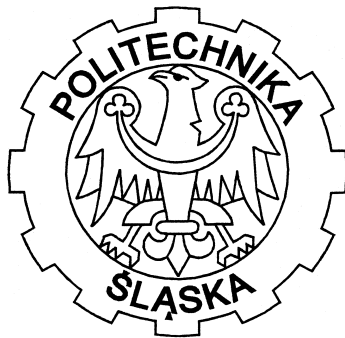


Politechnika Śląska
Wydział Automatyki, Elektroniki i Informatyki
Instytut Informatyki



mgr inż. Stanisław Wideł

Analiza architektury i wydajności Systemów Odpowiedzi Fonicznej

Autoreferat rozprawy doktorskiej

Promotor: prof. dr hab. inż. Bolesław Pochopień

Gliwice, 2013

Spis treści

1. Geneza rozprawy.....	3
2. Uzasadnienie wyboru tematu.....	4
3. Problem badawczy i tezy rozprawy	6
4. Układ rozprawy.....	7
Wykorzystanie mowy w systemach komputerowych.....	7
Architektura systemu odpowiedzi fonicznej.....	8
Analiza wydajności systemu odpowiedzi fonicznej.....	10
Analiza przestrzeni wydajności badanego modelu.....	11
5. Wnioski	13
6. Bibliografia.....	19

1. Geneza rozprawy

Rozprawa stanowi podsumowanie prac, wykorzystanie wiedzy i doświadczenia autora z zakresu sieci komputerowych [62, 63, 75, 184, 187, 188], systemów przemysłowych z ograniczeniami czasowymi [58, 59, 60, 94, 182, 186] i telefonii komputerowej [191, 193].

Genezą pracy było uczestnictwo w projekcie badawczym „Multimedialny rozproszony system zarządzania Urzędem Miasta Pszów” [74, 191, 192], w którym autor przeniósł doświadczenia zdobyte na stażu w firmie Cheyenne Communication USA [189, 190] w zakresie tworzenia systemów telefonii komputerowej. Po zakończeniu projektu badawczego prace z zakresu odpowiedzi fonicznej były kontynuowane [147, 181, 182, 183, 185, 194].

Stworzone oprogramowanie w ramach projektu zostało wdrożone między innymi w Urzędzie Miejskim w Gliwicach. Rozwój zastosowań systemu odpowiedzi fonicznej napotkał jednak na wiele barier. Z punktu widzenia wdrożeń pojawiła się bariera wynikająca z narzuconego modelu biznesowego w sieci telekomunikacyjnej [126]. Dla użytkownika korzystanie z serwisu, który jest płatny za każdą minutę połączenia staje się nieatrakcyjne jeżeli alternatywą jest możliwość korzystania z Internetu, gdzie opłaty za połączenie są zryczałtowane. Jedynie operatorzy telekomunikacyjni mogą, i to tylko w obrębie własnej infrastruktury, zwalniać użytkowników z opłat za korzystanie z udostępnionych serwisów. Dlatego też, w warunkach polskich, komercyjne implementacje systemów odpowiedzi fonicznej pracują niemal wyłącznie jako elementy infrastruktury operatorów telekomunikacyjnych.

Barierą stała się również automatyczna generacja poprawnie gramatycznej wypowiedzi ze względu na brak ogólnie dostępnych algorytmów i bibliotek API (ang. *Application Programming Interface*) [157] opracowanych przez językoznawców. Autor nie jest specjalistą w rozwiązywaniu zagadnień algorytmizacji generowania wypowiedzi w języku polskim. Jednak, aby skutecznie dokonać implementacji systemu odpowiedzi fonicznej konieczne było podjęcie prac nad dedykowanymi rozwiązaniami, tak jak to opisano w rozprawie.

Od lat 90 systemy telekomunikacyjne migrują w stronę sieci opartych na protokole TCP/IP. W konsekwencji systemy sieciowe i telekomunikacyjne wykorzystują te same technologie. Mowa jako interfejs z użytkownikiem jest wykorzystywana w systemach telekomunikacyjnych, ale również może być wykorzystywana w sieciach komputerowych. Mowa narzuca reżimy czasowe, które muszą być spełnione, by praca z aplikacją nie powodowała dyskomfortu dla użytkowników. Projekt Systemu Odpowiedzi Fonicznej wymaga połączenia wiedzy z zakresu sieci komputerowych z uwzględnieniem problemów związanych z ograniczeniami czasowymi.

2. Uzasadnienie wyboru tematu

Rozwój konwergentnych platform telefonicznych o otwartym kodzie źródłowym [111] oraz otwartych standardów takich jak opis serwerowych aplikacji za pomocą VoiceXML [81, 95] są czynnikami, które powodują, że systemy odpowiedzi fonicznej są elementem infrastruktury sieci opartej na protokole TCP/IP (ang. *Transmission Control Protocol/Internet Protocol*).

Tworzenie nowych systemów [178] wymaga opracowania metod i narzędzi do ich testowania i weryfikacji. Aplikacje foniczne muszą zachować rygor pracy czasu rzeczywistego [100] i to nie tylko w stosunku do warstwy transportowej przesyłanego dźwięku (czym zajmują się techniki kształtowania pasma w sieciach komputerowych), ale również do współdziałania elementów oprogramowania, często rozproszonych na kilka systemów komputerowych.

Jednym z podejść [21] jakie stosuje się do testowania i uruchamiania prototypowych aplikacji jest pomiar wydajności [22]. Zamiast dekompozycji systemu i weryfikowania poszczególnych składowych, podsystemów czy funkcji, wybiera się badanie wydajności całego systemu.

Opierając się na założeniu, że poprawna implementacja działa lepiej i wydajniej niż implementacja z błędami funkcjonalnymi można bardzo szybko zdiagnozować prototyp nowego systemu. Zastosowanie testów wydajności [113] do detekcji błędów i weryfikacji poprawności implementacji wymaga:

- opracowania kryterium ilościowej oceny wydajności systemów interaktywnej odpowiedzi fonicznych;
- opracowania metod pomiarowych i pomiarów na systemach referencyjnych;
- opracowania modelu systemu;
- wyznaczenie ograniczeń asymptotycznych;
- poszukiwania stanów wydajności systemu oraz metod ich identyfikacji;
- poszukiwanie miar wydajności na podstawie parametrów łatwo dostępnych pomiarowo;
- badania cech statystycznych rzeczywistych systemów;
- opracowania metod pomiarowych dla systemów niestacjonarnych.

W literaturze angielskiej [117] powszechnie stosuje się akronim IVR (ang. *Interactive Voice Response*) do określenia systemu, w którym wykorzystuje się mowę (ciąg poleceń głosowych) do komunikacji użytkownika z systemem komputerowym.

W rozprawie zamiast bezpośredniego tłumaczenia akronimu IVR, proponuje się przyjąć termin **System Odpowiedzi Fonicznej**. Zmiana w stosunku do bezpośredniego tłumaczenia skrótu IVR, polega na użyciu terminu fonia zamiast mowy czy głosu, co znacznie

przybliżyć opis funkcjonalny i zakres wymagań do współczesnych zastosowań mowy w systemach komputerowych.

Aktualnie wdrażane systemy odpowiedzi fonicznej są proste, nie przekraczają zwykle 10 modułów dialogowych, działają w oparciu o sieci telekomunikacyjne i są bardzo dobrze skalowalne. Aby zwiększyć zakres stosowania na przykład liczby modułów dialogowych, konieczne jest wprowadzenie technik pozwalających nadawać znaczenie semantyczne dynamicznie prowadzonemu dialogowi. Potoczne rozumienie dialogu prowadzonego za pomocą mowy to coś znacznie więcej niż wydawanie poleceń za pomocą pojedynczych słów, jakie system komputerowy może rozpoznać oraz odpowiedzi budowanych na podstawie stałych modułów dialogowych.

3. Problem badawczy i tezy rozprawy

Wykorzystanie mowy do budowy dialogu człowieka z systemem komputerowym prowadzi do rozwoju nowych metod projektowania aplikacji, które różnią się od aplikacji wykorzystujących GUI. Wynika to z konieczności uwzględnienia psychologicznych i fizjologicznych cech prowadzenia dialogu za pomocą mowy. Specyfika fonicznego interfejsu z użytkownikiem powoduje, że konieczne staje się proponowanie nowych rozwiązań architektury systemu, gdzie spełnienie wymagań czasowych narzucanych przez wykorzystanie toru fonicznego jest ważnym kryterium projektowym. Innym istotnym elementem jest równoległa obsługa wielu użytkowników w tym samym czasie oraz konieczność adaptacyjnych modyfikacji interfejsu z użytkownikiem w trakcie działania aplikacji.

Zasadniczym problemem badawczym w pracy jest analiza czasów reakcji wielowątkowego systemu odpowiedzi fonicznej opartej na sprzętowej implementacji równoległego przetwarzania rozproszonego z przesyłem komunikatów dla zmiennych parametrów wejściowych takich jak: czasy obsługi na poszczególnych zasobach, czas trwania komunikatu fonicznego i liczba wątków.

Rozważania przeprowadzone w rozprawie, potwierdzone odpowiednimi badaniami, służą wykazaniu prawdziwości następujących tez:

- I. Architektura Systemu Odpowiedzi Fonicznej definiuje sposób implementacji wielokanałowości, wielowątkowości i konfigurowalności systemu. Technika programowania zwana maszyną stanów pozwala zbudować System Odpowiedzi Fonicznej niezależnie od mechanizmów wielowątkowości systemu operacyjnego, w tym również na jednozadaniowym systemie operacyjnym.
- II. Ilościowym parametrem wydajności Systemu Odpowiedzi Fonicznej jest jego czas odpowiedzi, który można wyznaczyć, również dla procesu niestacjonarnego, na podstawie średniej liczby klientów, przepustowości systemu i średniego czasu trwania komunikatu fonicznego.
- III. Liczba klientów, czas trwania komunikatu fonicznego i czasy obsługi na stanowiskach wyznaczają przestrzeń zmiennych wejściowych dla modelu Systemu Odpowiedzi Fonicznej. Dla zmiennego czasu obsługi wybranego stanowiska i liczby klientów rozwiązanie modelu pozwala wyróżnić następujące stany wydajności systemu:
 - nieobciążenia,
 - maksymalnego obciążenia,
 - maksymalnego obciążenia, które spowodowało wybrane stanowisko,
 - stanu zrównoważenia.

4. Układ rozprawy

Celem rozprawy jest analiza architektury i wydajności sieciowego systemu, który do komunikacji z użytkownikiem wykorzystuje mowę.

