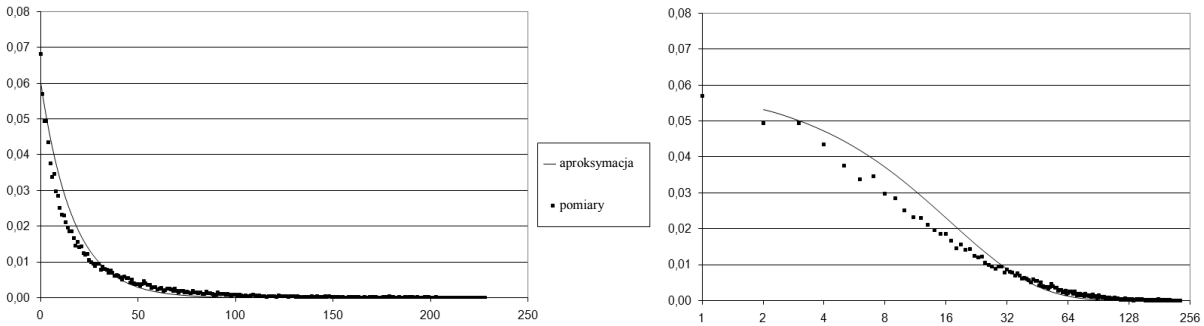
słanie ich do klienta. Użytkownik, po rozpoczęciu analizy płatnika, może zakończyć z nim pracę lub pobrać dodatkowe dokumenty. Liczba dokumentów do pobrania zostaje określona przez użytkownika w toku postępowania. Ta czynność może powtórzyć się kilka kilkakrotnie w ramach jednego płatnika. Następnie użytkownik może przejść do pobierania innego płatnika. W ten sposób określono cztery podstawowe czasy: pobrania płatnika, pobrania dokumentu, czas pomiędzy pobraniami dokumentu oraz pomiędzy zakończeniem jednego płatnika i rozpoczęciem pracy nad innym. Czasy, jak również liczba przetwarzanych dokumentów są losowe, a ich rozkłady zależą od rodzaju aplikacji. Rozróżniono pięć typów aplikacji, oznaczanych poniżej A1, ..., A5. Całkowite obciążenie systemu zależy od wszystkich wykorzystywanych aplikacji oraz ich udziału w poszczególnych etapach pracy. Dla wszystkich 5 aplikacji zmierzono rozkłady 4 zidentyfikowanych powyżej czasów. Przykładowe histogramy pomiarów przedstawiono na rys. 2 – 5.



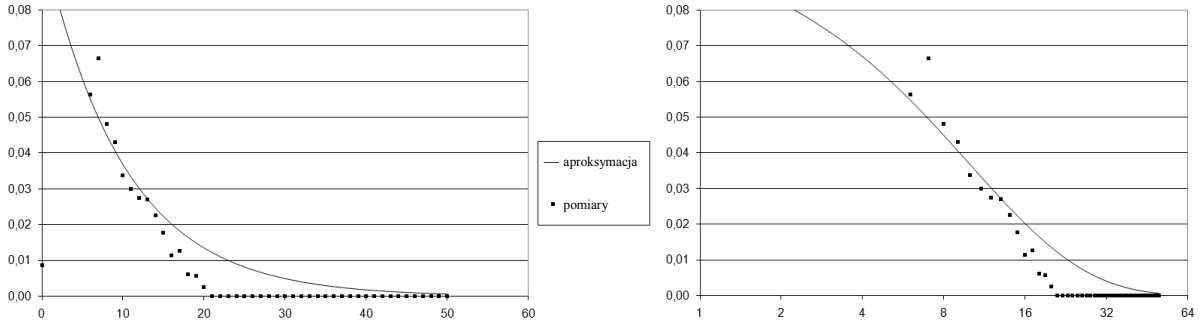

Rys. 1. Schemat działania aplikacji interakcyjnej
Fig. 1. Diagram of an application activities



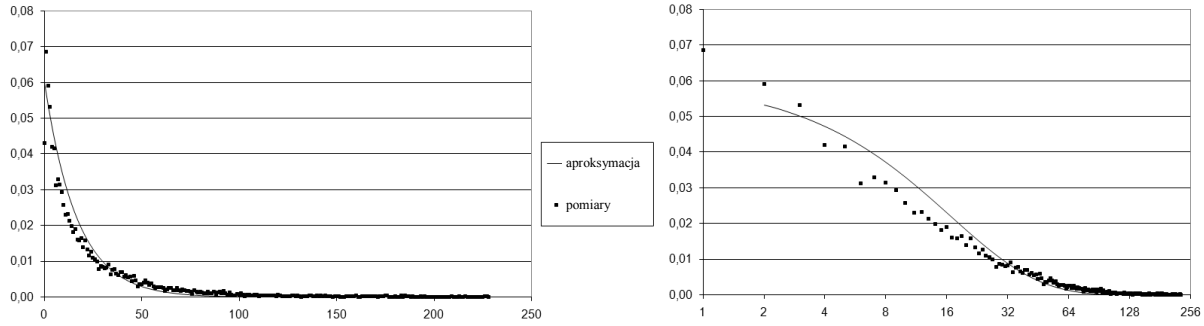

Rys. 2. Czas pobierania jednego płatnika – skale liniowa i logarytmiczna
Fig. 2. Distribution of the download time of a single payer – linear and logarithmic scale