Wykorzystanie mowy w systemach komputerowych

Mowa jako sposób komunikacji z systemem komputerowym stanowi jeden z obszarów badawczych [152], projektowania i rozwoju interfejsów systemów komputerowych HCI (ang. *Human Computer Interaction*) [162].

Badania [50] wykazują pozytywne nastawienie użytkowników do wykorzystywania mowy. Odsetek pozytywnych odpowiedzi na te same pytania jest większy przy użyciu telefonu niż formularzy WWW.

W rozprawie zdefiniowane zostało pojęcie fonicznego interfejsu z użytkownikiem jako ciągu poleceń i odpowiedzi, w których do wymiany informacji używa się toru fonicznego. Ze względu na wymagany duży potencjał zespołów badawczych zajmującymi się technikami syntezy mowy oraz konieczność pracy w interdyscyplinarnym zespole informatyków, lingwistów, akustyków, psychologów w rozprawie przyjęto założenie, aby bez prób teoretycznej analizy, utylitarnie wykorzystać istniejące generatory, czyli narzędzia syntezy mowy z tekstu. Współczesna cywilizacja wzmacnia przekonanie, że tekst jest podstawową formą języka. Jednak budowanie gramatycznie poprawnej wypowiedzi za pomocą mowy, na podstawie tekstu, nie jest jednoznaczne. Wynika to z faktu, że język mówiony i pisany różnią się pod wieloma względami. W rozprawie wprowadzono pojęcie tekstu fonicznego jako jednowymiarowego, w pełni odzwierciedlającego reguły gramatyczne, jednoznacznego zapisu sekwencji mowy.

W zależności od charakteru udostępnianej, bądź odbieranej od użytkownika informacji, wyróżniono szereg sposobów działania Systemów Odpowiedzi Fonicznej (AudioTekst, Foniczny dostęp do Baz Danych, Poczta Głosowa, Automatyczny Operator). Jeżeli System Odpowiedzi Fonicznej będzie realizował funkcje określonej aplikacji, to powstaje pytanie, czy istnieje pewien podzbiór algorytmów translacji na tekst foniczny za pomocą których można wygenerować dowolny komunikat wymagany w danej klasie aplikacji? Stopień skomplikowania gramatyki języka polskiego powoduje, że dynamiczna generacja prostych komunikatów wymaga zastosowania wiedzy lingwisty, którą nie dysponuje przeciętny informatyk. W pracy przedstawiono algorytmy budowania gramatycznie poprawnej wypowiedzi w języku polskim, na podstawie danych przechowywanych w systemach komputerowych (np. dane liczbowe, pola w bazach danych).

Architektura systemu odpowiedzi fonicznej

W rozprawie omówiono podstawowe założenia dla projektowania wielodostępnego Systemu Odpowiedzi Fonicznej oraz technicznych cech zasobów sprzętowo - programowych umożliwiającym udostępnianie usług w środowisku sieci za pośrednictwem interfejsu fonicznego. Zdefiniowano pojęcie Serwera Fonicznego jako systemu komputerowego (zwykle rozproszonego) wyposażonego w wielokanałowe interfejsy komunikacyjne, takie jak: łącza telefonii stacjonarnej, komórkowej bądź sieci komputerowej, który do dialogu z użytkownikiem posługuje się mową.

W rozprawie omówiono podstawowe funkcje kanału transmisyjnego, i wielokanałowej karty komunikacyjnej firmy Intel/Dialogic w tym celu wbrano opis karty D/41D na podstawie katalogu z 1996 roku by podkreślić fakt, iż wykorzystanie mowy do dialogu z systemami komputerowymi jest technologicznie dostępne już od kilkunastu lat. Istnieje wiele różnych kart komunikacyjnych jakich można użyć do budowy Sewera Fonicznego. Sprzęt taki jest wyposażony przez producenta w odpowiednie oprogramowanie systemowe. Oczywiście oprogramowanie systemowe nie jest bezpośrednio przenośne pomiędzy sprzętem pochodzącym od różnych producentów. Istnieje zatem pytanie, czy można opracować model sprzętu dla Serwera Fonicznego tak, aby przenosić aplikacje pomiędzy rozwiązaniami sprzętowymi różnych producentów. W rozprawie dokonano porównania oprogramowania systemowego dostarczonego wraz z kartami różnych producentów. Analizie poddano również zdarzenie pochodzące od karty fonicznej. Zaprojektowano odpowiednie struktury parametrów w celu wymiany danych pomiędzy sprzętem a oprogramowaniem aplikacyjnym. Parametry globalne dotyczące całej karty sprzętowej, parametry kanału odnoszące się do poszczególnych kanałów fonicznych, funkcje służące do inicjowania określonych zadań oraz związane z nimi zdarzenia informujące o sposobach zakończenia zainicjowanych funkcji pozwoliły na zbudowanie biblioteki API, pozwalającej na przenośność oprogramowania Serwerów Fonicznych pomiędzy kartami różnych producentów.

Rezultatem przeprowadzonych prac jest biblioteka pozwalająca na uruchomienie zaprojektowanych aplikacji fonicznych na kartach wybranych producentów. Stworzenie takiej biblioteki umożliwia budowę aplikacji, które byłyby przenośne. Bibliotekę zawierającą wielowątkowe procedury i definicje struktur danych nazwano skrótem UH. W rozprawie omówiono sposoby implementacji współbieżnego sterowania. Program Serwera Fonicznego może obsługiwać zdarzenia przychodzące z kanału poprzez funkcje blokujące działanie aplikacji na czas obsługi zdarzenia. Program taki jest łatwy do zaimplementowania, gdy w wielozadaniowym systemie operacyjnym, równolegle pracować będzie tyle programów Serwera Fonicznego ile jest kanałów. Możliwe też jest zastosowanie funkcji nieblokujących, czyli takich, które tylko inicjują obsługę zdarzenia, a następnie oddają sterowanie aplikacji. Sposób ten pozwala na napisanie wielowątkowej aplikacji niezależnej od mechanizmów obsługujących wielozadaniowość w konkretnym systemie operacyjnym. W rozprawie uzasadniono wybór asynchronicznego modelu progra-

mowania. Takie założenie daje możliwość implementacji zarówno dla systemów wielozadaniowych jak i dla systemów jednozadaniowych.

Tworzenie aplikacji dla Serwerów Fonicznych jest procesem złożonym, ze względu na specyfikę związaną z projektem i testowaniem fonicznego interfejsu z użytkownikiem. Rozwiązaniem docelowym jakie należy przyjąć przy tworzeniu tych aplikacji dla Serwerów Fonicznych jest wprowadzenie podziału na odpowiednie moduły funkcjonalne. Należy przyjąć, że definicja interfejsu fonicznego w danej klasie aplikacji może być zewnętrzna w stosunku do kodu programu. Tworzeniem programu będzie zajmować się programista, natomiast tworzeniem interfejsu fonicznego – administrator systemu wyposażony w odpowiednie środki do nagrywania i edycji komunikatów fonicznych. Metoda opisu interfejsu fonicznego jest podstawowym zagadnieniem wpływającym na architekturę systemu. Opis interfejsu fonicznego jest to struktura danych zawierająca informacje sterujące oraz zapis dźwiękowy. Najprostszą formą takiego opisu jest tekst w postaci zapisu alfanumerycznego. Aby zamienić tekst alfanumeryczny na postać foniczną należy użyć konwertera tekstu na mowę. Z punktu widzenia jakości dźwięku systemy syntezy dźwięku bezpośrednio z tekstu ustępują systemom, które wykorzystują nagrania składające się z ciągłych wypowiedzi. Wynika to również z faktu, że poza treścią alfanumeryczną interfejs foniczny przekazuje informacje poprzez intonację, szybkość wypowiedzi i dobór głosu lektora. Możliwe jest przejście koncepcji opartej na pliku tekstowym z opisem lub definicją interfejsu. W momencie uruchomienia aplikacji tworzy się strukturę interfejsu fonicznego na podstawie informacji zawartych w pliku konfiguracyjnym. Tworzone są wszystkie powiązania między punktami menu użytkownika (warunki tranzycji) oraz kolejność odczytywania komunikatów dla poszczególnych punktów menu. Przy większej liczbie użytkowników oraz wykorzystaniu serwera do różnych aplikacji może okazać się, że zmiany opisu interfejsu fonicznego muszą być dokonane dynamicznie. Aplikacja musi sama zmodyfikować opis interfejsu fonicznego, gdy użytkownik bez ponownego uruchomienia chce dokonać zmian w systemie (na przykład poprzez interfejs foniczny). W ramach etapu implementacji, zbadano dostępne technologie, w tym sprzęt do budowy systemu, zaproponowano wielokanałową, wielowątkową i konfigurowalną architekturę oraz zbudowano system, który miał zastosowania komercyjne.

Dysponując własną implementacją systemu przyjęto założenia dotyczące autonomicznej pracy karty, opisu aplikacji fonicznej, jednoczesnej pracy torów fonicznych, zachowania reżimów czasowych. Powyższe założenia pozwalają również na analizę rozwiązań technicznych innych producentów nie ograniczając się tylko do rozwiązania zaproponowanego przez autora.

Analiza wydajności systemu odpowiedzi fonicznej

Proponuje się przyjąć czas odpowiedzi (reakcji) systemu jako podstawowy parametr opisujący wydajność Systemu Odpowiedzi Fonicznej.

W rozprawie zaproponowano model kolejkowy Systemu Odpowiedzi Fonicznej. Zasoby z eksperymentalnego systemu w modelu odpowiadają stanowiskom obsługi. W dalszej części rozprawy przyjęto założenie, że dane do modelu czyli lista operacji wraz z czasem ich wykonania zostaną pozyskane z dziennika. Poza danymi o komunikatach fonicznych dziennik zawiera informacje o zdarzeniach, warunkach przejścia do następnych stanów. Dysponując własną implementacją, po zbudowaniu modelu, format dziennika zmodyfikowano specjalnie tak, by dostarczał wszystkie niezbędne dane dla modelu. Jednocześnie, aby zmniejszyć błędy pomiarowe uruchomiono System Odpowiedzi Fonicznej na jednozadaniowym systemie operacyjnym. W pracy została wprowadzona zależność na ograniczenia asymptotyczne czasu odpowiedzi systemu. Rzeczywisty średni czas odpowiedzi znajduje się w obszarze wyznaczonym przez ograniczenie asymptotyczne górne i ograniczenie dolne. Dla analizowanego modelu, który jest programem sterującym przyjęto, że analiza wartości średnich MVA (ang. *Mean Value Analysis*) stanowić będzie podstawową metodę rozwiązania modelu analitycznego.

Jednym z wniosków zastosowania metody MVA, jest możliwość odpowiedniego doboru czasu trwania komunikatu fonicznego, który może spowodować znaczną poprawę czasu reakcji systemu, bez wprowadzania innych, bardziej kosztownych, działań (ingerencja w kod bądź architekturę sprzętową systemu). Pozyskanie dokładnych danych jest warunkiem poprawnego wykorzystania modeli dla oceny efektywności. Modelowanie, niezależnie od typu (analityczne, symulacyjne), musi opierać się na prawidłowo wyznaczonych parametrach liczbowych. Wszystkie współczesne serwery sieciowe rejestrują dane do specjalnych plików systemowych zwanych dziennikami. Wpis do dziennika w zaprojektowanym systemie odbywa się po wywołaniu systemowej funkcji. Funkcja ta wywoływana jest przez procedurę, która chce umieścić dane w dzienniku. Wykazano, że odczyt czasu wykonania operacji z dziennika obarczony jest błędami systematycznym – metody i stochastycznym.

Dla wartości stałych rozwiązanie problemu korekty błędu metody wyznaczania czasów wykonywania operacji na podstawie dziennika jest proste. Natomiast przedstawiona w pracy analiza na podstawie dyskretnych funkcji gęstości prawdopodobieństwa czasów odpowiedzi może mieć znacznie szersze zastosowania niż tylko korekta błędu metody zapisu danych do dziennika. Jeżeli z sygnału czasowego pozyska się dyskretne funkcje gęstości prawdopodobieństwa, to wynikiem złożenia niezależnych systemów jest splot dyskretnych funkcji gęstości prawdopodobieństwa.

Splot i rozplot dyskretnej funkcji gęstości prawdopodobieństwa parametrów systemu komputerowego są rozszerzeniem metod i w pełni pokrywają obszary zastosowań obejmujące analizę najgorszego przypadku, jaka jest stosowana w systemach czasu rzeczywistego oraz badań na podstawie wartości średniej otrzymanej z testów typu „benchmark”.

W trakcie pracy z dyskretnymi funkcjami gęstości prawdopodobieństwa otrzymanymi z eksperymentalnego systemu w wyniku pomiarów okazało się, że metody rozplotu wykazywały bardzo dużą wrażliwość na błędy pomiarowe. W związku z tym dla rozkładu z błędem pomiarowym został zdefiniowany błąd bezwzględny oraz zostały rozróżnione sytuacje, w których powstają błędy pomiarowe.

Problem rozplotu przedstawiony można rozwiązać jako układ równań algebry liniowej. Metoda ta nosi nazwę rozplotu za pomocą najmniejszych kwadratów (ang. *least squares deconvolution*) i wykorzystuje większą liczbę równań, niż algorytmy obliczeniowe rozplotu bezpośrednio z definicji lub za pomocą dzielenia wielomianów. Ze zbadanych metod rozplotu najlepiej do zastosowań korekcji danych pomiarowych nadawała się metoda rozplotu za pomocą najmniejszych kwadratów.

W pracy przedstawiono wpływ szerokości przedziału histogramu na dokładność uzyskiwanego rozkładu oraz problem niestacjonarnych pomiarów.

Analiza przestrzeni wydajności badanego modelu

Celem przeprowadzonej analizy było maksymalne uproszczenie założeń i poszukiwanie podstawowych zależności, w stanie ustalonym. Wykorzystując analityczne rozwiązanie modelu, w którym krąży stała liczba klientów, można za pomocą metody MVA obliczyć wartości średnie parametrów opisujących stan systemu. Przy czym zamiast wyznaczenia jednej wartości liczbowej dla konkretnych wartości danych wejściowych, obliczenia można przeprowadzić dla wszystkich punktów znajdujących się na płaszczyźnie wyznaczonej przez czas obsługi wybranego stanowiska i liczbę klientów.

Na podstawie zależności wykorzystania zasobu jako funkcji czasu obsługi i liczby klientów przeprowadzono analizę wykorzystania zasobu, wprowadzono definicję maksymalnego wykorzystania zasobu oraz porównano wykorzystanie zasobu z wynikami otrzymanymi dla systemu otwartego.

W przestrzeni rozwiązań modelu zostały wyróżnione obszary:

- nieobciążony α ;
- maksymalnego wykorzystania β ;
- maksymalnego wykorzystania stanowiska wybranego do obserwacji β_0 ;
- zrównowazenia ρ .

Analizując przepustowość wprowadzono ograniczenia asymptotyczne oraz definicję stanu wydajności α . Poza wprowadzeniem ograniczeń asymptotycznych podano zależność określającą, kiedy wybrane stanowisko nie będzie wąskim gardłem. Różnica pomiędzy maksymalną a minimalną wartością czasu odpowiedzi systemu zależy również od czasu

trwania komunikatu fonicznego. Aby otrzymać zakładany poziom odpowiedzi systemu można przyspieszać wszystkie elementy systemu, ale możliwe jest również dotrzymanie reżimów czasowych poprzez wydłużenie czasu trwania komunikatu głosowego. Informacje, które można odczytać z wykresów są bardzo istotne z punktu widzenia zaplanowania strategii optymalizacji wydajności poszczególnych zasobów. Nie zawsze przyspieszanie (optymalizacja) wybranego stanowiska powoduje zmniejszenia czasu odpowiedzi systemu. Na podstawie przyrostu czasu odpowiedzi w funkcji liczby klientów i czasu obsługi została zaproponowana aproksymacja poszczególnych stanów wydajności za pomocą ograniczeń asymptotycznych

W pracy przedstawiono w postaci wykresów zależność przyrostu czasu odpowiedzi systemu w funkcji czasu obsługi klienta i liczby klientów. Zmiana przyrostu odpowiada drugiej pochodnej funkcji ciągłej. Maksimum otrzymanego wykresu pokrywa się z definicją zrównoważenia, jednak nie jest z nim tożsamy. Za pomocą warunku, gdzie zmiana przyrostu czasu odpowiedzi jest większa od zera otrzymuje się obszar, a nie tylko punkt zrównoważenia.

Liczbę klientów na badanych stanowiskach można odnieść do wykorzystania liczby klientów otwartego stanowiska obsługi będącego w stanie równowagi. Obserwację liczby klientów N na wybranym stanowisku wykorzystuje się jako parametr obciążenia w systemach operacyjnych. Dla systemu Linux zaimplementowano możliwość obserwacji parametru nazywanego obciążeniem (ang. *load average*) czyli średniej liczby procesów, jakie znajdują się w kolejce. Z przeprowadzonej analizy wynika, że średni czas odpowiedzi systemu wyznaczony metodą MVA jest znacznie bliższy ograniczeniom dolnym niż ograniczeniom górnym. Jeżeli czas odpowiedzi systemu przybliżony zostanie dolnym ograniczeniem asymptotycznym, to różnica pomiędzy tym ograniczeniem a średnią wartością, wyznaczoną z obliczeń modelu, będzie błędem takiej przybliżonej aproksymacji. Dla zilustrowania tego błędu zostały pokazane wykresy różnicy pomiędzy średnim czasem odpowiedzi systemu a dolnym ograniczeniem asymptotycznym.

5. Wnioski

Mowa dla człowieka jest środkiem budowania dialogu, który wymaga rozumienia przekazywanej informacji. Od strony sprzętowej systemy komputerowe pozwalające na generację komunikatów głosowych są dostępne technologicznie już od kilkudziesięciu lat. Pomimo rozbudowanych możliwości sprzętowych okazuje się, że z punktu widzenia oprogramowania wprowadzenie mowy do systemów komputerowych, tak jak rozumie się to potocznie, jest bardzo trudnym zadaniem.

System o sztywnych modułach dialogowych generowanych za pomocą technik TTS lub predefiniowanych komunikatów został w pracy nazwany Systemem Odpowiedzi Fonicznej. Natomiast interakcyjnym systemem wykorzystującym mowę (ang. *IVR*) zaproponowano nazywać systemy, które poszczególnym elementom prowadzonego dialogu za pomocą urządzeń fonicznych nadają znaczenie semantyczne.

W rozprawie zaproponowano wprowadzenie pojęcia tekstu fonicznego. Stopień skomplikowania gramatyki języka polskiego powoduje, że problemu translacji na tekst foniczny danych pochodzących z systemu komputerowego nie można rozwiązać poprzez proste nagranie komunikatów, bądź bezpośrednie zastosowanie metod TTS. Upowszechnienie Systemów Odpowiedzi Fonicznej wymaga połączenia wiedzy językoznawców z zespołami informatyków w celu opracowania bibliotek do budowy i translacji wypowiedzi na tekst foniczny. W pracy zastosowano analizę syntaktyczną dla opracowania algorytmu translacji na tekst foniczny wybranych komunikatów poczty głosowej na podstawie predefiniowanych słowników. Do opracowanego schematu podano klasyfikacje i algorytmy.

Cechą charakterystyczną aplikacji fonicznej jest rozdzielenie algorytmów przetwarzania kodu programu od budowy interfejsu z użytkownikiem. Taka architektura aplikacji jest naturalna dla mowy, gdyż budowa interfejsu z użytkownikiem wymaga użycia innych narzędzi niż środowisko tworzenia programów. Rozdzielenie kodu programu od zastosowanego interfejsu z użytkownikiem powoduje, że zamiast rozwoju odrębnych metod i narzędzi do tworzenia aplikacji na graficzny, głosowy czy multimedialny interfejs można zaprojektować wspólny kod i "rzutowanie" aplikacji na odpowiedni typ dialogu.

Dla wspomaganie budowy aplikacji wykorzystujących mowę od późnych lat 90-tych na świecie prowadzone są prace nad przejściem z rozwiązań firmowych (komercyjnych) na standardy otwarte.

W ramach rozprawy zbudowano własny System Odpowiedzi Fonicznej, którego model programowania, oparty o zdarzenia i Maszynę Stanów, można było zrealizować dla sprzętu dostępnego już od kilkunastu lat.

Analiza wydajności [113] jest bardzo dobrym testem detekcyjnym w trakcie procesu tworzenia i uruchamiania oprogramowania - szczególnie dla systemów pracujących w trybie ciągłym. Monitorowanie i badanie wydajności jest podstawowym testem stosowanym przez działy kontroli jakości w firmach tworzących systemy internetowe, na przykład Go-

ogole [27]. Postęp w zastosowaniach metod oceny wydajności systemów komputerowych dla działów kontroli jakości polega na przekształceniu testów detekcyjnych w metody diagnostyczne.

Ilościowym kryterium oceny wydajności systemu fonicznego jest czas odpowiedzi R wyznaczony

$$\bar{R} + \bar{Z} = \frac{\bar{N}}{X} \quad (1)$$

gdzie:

\bar{Z} średni czas generacji komunikatu głosowego,

\bar{N} średnia liczba pracujących kanałów w systemie, X przepustowość systemu.