Rys. 3. Czas pobierania dokumentów – skale liniowa i logarytmiczna
Fig. 3. Distribution of the download time of documents – linear and logarithmic scale



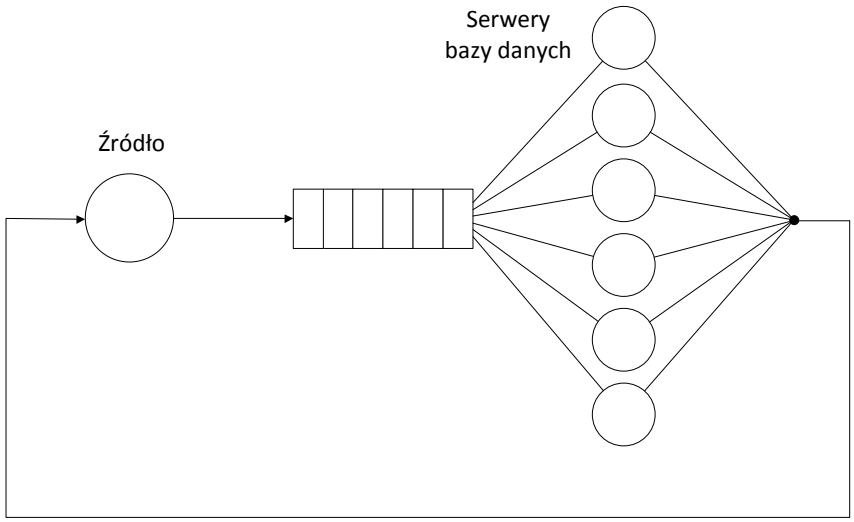

Rys. 4. Liczba dokumentów dla jednego płatnika – skale liniowa i logarytmiczna
Fig. 4. Distribution of the number of document downloads for a single payer – linear and logarithmic scale




Rys. 5. Przerwa pomiędzy dokumentami – skale liniowa i logarytmiczna
Fig. 5. Distribution of the gap length between downloaded documents – linear and logarithmic scale

# 3. Model systemu

Pomiary wskazały, że czas transmisji w sieci jest pomijalnie mały w porównaniu do czasów wyszukiwania dokumentów w bazie danych. Dlatego w modelu systemu uwzględniono jedynie czasy wyszukiwania i odczytu danych z bazy. W badanej konfiguracji występowało 6 serwerów, obsługujących po jednym żądaniu. System może być więc przedstawiony za pomocą modelu kolejkowego, zawierającego 6 równoległych kanałów obsługi (rys. 6).



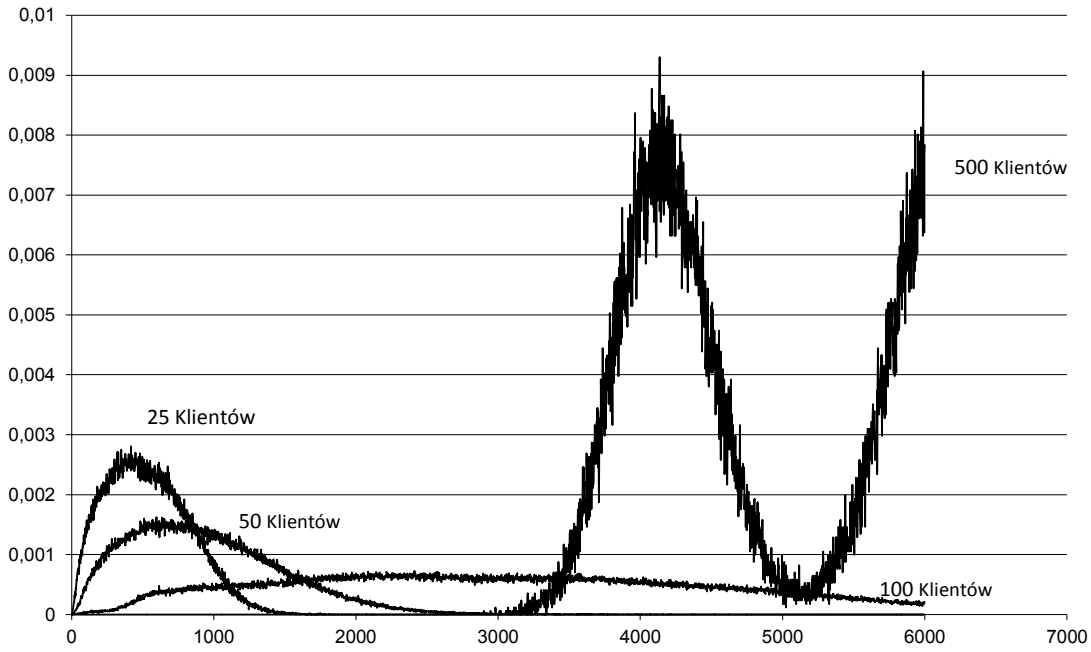

Rys. 6. Model kolejkowy systemu
Fig. 6. Queueing model of the system

Każdą aplikację reprezentują dwa rodzaje klientów – pierwszy dotyczy pobierania głównego płatnika do kontekstu, drugi rodzaj to pobieranie dokumentu w kontekście konkretnego płatnika. Są to dwie główne zależności występujące w modelu. Cały zestaw aplikacji zawiera dziesięć typów klientów.

Model został zbadany za pomocą trzech metod: symulacji, łańcuchów Markowa oraz połączenia aproksymacji dyfuzyjnej z metodą analizy wartości średnich (MVA).

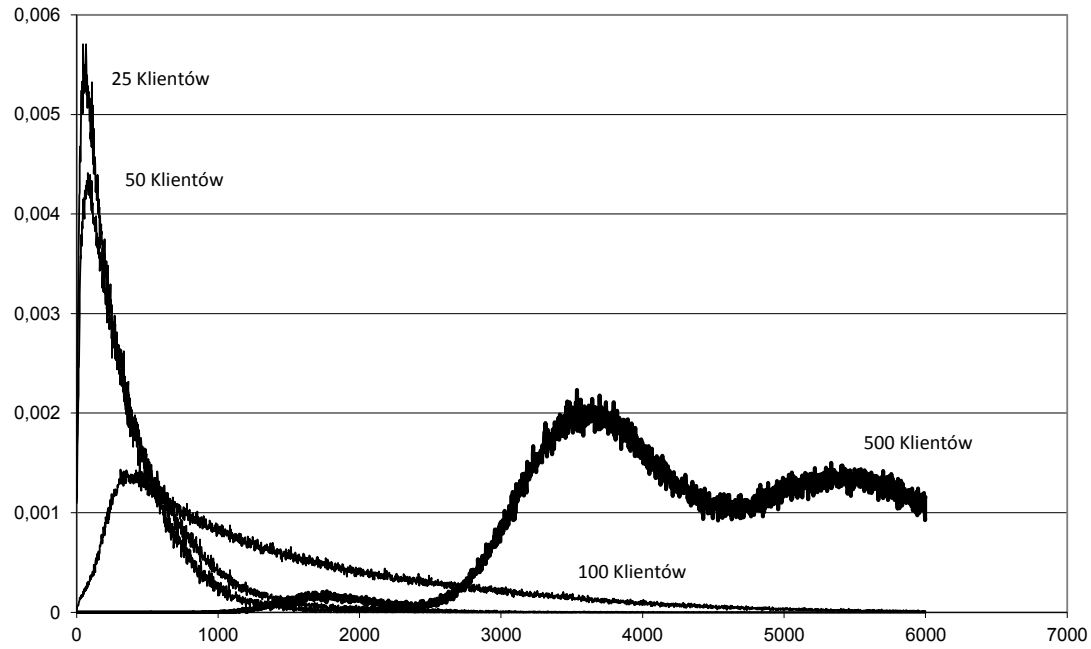
## 3.1. Model symulacyjny

Model symulacyjny został wykonany za pomocą oprogramowania OMNET++[3]. Wykorzystano w nim histogramy zgromadzone w czasie pomiarów w badanym systemie, oddzielnie dla każdej z pięciu aplikacji. Następnie przeprowadzono symulacje biorąc pod uwagę mieszaninę aplikacji oraz częstotliwość ich występowania. Na rys. 7 i 8 przedstawiono niektóre wyniki symulacji. Ukazują one rozkład łącznego czasu pobrania wszystkich dokumentów w kontekście jednego płatnika, w zależności od liczby klientów. Wyniki przedstawiono dla jednego rodzaju aplikacji.




Rys. 7. Rozkład czasu odszukania wszystkich dokumentów jednego płatnika w zależności od liczby aktywnych klientów dla aplikacji $A_1$, wyniki symulacji

Fig. 7. Density of total time for retrieval of all documents related to one payer as a function of the number of active stations, application $A_1$, simulation