Do wyznaczenia kryterium ilościowego systemu można zastosować podejście operacyjne, to znaczy pomiar może być przeprowadzany w wybranym czasie obserwacji. Niepotrzebne jest przyjmowanie dodatkowych założeń co do opisu określonym typem rozkładu wielkości losowej czasów obsługi zadań, czy czasów pomiędzy nadejściem zadań, a dodatkowo obserwowalne procesy nie muszą być stacjonarne.

Zaprojektowanie i uruchomienie własnego Systemu Odpowiedzi Fonicznej pozwoliło na wszechstronną analizę architektury. Na podstawie implementacji zbudowano model systemu. Po opracowaniu modelu ponownie zmodyfikowano kod źródłowy i treść zapisów do dziennika tak, aby dostarczał ilościowe dane niezbędne do wprowadzenia na wejście modelu. W ten sposób zapewniono pełną kompatybilność pomiędzy dziennikiem i modelem systemu.

Założono, że danymi wejściowymi do modelu będzie wprowadzenie rozkładów funkcji gęstości prawdopodobieństwa bądź wartości liczbowych (wartości średnie, maksymalne).

Wykazano, że pozyskanie danych o czasach wykonania operacji z dziennika obarczone jest błędem metody i błędem stochastycznym. Okazało się, że w wyniku pomiarów otrzymano dane niestacjonarne. Rzeczywisty system był oparty na kartach dwukanałowych. Pierwotnie model miał za zadanie odpowiedzieć jak system będzie się zachowywał dla większej ilości kanałów, by określić wymaganie projektowe dla Systemu Odpowiedzi Fonicznej jako słabo uwarunkowanego systemu czasu rzeczywistego.

Proces walidacji domyka się i jest zakończony sukcesem, gdy model odpowiedzi zgadza się z wielkościami otrzymanymi w wyniku pomiarów rzeczywistego systemu. Niestety w realnym systemie komputerowym realizacja tej samej operacji nie przebiega zawsze tak samo. Przyjęcie danych z jednokanałowej realizacji daje możliwość podjęcia obliczeń, jednak otrzymane wyniki opisują w sposób przybliżony realizację wielokanałową i nie zawsze będą nadawały się do zamknięcia procesu walidacji dla wielu kanałów.

Założenie, że dziennik jest źródłem danych do modelu jest założeniem praktycznym. Zamiast korzystać z dziennika można budować specjalne procedury pomiaru, jednak nie są one powszechnie stosowane i nie występują w systemach, w których badacz nie ma dostępu do kodu źródłowego.

W pracy nie udało się pozyskać z wystarczającą dokładnością danych, tak by móc dokonać walidacji modelu dla większej liczby kanałów, pomimo pełnego dostępu do kodu źródłowego, możliwości wyprowadzenia informacji zaprojektowanej dla akwizycji danych do modelu kolejkowego i uruchomieniu badanego systemu na jednozadaniowym systemie operacyjnym.

Przeprowadzone badania pokazują, że pomiary czasów wykonania prostych operacji, które przed przystąpieniem do eksperymentu zgodnie z oczekiwaniem miały mieć charakter wartości stałych, mają dużo bardziej złożony opis statystyczny. Zamiast stałej wartości, funkcja zapisu do dziennika charakteryzuje się dwoma maksimumami. Interpretacja fizyczna dyskretnej funkcji gęstości prawdopodobieństwa czasu odpowiedzi o dwóch maksimach opisana została w publikacji [180]. W czasie obserwacji systemu komputerowego występuje wiele zjawisk, które powodują, że pozyskanie wiarygodnych danych o wydajności systemu jest bardzo złożone. Do zjawisk takich należą: buforowanie, realizacja wielu stanów w czasie obserwacji, niestacjonarność wykonywania programu.

Buforowanie jest powszechnie stosowaną techniką przyspieszania wykonywania operacji. Jednak z punktu widzenia pomiarów buforowanie powoduje, że czasy wykonywania tych samych operacji różnią się od siebie w zależności od realizacji, gdyż zależą od wewnętrznego stanu bufora.

Obserwowalne elementy mogą pracować w różnych trybach w tym samym okresie obserwacji, co z praktycznego punktu widzenia jest szczególnie widoczne przy porównaniu wartości średnich pomiarów z wartościami maksymalnymi. Dla przykładu, dysk poza operacją zapisu, która zależy od położenia głowicy magnetycznej, może wykonać również asynchronicznie do pomiaru operację kalibracji położenia głowicy.

Dokonano analizy niestacjonarności, której powodem jest wprowadzanie stanów oczekiwania. Problemy te wymagają rozwiązania poprzez poszukiwanie nowych form wieloparametrowego opisu wydajności współczesnych systemów komputerowych (na przykład wykorzystanie przestrzeni ortogonalnych [11]).

Okazuje się, że wykorzystanie modelowania do analizy systemów komputerowych jest umiejętnością, którą zdobywa się poprzez doświadczenie [97]. Nie wystarcza tu tylko wiedza teoretyczna. Współczesny system operacyjny czy procesor jest dużo bardziej skomplikowany niż używane w modelowaniu stanowisko obsługi. Odpowiednie pominięcie szczegółów, czyli przedstawienia ogólnych zależności charakterystyk i ograniczeń, jest kluczem do sukcesu.

Wiele zależy od intuicyjnego przyjęcia założeń dotyczących:

- uproszczeń;
- adekwatności pomiarów;
- prostoty analizy weryfikacji.

Model systemu kolejkowego będąc pewną abstrakcją i przy odpowiednio dobranych założeniach może, ale nie musi, odwzorowywać podstawowe reguły działania modelowanego systemu, pomijając nieistotne szczegóły.

Działanie systemów komputerowych jest zdeterminowane przez oprogramowanie. Oczywiście w programie mogą znaleźć się błędy, wtedy może ono nie być takie jak planował je programista, jednak nie zmienia to faktu, że działanie systemu komputerowego jest całkowicie zdeterminowane. Odwołując się do doświadczeń fizycznych jesteśmy przyzwyczajeni, że zjawiska, które są zdeterminowane można opisać ścisłymi zależnościami matematycznymi (na przykład proces rozładowania kondensatora w obwodzie elektrycznym). Informatyk bazując na doświadczeniu programistycznym, dodatkowo wspomagany przez dyscypliny pokrewne (np. elektronika, automatyka) naturalnie oczekuje opisanego funkcjonowania, w tym również wydajności systemu komputerowego za pomocą ściśle deterministycznych zależności.

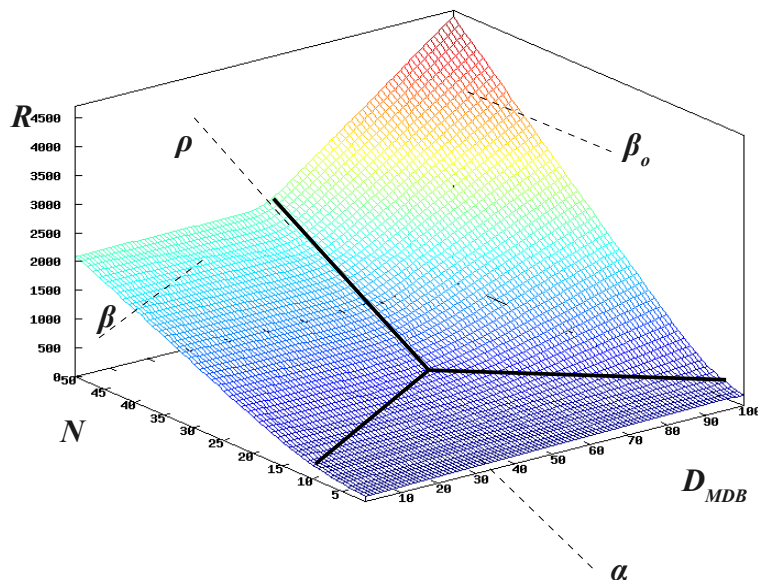
W literaturze [16] przyjmuje się, że praktyczne rozstrzygnięcie o zdeterminowanym bądź losowym charakterze sygnału dokonuje się rozpatrując możliwość lub niemożliwość odtworzenia go w danych warunkach. Sygnał jest deterministyczny, jeżeli otrzymuje się te same wyniki dla wielokrotnie powtarzanego doświadczenia. Jeżeli w tych samych warunkach powtarzane doświadczenie doprowadza do różnych wyników, to przyjmuje się, że natura sygnału jest losowa.

Budując system komputerowy wiadomo, że nie ma on natury czysto losowej. Jeżeli wyniki obserwowalnych doświadczeń nie są powtarzalne, to zawsze można postawić hipotezę, że niemożność opisanego systemu komputerowego za pomocą ścisłych zależności deterministycznych wynika z za mało dokładnego poznania praw.

Postawiony w rozprawie problem oceny systemu komputerowego, który składa się z niezależnych elementów x i y , gdzie wyizolowanie podsystemu y jest niemożliwe bądź niemierzalne, jest znacznie szerszy niż przedstawiona w pracy korekta błędu zapisu do dziennika i często występuje w zagadnieniach oceny wydajności systemów komputerowych.

Jak okazuje się, na podstawie praktycznych obserwacji, krytycznym elementem dla rozwiązania tego problemu jest znalezienie odpowiedniej metody rozplotu, która jest odporna na błędy powstające w trakcie pomiaru.