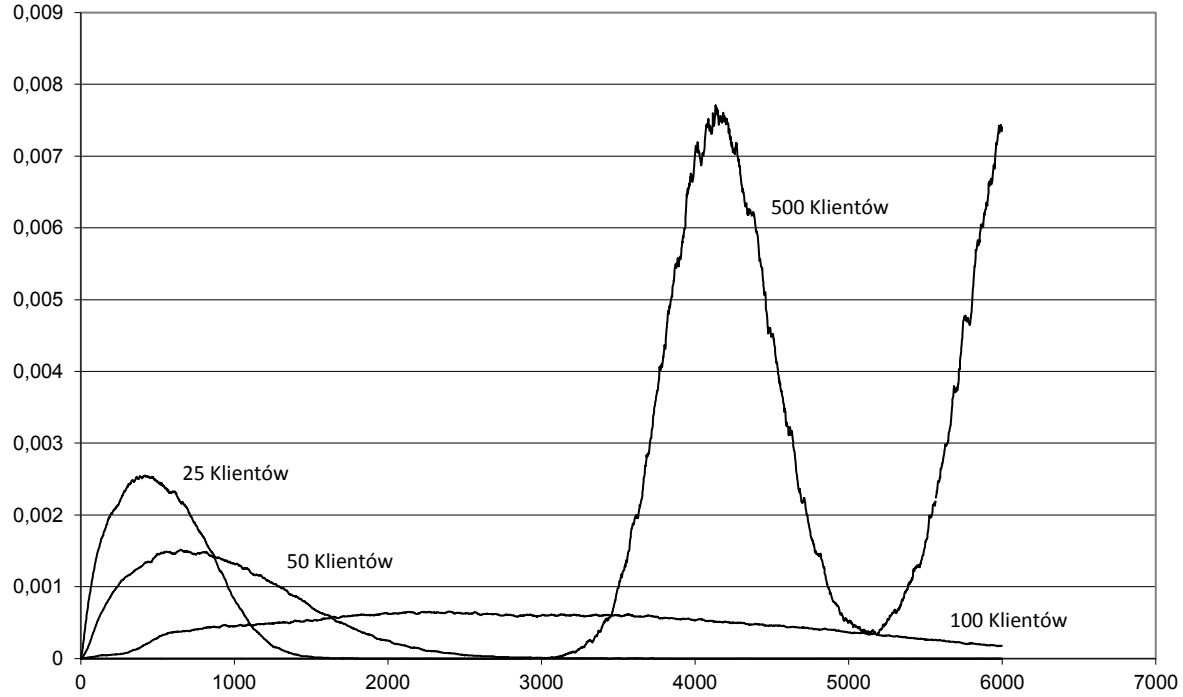



Rys. 8. Gęstość łącznego czasu odszukania wszystkich dokumentów jednego płatnika w zależności od liczby aktywnych klientów dla aplikacji $A_2$, wyniki symulacji

Fig. 8. Density of total time for retrieval of all documents related to one payer as a function of the number of active stations, application $A_2$, simulation

### 3.2. Model oparty na łańcuchach Markowa

Dla wykorzystania łańcuchów Markowa, zmierzone rozkłady czasów działania aplikacji zostały przybliżone przez rozkłady wykładnicze drugiego i trzeciego stopnia oraz rozkłady Coxa drugiego i trzeciego stopnia, których parametry dobrano metodą najmniejszych kwadratów. Minimalna suma kwadratów odległości między wartościami zmierzonymi a rozkładem dopasowanym została ustalona za pomocą funkcji Matlab *LSQNONLIN* oraz *FMINCON*. Aby zapewnić znalezienie minimum globalnego, dla każdego dopasowania określono 1600 punktów startowych, które zostały losowo wybrane w przedziałach [0, 1], [0, 10], [0, 100], [0, 1000], [0, 10000] oraz [0, $10^{15}$]. Następnie zbadana została hipoteza o rozkładzie przy użyciu testów chi-kwadrat oraz zgodności $\lambda$ Kołmogorowa, np. [4]. W większości przypadków (lecz nie we wszystkich) osiągnięto pozytywne wyniki. Daje nam to podstawę do budowy macierzy przejść łańcucha Markowa. Powstały model kolejkowy został rozwiązany za pomocą narzędzia OLYMP [7], zaprojektowanego w IITiS PAN, które umożliwia rozwiązywanie olbrzymich (do setek milionów układów) łańcuchów Markowa. Na rys. 9 przedstawiono wyniki modelu Markowa dla tej samej aplikacji, jak na rys. 7. Wyniki są zbliżone, chociaż różnice są widoczne dla dużej liczby (500) użytkowników.




Rys. 9. Rozkład całkowitego czasu wyszukiwania wszystkich dokumentów powiązanych z jednym płatnikiem, jako funkcja liczby aktywnych stanowisk, aplikacja $A_1$, model Markowa

Fig. 9. Density of total time for retrieval of all documents related to one payer as a function of the number of active stations, application $A_1$, Markov model

# 4. Model aproksymacji dyfuzyjnej

Aproksymacja dyfuzyjna jest klasyczną metodą, często stosowaną, opisaną m.in. w [1], [2] dla modeli pojedynczych stacji typu GI/GI/1, GI/GI/1/N oraz ich sieci. Poniżej wprowadzamy jej modyfikację – model stanowiska z wieloma kanałami obsługi i z ograniczoną, wieloklasową populacją klientów.

## 4.1. Zasady aproksymacji dyfuzyjnej

Niech $A(x)$, $B(x)$ oznaczają odpowiednio rozkład między nadejściem klientów w strumieniu wejściowym i rozkład czasu obsługi. Rozkłady są ogólne, zakłada się, że ich pierwsze dwa momenty są znane: $E[A]=1/\lambda, E[B]=1/\mu, Var[A]=\sigma_A^2, Var[B]=\sigma_B^2$. Oznaczmy kwadraty współczynników zmienności tych rozkładów jako $C_A^2=\sigma_A^2\lambda^2, C_B^2=\sigma_B^2\mu^2$. Niech $N(t)$ jest liczbą klientów obecnych w systemie, w czasie $t$. Dla pojedynczej kolejki typu FIFO, zmiany $N(t+\Delta t)-N(t)$ mają (w przybliżeniu) rozkład normalny ze średnią $(\lambda-\mu)\Delta t$ i wariancją $(\sigma_A^2\lambda^3+\sigma_B^2\mu^3)\Delta t$, pod warunkiem, że czas $\Delta t$ jest wystarczająco długi, a stacja pracuje bez jakichkolwiek przerw. Aproksymacja dyfuzyjna zastępuje proces $N(t)$ przez ciągły proces dyfuzji $X(t)$, którego przyrostowe zmiany $dX(t)=X(t+dt)-X(t)$ mają rozkład normalny o średniej $\beta dt$ i wariancji $\alpha dt$, gdzie $\beta$, $\alpha$ są współczynnikami równania dyfuzji:

$$\frac{\partial f(x,t;x_0)}{\partial t}=\frac{\alpha\partial^2 f(x,t;x_0)}{\partial x^2}-\beta\frac{\partial f(x,t;x_0)}{\partial x}, \tag{1}$$

które określa funkcję gęstości prawdopodobieństwa:

$$f(x,t;x_0)dx=P[x\le X(t)<x+dx\,|\,X(0)=x_0] \text{ dla } X(t).$$

Oba procesy $X(t)$ i $N(t)$ mają więc rozkład normalny zmian w czasie; wybór $\beta=\lambda-\mu, \alpha=\sigma_A^2\lambda^3+\sigma_B^2\mu^3=C_A^2\lambda+C_B^2\mu$ zapewnia ten sam stosunek wartości średniej i wariancji tych rozkładów do czasu obserwacji. Funkcja $f(n,t;n_0)$ przybliża rozkład $p(n,t;n_0)$ liczby klientów wszystkich klas obecnych w kolejce. Jeżeli strumień wejściowy $\lambda$ składa się z $K$ klas klientów mających natężenia $\lambda^{(k)}$, z całkowitą intensywnością $\lambda=\sum_{k=1}^{K}\lambda^{(k)}$, a parametry rozkładu czasu obsługi dla klasy $k$ wynoszą $E[B^{(k)}]=1/\mu^{(k)}, Var[B^{(k)}]=\sigma_B^{(k)^2}$, wtedy $B(x)$ – łączny czas rozkładu dla wszystkich klas – jest wyrażony jako:

$$B(x)=\sum_{k=1}^{K}\frac{\lambda^{(k)}}{\lambda}B^{(k)}(x), \qquad \frac{1}{\mu}=\sum_{k=1}^{K}\frac{\lambda^{(k)}}{\lambda}\frac{1}{\mu^{(k)}},$$

$$C_B^2=\mu^2\sum_{k=1}^{K}\frac{\lambda_{(k)}}{\lambda}\frac{1}{\mu^{(k)^2}}(C_B^{(k)^2}+1)-1. \tag{2}$$

Jeżeli założymy, że strumienie wejściowe klientów są niezależne, to liczba klientów przychodzących w $\Delta t$ ma rozkład normalny o wariancji $\lambda C_A^2 \Delta t = \sum_{k=1}^{K} \lambda^{(k)} C_A^{(k)^2} \Delta t$, stąd:

$$C_A^2 = \sum_{k=1}^{K} \frac{\lambda^{(k)}}{\lambda} C_A^{(k)^2} \ . \tag{3}$$

Powyższe równania określają parametry $\alpha, \beta$ równania dyfuzyjnego, opisującego liczbę klientów wszystkich *K* klas w systemie.

Trzeba również określić graniczne warunki dla równania (1). W [1] aproksymację dyfuzyjną stacji $G/G/1/N$ określono jako proces $X(t)$, na zamkniętym przedziale $x \in [0, N]$. Kiedy proces osiąga $x = 0$, pozostaje tam przez czas o rozkładzie wykładniczym z parametrem $\lambda$, po czym powraca do $x = 1$; kiedy dochodzi do $x = N$ pozostaje tam przez czas o rozkładzie wykładniczym z parametrem $\mu$, a następnie rozpoczyna się od $x = N - 1$. Równanie dyfuzyjne jest uzupełniane przez równania bilansu prawdopodobieństwa pobytu procesu w barierach $p_0(t) = P[X(t) = 0]$, $p_N(t) = P[X(t) = N]$ i przybiera postać:

$$\begin{aligned}
&\frac{\partial f(x,t;x_0)}{\partial t} = \frac{\alpha}{2} \frac{\partial^2 f(x,t;x_0)}{\partial x^2} - \beta \frac{\partial f(x,t;x_0)}{\partial x} + \lambda p_0(t) \delta(x-1) + \mu p_N(t) \delta(x-N+1), \\
&\frac{dp_0(t)}{dt} = \lim_{x \to N} [\frac{\alpha}{2} \frac{\partial f(x,t;x_0)}{\partial x} - \beta f(x,t;x_0)] - \lambda p_0(t), \\
&\frac{dp_N(t)}{dt} = -\lim_{x \to N} [\frac{\alpha}{2} \frac{\partial f(x,t;x_0)}{\partial x} - \beta f(x,t;x_0)] - \lambda p_N(t).
\end{aligned} \tag{4}$$

### 4.2. Model serwera aplikacji

W modelu dyfuzyjnym rozpatrywanego systemu powinniśmy brać pod uwagę skończony wymiar źródła klientów oraz wielokrotność kanałów usług – oba te fakty wywierają wpływ na parametry dyfuzji i sprawiają, że parametry są zależne od wartości procesu. Rozważmy na początku model jednej klasy. Niech $N$ będzie liczbą użytkowników, a *L* liczbą równoległych kanałów obsługi, natomiast parametry $\lambda$ i $C_A^2$ odnoszą się do czasu pobytu pojedynczego klienta w źródle. Wyznaczamy wartości $\alpha(x)$ oraz $\beta(x)$:

$$\beta(x) = \beta_n \ dla \ x \in (n-1, n], \quad n = 1, ...N,$$

$$\alpha(x) = \alpha_n \ dla \ x \in (n-1, n], \quad n = 1, ...N,$$

oraz określamy te wartości w następujący sposób:

$$\begin{aligned}
&\beta_n = (N - n + 1)\lambda - n\mu, \ 1 \le n \le L, \\
&\alpha_n = (N - n + 1)\lambda C_A^2 + n\mu C_B^2, \ 1 \le n \le L, \\
&\beta_n = (N - n + 1)\lambda - K\mu, \ n \ge L, \\
&\alpha_n = (N - n + 1)\lambda C_A^2 + K\mu C_B^2, \ n \ge L.
\end{aligned}$$

Rozwiązanie stanu ustalonego dla tego modelu ma postać:

$$f_i(x) = C_{1,i} + C_{2,i}e^{z_i x}, \quad gdzie \quad z_i = \frac{2\beta_i}{\alpha_i}, \quad i-1,...N.$$

Stałe $C_{1,i}, C_{2,i}$ można obliczyć z warunków ciągłości funkcji na granicach przedziałów: $f_n(n) = f_{n+1}(n), n = 1,...N-1$ i warunków bilansu przepływu: dla każdego przedziału (z wyjątkiem pierwszego i ostatniego) przepływ masy prawdopodobieństwa, określony jako $\frac{\alpha}{2}\frac{\partial f_n(x,t;x_0)}{\partial x} - \beta f_n(x,t;x_0)$, powinien być zerowy, a w przedziałach pierwszym i ostatnim powinien być zbilansowany z przepływem masy prawdopodobieństwa, wynikającym ze skoków z 0 do 1 oraz z $N$ do $N-1$. Trzeba też uwzględnić warunek normalizacyjny. Całka funkcji $f_i$ w przedziale $(x_{i-1}, x_i)$ daje:

$$\int_{x_{i-1}}^{x_i} f_i(x)dx = C_{1,i}(x_i - x_{i-1}) + C_{2,i}(1/z_1)[e^{z_i x_i} - e^{z_i x_{-1i}}],$$

stąd warunek normalizacji ma postać:

$$p_0 + \sum_{i=1}^{N}\{C_{1,i}(x_i - x_{i-1}) + C_{2,i}(1/z_1)[e^{z_i x_i} - e^{z_i x_{i-1}}]\} + p_N = 1. \tag{5}$$

Analitycznie można otrzymać wartości stałych $C_{1,i}, C_{2,i}$ uzyskując $f(x)$ w postaci:

$$\begin{aligned}
&f_1(x) = \frac{N\lambda p_0}{-\beta_1}(1-\exp(z_1)) \quad \text{dla} \quad \beta_1 \neq 0\\
&f_1(x) = \frac{2N\lambda p_0}{\alpha_1}x \quad \text{dla} \quad \beta_1 = 0, \quad 0 < x \leq 1\\
&f_i(x) = f_{i-1}(i-1)\exp(z_i(x-i+1)) \quad \text{dla} \quad \beta_i \neq 0,\\
&f_i(x) = f_{i-1}(i-1) \quad \text{dla} \quad \beta_i = 0, \quad i-1 \leq x \leq i, \quad i = 2,...N-1\\
&f_N(x) = \frac{K\mu\lambda p_N}{-\beta_N}(1-\exp(z_N(x-N))) \quad \text{dla} \quad \beta_N \neq 0\\
&f_N(x) = \frac{-2K\mu p_N}{\alpha_N}(x-N) \quad \text{dla} \quad \beta_N = 0, \quad N-1 \leq x < i
\end{aligned} \tag{6}$$

W rozważanym modelu mamy do czynienia z dwoma rodzajami klientów dla każdego typu aplikacji: klasa (1) *pobieranie płatników* ze średnim czasem obsługi $1/\mu^{(1)}$ i wariancją $\sigma_B^{(1)^2}$ oraz czasem pobytu w źródle *czas między płatnikami*, mającym średnią $1/\lambda^{(1)}$ i wariancję $\sigma_A^{(1)^2}$ i klasa (2) *pobieranie dokumentów* o średniej czasu obsługi $1/\mu^{(2)}$ i wariancji $\sigma_B^{(2)^2}$ oraz z czasem pobytu w źródle *przerwa pomiędzy pobieraniem dokumentów* o średniej $1/\lambda^{(2)}$ i wariancji $\sigma_A^{(2)^2}$. Każde pobranie płatnika odpowiada pewnej liczbie $r$ pobranych dokumentów. W tym miejscu pomijamy rozkład *r*, mając na względzie tylko jego wartość śred-

nią $\bar{r}$. Na przykład, w przypadku aplikacji $A_1$ mamy (średnie czasy są wyrażone w sekundach, wariancje w $s^2$):

$$1/\mu^{(1)} = 30{,}894;\ \sigma_B^{(1)^2} = 5040{,}71;\ 1/\lambda^{(1)} = 327{,}901;\ \sigma_A^{(1)^2} = 126329 \text{ oraz}$$

$$1/\mu^{(2)} = 26{,}2426;\ \sigma_B^{(2)^2} = 2906{,}68;\ 1/\lambda^{(2)} = 26{,}23;\ \sigma_A^{(2)^2} = 2906{,}82;\ \bar{r} = 6{,}19.$$