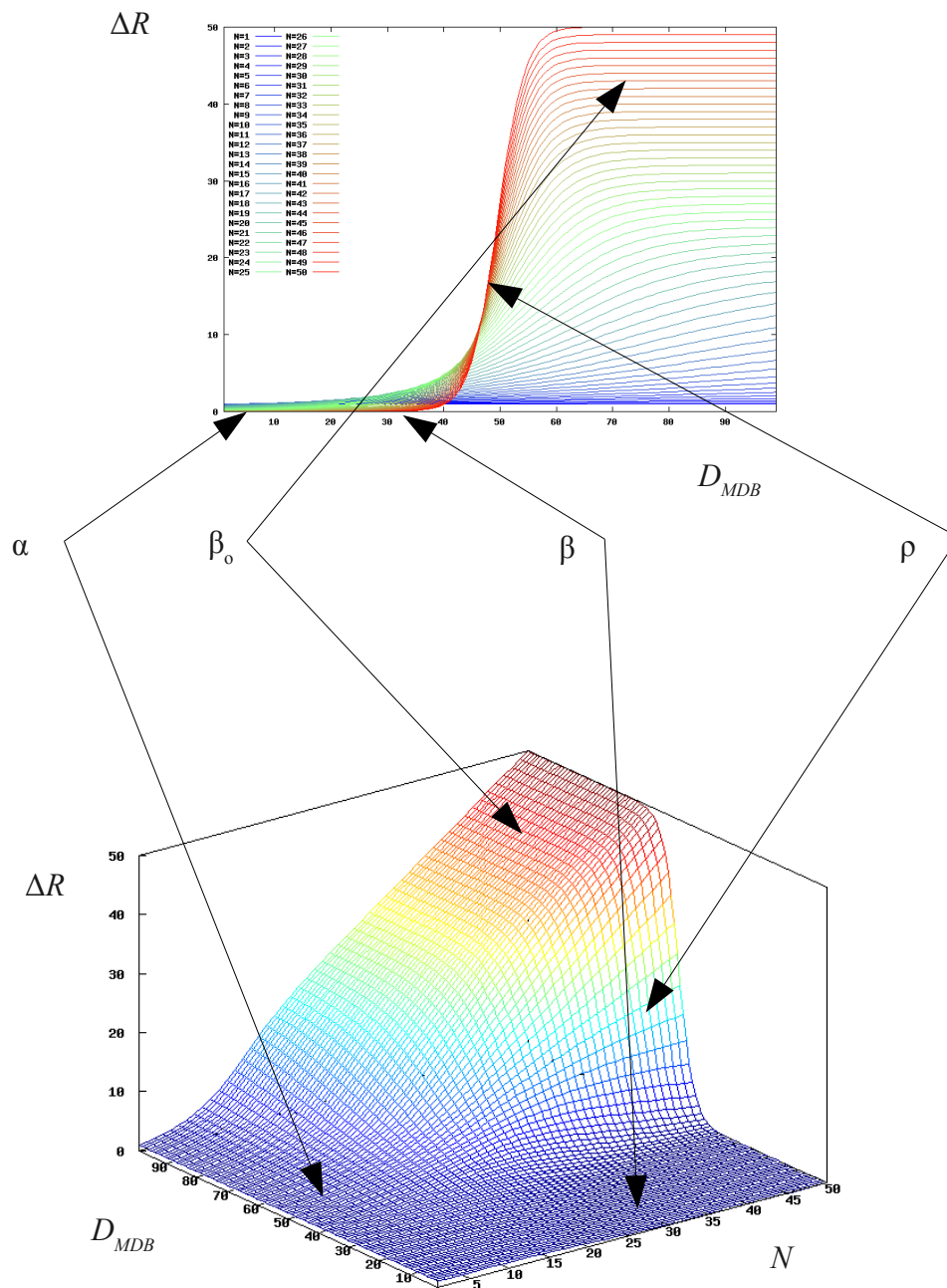
Jeżeli w wyniku pomiarów czasów odpowiedzi elementów z i x otrzymamy funkcje gęstości prawdopodobieństwa czasów odpowiedzi p_z i p_x , a system z powstał jako złożenie x i y , to w obliczeniach p_y błąd pomiarowy nie spleciony z funkcją gęstości prawdopodobieństwa p_x powoduje bardzo dużą niestabilność obliczeniową. Wzmocnienie błędów pomiarów obserwowane jest dla rozplotów wykonanych z definicji, na podstawie dzielenia wielomianów lub przez transformatę. Największą odporność na błędy pomiarowe wykazała metoda rozplotu na pomocą najmniejszych kwadratów.



Rys. 1. Czas odpowiedzi systemu R z zaznaczonymi płaszczyznami stanów wydajności α , β , β_0 , ρ w funkcji liczby kanałów fonicznych N i czasu obsługi stanowiska odpowiedzialnego za przygotowanie komunikatów fonicznych D_{MDB}

W rozprawie wyznaczono przestrzeń parametrów wejściowych modelu i dla każdego punktu w tej przestrzeni rozwiązano model obliczając wartości, parametrów wyjściowych. Dla przestrzeni rozwiązań modelu wprowadzono definicje stanów wydajności systemu.

Wyróżniono stany pracy systemu jako: nieobciążony α , maksymalnie obciążony β , maksymalnie obciążony przez wybrane do obserwacji stanowisko β_0 i zrównoważenia ρ (rys. 1). Przejście pomiędzy stanami nie jest ostre, dlatego kontynuacja badań i wyznaczenie granic pomiędzy stanami jest istotne z punktu widzenia identyfikacji i pomiarów rzeczywistych systemów.



Rys. 2. Przyrost czasu odpowiedzi systemu ΔR z zaznaczonymi płaszczyznami stanów wydajności α , β , β_0 , ρ w funkcji liczby kanałów fonicznych N i czasu obsługi stanowiska odpowiedzialnego za przygotowanie komunikatów fonicznych D_{MDB}

Wyniki otrzymywane z rozwiązania modelu pokazują, że dolne ograniczenia asymptotyczne R_{min} lepiej szacują działanie systemu niż górne ograniczenia asymptotyczne R_{max} . Poza stanem zrównoważenia ograniczenie asymptotyczne R_{min} jest dobrym przybliżeniem działania systemu. Największy błąd oszacowania czasu odpowiedzi systemu za pomocą ograniczeń asymptotycznych R_{min} występuje w stanie nieobciążenia α i w stanie zrównowa-

żenia ρ . Dla stanów maksymalnego wykorzystania β i β_0 , błąd oszacowania jest bliski zeru.

W rozprawie wykazano, że zamiast obserwowania takich podstawowych parametrów jak przepustowość, czy czas odpowiedzi, łatwiej jest identyfikować stany wydajności systemu rejestrując (rys. 2) przyrost czasu odpowiedzi ΔR w funkcji zmian czasu obsługi zadania dla wybranego stanowiska (podsystemu).

W rozprawie nie wyczerpano pierwotnego tematu badawczego. Wymagane jest również pogłębienie współpracy polskich lingwistów z zespołami informatyków w celu upowszechnienia algorytmów translacji wypowiedzi na tekst foniczny. Z punktu widzenia analizy wydajności potrzeba jest opracowania specjalizowanych metod pomiarowych. Tu otwiera się wiele obszarów badawczych wśród których należy wyróżnić: metody poszukiwania zależności liniowych w analizie wydajności systemów komputerowych, metody klasyfikacji różnych stanów, trybów pracy w czasie obserwacji systemu, formy wieloparametrycznego opisu wydajności oraz analizy źródeł i metody wykrywania zjawisk powodujących niestacjonarne zachowanie się systemów.

6. Bibliografia

- [1] Syed **Ahson**, Mohammad Ilyas, SIP handbook: services, technologies, and security of Session Initiation Protocol, CRC Press, Boca Raton, USA, 2008
- [2] Michael **Amundsen**, Mapi, Sapi, and Tapi: developer's guide, Sams Publishing, Indianapolis, USA, 1996
- [3] Gregory R. **Andrews**, Foundations of multithreaded, parallel, and distributed programming, Addison-Wesley, University of Arizona, Tucson, USA, 2000
- [4] Howard **Anton**, Chris Rorres, Elementary Linear Algebra: Applications Version, John Wiley and Sons, Philadelphia, USA, 2005
- [5] Tarmo **Anttalainen**, Introduction to telecommunications network engineering, Artech House, Norwood, USA, 2003
- [6] Søren **Asmussen**, Applied probability and queues, Springer-Verlag, New York, USA, 2003
- [7] Reza **B'Far**, Mobile computing principles: designing and developing mobile applications with UML and XML, Cambridge University Press, Cambridge, UK, 2005
- [8] Paolo **Baggia**, Speech Technologies and Platforms Present and Future Evolution, Google Tech Talks, <http://research.google.com/video.html>, March 3th, 2008
- [9] Alicja **Balcerak**, Walidacja Modeli Symulacyjnych - Źródła Postaw Badawczych, Prace Naukowe Instytutu Organizacji Zarządzania Politechniki Wrocławskiej, Wyd. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 27-44, 2003
- [10] Osman **Balci**, Verification validation and accreditation of simulation models, Winter Simulation Conference, Proceedings of the 29th conference on Winter simulation, Atlanta, USA, 135 - 141, 1997

-
- [11] Stefan **Banach**, Theory of linear operations, Elsevier, Amsterdam, 1987
- [12] Regis J. **Bates**, Broadband telecommunications handbook, McGraw-Hill Professional, Phoenix, USA, 2002
- [13] Regis J. **Bates**, Donald W. Gregory, Voice and data communications handbook, McGraw-Hill Professional, New York, USA, 2007
- [14] Michael Thomas **Bayer**, Computer Telephony Demystified: Putting CTI, Media Services, and IP Telephony to Work, McGraw-Hill Professional, New York, USA, 2000
- [15] John **Bellamy**, Digital Telephony, John Wiley & Sons, Inc., New York, USA, 2000
- [16] J. S. **Bentat**, A.G.Piersol, Metody analizy i pomiaru sygnałów losowych, Państwowe Wydawnictwo Naukowe, Warszawa, 1976
- [17] F. J. **Beutler**, Mean sojourn times in Markov queueing networks: Little's formula revisited, IEEE Transactions on Information Theory 29, 2, March, 233-241, 1983
- [18] U. Narayan **Bhat**, An Introduction to Queueing, Theory Modeling and Analysis in Applications, Springer Science+Business Media, Boston, USA, 2008
- [19] Arkadiusz **Biernacki**, Modelowania transmisji VoIP w bramie multimedialnej, Rozprawa doktorska, Politechnika Śląska Wydział Automatyki, Elektroniki i Informatyki, Instytut Informatyki, Gliwice, 2007
- [20] Kurt **Bittner**, Ian Spence, Managing iterative software development projects, Pearson Education, Inc., Boston, USA, 2007
- [21] Goranka **Bjedov**, Using open source tools for performance testing, Google London Test Automation Conference (LTAC) Google Tech Talks, <http://video.google.com/videoplay?docid=-6891978643577501895#>, September 8th, 2006
- [22] Richard **Blum**, Network Performance Open Source Toolkit: Using Netperf, Tcptrace, Nist Net, and Ssfnet, John Wiley and Sons, Indianapolis, USA, 2003
- [23] Leszek **Borzemski**, Marek Rodkiewicz, Gabriel Starczewski, Miary wydajności połączeń w systemach SOA, Interdyscyplinarność badań naukowych 2010 [Dokument elektroniczny] : praca zbiorowa pod red. Jarosława Szreka, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2010
- [24] Daniel Pierre **Bovet**, Marco Cesati, Understanding the Linux Kernel, O'Reilly Media, Inc., Sebastopol, USA, 2005
- [25] Dave **Burke**, Speech processing for IP networks: Media Resource Control Protocol (MRCP), John Wiley and Sons, West Sussex, UK, 2007
- [26] David R. **Butenhof**, Programming with POSIX threads, Addison-Wesley, Boston, USA, 1997
- [27] Microsoft, Microsoft C/C++ **Class** Libraries Users Guide for MS-DOS and Windows Operating Systems, Microsoft Corporation, Irland,
- [28] Microsoft, Microsoft C/C++ **Environment** and Toos for MS-DOS and Windows Operating Systems, Microsoft Corporation, Irland,
- [29] Microsoft, Microsoft C/C++ Comprehensive Index and Errors Reference for MS-DOS and Windows Operating Systems, Microsoft Corporation, Irland, 1991
- [30] Microsoft Corporation, Microsoft C/C++ **Programming** Techniques for MS-DOS and Windows Operating Systems, Microsoft Corporation, Irland, 1991
-

-
- [31] Microsoft, Microsoft C/C++ **Run Time** for MS-DOS and Windows Operating Systems, Microsoft Corporation, Irland, 1991
- [32] Microsoft, Microsoft C/C++ **Tutorial** for MS-DOS and Windows Operating Systems, Microsoft Corporation, Irland, 1991
- [33] Carlton **Carden**, Understanding Computer Telephony: How to Voice Enable Databases from PCs to Lans to Mainframes, Telecom Books, New York, USA, 1997
- [34] Jeff **Carter**, Unified Communications 100 Success Secrets Discover the Best Way to Unify Your Enterprise, Covers Unified Messaging, Systems, Solutions, Software and Services, Emereo Pty Ltd, London, UK, 2008
- [35] W3C, Voice Browser Call Control **CCXML** Version 1.0, W3C Working Draft, <http://www.w3.org/TR/ccxml/>, 19 January 2007
- [36] Michael H. **Cohen**, Michael Harris Cohen, James P. Giangola, Jennifer Balogh, Voice user interface design, Addison-Wesley, Boston, USA, 2004
- [37] Thomas H. **Cormen**, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to algorithms, MIT Press, Cambridge, UK, 2001
- [38] D. R. **Cox**, P. A. W. Lewis, The statistical analysis of series of events, Methuen's Monographs on Applied Probability and Statistics (John Wiley), London, 1966
- [39] R.G. **Coyle**, Management System Dynamics, Wiley, London, New York, 1977
- [40] David J. **Cutler**, Numerical deconvolution by least squares: Use of prescribed input functions, Journal of Pharmacokinetics and Pharmacodynamics, Vol. 6, No. 3 / June, 227-241, 1978
- [41] Tadeusz **Czachórski**, Modele Kolejowe Systemów Komputerowych, Wydawnictwo Politechniki Śląskiej, Gliwice, 1999
- [42] Andrzej **Czyżewski**, Dźwięk cyfrowy, Akademicka Oficyna Wydawnicza EXIT, Gdańsk, 2001
- [43] Gabriele **D'Antona**, Alessandro Ferrero, Digital signal processing for measurement systems: theory and applications, Springer, Milano, Italy, 2006
- [44] Ramón López Cózar **Delgado**, Masahiro Araki, Spoken, multilingual and multimodal dialogue systems: development and assessment, John Wiley and Sons, West Sussex, UK, 2005
- [45] **Dialogic**, Dialogic Corporation, http://en.wikipedia.org/wiki/Dialogic_Corporation
- [46] **Dialogic** Corporation, Dialogic D/42 Series Boards User's Guide, Dialogic Corporation, New Jersey, USA, 2008
- [47] **Dialogic** Corporation, Dialogic Products & Service Guide , Dialogic Corporation, New Jersey, USA, 1995
- [48] **Dialogic** Corporation, System Release Software Instalation Reference , Dialogic Corporation, New Jersey, USA, 1994
- [49] **Dialogic** Corporation, Voice Hardware Reference, Dialogic Corporation, New Jersey, USA, 1994
- [50] **Dillman**, D. A.; Phelps, G., Tortora, R. D., Swift, K., Kohrell, J., Berck, J., Messer, B. L., Response rate and measurement differences in mixed-mode surveys using mail, telephone, interactive voice response (IVR) and the Internet, *Social Science Research*, 38, 1, 1-18, 2009
- [51] Joanna **Domańska**, Procesy Markowa w modelowaniu natężenia ruchu w sieciach komputerowych, Rozprawa doktorska, Instytut Informatyki Teoretycznej i Stosowanej Polskiej Akademii Nauk, Gliwice, 2003
- [52] Pál **Dömösi**, Chrystopher L. Nehaniv, Algebraic theory of automata
-

- networks: an introduction, SIAM, Philadelphia, USA, 2004
- [53] Thierry **Dutoit**, An Introduction to Text-To-Speech Synthesis, Kluwer Academic Publishers, Dordrecht, Netherlands, 2001
- [54] John W. **Eaton**, David Bateman, Søren Hauberg, GNU Octave Manual Version 3, Network Theory Ltd., UK, 2008
- [55] Bob **Edgar**, PC-Based Voice Processing. How to Design, Build and Program Systems Using Industry-Standard Dialogic Hardware, Flatiron Publishing Inc., 12 West 21 Street, New York, USA, 1994
- [56] Margulies **Edwin**, Client Server Computer Telephony, Flatiron Publishing, New York, USA, 1997
- [57] P. G. **Emma**, Understanding some simple processor performance limits, IBM J. Res. Develop., Vol. 41, No. 3, 215-232, 1997
- [58] Michał **Ferenc**, Stanisław Wideł, Opracowanie programu układu mikroprocesorowego do sterowania obiektów energetycznych na przykładzie średnioobrotowego silnika wysokoprężnego, Międzyresortowy Program Badawczo Rozwojowy Praca badawcza NB-400/RAU-2/RME-2/88 zad 30.3, Gliwice, 1989
- [59] Michał **Ferenc**, Stanisław Wideł, Michał Fiutkowski, Zasady doboru charakterystyki dynamicznej regulatora prędkości obrotowej średnio-obrotowego silnika wysokoprężnego napędzającego prądnicę, Silniki Spalinowe, 3-4, 108-109, 15-19, 1990
- [60] Michał **Ferenc**, Jan Sochański, Stanisław Wideł, Komputerowa symulacja układu sterowania procesem odzysku ciepła ze ścieków w przemyśle włókienniczym, Zeszyty Naukowe Politechniki Śląskiej, Zeszyt 114, Energetyka, 61-87, 1993
- [61] Domenico **Ferrari**, Songnian Zhou, An Empirical Investigation of Load Indices For Load Balancing Applications, Proc. Performance '87, the 12th Int'l Symp. On Computer Performance Modeling, Measurement, and Evaluation, North Holland Publishers, Amsterdam, Netherlands, 515-528, 1988
- [62] M. **Fiuk**, R. Hausman, S. Dalal, S. Wideł, Developing a toolkit for extracting and maintaining data from WEB pages, Studia Informatica, IX Konferencja Sieci Komputerowe, Vol. 23, No. 2B (49), 83-93, 2002
- [63] Marek **Fiuk**, Stanisław Wideł, Jan Machniewski, Systemy Informatyczne z Ograniczeniami Czasowymi: Analiza działania i modelowanie sieci UPnP, Wydawnictwa Komunikacji i Łączności, Warszawa, 2006
- [64] J.W. **Forrester**, Senge P.M., Tests for building confidence in System Dynamics models, System Dynamics, TIMS Studies in the Management Sciences 14, North-Holland, New York, 209-228, 1980
- [65] Roger L. **Freeman**, Fundamentals of telecommunications, John Wiley and Sons, Hoboken, New Jersey, USA, 2005
- [66] D. H. **Fremlin**, Measure theory, Torres Fremlin, Colchester, UK, 2003
- [67] Aileen Frisch, Essential system administration, O'Reilly Media, Inc., Sebastopol, USA, 2003
- [68] Sadaoki **Furui**, Digital speech processing, synthesis, and recognition, Marcel Dekker Inc., New York, USA, 2001
- [69] Daryle **Gardner**-Bonneau, Harry E. Blanchard, Human factors and voice interactive systems, Signals and communication technology, Kluwer international series in engineering and computer science, Springer Science + Business Media, New York, USA, 2008
- [70] Steve **Gladstone**, Testing computer telephony systems and networks,

- Telecom Books, New York, USA, 1996
- [71] P. W. **Glynn**, Whitt W., Extensions of the queueing relations $L = \lambda W$ and $H = \lambda G$, *Operations Research*, 37, 1989, 634–644, 1989
- [72] Krzysztof **Grochla**, Dobór Parametrów Mechanizmów Kształtowania Ruchu Protokołu Internetowego IP, Rozprawa doktorska, Instytut Informatyki Teoretycznej i Stosowanej Polskiej Akademii Nauk, Gliwice, 2006
- [73] Jonathan L. **Gross**, Jay Yellen, Handbook of graph theory, Discrete mathematics and its applications, CRC Press, USA, 2004
- [74] **Andrzej Grzywak**, Stanisław Wideł, Multimedialny rozproszony system zarządzania Urzędem Miasta Pszów, *Komputerowe Systemy Wielodostępne*, Bydgoszcz, 1998
- [75] **Andrzej Białas**, **Andrzej Grzywak**, Wojciech Mielczarek, Michał Rozmus, Stanisław Wideł, Lokalne Sieci Komputerowe Mikrokomputerów Personalnych, Wydawnictwo Politechniki Śląskiej, Gliwice, 1999
- [76] Neil J. **Gunther**, Analyzing Computer System Performance with Perl PDQ, Springer Verlag, Heidelberg, Germany, 2005
- [77] Eitan **Gurari**, An introduction to the theory of computation, Computer Science Press, Ohio, USA, 1989
- [78] Yuri **Gurevich**, Philipp W. Kutter, Martin Odersky, Lothar Thiele (Eds.), Abstract state machines: theory and applications, international workshop : proceedings, ASM 2000, Monte Verità, Switzerland, March 19-24, 2000
- [79] Marc **Hamilton**, Software development: building reliable systems Enterprise computing series, Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999
- [80] Randy Allen **Harris**, Voice interaction design: crafting the new conversational speech systems, Elsevier, San Francisco, USA, 2005
- [81] **Adam Holec**, David Cuddihy, Definitive VoiceXML, The Charles F. Goldfarb definitive XML series, Prentice Hall, Upper Saddle River, New Jersey, USA, 2003
- [82] J. N. **Holmes**, Wendy J. Holmes, Speech synthesis and recognition, Aspects of information technology, Taylor and Francis, London, UK, CRC Press, 2001
- [83] Grzegorz **Hryń**, Badanie możliwości dostosowania charakterystyki serwera WWW do potrzeb Klienta, Rozprawa doktorska, Politechnika Śląska Wydział Automatyki, Elektroniki i Informatyki, Gliwice, 2005
- [84] Vinay K. **Ingle**, John G. Proakis, Digital Signaling Processing Using MATLAB, PWS BookWare companion series, Brooks/Cole, Northeastern University, Boston, USA, 2000
- [85] **IVO Software Sp. z o.o.**, IVO Software, <http://www.ivona.com/index.php>
- [86] Frederick **Jelinek**, Statistical methods for speech recognition, Language, speech, and communication, MIT Press, Massachusetts Institute of Technology, Cambridge, USA, 2001
- [87] Stanisław **Jędrus**, Modelowanie natężenie ruchu pakietów w sieciach komputerowych z wykorzystaniem miar multifrakalnych, Rozprawa doktorska, Instytut Informatyki Teoretycznej i Stosowanej PAN, Gliwice, 1999
- [88] Sandra K. **Johnson**, Gerrit Huizenga, Badari Pulavarty, Performance Tuning for Linux Servers, IBM Press, Upper Saddle River, NJ, USA, 2008
- [89] Jean-Claude **Junqua**, Robust speech recognition in embedded systems and PC applications, Kluwer Academic Publisher, Dordrecht, Netherlands, 2000
- [90] Steven T. **Karris**, Signals and systems with MATLAB applications, Orchard