Dla określenia przepływów wejściowych używamy iteracyjnego modelu na podstawie prezentowanego powyżej modelu dyfuzyjnego i analizy wartości średnich. Niech $M$ będzie liczbą aktywnych terminali, obsługujących jeden rodzaj aplikacji (pomijamy w tej chwili pozostałe aplikacje). To znaczy, że mamy $N^{(1)} = M$ klientów pierwszego typu i $N^{(2)} = \bar{r}M$ klientów drugiego typu ($\bar{r}M$ jest zaokrąglone do najbliższej liczby naturalnej). Parametry dyfuzji dla przypadku, w którym obie klasy są brane pod uwagę razem wyznacza się z równań (2) i (3). W związku ze skończoną populacją klientów, prawdopodobieństwo znalezienia klienta klasy $k$ w źródle jest inne niż znalezienia go w systemie. Niech $p^{(k)}$ oznacza prawdopodobieństwo, iż klient w systemie należy do klasy $k$, a $q^{(k)}$ jest prawdopodobieństwem, że klient wewnątrz źródła należy do klasy $k$. Postępujemy zgodnie z poniższym algorytmem:

1. Na początku przyjmij $p^{(k)} = q^{(k)} = \dfrac{N^{(k)}}{N}$.
2. Używając równań (2), (3) i modelu dyfuzyjnego (6) oblicz rozkład $p(n)$ liczby klientów wszystkich klas w systemie, średnią długość kolejki (nie wliczając klientów w obsłudze):

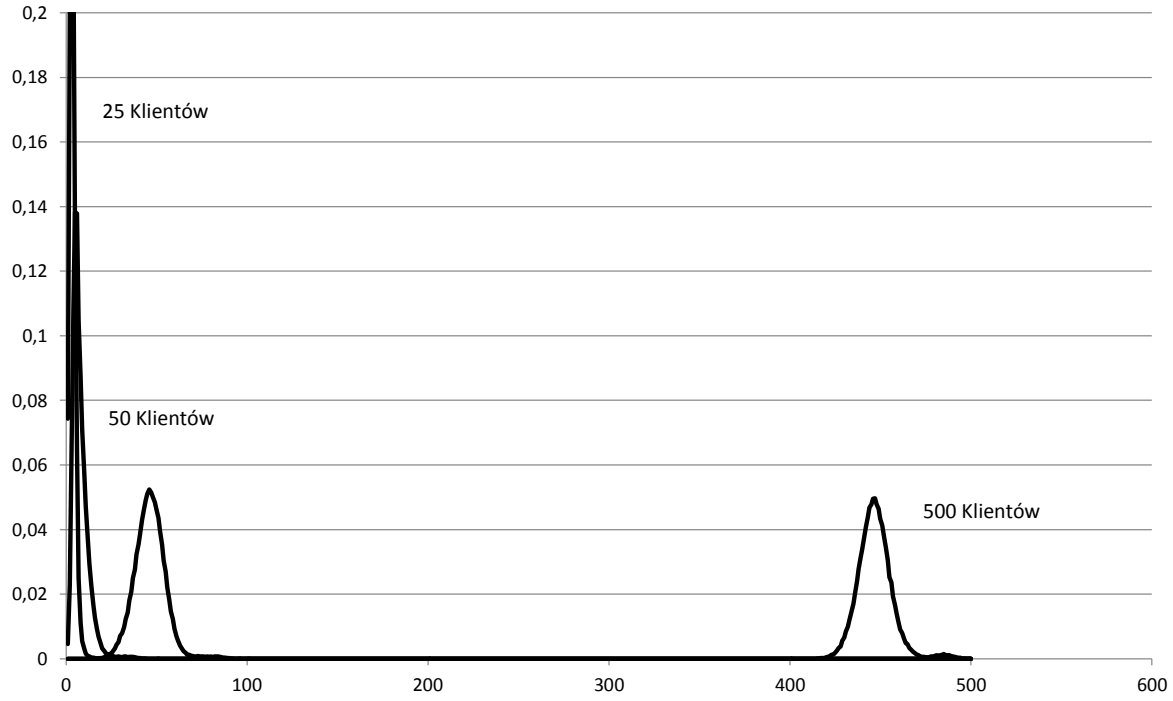
   $$E[k] = \sum_{n=K+1}^{N} p(n)(n-K)$$

   oraz średni czas oczekiwania $E[w] = E[k]/\mu$.
3. Oblicz nową wartość dla $p^{(k)}, q^{(k)}$

$$p^{(k)} = \frac{\dfrac{[E[w]+1/\mu^{(k)}]N^{(k)}}{1/\lambda^{(k)} + E[w] + 1/\mu^{(k)}}}{\sum_{k=1}^{K} \dfrac{[E[w]+1/\mu^{(k)}]N^{(k)}}{1/\lambda^{(k)} + E[w] + 1/\mu^{(k)}}}, \qquad q^{(k)} = \frac{\dfrac{N^{(k)}/\lambda^{(k)}}{1/\lambda^{(k)} + E[w] + 1/\mu^{(k)}}}{\dfrac{N^{(k)}/\lambda^{(k)}}{1/\lambda^{(k)} + E[w] + 1/\mu^{(k)}}}$$

   i wróć do punktu 2.

Powtarzamy tę pętlę aż do osiągnięcia zbieżności. Nie istnieją żadne dowody na zbieżność, ale w więcej niż stu obliczonych numerycznych przykładach procedura była zawsze zbieżna w mniej niż dziesięciu powtórzeniach. Rysunek 10 przedstawia wyniki tego algorytmu, tzn. ostateczną postać funkcji $f(x)$, przestawioną równaniem (6) dla aplikacji $A_1$, biorąc pod uwagę podane wyżej dane liczbowe – jest to aproksymacja długości kolejki podczas pojedynczego dostępu do serwera, które powinno być następnie połączone z innym rozkładami, wpływającymi na całkowity czas obsługi dla całej aplikacji.




Rys. 10. Aproksymacja dyfuzyjna liczby klientów w systemie obsługi, $f(x)$ wyrażona równaniem (6), aplikacja $A_1$
Fig. 10. Diffusion approximation of the number of customers in the service system, $f(x)$ given by Eq. (6), application $A_1$

## 5. Wnioski

Artykuł opisuje próby ujęcia cech charakterystycznych czynności klienta w wielkim komputerowym systemie bazy danych oraz skonstruowania modelu dla analizy wydajności systemu w funkcji liczby klientów. Przedstawiono wyniki na podstawie pomiarów rzeczywistej sieci, modelu symulacyjnego, modelu Markowa i aproksymacji dyfuzyjnej. Wyniki wykazują dużą zgodność opracowanych modeli.

Aproksymacja dyfuzyjna, jako że działa tylko na podstawie średnich wartości i wariancji mierzonego czasu rozkładu oraz agreguje stany widziane oddzielnie przez łańcuch Markowa, wymaga znacznie mniejszego czasu programowania i potrzebuje mniej mocy obliczeniowych niż inne metody. Autorzy skupili się na skalowaniu tylko jednego parametru badanego systemu – liczbie użytkowników i jej wpływie na pracę systemu, zdając sobie sprawę z większej liczby stopni swobody w kształtowaniu jego działania.

**BIBLIOGRAFIA**

1. Gelenbe E.: On Approximate Computer System Models. J.ACM, Vol. 2, No. 2, 1975.
2. Gelenbe E., Pujolle G.: The Behaviour of a Single Queue in a General Queueing Network, Acta Informatica, Vol. 7, Fasc. 2, 1976, s. 123÷136.
3. OMNET++ site: http://www.omnetpp.org
4. DeGroot M. H., Schervish M. J.: Probability and Statistics, third edition, Addison Wesley, Boston 2002.
5. Kleinrock L.: Queueing Systems, Vol. II, Wiley, New York 1976.
6. Newell G. F.: Applications of Queueing Theory. Chapman and Hall, London 1971.
7. Pecka P.: An object-oriented software system for numerical solution of Markovian queueing models, PhD Thesis, IITiS PAN Gliwice, 2002.

Recenzenci: Prof. dr hab. inż. Roman Gielerak
Prof. dr hab. inż. Zbigniew Huzar

Wpłynęło do Redakcji 14 kwietnia 2011 r.

**Abstract**

The article presents a queueing model for performance evaluation of a large database system at an assurance company. The system includes a server with a database and a local area network with a number of terminals where the company employees run applications that introduce documents or retrieve them from the database. We discuss a model of clients' activities. Measurements were collected inside the working system: the phases at each user application performance were identified and their duration was measured. The collected data are used to construct a synthetic model of applications activities which is then applied to predict the system behaviour in case of the growth of the number of users. We apply simulation, Markov chain and diffusion models – their comparison, based on real data, may better verify the utility of particular methods than usual academic examples.

**Adresy**

Tadeusz CZACHÓRSKI: Instytut Informatyki Teoretycznej i Stosowanej Polskiej Akademii Nauk, ul. Bałtycka 5, 44-100 Gliwice, Polska, tadek@iitis.gliwice.pl

Krzysztof GROCHLA: Instytut Informatyki Teoretycznej i Stosowanej Polskiej Akademii Nauk, ul. Bałtycka 5, 44-100 Gliwice, Polska, kil@iitis.gliwice.pl

Adam JÓZEFIOK: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16, 44-100 Gliwice, Polska, jozefiok@gmail.com

Tomasz NYCZ: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16, 44-100 Gliwice, Polska, tomasz.nycz@polsl.pl