- Publications, Fremont, USA, 2003
- [91] Wolfgang von **Kempelen's** speaking machine and its successors, <http://www2.ling.su.se/staff/hartmut/kemplne.htm>
- [92] Brian W. **Kernighan**, Dennis M. Ritchie, Jezyk C, Wydawnictwa Naukowo-Techniczne, Warszawa, 1988
- [93] Dennis H. **Klatt**, Review of Text-to-Speech Conversion for English, Journal of the Acoustical Society of America 82, 82(3), <http://www.cslu.ogi.edu/tts/research/history/>, 737-793, 1987
- [94] Marek **Kurowicz**, Stanisław Wideł, System monitorowania stanu dynamicznego maszyn wirnikowych, Problemy i Innowacje w Remontach Energetycznych, VII, Konferencja Naukowo Techniczna, 155-167, 2005
- [95] James A. **Larson**, VoiceXML: Introduction to Developing Speech Applications, Prentice Hall, Upper Saddle River, New Jersey, USA, 2003
- [96] S. S. **Lavenberg**, M. Reiser, Stationary State Probabilities at Arrival Instants for Closed Queueing Networks with Multiple Types of Customers, Journal of Applied Probability, Vol. 17, No. 4 Dec, 1048-1061, 1980
- [97] Edward D. **Lazowska**, John Zahorjan, G. Scott Graham, Kenneth C. Sevcik, Quantitative System Performance, Computer System Analysis Using Queueing Network Models, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, USA, 1984
- [98] Peter A. W. **Lewis**, Stochastic point processes: statistical analysis, theory, and applications, Wiley-Interscience, New York, USA, 1972
- [99] David J. **Lilja**, Measuring Computer Performance: A Practitioner's Guide, Cambridge University Press, Cambridge, USA, 2005
- [100] **Matthew Liotine**, Mission-critical network planning, Wydawnictwo Komunikacji i Łączności, , 2003
- [101] Lester **Lipsky**, Queueing Theory: A Linear Algebraic Approach, Springer, Connecticut, USA, 2009
- [102] John D. C. **Little**, A proof for the queueing formula $L=\lambda W$, Operations Research, Vol.9, No.3, 383-387, 1961
- [103] Sergey Edward **Lyshevski**, Engineering and scientific computations using MATLAB, Wiley-IEEE, Hoboken, NJ, USA, 2003
- [104] Piotr **Majewski**, Syllable Based Language Model for Large Vocabulary Continuous Speech Recognition of Polish, Text, Speech and Dialogue, Vol. 5246/2008, Lecture Notes in Computer Science, 397-401, 2008
- [105] J. P. **Marques de Sá**, Applied statistics: using SPSS, Statistica, MATLAB, and R, Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, Germany, 2007
- [106] Robert Lee **Mason**, Richard F. Gunst, James L. Hess, Statistical Design Analysis of Experiments, Wiley, Hoboken, New Jersey, USA, 2003
- [107] Ignatius G. **Mattingly**, Speech Synthesis for Phonetic and Phonological Models, The Hague, Vol. 12, Thomas A. Sebeok, Editor, 2451-2487, 1974
- [108] Arthur **Mattuck**, Introduction to analysis, Prentice Hall, Cambridge, USA, 1998
- [109] James D. **McCabe**, Network Analysis, Architecture, and Design, Morgan Kaufmann, Burlington, USA, 2007
- [110] Michael **McTear**, Spoken dialogue technology: toward the conversational user interface, Springer-Verlag, London, UK, 2004
- [111] **Jim Van Meggelen**, Jared Smith, Leif Madsen, Asterisk: the future of telephony. Join the Open Source Revolution, O'Reilly Media, Inc., Sebastopol, USA, 2007

- [112] Daniel A. **Menascé**, Virgilio A. F. Almeida, Capacity planning for Web services: metrics, models, and methods, Prentice Hall, University of Michigan, Ann Arbor, USA, 2002
- [113] Daniel A. **Menascé**, Virgilio A. F. Almeida, Lawrence W. Dowdy, Larry Dowdy, Performance by design: computer capacity planning by example, Prentice Hall PTR, New Jersey, USA, 2004
- [114] D. L. **Mills**, Network Time Protocol (version 3): Specification, implementation, and analysis. IETF Network Working Group RFC 1305 Cited Jun 12, 2004, www.ietf.org/rfc/rfc1305.txt
- [115] Sanjit Kumar **Mitra**, Digital signal processing: a computer based approach, McGraw-Hill Higher Education, Santa Barbara, USA, 2005
- [116] Renato De Mori, Spoken dialogues with computers, Academic Press, San Diego, USA, 1998
- [117] Nathan J. **Muller**, Desktop encyclopedia of telecommunications, McGraw-Hill Professional, Telecommunications Series, New York, USA, 2002
- [118] Gian-Paolo D. **Musumeci**, Michael Kosta Loukides, System performance tuning, O'Reilly Media, Inc., Sebastopol, USA, 2002
- [119] Stochastic Language Models (**N-Gram**) Specification, W3C Working Draft 3 January 2001, <http://www.w3.org/TR/ngram-spec/>
- [120] Bradford **Nichols**, Dick Buttlar, Jacqueline Proulx Farre, Pthreads programming, O'Reilly Media, Inc., Sebastopol, USA, 1996
- [121] Mateusz **Nowak**, Dynamiczne zarządzanie realizacją procesów równoległych w środowisku rozproszonym, Rozprawa doktorska, Instytut Informatyki Teoretycznej i Stosowanej PAN, Gliwice, 2006
- [122] Daniel **Page**, A Practical Introduction to Computer Architecture, Springer, Bristol, UK, 2009
- [123] Athanasios **Papoulis**, S. Unnikrishna Pillai, Probability, random variables, and stochastic processes, McGraw-Hill, Stanford, USA, 2002
- [124] Derek **Partridge**, Engineering artificial intelligence software, Intellect Books, 1992, Oxford, 1992
- [125] David A. **Patterson**, John L. Hennessy, Computer organization and design: the hardware/software interface, Morgan Kaufmann, Burlington, USA, 2009
- [126] Kerstin Day **Peterson**, Business telecom systems: a guide to choosing the best technologies and services, Focal Press, New York, USA, 2000
- [127] Charles **Petzold**, Programowanie Windows, Read Me, Warszawa, 2009
- [128] W3C, Pronunciation Lexicon Specification (**PLS**) Version 1.0, W3C Recommendation 14 October 2008, <http://www.w3.org/TR/pronunciation-lexicon/>
- [129] Pod red. K **Polańskiego**, Słownik syntaktyczno-generatywny czasowników polskich, PAN. IJP, Zakład Narodowy im. Ossolińskich, Kraków, 1992
- [130] Shashi **Prasad**, Multithreading programming techniques, McGraw-Hill, University of Virginia, Charlottesville, USA, 1997
- [131] George E. **Priest**, Lynn Wallace, Voice mail: much more than an answering machine, Association for Information and Image Management, Silver Spring, USA, 1989
- [132] Zheng **Qin**, Jiankuan Xing, Xiang Zheng, Software architecture, Springer, Berlin, Heidelberg, New York, 2008
- [133] M. **Reiser**, H. Kobayashi, Mean Value Analysis of Closed Multichain Queueing Networks, ACM, Vol. 27, No.2, 313-322, 1980
- [134] **Rhetorex**, AccuCall Plus User Guide, Rhetorex, Inc., Campbell, USA, 1995
- [135] **Rhetorex**, AccuSpan User's Guide, Rhetorex, Inc., Campbell, USA, 1995

-
- [136] **Rhetorex**, Inc, API Reference for Windows 95 and Windows NT, Rhetorex, Inc., Los Gatos, USA, 1996
- [137] **Rhetorex**, Configuration Manual for DOS, Rhetorex, Inc., Campbell, USA, 1995
- [138] **Rhetorex**, Data Structures Reference Manual for DOS, Rhetorex, Inc., Campbell, USA, 1995
- [139] **Rhetorex**, Function Reference Manul for DOS, Rhetorex, Inc., Campbell, USA, 1995
- [140] **Rhetorex** , Programing Guide for DOS, Rhetorex, Inc., Campbell, USA, 1995
- [141] **Rhetorex**, RDSPTest User's Guide, Rhetorex, Inc., Campbell, USA, 1995
- [142] **Rhetorex**, RealCT User' Guide for Windows NT and Windows 95, Rhetorex, Inc., Campbell, USA, 1996
- [143] **Rhetorex**, Software and Firmware Instalation Manual for DOS, Rhetorex, Inc., Campbell, USA, 1995
- [144] **Rhetorex**, Vantage Hardware Manual, Rhetorex, Inc., Campbell, USA, 1995
- [145] Elaine **Rich**, Automata, computability and complexity: theory and applications, Prentice Hall, Upper Saddle River, NJ, USA, 2008
- [146] Kenneth H. **Rosen**, Discrete Mathematics and Its Applications, McGraw Hill Book, Holmdel, NJ, USA, 2006
- [147] Halina **Rychlicka**, Nina Suszczańska, Stanisław Wideł, Generacja wybranych komunikatów poczty głosowej w postaci tekstów fonicznych w języku polskim, *Studia Informatica*, VII Konferencja Sieci Komputerowe , Vol. 21, No. 39, 287-315, 2000
- [148] David **Salomon**, Data compression: the complete reference, Springer-Verlang, London, UK, 2007
- [149] Jan P. H. Van **Santen**, Progress in speech synthesis, Springer-Verlang, New York, USA, 1997
- [150] S. **Schlesinger**, Crosbie, R.E., Gagne, R.E., Innis, G.S., Lalwani, C.S., Loch, J., Sylvester, R.J., Wright, R.D., Kheir, N., Bartos D., Terminology for model credibility, *Simulation*, March 1979, Series 03, 103-104, 1979
- [151] W3C, State Chart XML (**SCXML**): State Machine Notation for Control Abstraction, W3C Working Draft 7 May 2009, <http://www.w3.org/TR/scxml/>
- [152] Andrew **Sears**, Julie A. Jacko, The human-computer interaction handbook: fundamentals, evolving technologies, and emerging applications Human factors and ergonomics, CRC Press, New York, USA, 2007
- [153] Chris **Sells**, Windows telephony programming: a developer's guide to TAPI, Addison-Wesley, Upper Saddle River, NJ, USA, 2002
- [154] Kenneth C. **Sevcik**, Isi Mitrani, The Distribution of Queuing Network States at Input and Output Instants, Performance of Computer Systems, Proceedings of the Third International Symposium on Modelling and Performance Evaluation of Computer Systems, Vienna, Austria, February 6-8, 358-371, 1979
- [155] Jeffrey R. **Shapiro**, Computer Telephony Strategies, IDG Books Worldwide, Foster City, USA, 1996
- [156] W3C, Semantic Interpretation for Speech Recognition (**SISR**) Version 1.0, W3C Recommendation 5 April 2007, <http://www.w3.org/TR/semantic-interpretation/>
- [157] Ian **Sommerville**, Software engineering, Pearson Education, Essex, UK, 2007
-

-
- [158] Charles E. **Spurgeon**, Ethernet: the definitive guide, O'Reilly Media, Sebastopol, USA, 2000
- [159] Andrew Hunt, Scott McGlashan, Speech Recognition Grammar Specification (**SRGS**) Version 1.0, W3C Recommendation 16 March 2004, <http://www.w3.org/TR/speech-grammar/>
- [160] W3C, Speech Synthesis Markup Language (**SSML**) Version 1.1, W3C Candidate Recommendation 27 August 2009, <http://www.w3.org/TR/speech-synthesis11/>
- [161] Speech Synthesis for Phonetic and Phonological Models, Speech Synthesis for Phonetic and Phonological Models (**SSSHP**), http://americanhistory.si.edu/archives/speechsynthesis/im_home.htm
- [162] **Constantine Stephanidis**, Universal Access in Human-computer Interaction: 4th International Conference on Universal Access in Human-Computer Interaction, Springer, Pekin, China, 2007
- [163] S. Jr. **Stidham**, A last word on $L = \lambda W$, Operations Research, 22, 1974, 417–421, 1974
- [164] Raik **Stolletz**, Approximation of the non-stationary $M(t)/M(t)/c(t)$ -queue using stationary queueing models: The stationary backlog-carryover approach, European Journal of Operational Research, Vol. 190, Issue 2, 478–493, 2008
- [165] Gilbert **Strang**, Linear Algebra and Its Applications, Thomson, Brooks/Cole, Belmont, USA, 2006
- [166] Joseph D. **Straubhaar**, Robert LaRose, Communications media in the information society, Wadsworth Pub. Co., Belmont, USA, 1997
- [167] Aleksander **Szlam**, Ken Thatcher, Predictive Dialing Fundamentals An Overview Predictive Dialing Technologies, their Applications and Usage, Flatiron Publishing Inc., New York, USA, 1996
- [168] Andrew S. **Tanenbaum**, Sieci Komputerowe, Helion, Gliwice, 2004
- [169] Mark **Tatham**, Katherine Morton, Developments in speech synthesis, John Wiley and Sons, West Sussex, UK, 2005
- [170] Ed **Tittel**, Dawn Rader, Computer telephony: automating home offices and small businesses, AP Professional, San Francisco, USA, 1996
- [171] W. **Tucker** i zespół, A Glossary of Modeling and Simulation Terms for Distributed Interactive Simulation, <http://www.tmpo.nima.mil/guides/Glossary/>
- [172] Dialogic, **Voice** Software Reference for MS-DOS Volume 1, Voice Features Guide, Dialogic Corporation, New Jersey, USA, 1994
- [173] Dialogic, **Voice** Software Reference for MS-DOS Volume 2, Voice Programmer's Guide, Dialogic Corporation, New Jersey, USA, 1994
- [174] W3C, **Voice** Extensible Markup Language (VoiceXML) 3.0, W3C Working Draft 25 August 2009, <http://www.w3.org/TR/voicexml30/>
- [175] W3C, **Voice** Browser Activity, <http://www.w3.org/Voice/>
- [176] Jim A. Larson, Intel Architecture Labs, Introduction and Overview of W3C Speech Interface Framework, W3C Working Draft 4 December 2000, <http://www.w3.org/TR/voice-intro/>
- [177] Ferdinand **Wagner**, Ruedi Schmuki, Thomas Wagner, Peter Wolstenholme, Modeling software with finite state machines: a practical approach, CRC Press, Boca Raton, USA, 2006
- [178] **Ted Wallingford**, Switching to VoIP, O'Reilly Media, Inc., Sebastopol, USA, 2005
- [179] Stefan **Węgrzyn**, Podstawy informatyki, Państwowe Wydawnictwo
-

- Naukowe, Warszawa, 1982
- [180] Stanisław Widel, Jarosław Flak, Piotr Gaj, Interpretation of Dual Peak Time signal Measured in Network systems, Computer Networks, Communications in Computer and Information Science 79, 17th Conference, CN210, 141-152, 2010
- [181] Stanisław Widel, Halina Rychlicka, Wybrane problemy architektury serwerów fonicznych, Studia Informatica, VIII Konferencja Sieci Komputerowe, Vol. 22, No. 2 (44), 245-260, 2001
- [182] Stanisław Widel, Jan Machniewski, Wykorzystanie algorytmu rsync do budowy serwerów synchronizujących dane, Wydawnictwa Naukowo Techniczne, Warszawa, 2004
- [183] Stanisław Widel, Jan Machniewski, Marek Fiuk, Systemy czasu rzeczywistego Tom 1: Kierunki badań i rozwoju, Wykorzystanie metody analizy wartości średnich MVA do wyznaczania czasów odpowiedzi Interakcyjnych systemów głosowych, Wydawnictwo Komunikacji i Łączności, Warszawa, 2005
- [184] Stanisław Widel, Jan Machniewski, Marek Fiuk, Wyznaczanie czasu wykonania operacji na podstawie dziennika systemowego serwera, Wysokowydajne sieci komputerowe, Nowe technologie, Praca zbiorowa, Wydawnictwa Komunikacji i Łączności, Warszawa, 2005
- [185] Stanisław Widel, Jan Machniewski, Measurement and data acquisition of execution time from application log, Polish Teletraffic Symposium, 16th, September 24-25, Łódź, 145-149, 2009
- [186] Stanisław Widel, Piotr Gaj, Jarosław Flak, Metoda analizy czasowej systemów czasu rzeczywistego za pomocą splotu, Metoda wytwarzania i zastosowania systemów czasu rzeczywistego, Konferencja Systemy Czasu Rzeczywistego, SCR 2010, 173-186, 2010
- [187] Stanisław Widel, Koncepcja rozwiązania sieci lokalnej Local SM/PC w oparciu o krajową konstrukcję sieci, Konferencja: Mikrokomputery w sieci lokalnej, Gdańsk, Sopot, 61-72, 1989
- [188] Stanisław Widel, Konstrukcja sterownika Sieci Ethernet opartego na koprocesorze sieci lokalnej Intel 82586, Zeszyty Naukowe Politechniki Śląskiej, Seria: Informatyka, zeszyt 20, nr 1190, 33-48, 1992
- [189] Stanisław Widel, Tomasz Radko, Rozwiązania faksowe w sieciach lokalnych, Computerworld, 41, ISSN 08672334, 32, 1995
- [190] Stanisław Widel, Faksowanie w sieciach NetWare, Computerworld, 3, ISSN 08672334, 21, 1996
- [191] Stanisław Widel, Piotr Stera, Jacek Piasecki, Interakcyjne Systemy Telefoniczne, Politechnika Śląska, Instytut Informatyki, Tom VI, Dokumentacja techniczna, 1997
- [192] Stanisław Widel, Piotr Stera, Założenia wstępne do projektu komputerowego systemu telefonicznego w projekcie Pszów, Politechnika Śląska, Instytut Informatyki, Tom V, Dokumentacja techniczna, Gliwice, marzec 1997
- [193] Stanisław Widel, Jan Machniewski, Marek Fiuk, Zdalny dostęp do danych za pomocą interakcyjnych systemów telefonicznych w jednostkach administracji publicznej na przykładzie Urzędu Miasta Pszów, Zeszyty Naukowe Politechniki Śląskiej, Seria: Informatyka, Zeszyt 34, Nr 1381, 465-475, 1998
- [194] Stanisław Widel, Halina Rychlicka, Algorytmy translacji podstawowych formatów danych na postać głosową dla Interaktywnych Systemów

- Telefonicznych, Zeszyty Naukowe Politechniki Śląskiej, Seria: Informatyka, Zeszyt 36, Nr 1414, 633-649, 1999
- [195] H. B. **Wilson**, Louis H. Turcotte, David Halpern, Advanced mathematics and mechanics applications using MATLAB, CRC Press, Boca Raton, USA, 2002
- [196] Michael **Wooldridge**, Paolo Ciancarini, Agent-Oriented Software Engineering: The State of the Art, Computer Science, Volume 1957, Lecture Notes in Computer Science, 55-82, 2001
- [197] Robert **Wójcicki**, Wpływ mechanizmów kontroli natężenia ruchu na jakość usług w sieciach TCP/IP ATM, Rozprawa doktorska, Politechnika Śląska, Wydział Automatyki Elektroniki i Informatyki, Instytut Informatyki, Gliwice, 2005
- [198] William **Yarberry**, Computer telephony integration, CRC Press, Boca Raton, USA, 2002
- [199] J. **Yeh**, Real analysis: theory of measure and integration, World Scientific, Danvers, USA, 2006
- [200] Tomasz P. **Zieliński**, Cyfrowe przetwarzanie sygnałów. Od teorii do zastosowań, Wydawnictwa Komunikacji i Łączności, Warszawa, 2005